Machine Learning Engineer Nanodegree

Capstone Proposal by Madhan Merugu

Date: 12/14/2017

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

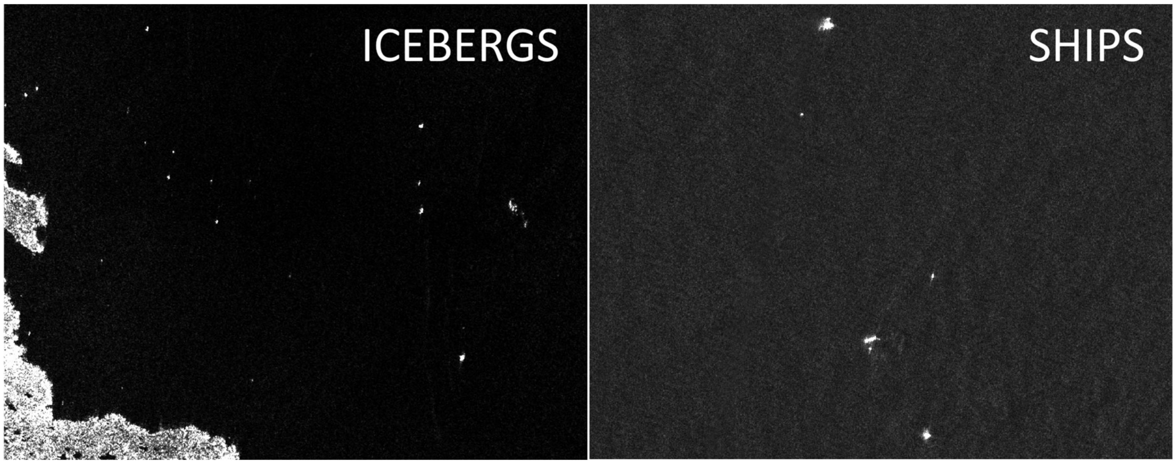
Proposal Summary

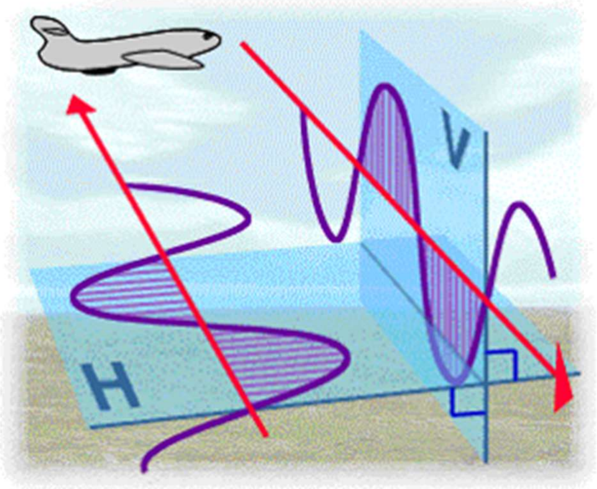
Domain Background

There are several satellites in the sky, which usually orbit around at several kilometers high, have a capability for “remote sensing” objects on earth. They can capture object images regularly at any given place or location. They are equipped with radars that can even “see through” any weather conditions like “rain, cloud or fog” as these radars can emit own energy to receive images in any weather conditions.

Figure Reference: https://www.thedailybeast.com/now-you-can-hunt-for-malaysia-airlines-flight-370 (GETTY)

These satellite radars use much like a “SONAR” technology which can send and detect signals bounced off an object and then analyze these signals to identify to translate to an image. Inside the translated image, bright spots are usually the objects representation because objects can reflect more energy back when compared to the surrounding environment. However, the main challenge that scientists face is classification of these objects. These bright spots can be anything. When scanning the big oceans or seas, these bright objects can be small islands, icebergs, big ships, small boats etc.





Reference: https://www.kaggle.com/c/statoil-iceberg-classifier-challenge

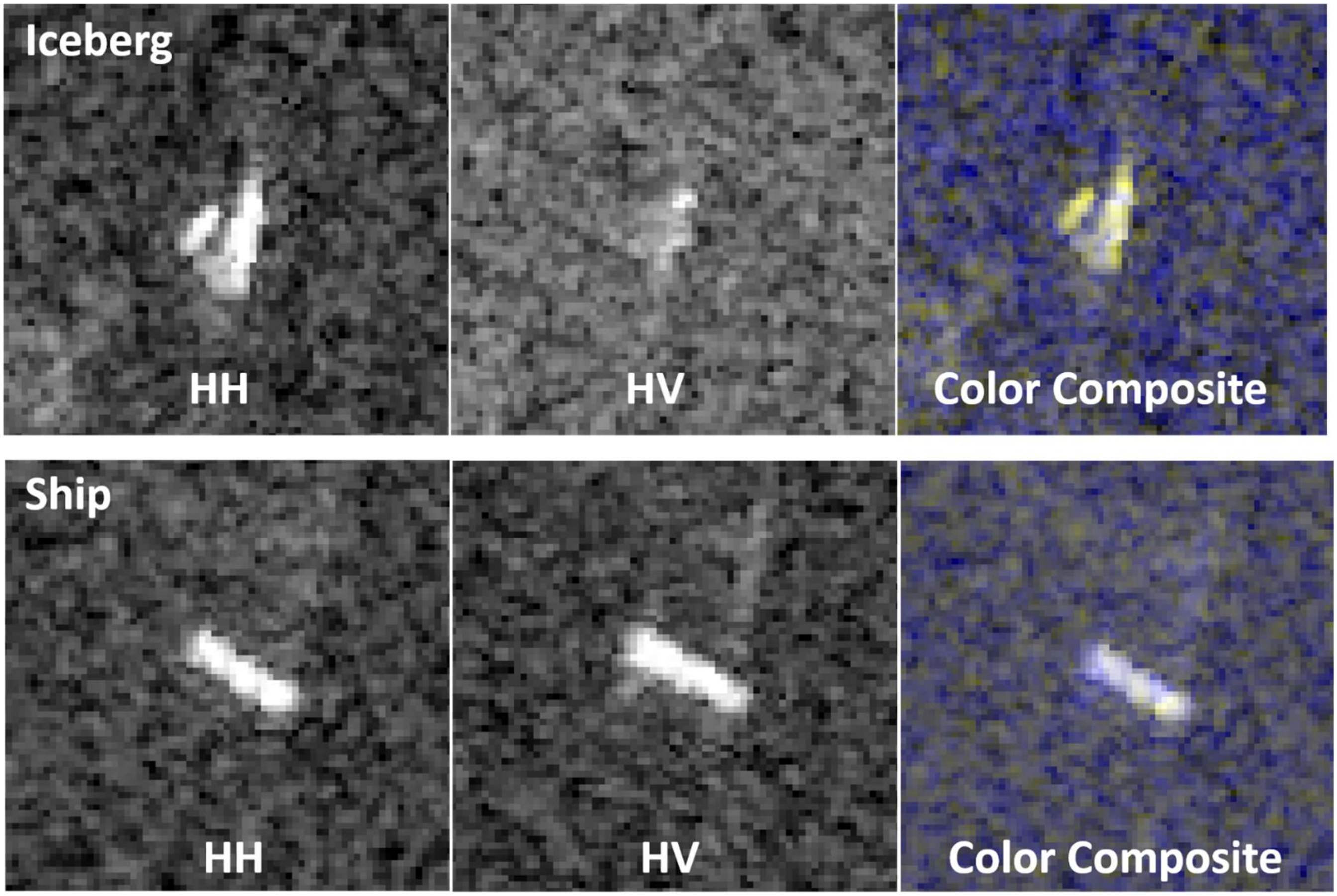
Having our radar just detecting these objects in ocean won’t be of much help unless we analyze the background environment surrounding these objects. In our case, we need to analyze the backscatter of the ocean or surrounding area of the object in order to identify the object type. But again, another challenge that we face is that this backscatter can be inconsistent and can be affected by weather conditions like high winds produce brighter background while low winds generate a darker kind of background of the image.

There are other factors to consider that affect the image background like the radar incident angle and radar polarization like how radars send and receive the energy back from the object. Some satellite radars can send and receive images back in both horizontal and vertical planes (dual-polarization).

In our data domain to analyze, we have data with two channels to consider:

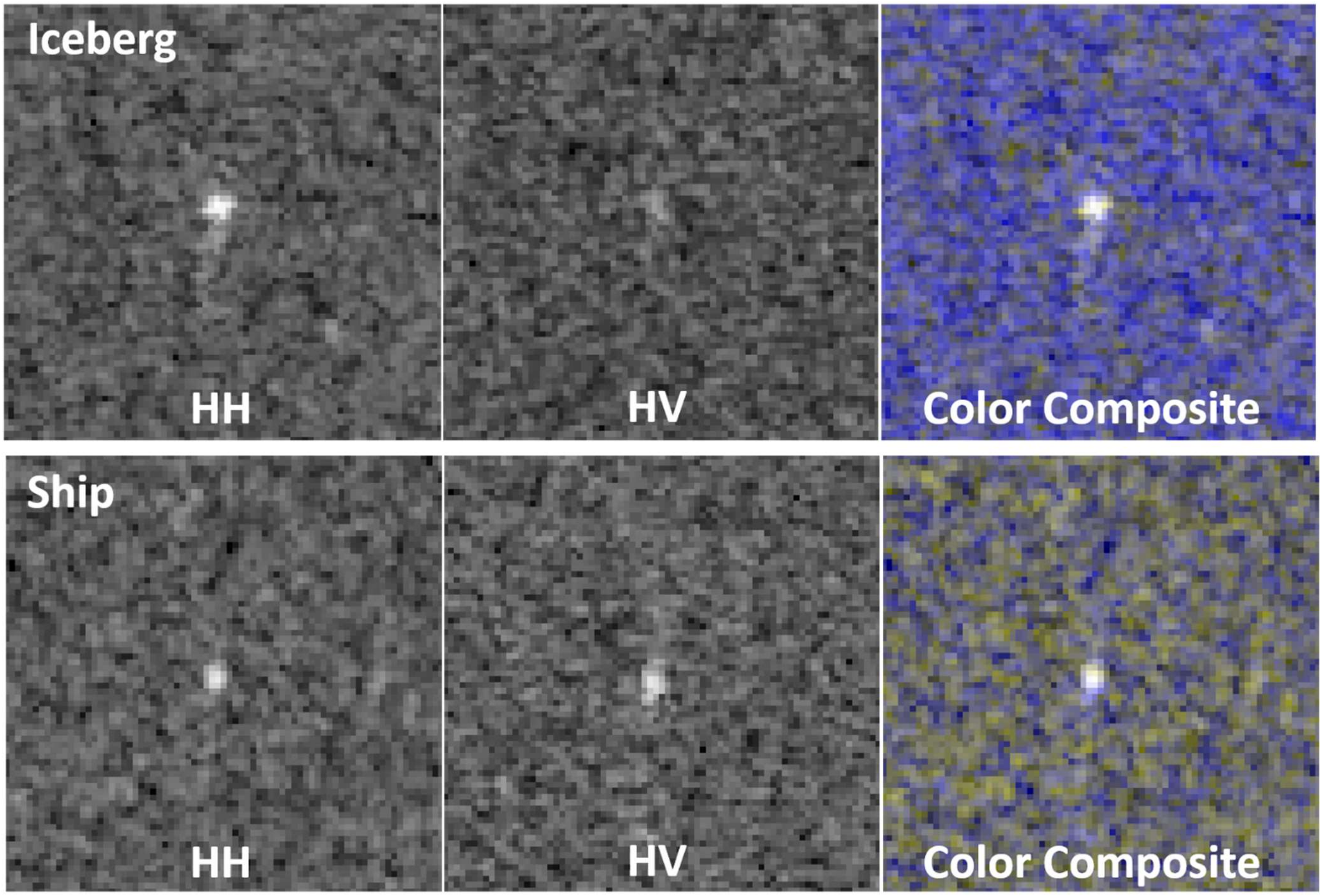
- HH (transmit/receive horizontally)
- HV (transmit horizontally and receive vertically).

As you can see from the below examples (courtesy: C-CORE and Statoil), objects like Iceberg and Ships look entirely different in these 2 different channels (HH & HV) due to image reflections are differences in these channels. Our human brains are trained to classify some of these images but it is going to impractical when we face millions of such images to process and classify in a limited time.



Reference: https://www.kaggle.com/c/statoil-iceberg-classifier-challenge

Below are few more challenging objects to classify. Is seems so difficult to automate the process of identifying whether it is a Ship or is it an Iceberg.



Reference: https://www.kaggle.com/c/statoil-iceberg-classifier-challenge

Problem Statement

There were many incidents in the past that drifting icebergs posed many challenges and even threats or catastrophic damages (major being Titanic syncing) especially near polar regions like East Coast of Canada.

Iceberg Picture - Reference: http://www.cbc.ca/radio/thisisthat/mp-forgets-names-iceberg-150-street-art-trashed-cross-country-skiing-1.3865163/400-ton-iceberg-to-be-sent-on-cross-country-tour-to-celebrate-canada-150-1.3865168

Some companies use aerial reconnaissance and other navigational tools to monitor the icebergs to prevent any potential dangers to ships and other vessels. But, this process proves to be difficult to implement in remote places where harsh and unformidable conditions prevail. In such cases, monitoring via satellites proves to be an effective solution.

I think this is where we can leverage emerging “Machine Learning” technologies including deep learning capabilities to address such issues with an objective to help the mankind by accurately detecting and differentiating these life-threatening icebergs as soon as possible.

In this project, our challenge is to build machine learning algorithms and pipelines that can automatically identify and classify ships & icebergs by processing satellite radar images of such objects. This is will not only help in cutting operational costs but also ensures the safe working environment for many.

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.

References:

This project has been part of ongoing **Kaggle competition** and below are the references:

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

http://www.cbc.ca/radio/thisisthat/mp-forgets-names-iceberg-150-street-art-trashed-cross-country-skiing-1.3865163/400-ton-iceberg-to-be-sent-on-cross-country-tour-to-celebrate-canada-150-1.3865168