NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

CHANGE DETECTION OF MARINE ENVIRONMENTS USING MACHINE LEARNING

by

Theodore A. Ayoub III

June 2020

Thesis Advisor: Neil C. Rowe
Co-Advisor: Mara S. Orescanin

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2020	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE
CHANGE DETECTION OF MARINE ENVIRONMENTS USING MACHINE LEARNING

5. FUNDING NUMBERS

6. AUTHOR(S) Theodore A. Ayoub III

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release. Distribution is unlimited.

12b. DISTRIBUTION CODE
A

13. ABSTRACT (maximum 200 words)

Change detection in aerial imagery is important for many disciplines. The Navy and Marine Corps have used it for planning missions in coastal areas. Current change-detection methods analyze registered images pixel by pixel. We trained two convolutional neural networks, VGG19 and DenseNet121, to distinguish eight coastal classes using 8,000 oblique aerial images. We then used both commercial satellite images and orthorectified imagery from small unmanned aerial systems with structure-from-motion methods to test whether learning could transfer from the oblique images. Our results showed significant effects of the tile size for coastal classification in the orthorectified images. We also tested change detection by comparing the highest confidence coastal class between images taken in different time periods, but this was often unsuccessful due to the errors in the classifications.

14. SUBJECT TERMS
machine learning, computer vision, oceanography, change detection, neural network

15. NUMBER OF PAGES
67

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

CHANGE DETECTION OF MARINE ENVIRONMENTS USING MACHINE LEARNING

Theodore A. Ayoub III
Captain, United States Marine Corps
BSME, State Univ of New York Inst of Tech at Utica-Rome, 2012
MS, University of Arkansas, 2017

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
June 2020

Approved by: Neil C. Rowe
Advisor

Mara S. Orescanin
Co-Advisor

Peter J. Denning
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Change detection in aerial imagery is important for many disciplines. The Navy and Marine Corps have used it for planning missions in coastal areas. Current change-detection methods analyze registered images pixel by pixel. We trained two convolutional neural networks, VGG19 and DenseNet121, to distinguish eight coastal classes using 8,000 oblique aerial images. We then used both commercial satellite images and orthorectified imagery from small unmanned aerial systems with structure-from-motion methods to test whether learning could transfer from the oblique images. Our results showed significant effects of the tile size for coastal classification in the orthorectified images. We also tested change detection by comparing the highest confidence coastal class between images taken in different time periods, but this was often unsuccessful due to the errors in the classifications.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

GPS Global Positioning System

VGG19 Visual Geometry Group's 19

DenseNet121 Dense Convolutional Network 121

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

1.1 Motivation

The Marine Corps and the other services have been conducting amphibious landings since the formation of the United States. In World War II, American forces stormed the beaches of France in Operation Neptune. As the invasion went on, setbacks from the littoral environment influenced continuing operations. Coastal changes, such as the difference between low tide and high tide, stranded some ships, leaving them immobile and making them easy targets. When the United States made the decision to land forces into Korea in the early 1950s, Navy LCDR Eugene Clark went on a month-long secret mission to study the landing areas of Inchon to provide a beachhead for forces to land [1]. During the invasion of Iraq in 2003, the amphibious assault on the Al-Faw peninsula suffered setbacks from poor visibility and littoral landing operations. These operations relied on geospatial intelligence and knowledge of coastal change detection.

1.2 How Computer Vision Helps Change Detection

Computer vision and neural networks have inspired much research and innovation over the past few decades. Computer vision can sift through large volumes of data faster than a human to classify or locate specific objects in images. Examples of computer vision include determining which buildings survived a tsunami [2] or whether an image contains a helicopter [3].

Computer vision can aid in surveillance, intelligence gathering, and mission-specific planning. We generate volumes of surveillance data from satellites that are ignored because we analyze them manually, but that could be processed quickly through computer vision. Neural networks can be trained to look for specific terrain in satellite imagery, picking out anomalies for a human to review. Computer vision should reduce the workload of image data processing for human intelligence personnel.

1.3 Flawed Change Detection Methodology in the Military

The coastal-landscape change-detection problem is important to the military for two reasons. The first is the big-data problem. The U.S. government collects large amounts of aerial and satellite images for change detection and sorting. The review of this data requires a substantial investment of time. An automated program that could address first-phase processing of satellite data could distinguish the important data from the less important data, as for example morphological change from storms from typical seasonal data. The second reason is the human-effort problem. Currently, personnel create change detection maps by hand; an automated program could free up personnel to focus on higher-level analytics.

Automated coastal change-detection techniques could permit the United States to keep a constant surveillance of changes to coastal zones suitable for landing forces. This process could improve landing forces' speed in deployment and increase the number of options available to them. The two technologies of computer vision and aerial surveillance could be combined to gather, analyze, and classify coastal landscapes without sending troops behind enemy lines in risky missions to gather intelligence.

1.4 Research Focus

The objective of this research is to test computer-vision methods that use deep neural networks to detect changes in the coastal-environment classification between two time periods. A littoral (coastal) environment could undergo a drastic change after a significant weather event. The main research question is whether a coastal-classification change-detection method can sufficiently differentiate coastal classes between two registered images. The hypothesis of this thesis is that it can. It will be tested by analyzing registered aerial images and Google Earth satellite imagery of the same area at different times. Neural networks will classify the images.

We tested images of Carmel River State Beach because it is subject to frequent storm-induced change. Its close proximity to Naval Postgraduate School permitted on-site inspection to provide ground truth for the classifications that the neural networks produce.

1.5 Thesis Outline

Chapter 2 reviews related work in the area of change-detection methods. This chapter also explains the basics of change detection as well as all terminology. Chapter 3 explains classification of coastal terrain and why it matters to the military. Chapter 4 describes the neural-network architectures, classification methods, and change-detection methodology used in the experiments. Chapter 5 discusses the experimental results and compares them to ground truth. Chapter 6 summarizes the thesis and suggests further research.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Background

2.1 Neural-Network Concepts

An artificial neuron (or node) computes the weighted sum of a set of inputs and applies a nonlinear activation function to it to get an output [4]. Its inputs and outputs are often expressed as vectors or arrays of vectors. An artificial neural network is an interconnected set of neurons. A neural-network layer is a collection of neurons at the same distance from the inputs. Neural networks have an input layer, hidden layers, and an output layer, as shown in Figure 2.1. In the Figure, squares denote the input layer and circles denote the hidden and output layers. A neural network is considered a deep neural network if it contains one or more hidden layers. Neural network research has explored several types of hidden layers including convolutional, recurrent, pooling, and flattening.

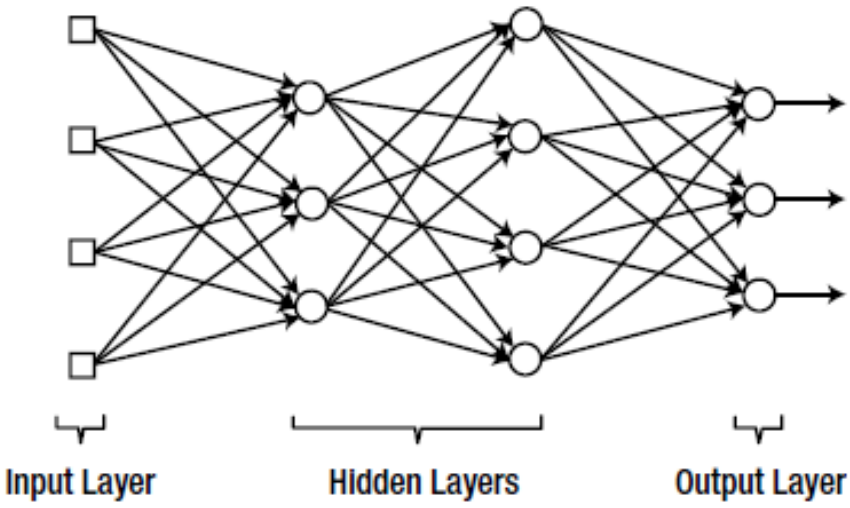

Figure 2.1. Basic neural network diagram. Source: [4]

When processing images, the inputs to a neural network are either single brightness values for each pixel or sets of red-green-blue triples for each pixel. A pixel is the simplest element of an image [5]. This thesis used color (red-green-blue) images, so the inputs for neural networks were red-green-blue triples.

Feature vectors are the outputs from a hidden layer in a neural network. We only discuss the hidden layers that are of interest to this thesis, such as convolutional layers. In this thesis, we are especially interested in the feature vectors at the output of the last convolutional layer (defined later) in the network. Collectively, these feature vectors form a **feature map**, which correspond to geographical locations.

In a neural network designed for classification like ours, the **prediction vector** represents the computed likelihoods that the neural network has for each possible classification of an image. It is the vector of the outputs of the final layer. The prediction vector had a length of 8 for the number of terrain classifications in this thesis; terrain classifications will be described in Chapter 3.

Convolutional neural networks get their name from the mathematical functions they use to process data, the convolutions in the convolutional layers [5]. A convolutional function is a linear function that computes the sum of the weighted values for data in a local area [6], where weights can be negative. For images, the local areas are subregions of the image, and the convolutional layers extract features of those subregions and produce a feature map. This can also be described as passing a two-dimensional-matrix called a filter over the image for each layer. Figure 2.2 shows the basic idea.

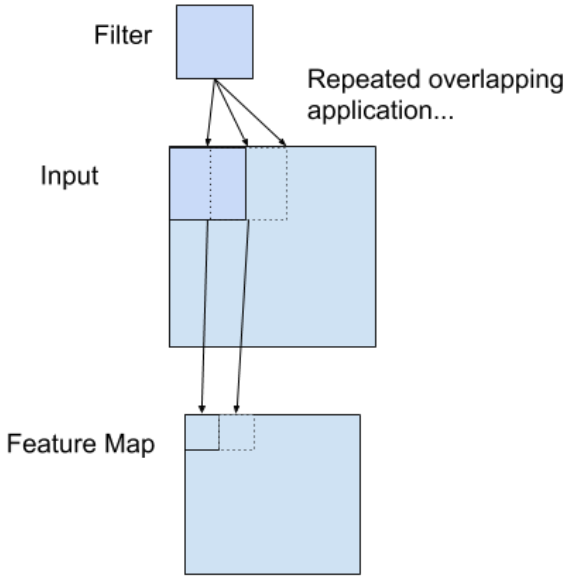

Figure 2.2. Convolutional filter to feature map diagram. Source: [7]

When a neuron in one layer does an operation it passes its output to all the neurons in the next layer as an input. In the convolutional layers, the same linear weighted sum is applied to every neuron in the same layer, but the neurons of the later non-convolutional layers can all compute different weighted sums. Figure 2.3 is another illustration of how a convolutional filter passes over an input image to extract information.

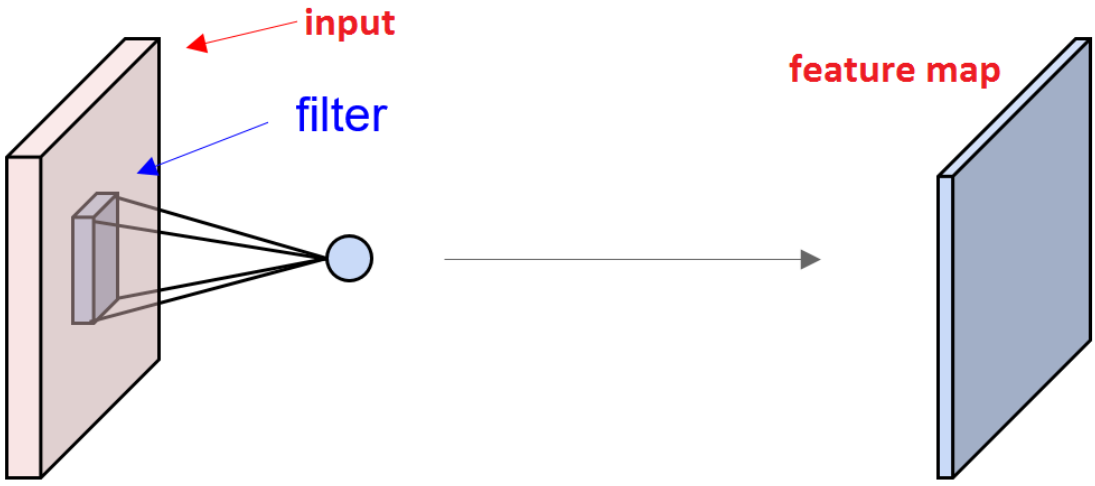

Figure 2.3. Convolutional filter to neuron diagram. Source: [8]

Back propagation is most popular way for a neural network to learn [9]. It works backward from the output layers to the input layer and applies an adjustment formula to the weights. **Transfer learning** is using a network that is trained for one task and repurposing it to another task [5]. This is often done by adjusting the weights in the network to handle different input. Networks that were originally trained for very different inputs require more training time and larger datasets to acquire adequate classification with transfer learning.

Transfer learning typically requires a training set, a validation set, and a test set [10]. A **training set** is for fitting the weights of a neural-network model. A **validation set** is for evaluating how well the model parameters are adjusting during training. A **test set** is for evaluating performance of the model after it is completely trained.

2.2 Neural Network Models

Our thesis research used two neural-network models (implementations), the Visual Geometry Group's 19 (VGG19) and the Dense Convolutional Network 121 (DenseNet121). The VGG19 convolutional neural network is a general-purpose tool developed for computer-vision applications [11]. It has nineteen layers. The model was part of the ImageNet challenge in 2014, where it scored high marks. Since then, the model has been available as an open- source network for other researchers to use to experiment with computer vision. DenseNet [12], developed in 2016, comes in three sizes: 121 layers, 169 layers, and 201 layers [13]. This thesis used the 121-layer version during experimentation. It uses the term **block** for important sequences of layers.

Siamese neural networks can compare data for change detection. They consist of two subnetworks that are joined by a few additional layers. Pseudo-Siamese networks use different parameters in the two subnetworks. Figure 2.4 shows the basic layout. At the output end, the base networks are joined by a similarity-measuring subnetwork. The simplest similarity function is a Euclidean distance between two feature vectors. In [14], the similarity function was a two-layer decision network with 512 nodes.

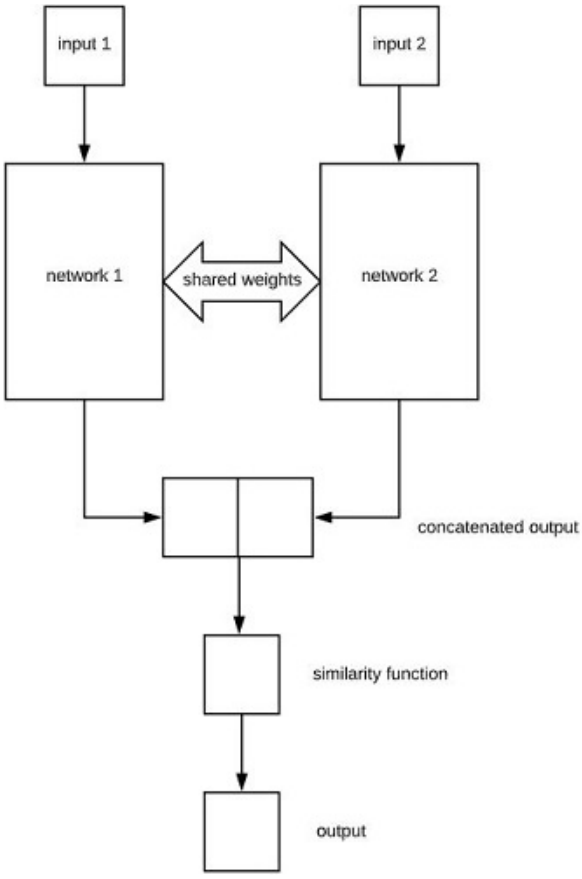

Figure 2.4. Basic Siamese neural network diagram.

One study compared a Siamese network that shared parameters, a pseudo-Siamese network that did not share parameters, and a center-surround two-stream network with four branches that join; the two-channel network did the best [15]. Subsequent work [2] did change detection for analyzing photographs after a tsunami. The Siamese network was trained to distinguish "washed away" and "surviving"; the pseudo-Siamese network did the best, with accuracy in the 94-95% range.

Siamese networks have been trained for helicopter detection in images [3]. Their dataset was computer-generated imagery with "bootstrapping" to augment it by reusing misclassified data. The best results happened when they used two Siamese networks to feed into a two-layer decision network. Research [15] and [16] produced a neural network that could learn a similarity function automatically for change detection. To clean up their output in the final output, they used a contrastive loss function adapted from [17], a distance-based loss function.

2.3 Image Registration

Image registration means putting locations in two images into the same coordinate system. Good image registration is important for change detection because spurious changes can be found at misaligned points. In registering, calculations are made of the overlap area, scale change, orientation change, and rotation change [18]. Registration is aided by matching distinctive features of the images such as roads, road intersections, and buildings. Manmade features are preferable for matching because they change less over time than environmental features such as rivers, coastlines, and tree lines.

Orthorectification is a process for registering images that have elevation, scale, or viewing angle discrepancies [19]. It enables us to match images to the same location coordinate system when they are from different times. Orthorectified images are often stitched together from aerial images in a process called **structure-from-motion**, a photogrammetry method that combines aerial photographs to cover a larger region [20].The Global Positioning System (GPS) surveyed "ground control points" are recognized and recorded when the images were being captured as reference points that help with orthorectification [21]. Elevation differences are a challenge for orthorectification; errors are minimized between ground and image coordinates to overcome this challenge.

2.4 Change Detection Methods

Image **change detection** identifies features or regions in two registered images that differ significantly [22]. Changes in color and object shape between two matched images are important, as well as the presence of an object in one image that is not in the other. Differences between images can also be detected at the most basic level as differences in corresponding pixels (image points), as with red, green, and blue values for conventional photographs. A difference image is produced by subtracting corresponding pixel vectors (red-green-blue values) of two images. To mark where changes are significant, a threshold can be applied to the norms (magnitudes) of the difference-image vectors to get an image-change mask, which is a binary image indicating pixels with significant change.

Ratios of frequency bands instead of differences can also detect significant changes in an image. One study detected coastal changes in radio-frequency bands b2 (1900 megahertz), b4 (1700 megahertz), and b5 (850 megahertz) [23]. Band b5 shows the greatest contrast between land and water. The ratios between spectral bands b2 and b4 and bands b2 and b5 are also helpful in distinguishing land and water. This approach multiplied two binary images, an image created by thresholding band b5 on the coastline image, and an image created by taking the ratios of spectral bands b2 and b4 along with b2 and b5. Thresholding the result produced a map of changes to the shoreline.

A different project used satellite images to detect changes in soil composition over time in Brazil [24]. Researchers used self-organizing maps (a form of unsupervised learning) to define the classes. They produced a binary change-detection image that had over 80% precision and recall scores. Transfer learning was used to train a neural network for landscape classification in [25].The study used MobileNetV2 for the early layers, a model imported through the Keras library. The study also experimented with tiling techniques. The study yielded average classification accuracies between 91% and 98%. Classification can be a precursor to change detection.

2.5 Previous Work Conducted in This Study

This thesis built upon a previous thesis at our school [26] in which a VGG19 neural network was trained for coastal-image classification. It used a coastal-class dataset of over 10,000 oblique aerial images described in chapter 4. The images came from the United States Geological Survey [27] and the California Coastal Records Project [28]. All images in the dataset were scaled to 299x299 pixels. This work focused on classification and did not address change detection. The project did transfer learning on the neural network from the parameters obtained from a different kind of image. This project stopped training on some earlier layers while continuing to train the later layers, one form of transfer learning; freezing the first five layers appeared optimal.

CHAPTER 3:
The Coastal Classification Problem

Navigation safety in littoral regions for landing of military forces is not easy. A littoral environment can be assigned to one of eight **coastal classes**: tidal flats, sandy beaches, salt marshes, manmade structures, dunes, coastal waterways, coastal rocky, and coastal cliffs [29]. Each coastal class presents different problems for the Navy and Marine Corps. Amphibious landing forces prefer to land on stable surfaces such as sandy beaches when going from ship to shore. A force landing on a large area that contains tidal flats (also called mud flats) could experience a severe slowdown. Landing forces prefer to avoid salt marshes because they do not offer solid ground for troop movements. Manmade structures, dunes, coastal cliffs, and rocky areas present dangers to landing forces because they create obstacles which require energy to overcome and they could be made into fortified positions by the enemy. Coastal waterways and seasonal ephemeral rivers could be a problem because the area around the waterway could be unstable [30].

Coastal classification especially concerns a specific Navy community, the surface-connector fleet, or the ships designed to move troops from ship to shore. These include amphibious assault vehicles, Marine Corps infantry, and Marine Corps landing forces (logistic support agencies) that support sustained operations onshore. Analysis of coastal imagery could enhance landing-operation preparedness. With automated coastal-change detection, a military has timely information on coastal zones suitable for landing forces. This should increase speed of deployment for landing forces and increase the number of options they have available. This technology directly supports the Marine Corps' "Title-10" functions to land on foreign shores and establish a foothold in an area knowing that the littoral bottom can support incoming vehicles and Marines on foot. If the area cannot support vehicles, coastal analysis can suggest where to stop and dismount.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4:
Methodology

4.1 Introduction to the Image Data

Datasets for landscape change detection can include imagery from satellites, unmanned aerial systems and aircraft. Important benchmark datasets were created by [2], [14], [31], [32]. Our research required new datasets. For this thesis, we used three types of imagery: oblique aerial imagery, orthorectified unmanned-aerial-systems imagery, and Google Earth satellite imagery. The oblique aerial imagery dataset was just for training the neural network models. The orthorectified images and the Google Earth images were tested by the trained neural networks.

4.1.1 Oblique Aerial Images

The oblique aerial dataset contained over 8,000 color images of coastal landscapes. These images were collected from open-source databases of the California Coastal Records Project [28] and the US Geological Survey [27] and covered our 8 coastal classes well. It included data from the West, East, and Gulf Coasts. The images were taken from unmanned aerial system aircraft at different heights, with different cameras, and at different angles [26]. Figure 4.1 shows eight example images that were used for training the neural networks.

To use neural networks for classification, it helps if images have identical dimensions. All images were subsampled to 299x299 pixels. The images were first cropped to a square, then down-sampled with an anti-aliasing filter [33].

Figure 4.1. Coastal classification examples. Top row, from left to right: coastal cliffs, coastal rocky, coastal waterway, and dunes classes. Bottom row, from left to right: manmade structures, salt marsh, sandy beach, and tidal flats classes.

4.1.2 Orthorectified Imagery

The images tested in this thesis were created from November 2017 through February 2018 by [20]. They came from a DJI Phantom III Advanced quadcopter aircraft equipped with a twelve-megapixel-resolution camera. The images were taken using structure-from-motion to monitor coastal morphodynamics at Carmel River State Beach. Figure 4.2 shows images before and after a river breach flowing out to the ocean. A river breach is the kind of significant change that could have an impact on operating forces. This orthorectified image from unmanned aerial system imagery, while able to produce elevation data, also includes image artifacts owing to shadows created by obstacles (such as vegetation). In our work, we did not use the elevation data.

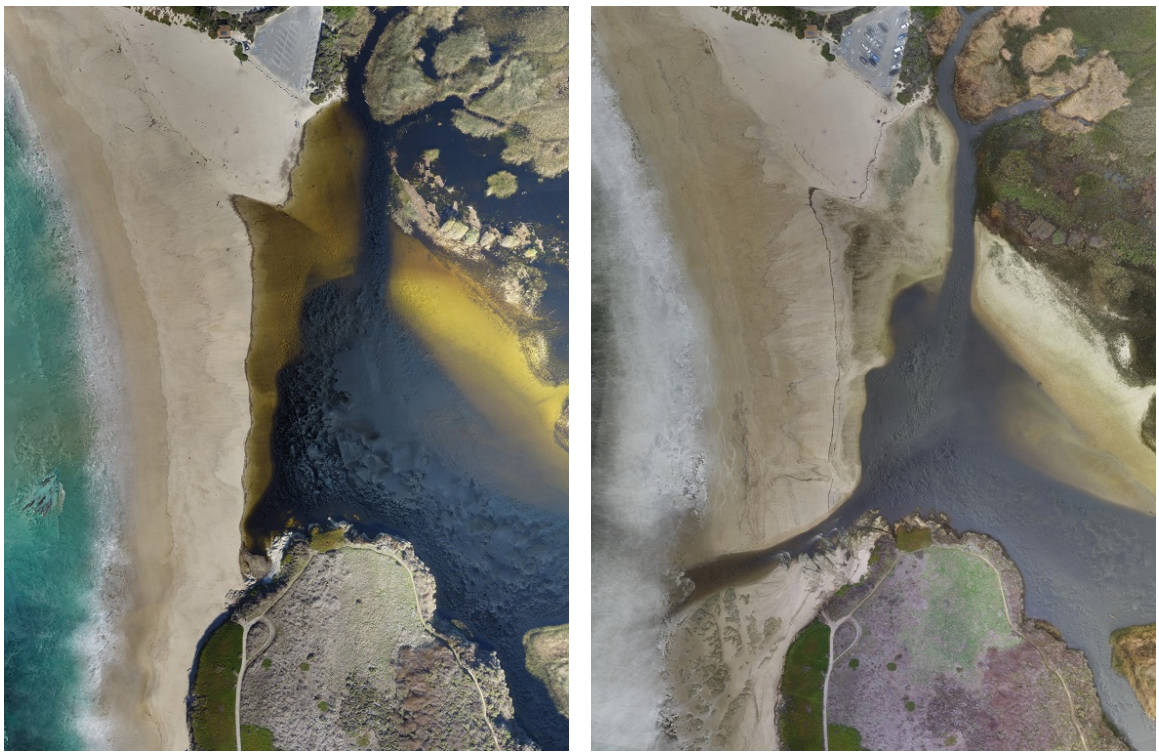

Figure 4.2. Orthorectified images created from unmanned aerial system imagery. Left image: December 6, 2017. Right image: January 10, 2018. The images are 5784x7621 pixels.

4.1.3 Google Earth Imagery

The satellite images studied were color Google Earth images taken of Carmel River State Beach. A sample of five images were taken from Google Earth representing one square mile of the Carmel river State Beach area. The images are not on the same days as the orthorectified images because the Google Earth application did not have those specific dates saved in its history. Google Earth depicts the earth in a three-dimensional space by using aerial and satellite imagery [34]. We did not use the elevation from the application explicitly. Satellite imagery offers larger scales (map scales) because the photos are taken at a greater distance from the Earth.

Figure 4.3. Google Earth images from (left) June 17, 2017 and (right) September 14, 2018 [35].

4.2 The Ground Truth

The oblique aerial imagery dataset described in section 4.1.1 served as labeled ground truth for coastal classification. This dataset was hand labeled and checked by [26].

The two images in Figure 4.4 were the basis for the only ground truth for the orthorectified images. They were labeled by Prof. Orescanin with support by Matlab software which allows a researcher to interactively label regions of a picture. These two images have four key features for military analysis: the beach in the middle, the peninsula on the lower end, the coastal waterway that runs from the lower right corner up through the middle, and the marsh in the upper right corner.

We do not have explicit ground truth for the Google Earth satellite imagery that we used, but Figure 4.4 can approximate it because it shows the same location. Eventually, students at the school in the oceanography department will do on-site observations to get more accurate ground truth.

Ground truth is necessary for testing of classification; however, there is a separate ground truth for change detection. For our test images, it was supplied by visual inspection. The main change that we looked for was the river breach seen in the lower portion of the January image since that has the most significance for use of the terrain.

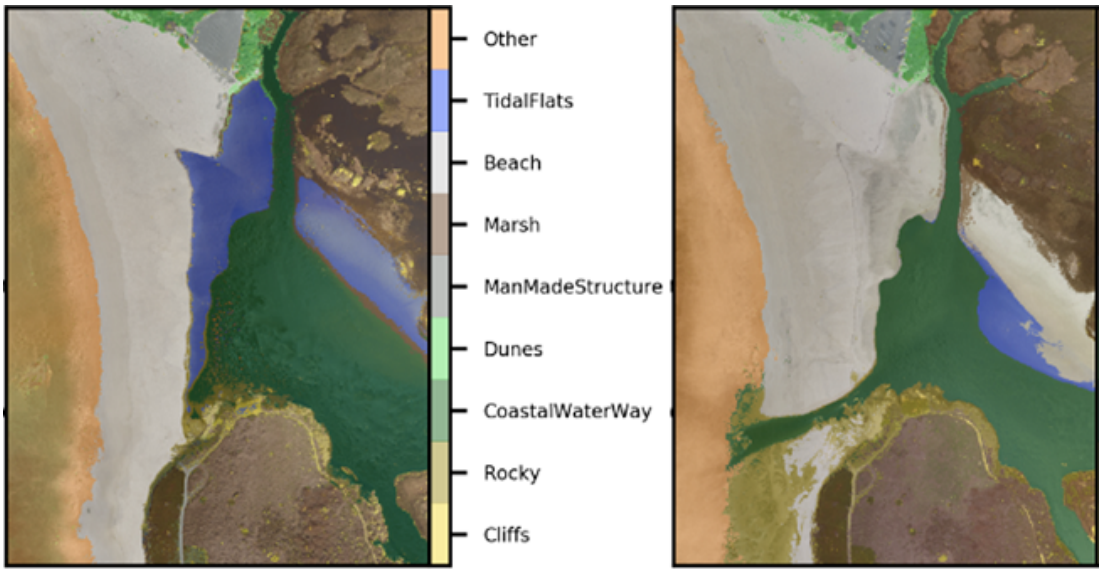

Figure 4.4. Ground truth labeled images. Left image: December 6, 2017, Right image: January 10, 2018.

4.3 Neural Network Setup and Description

4.3.1 Neural Network Hardware and Software Configuration

The neural networks were trained on two platforms, a Dell workstation with an i7 processor and 64 gigabytes of memory using an NVIDIA graphics processing unit, and Amazon Web Services Cloud service using a NVIDIA Tesla V100 graphics processing unit with 16 gigabytes of memory. The networks were also tested on the Dell workstation mentioned above and a Micro-Star International gaming laptop with an Intel i7 processor and 16 gigabytes of memory, using a NVIDIA GeForce GTX graphics-processing unit with 3 gigabytes of memory.

Both neural networks were imported into our Python environment using the Keras library. Keras is an open-source library used for neural-network experimentation [13]. Neural networks importable through Keras have been trained with the ImageNet database, a large labeled dataset of one million images of a wide range of subjects.

4.3.2 Neural Network Architecture

The VGG19 architecture has 16 convolutional layers and 3 subsequent fully-connected layers. There are 64 neurons in the first convolutional layer of the network where the network takes an image as input. Using our code, any image input gets downsampled to 256x256x3, which means that the image is 256x256 pixel size and has three color channels (red-blue-green) for a total number of inputs of 196,608 into the first layer of the neural network. There are 512 neurons in the last convolutional layer which we use to generate the class-likelihood maps. The final layer has 8 neurons (one for each coastal class).

The DenseNet121 architecture has 126 layers; 120 convolutional layers 1 fully connected layer, and 5 pooling layers. There are 64 neurons in the first convolutional layer of the network where the network takes the image as an input. As with VGG19, our code downsamples images for DenseNet121. The network takes an input image of 299x299 pixel size and has three color channels (red-blue-green) for a total number of inputs of 268,203 into the first layer of the neural network. There are 1024 neurons in the last convolutional layer (double that of VGG19) which we use to generate the class-likelihood maps. The final layer has 8 neurons (one for each coastal class).

4.3.3 Transfer Learning Dataset Configuration

We used the results of training on the oblique aerial imagery dataset (explained in section 4.1.1) as the source for transfer learning. There were 8,727 images for the eight coastal classes consisting of 6,400 training images, 1,600 validation images, and 727 test images.

4.3.4 Neural Network Transfer Learning Procedure

We reproduced the transfer learning of [26] for the VGG19 network and did our own transfer learning for DenseNet121 network. Transfer learning readjusted the neural network parameters to our specific task of predicting coastal classes from overhead imagery. The inputs for transfer learning were the training and validation images of the oblique aerial dataset; the output was a fitted model. This is the only time the oblique aerial dataset was used - to train the neural networks. The VGG19 model was 400 megabytes and the DenseNet121 model was 700 megabytes. The models took roughly a day each to train.

4.4 Processing Programs

Our programs used several Python libraries. The Keras library was for importing the networks, the TensorFlow library was for training neural networks on a graphics processing unit, the OpenCV library was for image processing, the Numpy library was for working with matrices, and the Matplotlib library was for plotting data in graphs.

4.4.1 Class-Likelihood Maps

Class-activation maps are visual representations of the feature maps from the last convolutional layer of a neural network [36], [37]. These show the degree to which a particular coastal class is present in an image. In this thesis, the class-activation maps will be referred to as class-likelihood maps. Class-likelihood maps were generated for the Google Earth satellite imagery and orthorectified imagery. Figure 4.5 shows how the class-likelihood maps are generated.

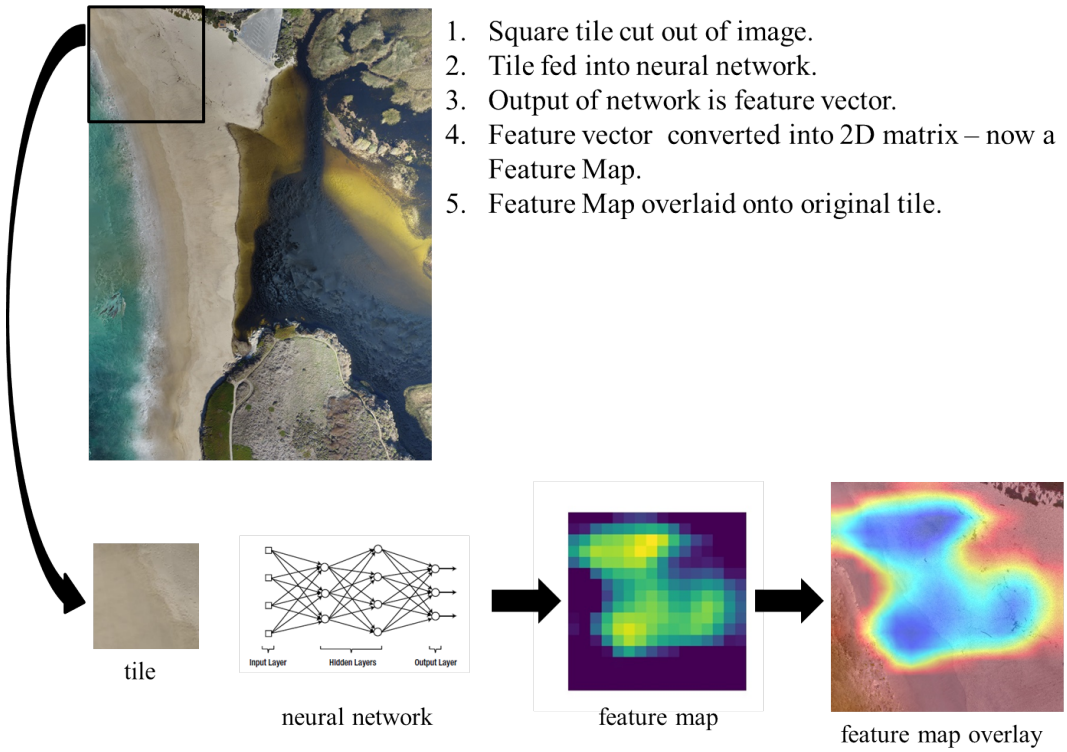

Figure 4.5. Class-likelihood map generation process flow diagram.

4.4.2 Tiling Graphics Programs

For the VGG19 we experimented with 256x256, 512x512, and 768x768 square pixel tiles; for DenseNet121 we experimented with 299x299, 598x598, and 897x897 square pixel tiles. We tested two types of tiling for disjoint tiles and sliding tiles. In the disjoint tiling, the images were tiled evenly with no overlap; in sliding tiling, the tiles were overlapped at a fraction of the tile size such as a half or quarter of the tile size. The sliding-window classifier gives higher granularity for classifications. We used smoothing was for reducing errors in the classification of tiles: If the highest likelihood for the tile was below 60% after normalization, the tile would have its classification replaced by the neighbor with the highest classification likelihood. We used the prediction vectors (outputs of the final layer) to show the best classification for each tile in an image, a different kind of graphics (Figure 4.6).

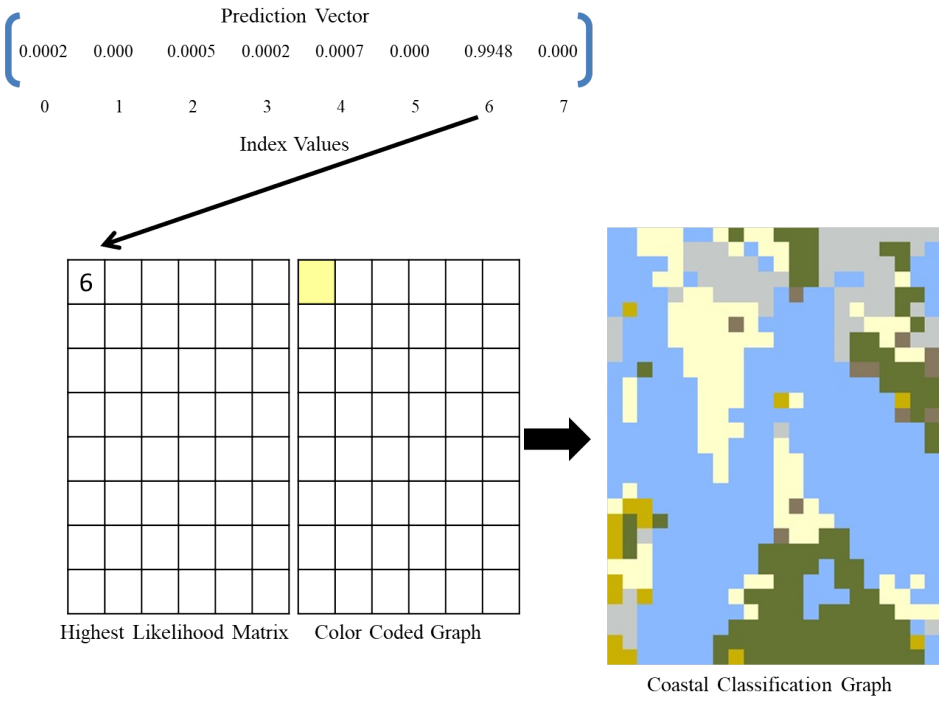

Figure 4.6. Coastal class graph generation

Change detection compared the classifications generated from the neural n etworks. Two displays were produced: a binary image that indicates those tiles that changed classification, and an images that show the coastal classes that developed.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5:
Results and Discussion

5.1 Neural Network Training with Oblique-Image

Neural network coastal-classification performance was calculated for the oblique aerial dataset described in section 4.1.1. Confusion matrices were constructed for the test set of 727 images: 93 tidal flats, 100 sandy beaches, 100 salt marshes, 81 manmade structures, 100 dunes, 53 coastal waterways, 100 coastal rocky, and 100 coastal cliffs. They showed which classes the network selected for each correct class.

Figure 5.1 shows the confusion matrix for the VGG19 neural network on the oblique aerial dataset. VGG19 had a 93.75% overall classification rate. The sandy beach class had 100 images in the test folder: there were 85 tiles that the neural network correctly identified as sandy beach, 4 tiles that it incorrectly identified as sandy beach, and 15 tiles that were sandy beach that it failed to identify as such. We thus calculate the precision as 96% and recall as 85% for the sandy beach class:

- True Positive — 85 tiles — numbers on the diagonal.
- False Positive — 4 tiles — sum of the column. (Type I error).
- False Negative — 15 tiles — sum of the row. (Type II error).

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} = \frac{85}{85 + 4} = 96\% \tag{5.1}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} = \frac{85}{85 + 15} = 85\% \tag{5.2}$$

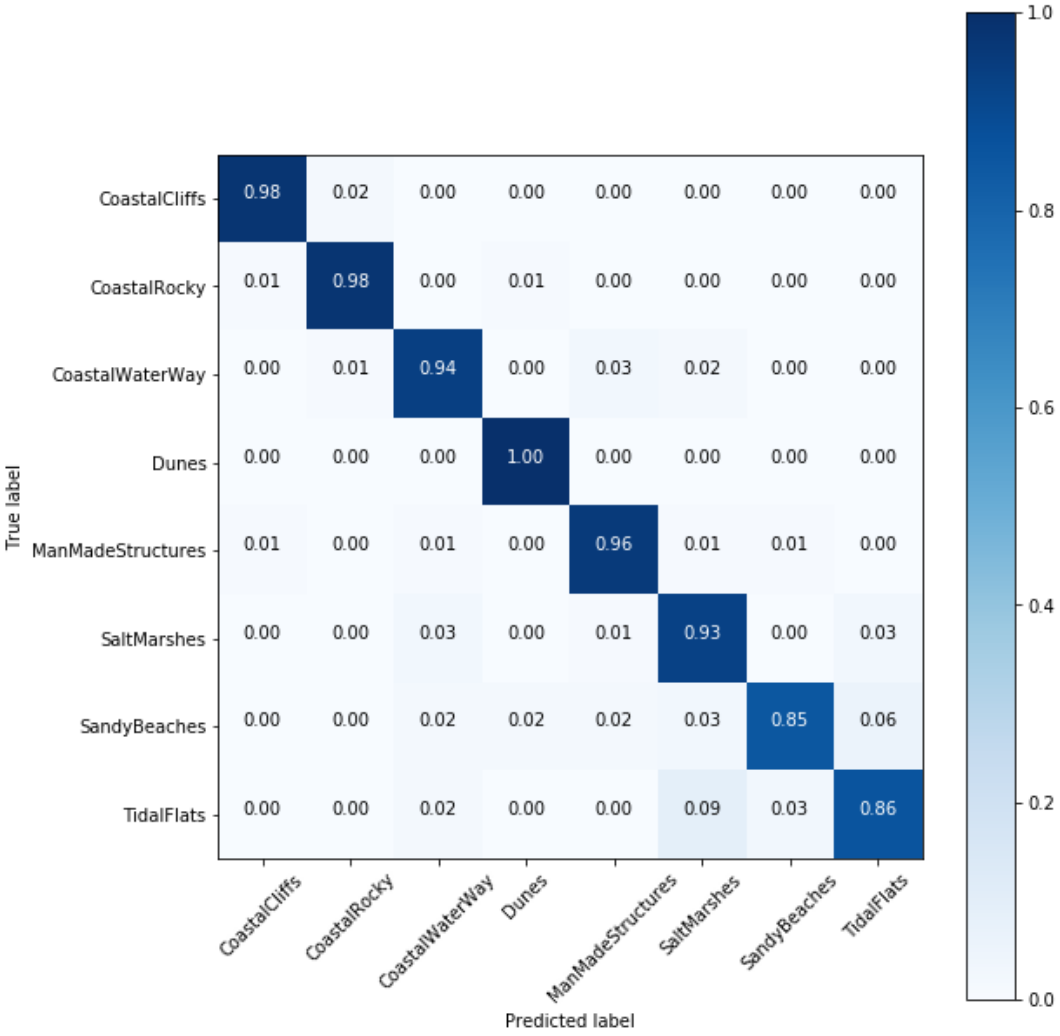

Figure 5.1. VGG19 confusion matrix showing 93.75% overall oblique aerial imagery classification rate.

Figure 5.2 shows the confusion matrix for DenseNet121 neural network on the oblique aerial dataset. It had a 95% overall classification accuracy. VGG19 classified sandy beaches at 85% and tidal flats at 86%; DenseNet121 classifies sandy beaches and tidal flats at 91%. Classifying these two classes is important because sandy beaches provide solid footing for landing forces while tidal flats slow movement down. The DenseNet121 network has double the number of neurons in the last convolutional layer of its network. This may be the reason that it has a higher success rate for classifying sandy beaches and tidal flats in the oblique aerial imagery.

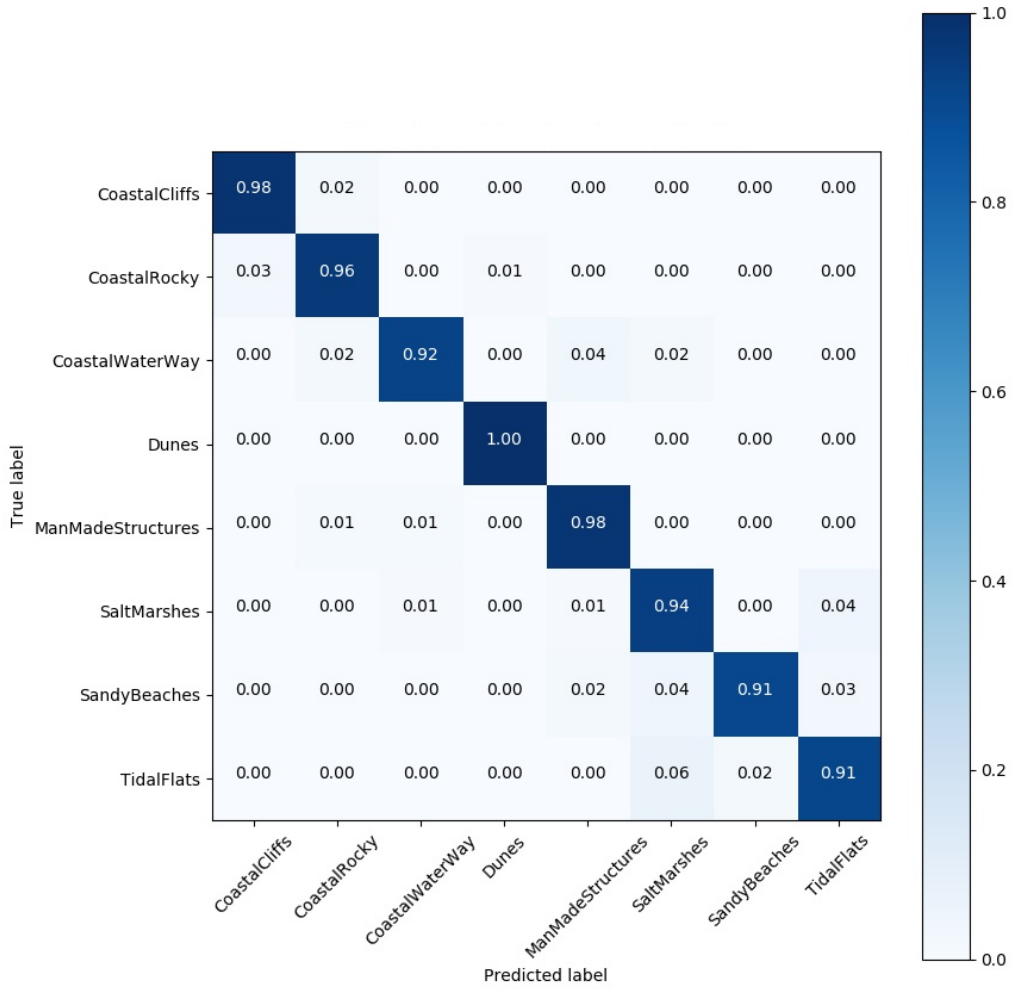

Figure 5.2. DenseNet121 confusion matrix showing 95% overall oblique aerial imagery classification rate.

5.2 Coastal Class-Likelihood Maps

Figures 5.3 and 5.4 show likelihood maps created for the orthorectified images by the VGG19 neural network; Figures 5.5 and 5.6 show the likelihood maps for DenseNet121. These show the degree to which a particular coastal class is present in an image. Blue means high likelihood of the class, and red means low likelihood. Note how similar tidal flats and sandy beach appear to the network (Figure 5.3 middle and right) since both are largely sandy.

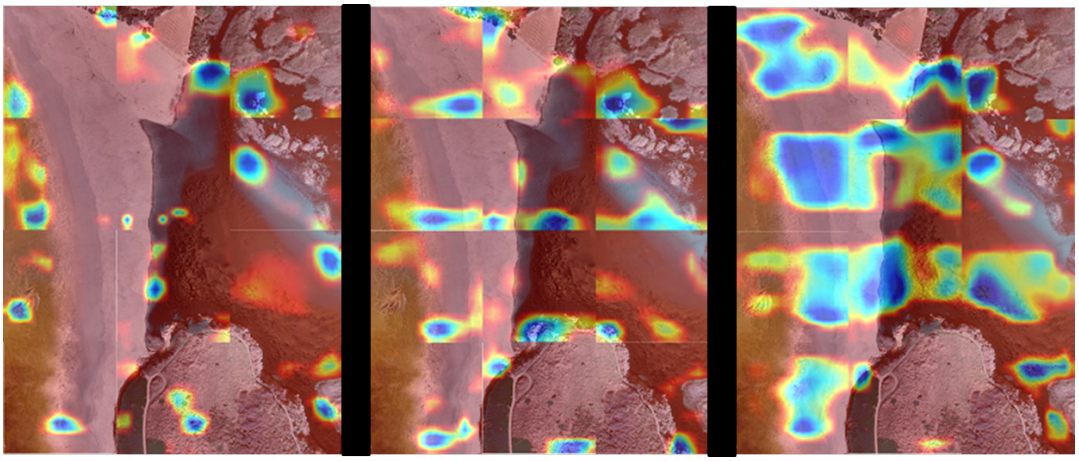

Figure 5.3. VGG19 December coastal class likelihood maps. Left image: likelihood of salt marsh. Middle image: likelihood of sandy beach. Right image: likelihood of tidal flats.

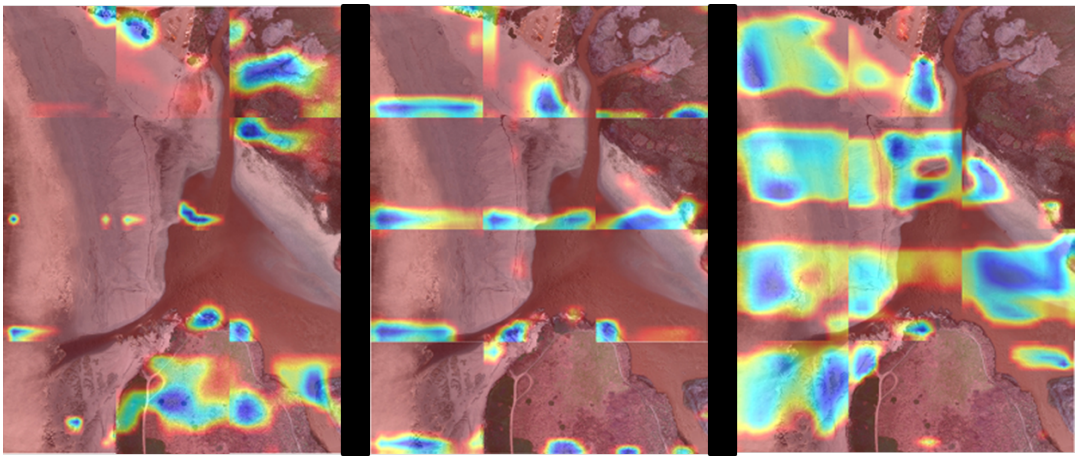

Figure 5.4. VGG19 January coastal class likelihood maps. Left image: likelihood of salt marsh. Middle image: likelihood of sandy beach. Right image: likelihood of tidal flats.

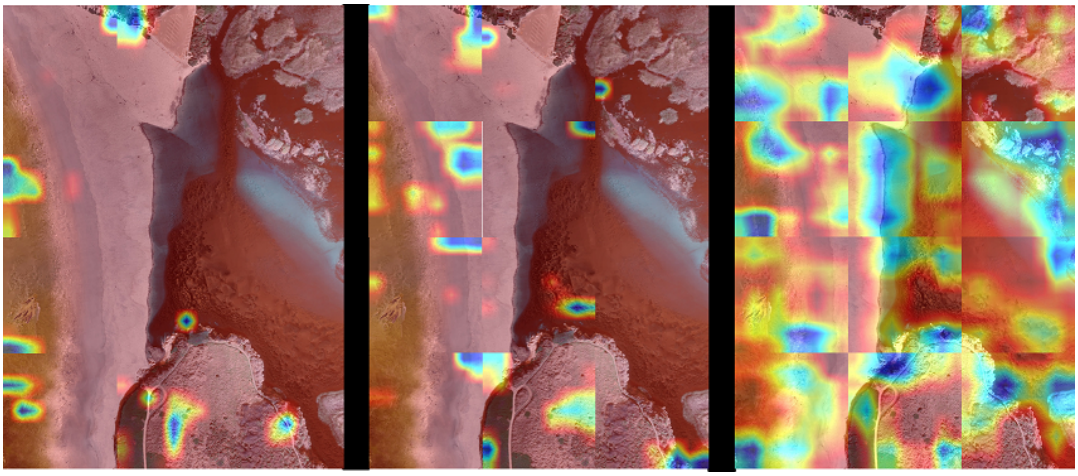

Figure 5.5. DenseNet121 December coastal class likelihood maps. Left image: likelihood of salt marsh. Middle image: likelihood of sandy beach. Right image: likelihood of tidal flats.

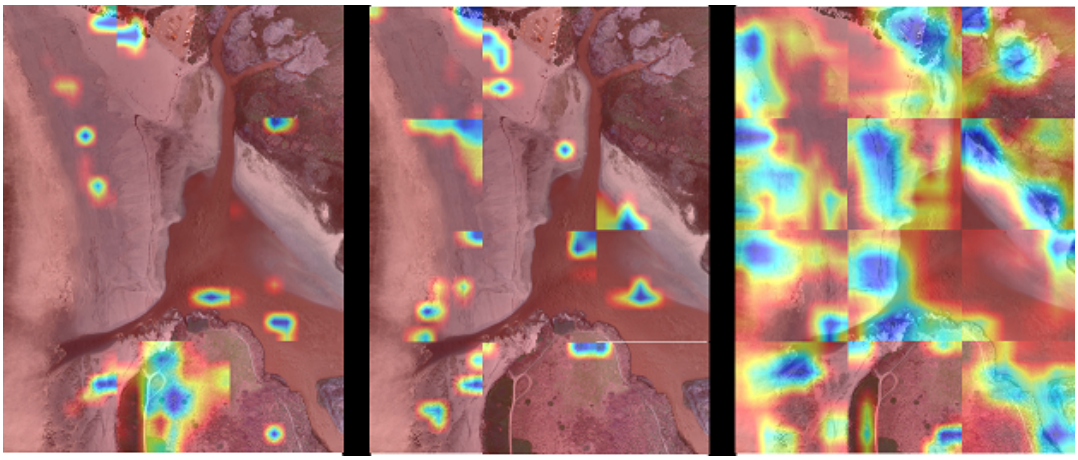

Figure 5.6. DenseNet121 January coastal class likelihood maps. Left image: likelihood of salt marsh. Middle image: likelihood of sandy beach. Right image: likelihood of tidal flats.

Figures 5.7 and 5.8 show likelihood maps created for Google Earth images by the VGG19 neural network. The leftmost image of Figures 5.7 and 5.8 show the original Google Earth image we used for generating the class likelihood maps, and the other subimages are likelihood maps for three specific coastal classes. The numbers in the leftmost subimage give tile numbers used in Tables 5.1 and 5.2. Figures 5.9 and 5.10 show the likelihood maps for the DenseNet121 network. Tables 5.3 and 5.4 correspond to Figures 5.9 and 5.10 respectively.

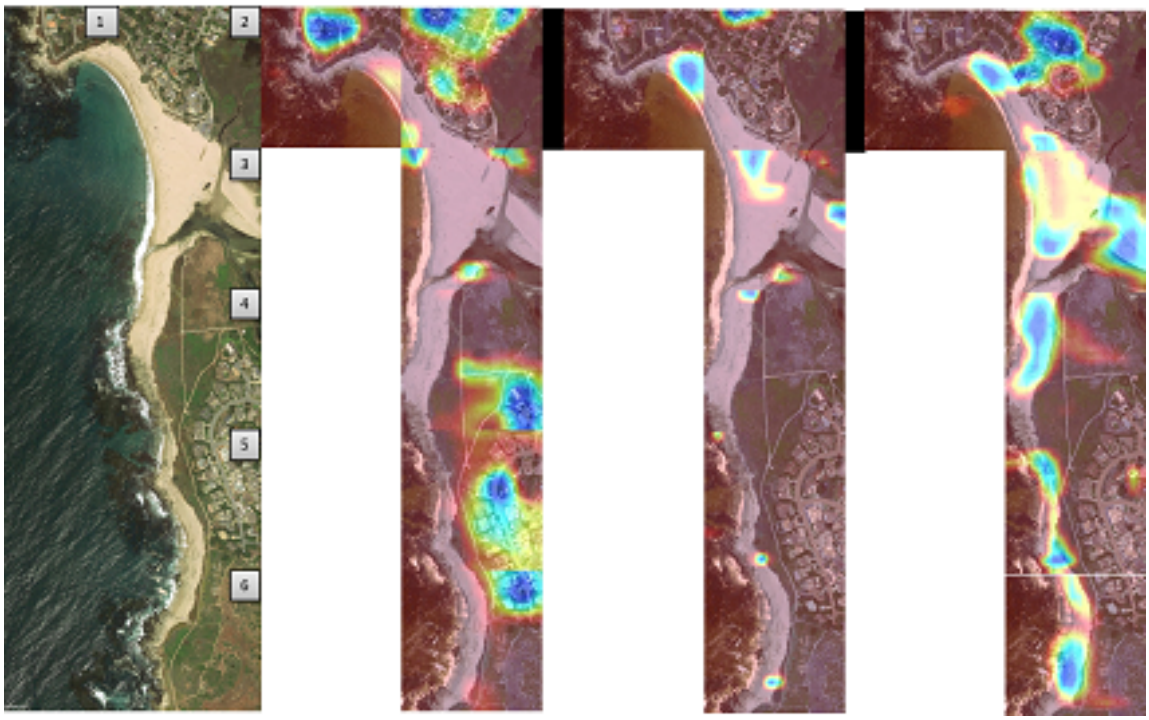

Figure 5.7. VGG19 June 2017 Google Earth class likelihood maps. Left image: original satellite image. Second image: likelihood of manmade structures. Third image: likelihood of sandy beach. Fourth image: likelihood of tidal flats.

Table 5.1. June 2017 class likelihood maps table for Figure 5.7. The table shows the prediction vectors for each tile, as labeled in Figure 5.7 (left).

Tile	Coastal Cliffs	Coastal Rocky	Coastal Water Way	Dunes	Man Made Structures	Salt Marshes	Sandy Beaches	Tidal Flats
1	0.02%	99.84%	0.05%	0.02%	0.07%	0.00%	0.00%	0.00%
2	0.00%	0.00%	0.02%	0.00%	2.09%	97.84%	0.00%	0.04%
3	0.00%	0.00%	0.08%	0.00%	0.00%	23.77%	0.04%	76.11%
4	0.03%,	0.08%,	24.10%,	0.00%,	25.00%,	50.59%,	0.01%,	0.19%
5	0.00%	0.77%	0.35%	0.03%	98.83%	0.02%	0.00%	0.00%
6	0.78%	56.67%	20.44%	1.25%	15.95%	4.44%	0.00%	0.47%

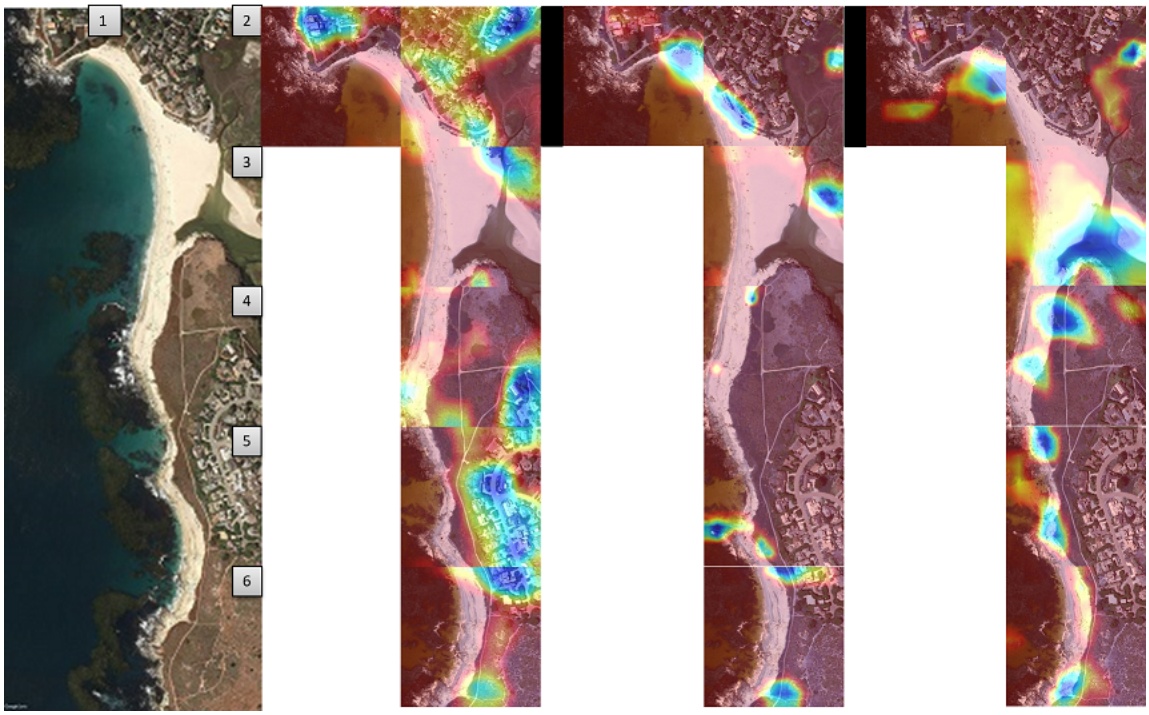

Figure 5.8. VGG19 September 2018 Google Earth class likelihood maps. Left image: original satellite image. Second image: likelihood of manmade structures. Third image: likelihood of sandy beach. Fourth image: likelihood of tidal flats.

Table 5.2. September 2018 class likelihood maps table for Figure 5.8. The table shows the prediction vectors for each tile, as labeled in Figure 5.8 (left).

Tile	Coastal Cliffs	Coastal Rocky	Coastal Water Way	Dunes	Man Made Structures	Salt Marshes	Sandy Beaches	Tidal Flats
1	0.07%	99.70%	0.01%	0.00%	0.21%	0.00%	0.01%	0.00%
2	0.00%	0.00%	0.01%	0.00%	99.97%	0.00%	0.01%	0.00%
3	0.00%	0.00%	0.00%	0.00%	0.40%	0.41%	0.41%	98.77%
4	0.02%	0.12%	11.17%	0.00%	88.66%	0.02%	0.00%	0.01%
5	0.00%	0.04%	0.00%	0.00%	99.96%	0.00%	0.00%	0.00%
6	0.08%	97.27%	0.48%	0.00%	2.04%	0.11%	0.00%	0.01%

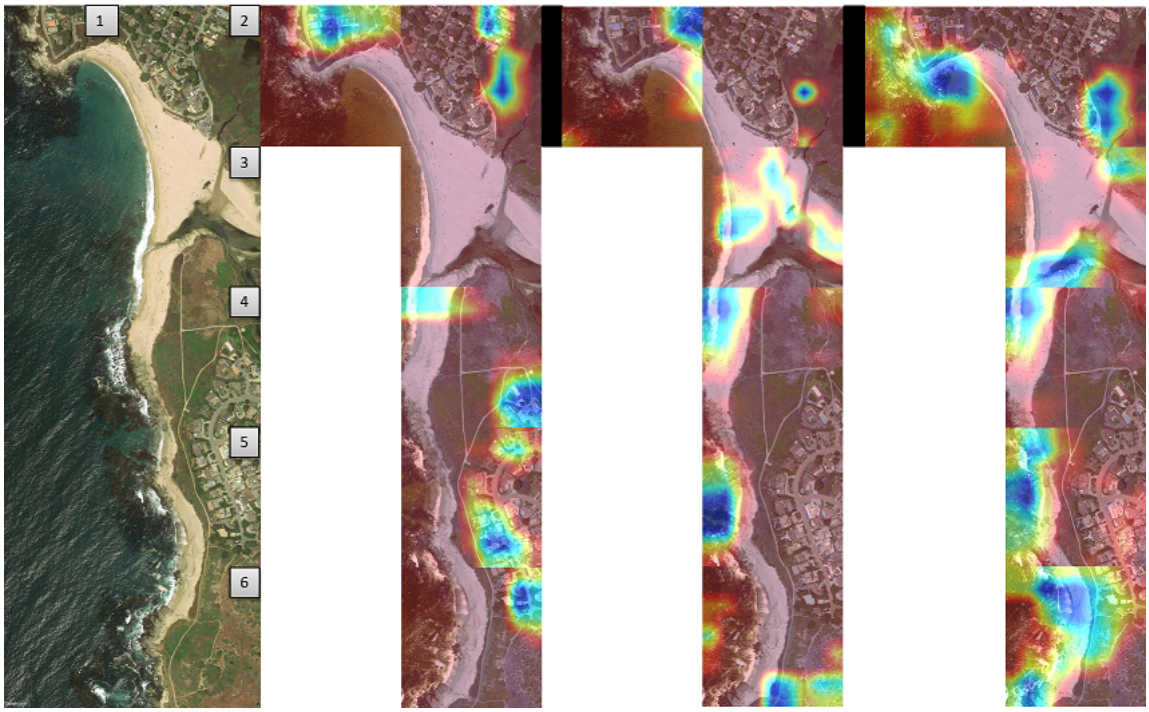

Figure 5.9. DenseNet121 June 2017 Google Earth class likelihood maps. Left image: original satellite image. Second image: likelihood of manmade structures. Third image: likelihood of sandy beach. Fourth image: likelihood of tidal flats.

Table 5.3. June 2017 class likelihood maps table for Figure 5.9. The table shows the prediction vectors for each tile, as labeled in Figure 5.9 (left).

Tile	Coastal Cliffs	Coastal Rocky	Coastal Water Way	Dunes	Man Made Structures	Salt Marshes	Sandy Beaches	Tidal Flats
1	0.00%	0.07%	0.00%	0.00%	2.83%	0.00%	5.21%	91.90%
2	0.00%	0.00%	0.00%	0.00%	0.13%	99.86%	0.00%	0.01%
3	0.00%	0.00%	0.00%	0.02%	0.00%	0.05%	17.10%	82.82%
4	1.65%,	15.84%,	5.06%,	0.06%,	71.65%,	1.20%,	0.60%,	3.94%
5	20.90%	38.01%	0.02%	0.04%	3.88%	0.09%	0.87%	36.18%
6	0.07%	0.41%	0.00%	0.01%	0.04%	0.03%	24.47%	74.97%

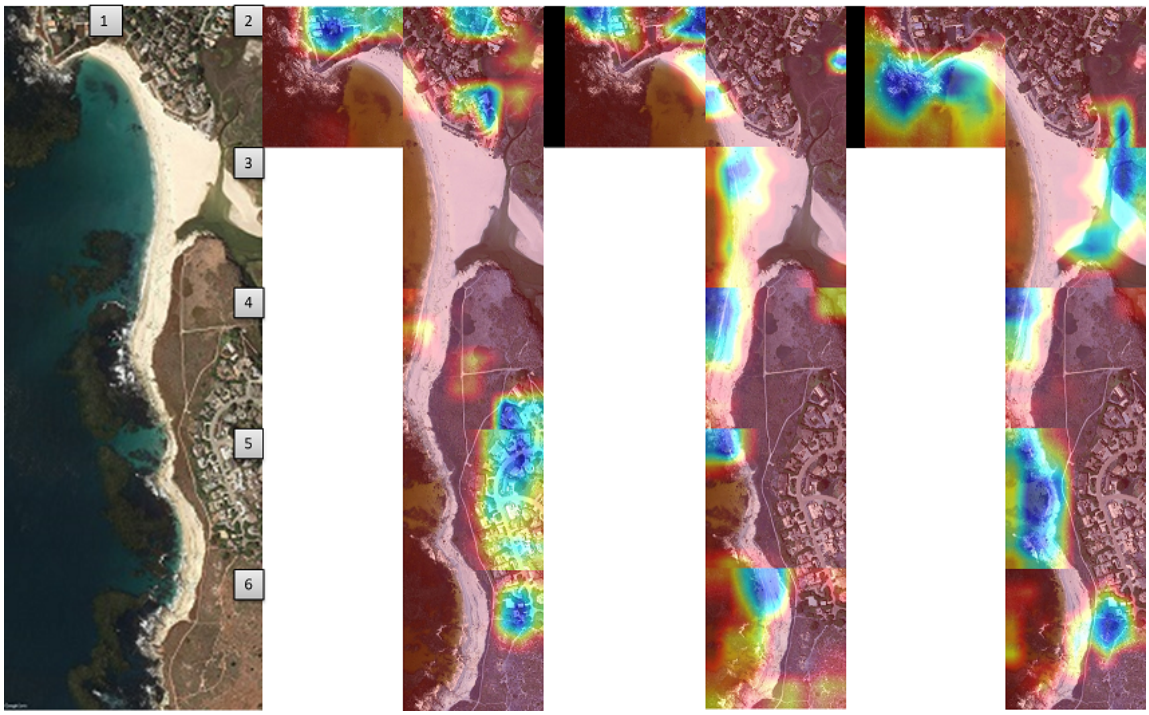

Figure 5.10. DenseNet121 September 2018 Google Earth class likelihood maps. Left image: original satellite image. Second image: likelihood of manmade structures. Third image: likelihood of sandy beach. Fourth image: likelihood of tidal flats.

Table 5.4. September 2018 class likelihood maps table for Figure 5.10. The table shows the prediction vectors for each tile, as labeled in Figure 5.10 (left).

Tile	Coastal Cliffs	Coastal Rocky	Coastal Water Way	Dunes	Man Made Structures	Salt Marshes	Sandy Beaches	Tidal Flats
1	0.00%	0.11%	0.00%	0.00%	4.64%	0.04%	7.63%	87.58%
2	0.00%	0.00%	0.00%	0.02%	47.61%	52.37%	0.00%	0.00%
3	0.02%	0.00%	0.01%	0.19%	0.04%	0.03%	14.20%	85.51%
4	6.86%	72.60%	5.05%	0.05%	12.52%	0.38%	0.09%	2.46%
5	0.13%	0.93%	0.00%	0.74%	52.79%	0.03%	0.22%	45.16%
6	0.43%	2.11%	0.00%	0.58%	0.36%	0.02%	88.25%	8.25%

5.3 Tile-Size Comparison and Test of Homogeneity

Comparing the ground truth (Figure 4.4) to the results for 256x256 disjoint tiling shown in Figure 5.11 indicate several errors in the tidal-flats class. Still, the overall structure of the terrain is recognized in both December and January in that sand is distinguished from marsh. The classifications do not match as well to the ground-truth labels. There are many errors on vegetation (marsh) with a propensity to misclassify it as manmade structures or rocks, perhaps because of distortion of these landscapes during orthorectification.

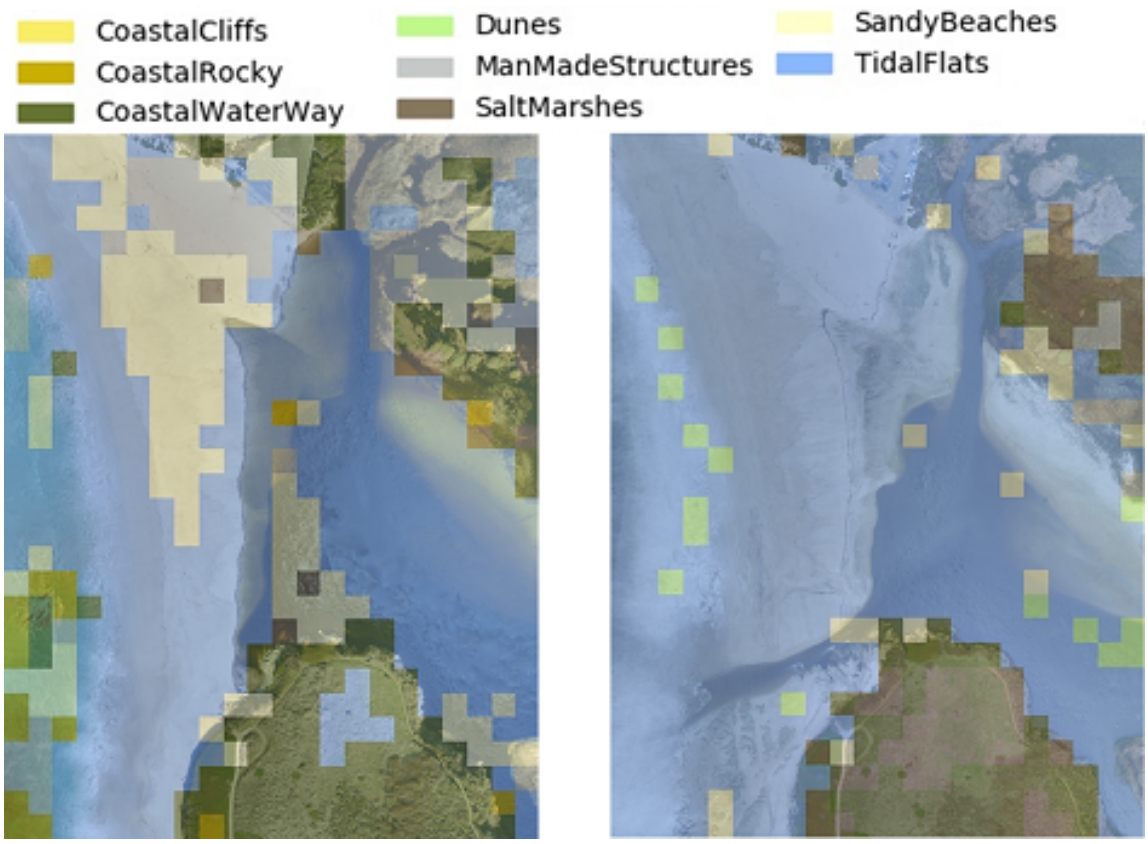

Figure 5.11. VGG19 256x256 disjoint pixel tile graphics for December 6, 2017 and January 10, 2018 overlaid the original red-green-blue orthorectified images. Note that each color in the caption is a single color, but shading is affected by the underlying red-green-blue pixel color.

Figure 5.12 shows example classifications found using 512x512 tiles and sliding-window tiling. Compared to results for the smaller tiles, more neighboring tiles seem to share class labels, though still much misclassification of vegetation as rocks occurs.

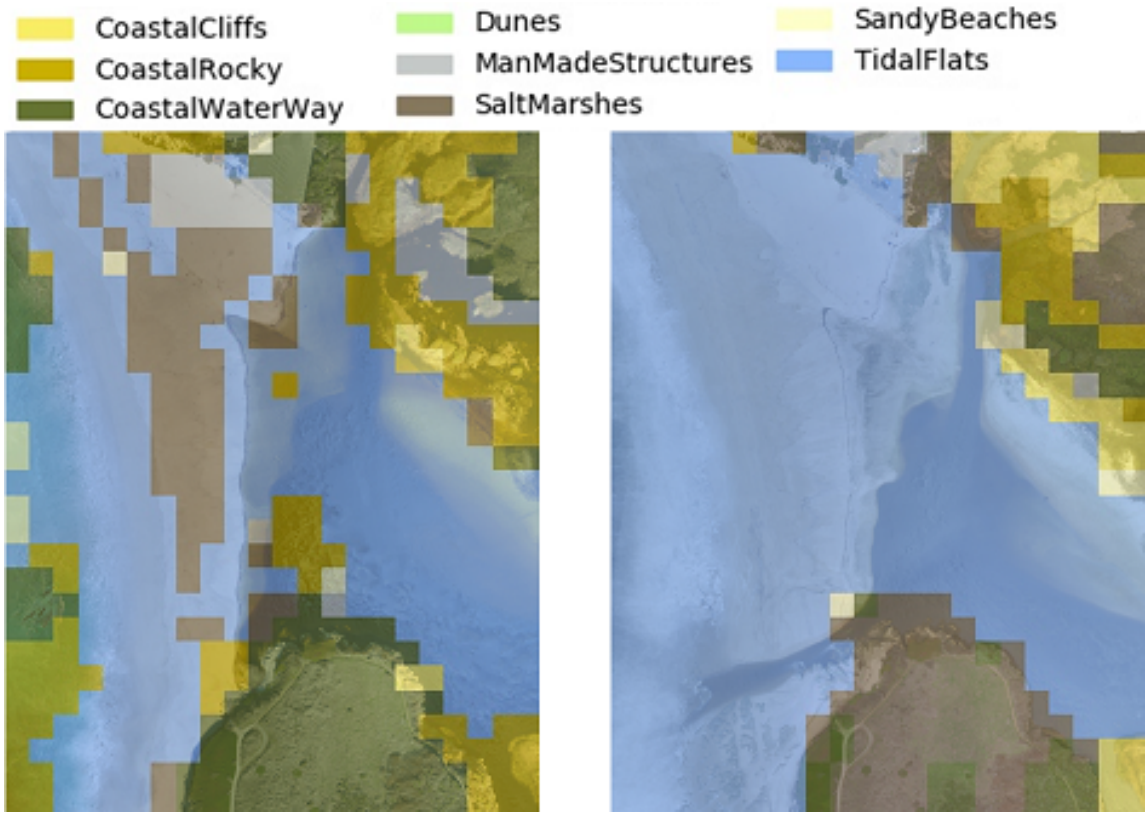

Figure 5.12. VGG19 512x512 sliding window pixel tile graphics from the sliding window program for December 6, 2017 and January 10, 2018.

Figure 5.13 shows example classifications for DenseNet121 299x299 disjoint tiles for the orthorectified images. The classification of the image is mostly sandy beach. It is difficult to say why a neural network favored the sandy beach class in both images. The network might be more sensitive to artifacts that are introduced when orthorectified images are created.

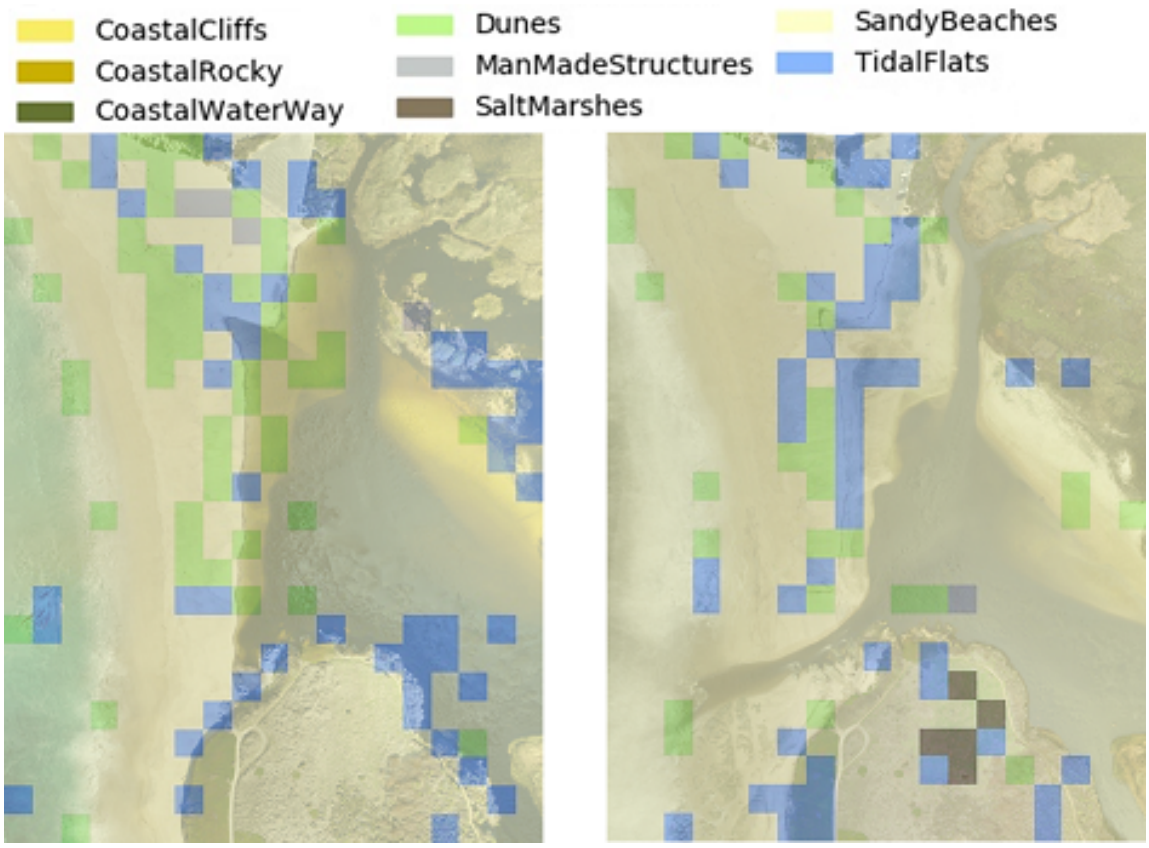

Figure 5.13. DenseNet121 299x299 pixel tile graphics. Left image: December 6, 2017. Right image: January 10, 2018.

Figure 5.14 shows the classifications for the DenseNet121 598x598 disjoint tile size. When we doubled the size of the tiles for DenseNet121, the network began to favor the tidal flats class more. Comparing between tile sizes (299 or 598 pixel size), it is difficult to say which does better because both graphics are far from the ground truth.

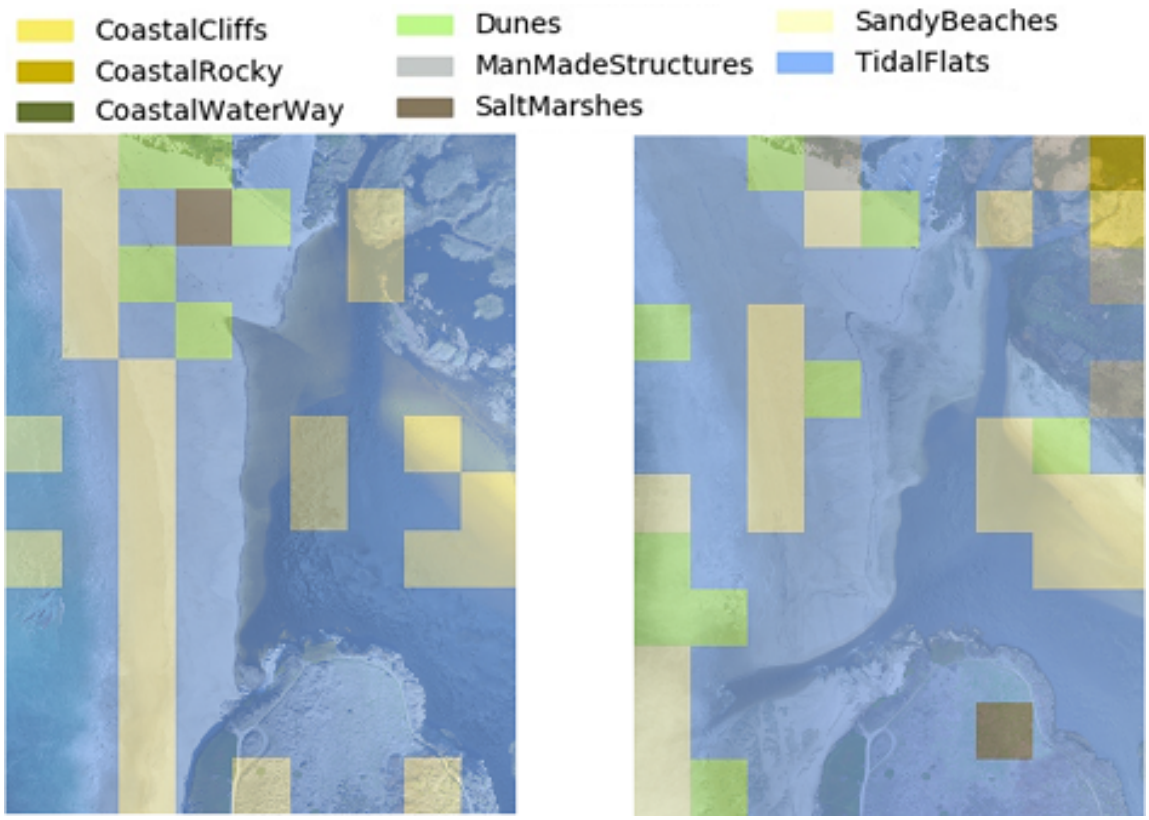

Figure 5.14. DenseNet121 598x598 disjoint pixel tile graphics. Left image: December 6, 2017. Right image: January 10, 2018.

5.4 Change Detection Method

The change detection method only used graphics the VGG19 classifier produced. Figure 5.15 shows changes in 512x512 disjoint tiles between December 6, 2017 and January 10, 2018. White represents no change in the coastal classification and the black represents a change. A transparent binary change image was overlaid on top of the January image. The ground truth was hand labeled for this image. There were 8 true positives, 76 false positives, and 22 false negatives for a precision of 0.095 and a recall of 0.266. This is not very good. We noted that areas with vegetation are commonly predicted as changing, which may be because of difficulty in stitching orthorectified vegetation together.

Figure 5.15. Example of VGG19 change detection with 512x512 tiles overlaid on the January orthorectified image.

5.4.1 Coastal Class Change Detection Color Graphics

Figures 5.16 and 5.17 show a different kind of display, the final inferred coastal class when a change occurred. Figure 5.16 shows it for the 256x256 tiles, and Figure 5.17 shows it for the 512x512 tiles. The color key indicates what the coastal class changed into when it was different from the previous class. The main difference between the two images is in the upper right corner, where Figure 5.16 finds mostly tidal flats and salt marshes, and Figure 5.17 finds mostly coastal cliff and coastal rocky; Figure 5.16 appears more correct.

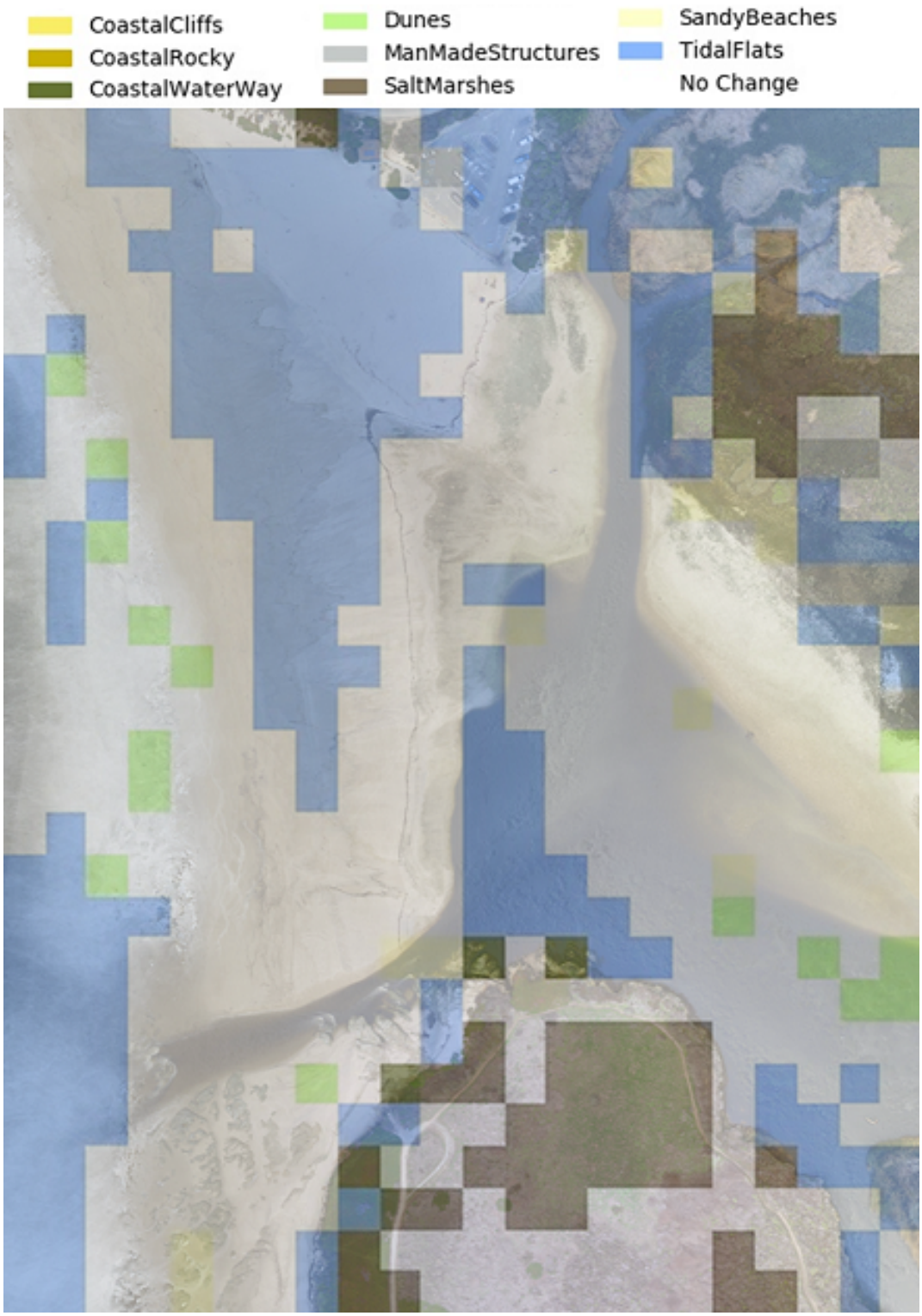

Figure 5.16. January 10, 2018 orthorectified image with 256x256 pixel tile change detection graphic overlay.

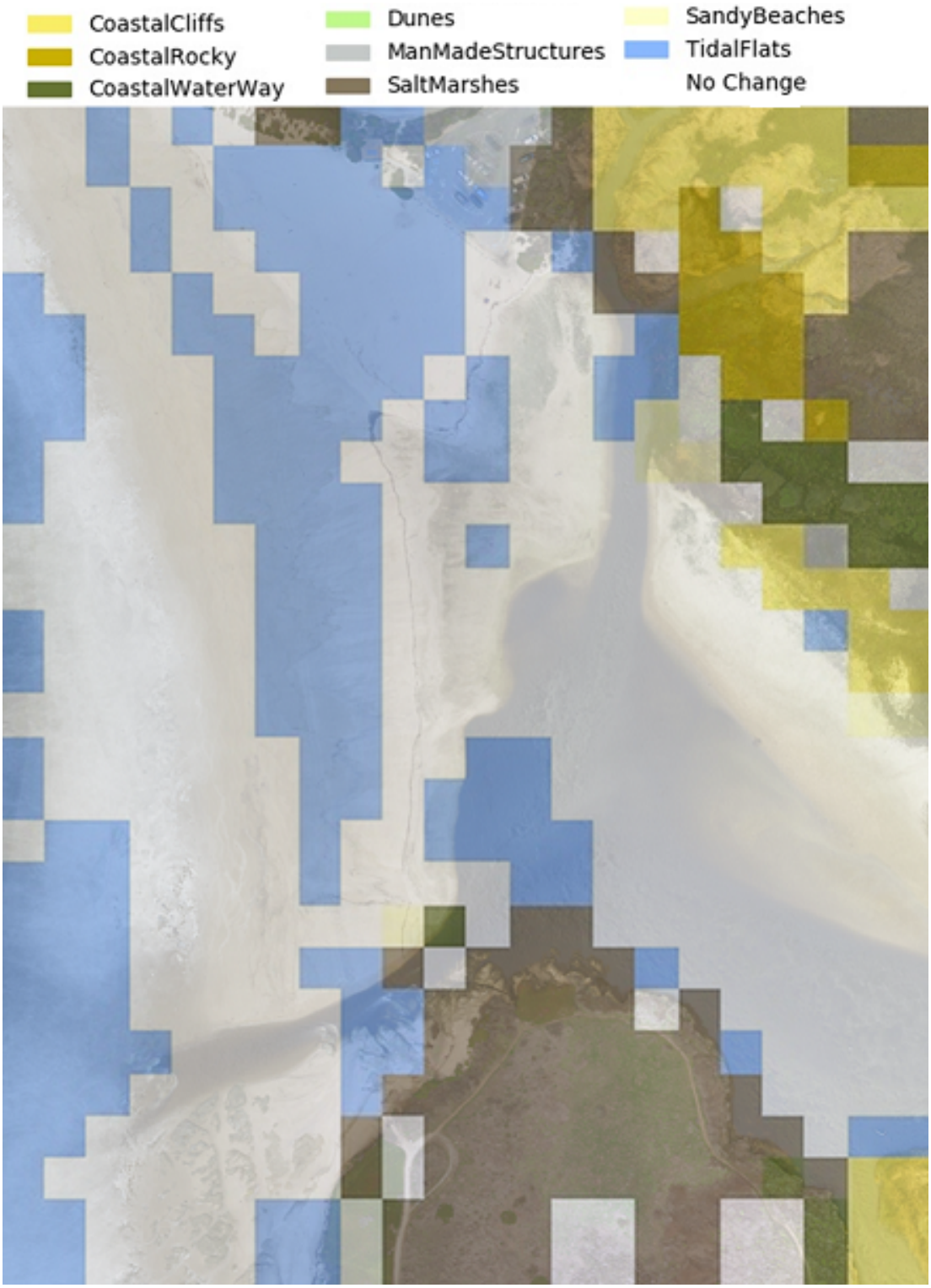

Figure 5.17. January 10, 2018 orthorectified image with 512x512 sliding window pixel tile change detection graphic overlay.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 6:
Conclusions

The main research question of this thesis was whether a coastal-classification change-detection method could sufficiently differentiate coastal classes between two registered images. We trained VGG19 and DenseNet121 neural networks to 93.75% and 95% overall coastal classification accuracy on oblique images. The classes that were the most difficult for both networks were the sandy beach and tidal flats. These two classes are important because one class could mean solid ground for landing forces, and the other could mean a slowdown in operational tempo. For change detection, we compared corresponding tile classifications in images taken at different times of the same terrain, and found they had low precision and recall scores that appeared because of poor coastal classifications. A problem with the orthorectified imagery was generation of artifacts near regions of vegetation which consistently led to misclassification of these regions as rocky or manmade structures.

Future work could include:

- Improve the classification programs used in this thesis. Different sizes of tiles and different sliding-window intervals might improve performance. Better photogrammetry of the orthorectified images could help.
- Try other neural networks in the Keras library.
- Improve network classification performance by generating more training images by flipping images horizontally or vertically or rotating them in 90-degree increments.
- Study the tiles misclassified by the neural networks and try giving them extra weight in training.
- Develop a change-detection method to compare class-likelihood maps between image tiles of different time periods.
- Implement Siamese network architectures.
- Test methods for change detection using a single neural network with input from both corresponding tiles. This could avoid the need to accurately classify tiles when all that is needed is to recognize changes.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

[1] T. Fleming and E. F. Clark, *The Secrets of Inchon: The Untold Story of the Most Daring Covert Mission of the Korean War*. New York, NY: Putnam Adult, 2002.

[2] A. Fujita, K. Sakurada, T. Imaizumi, R. Ito, S. Hikosaka, and R. Nakamura, "Damage detection from aerial images via convolutional neural networks," in *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, Nagoya, Japan, May 2017.

[3] F. Rahman, B. Vasu, J. V. Cor, J. Kerekes, and A. Savakis, "Siamese network with multi-level features for patch-based change detection in satellite imagery," in *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Anaheim, CA, Feb. 2018, pp. 958–962.

[4] P. Kim, *MATLAB Deep Learning: With Machine Learning, Neural Networks and Artificial Intelligence*. Seoul Korea: Apress, 2017.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[6] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Upper Saddle River, NJ: Pearson Education Inc., 2008.

[7] J. Brownlee. (May 12, 2020). How Do Convolutional Layers Work in Deep Learning Neural Networks? [Online]. Available: https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-network

[8] A. Ray. (May 12, 2020). What is meant by feature maps in convolutional neural networks? [Online]. Available: https://www.quora.com/What-is-meant-by-feature-maps-in-convolutional-neural-networks

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv 1512.03385v1*, pp. 1–12, Dec. 2015.

[10] Brian D. Ripley, *Pattern Recognition and Neural Networks*. Cambridge United Kingdom: Cambridge University Press, 2005.

[11] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv 1409.1556*, pp. 1–14, Sep. 2014.

[12] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *arXiv 1608.06993*, pp. 1–9, Aug. 2016.

[13] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[14] M. Brown, G. Hua, and S. Winder, "Discriminative learning of local image descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, pp. 43–57, Jan. 2011.

[15] S. Zagoruyko and N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, June 2015.

[16] M. Arabi and K. Djerriri, "Optical remote sensing change detection through deep Siamese network," in *IEEE International Symposium Geoscience and Remote Sensing IGARSS*, Valencia, CA, July 2018, pp. 5041–5044.

[17] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'06)*, New York, NY, June 2006.

[18] N. C. Rowe and L. L. Grewe, "Change detection for linear features in aerial photographs," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, pp. 1608–1612, July 2001.

[19] S. A. C. Nelson and S. Khorram, *Image Processing and Data Analysis with ERDAS IMAGINE*. Boca Raton, FL: CRC Press, 2019.

[20] W. R. Young, "Sediment transport associated with ephemeral river breaching and closing events," M.S. thesis, Naval Postgraduate School, Monterey, CA, 2018.

[21] J. N. Coughlin, "Morphology changes to Carmel River State Beach in relation to waves and river discharge," M.S. thesis, Naval Postgraduate School, Monterey, CA, 2019.

[22] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: A systematic survey," *IEEE Transactions on Image Processing*, vol. 14, pp. 3367–3371, Mar. 2005.

[23] A. A. Alesheikh, A. Ghorbanali, and N. Nouri, "Coastline change detection using remote sensing," *International Journal of Environmental Science and Technology*, vol. 4, pp. 61–66, Jan. 2007.

[24] M. D. L. Santos, E. H. Shiguemori, R. L. M. Mota, and A. C. B. Ramos, "Change detection in satellite images using self-organizing maps," in *2015 11th International Conference on Information Technology — New Generations*, Las Vegas, NV, Apr. 2015, pp. 662–667.

[25] D. Buscombe and A. C. Ritchie, "Landscape classification with deep neural networks," *Geosciences*, vol. 8, pp. 244–267, June 2018.

[26] D. W. Herrmann, "Morphodynamic classification of coastal regions using deep learning through digital imagery collection," M.S. thesis, Naval Postgraduate School, Monterey, CA, 2018.

[27] K. L. Morgan. Baseline Coastal Oblique Aerial Photographs Collected Ponte Vedra, Florida, to South Carolina/North Carolina Border. [Online]. Available: https: // coastal.er.usgs.gov/data-release/doi-F7N877XX/

[28] Kenneth and G. Adelman. California Coastal Records Project. [Online]. Available: https://www.californiacoastline.org/

[29] *Shoreline Assessment Manual*, Emergency Response Division, Office of Response and Restoration, National Oceanic and Atmospheric Administration, U.S. Dept. of Commerce, Seattle, WA, 2013, 73 pp + appendices.

[30] M. M. Orescanin and J. Scooler, "Observations of episodic breaching and closure at an ephemeral river," *Continental Shelf Research*, vol. 166, pp. 77–82, Aug. 2018.

[31] C. Benedek and T. Szirányi, "Change detection in optical aerial images by a multilayer conditional mixed Markov mode," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, pp. 3416–3430, Oct. 2009.

[32] C. Benedek and T. Szirányi, "A mixed Markov model for change detection in aerial photos with large time differences," in *International Conference on Pattern Recognition (ICPR)*, Tampa, FL, Dec. 2008.

[33] F. Lundh and A. Clark. Pillow 3.0.0. [Online]. Available: https://pillow.readthedocs.io/en/3.0.0/reference/Image.html

[34] Carleton College. What is Google Earth? [Online]. Available: https://serc.carleton.edu/introgeo/google_earth/what.html

[35] "Carmel River State Beach: 36.32 N, 121.55 W," Google Earth Application, March 6, 2020.

[36] P. Du, H. Zhang, and H. Ma, "Classifier refinement for weakly supervised object detection with class-specific activation map," in *2019 IEEE International Conference on Image Processing ICIP*, Taipei, Taiwan, Sep. 2019, pp. 5041–5044.

[37] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *2017 IEEE International Conference on Computer Vision ICCV*, Venice, Italy, Oct. 2017, pp. 336–359.

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California