Machine Learning Engineer Nanodegree

Capstone Proposal by Madhan Merugu

Date: 12/12/2017

Featured Prediction Competition
Statoil/C-CORE Iceberg Classifier Challenge
Ship or iceberg, can you decide from space?
Statoil

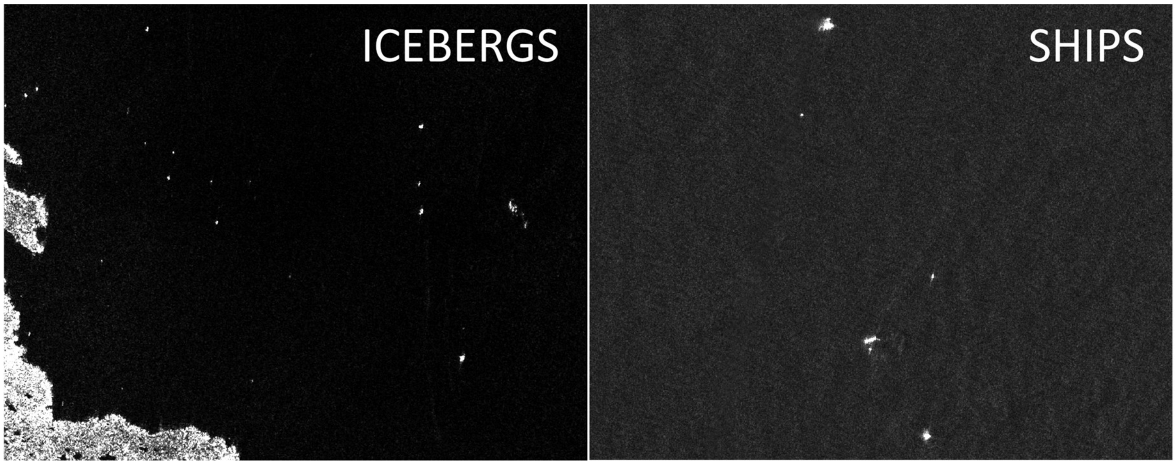
Proposal Summary

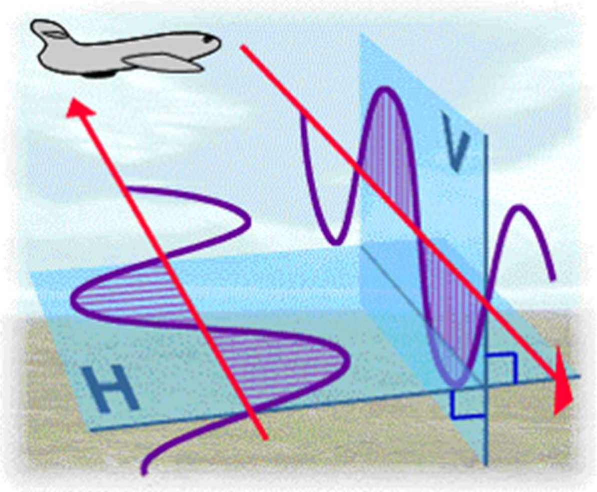
COMPETITION TITLE: Statoil Iceberg Identification Challenge
COMPETITION SPONSOR: C-CORE and Statoil
COMPETITION WEBSITE: https://www.kaggle.com/c/statoil-iceberg-classifier-challenge

Domain Background

The remote sensing systems used to detect icebergs are housed on satellites over 600 kilometers above the Earth. The Sentinel-1 satellite constellation is used to monitor Land and Ocean. Orbiting 14 times a day, the satellite captures images of the Earth's surface at a given location, at a given instant in time. The C-Band radar operates at a frequency that "sees" through darkness, rain, cloud and even fog. Since it emits it's own energy source it can capture images day or night.

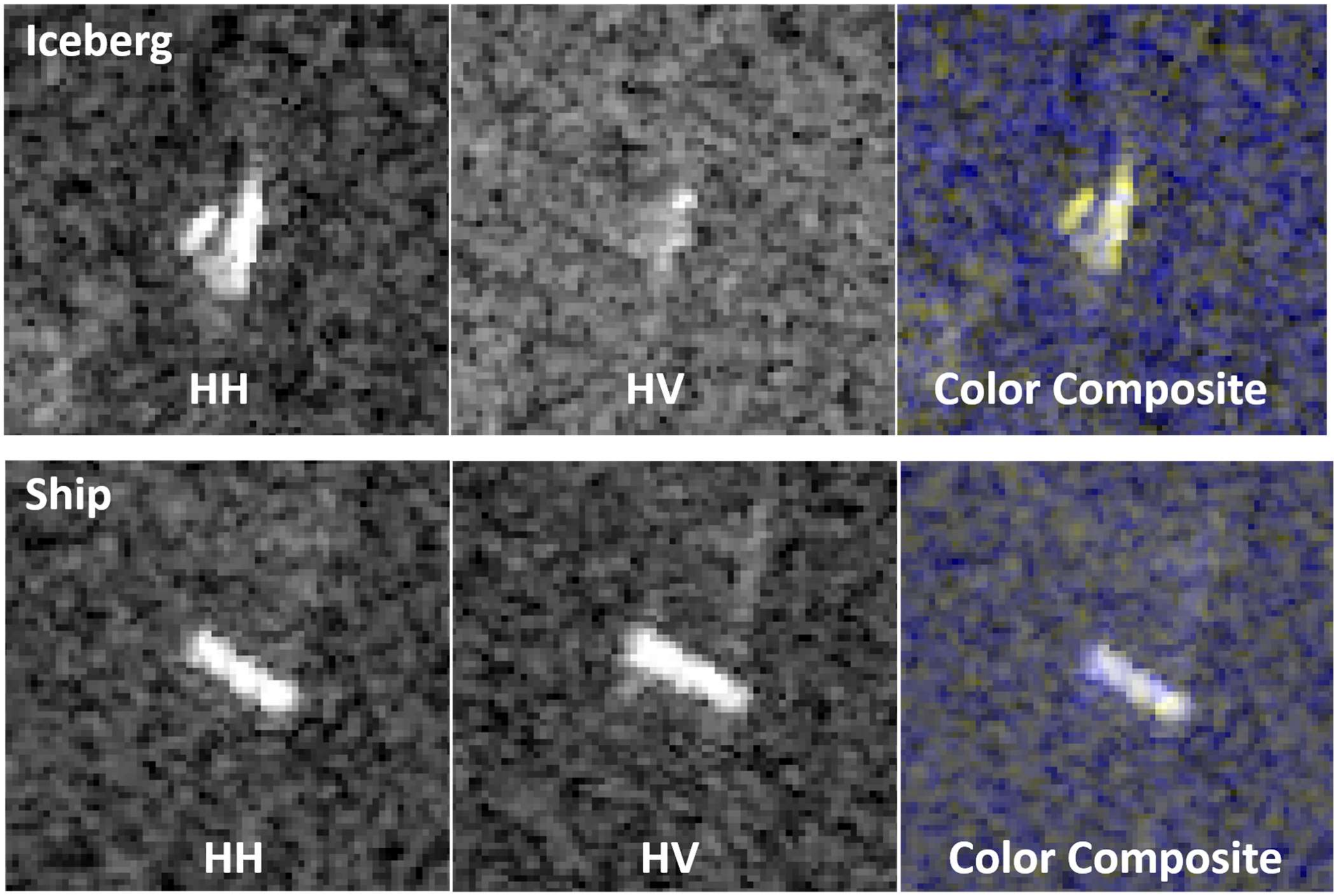
Satellite radar works in much the same way as blips on a ship or aircraft radar. It bounces a signal off an object and records the echo, then that data is translated into an image. An object will appear as a bright spot because it reflects more radar energy than its surroundings, but strong echoes can come from anything solid - land, islands, sea ice, as well as icebergs and ships. The energy reflected back to the radar is referred to as backscatter.



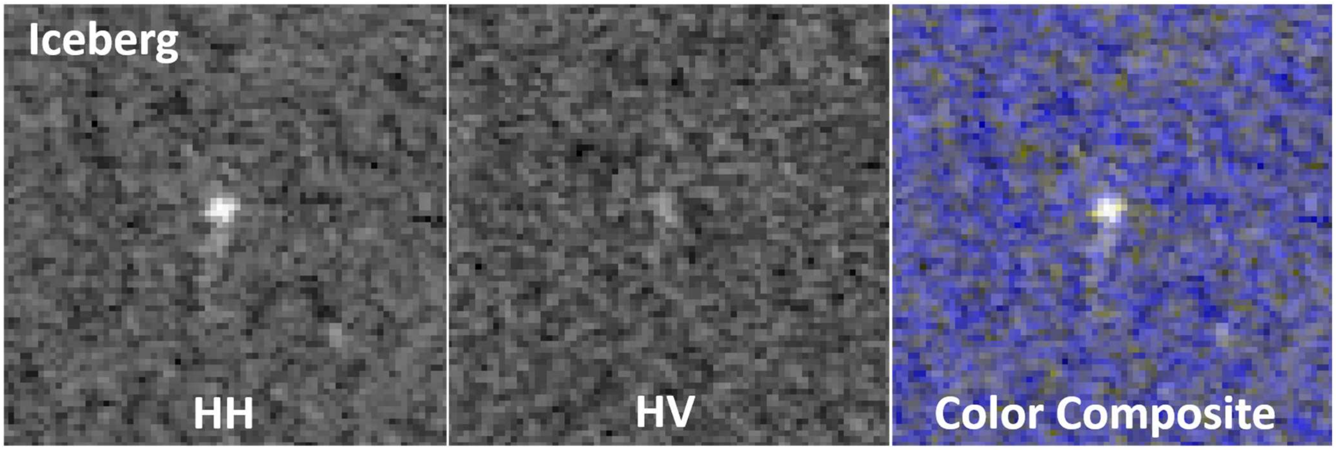


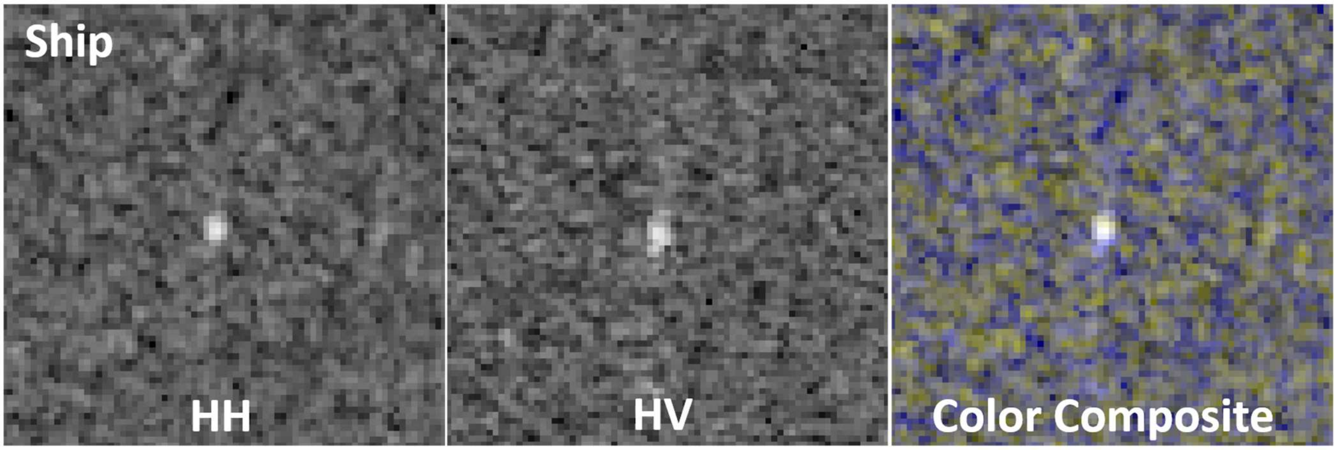
When the radar detects a object, it can't tell an iceberg from a ship or any other solid object. The object needs to be analyzed for certain characteristics - shape, size and brightness - to find that out. The area surrounding the object, in this case ocean, can also be analyzed or modeled. Many things affect the backscatter of the ocean or background area. High winds will generate a brighter background. Conversely, low winds will generate a darker background. The Sentinel-1 satellite is a side looking radar, which means it sees the image area at an angle (incidence angle). Generally, the ocean background will be darker at a higher incidence angle. You also need to consider the radar polarization, which is how the radar transmits and receives the energy. More advanced radars like Sentinel-1, can transmit and receive in the horizontal and vertical plane. Using this, you can get what is called a dual-polarization image.

For this contest you will see data with two channels: HH (transmit/receive horizontally) and HV (transmit horizontally and receive vertically). This can play an important role in the object characteristics, since objects tend to reflect energy differently. Easy classification examples are see below. These objects can be visually classified. But in an image with hundreds of objects, this is very time consuming.



Here we see challenging objects to classify. We have given you the answer, but can you automate the answer to the question .... Is it a Ship or is it an Iceberg?





Problem Statement

Drifting icebergs present threats to navigation and activities in areas such as offshore of the East Coast of Canada.

Currently, many institutions and companies use aerial reconnaissance and shore-based support to monitor environmental conditions and assess risks from icebergs. However, in remote areas with particularly harsh weather, these methods are not feasible, and the only viable monitoring option is via satellite.

Statoil, an international energy company operating worldwide, has worked closely with companies like C-CORE. C-CORE have been using satellite data for over 30 years and have built a computer vision based surveillance system. To keep operations safe and efficient, Statoil is interested in getting a fresh new perspective on how to use machine learning to more accurately detect and discriminate against threatening icebergs as early as possible.

In this competition, the challenge is to build an algorithm that automatically identifies if a remotely sensed target is a ship or iceberg. Improvements made will help drive the costs down for maintaining safe working conditions.

Datasets and Inputs

train.json, test.json

The data (`train.json`, `test.json`) is presented in `json` format. The files consist of a list of images, and for each image, you can find the following fields:

- **id** - the id of the image
- **band_1, band_2** - the flattened image data. Each band has 75x75 pixel values in the list, so the list has 5625 elements. Note that these values are not the normal non-negative integers in image files since they have physical meanings - these are float numbers with unit being dB. Band 1 and Band 2 are signals characterized by radar backscatter produced from different polarizations at a particular incidence angle. The polarizations correspond to HH (transmit/receive horizontally) and HV (transmit horizontally and receive vertically). More background on the satellite imagery can be found here.
- **inc_angle** - the incidence angle of which the image was taken. Note that this field has missing data marked as "na", and those images with "na" incidence angles are all in the training data to prevent leakage.

- **is_iceberg** - the target variable, set to 1 if it is an iceberg, and 0 if it is a ship. This field only exists in `train.json`.

https://www.kaggle.com/c/statoil-iceberg-classifier-challenge/download/sample_submission.csv.7z

https://www.kaggle.com/c/statoil-iceberg-classifier-challenge/download/test.json.7z

https://www.kaggle.com/c/statoil-iceberg-classifier-challenge/download/train.json.7z

sample_submission.csv

The submission file in the correct format:

- id - the id of the image
- is_iceberg - your predicted probability that this image is iceberg.

Solution Statement

A deep learning algorithm will be developed using Google Tensorflow/Keras libraries and will be trained with the given training data.

Specifically, CNN (Convolutional Neural Network) will be implemented using Tensorflow/Keras and will be optimized to minimize multi-class logarithmic loss as defined in the Evaluation Metrics section. Predictions will be made on the test data set and will be evaluated.

I am also thinking of using “Tranfer Learning” techniques like using VGG16, Resnet etc., to better make use of pre-trained models for faster convergence and

Benchmark Model

The model with the Public Leaderboard current top score of multi-class logarithmic loss value 0.1029 of will be used as a benchmark model.

Attempt will be made so that score (multi-class logarithmic loss) obtained will be among the top 50% of the Public Leaderboard submissions.

Evaluation Metrics

Submissions are evaluated on the log loss between the predicted values and the ground truth.

This is the multi-class version of the Logarithmic Loss metric. Each observation is in one class and for each observation, you submit a predicted probability for each class. The metric is negative the log likelihood of the model that says each test observation is chosen independently from a distribution that places the submitted probability mass on the corresponding class, for each observation.

$$logloss = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{i,j}\log(p_{i,j})$$

where N is the number of observations, M is the number of class labels, loglog is the natural logarithm, yi,j is 1 if observation ii is in class jj and 0 otherwise, and pi,j is the predicted probability that observation ii is in class jj.

Both the solution file and the submission file are CSV's where each row corresponds to one observation, and each column corresponds to a class. The solution has 1's and 0's (exactly one "1" in each row), while the submission consists of predicted probabilities.

The submitted probabilities need not sum to 1, because they will be rescaled (each is divided by the sum) so that they do before evaluation.

(Note: the actual submitted predicted probabilities are replaced with max(min(p,1−10−15),10−15)max(min(p,1−10−15),10−15).)

Submission File

For each id in the test set, you must predict the probability that the image contains an iceberg (a number between 0 and 1). The file should contain a header and have the following format:

```
id,is_iceberg
809385f7,0.5
7535f0cd,0.4
3aa99a38,0.9
etc.
```

Project Design

From the description and problem statement it can be inferred that computer vision can be used to arrive at a solution. CNN class of deep learning algorithm can be employed for this problem.

Initially data exploration will be carried out to understand possible labels, range of values for the image data and order of labels. This will help preprocess the data and can end up with better predictions.

After this necessary preprocess functions will be implemented, data will be randomized and CNN will be implemented in Tensorflow/Keras.

Finally, necessary predictions on the test data will be carried out and these will be evaluated.