Stanford University

CS 229

Machine Learning

Iceberg-Ship classifier using SAR Image Maps

Authors:
Li Jun, Parulekar Atharva, Samant Dhruv

January 14, 2018

Contents

1. Introduction

Drifting icebergs present a threat to navigation and activities in areas such as offshore of the East Coast of Canada, thus the Iceberg and ship identification in satellite synthetic aperture radar (SAR) data has been an essential part of offering an operational iceberg surveillance program. Because coarser resolution satellite SAR data is at times not as intuitive as satellite optical data for manual human interpreted target classification, this process can be labor intensive, subjective, and error prone. Therefore, it is desired that an automated method for iceberg or ship identification be implemented [1]. In this project, we build an algorithm that automatically identifies whether a remotely sensed target is a ship or iceberg. This project is inspired by the kaggle challenge posed by Startoil [14], an international energy company who have acquired satellite images by working closely with companies like C-CORE. When the radar detects an object, it needs to be analyzed for specific characteristics - shape, size and brightness to classify it as an iceberg, or a ship or any other solid object. The Sentinel-1 satellite constellation, with remote sensing systems at over 600 km above the earth, transmits and receives energy in the horizontal and vertical plane, generating a dual-polarization image. In this project, we use raw images with two channels HH (transmit/receive horizontally) and HV (transmit horizontally and receive vertically) and try to output a probability of the radar image being an iceberg. We use a variety of algorithms namely Support Vector Machines, CNNs, ResNets and feature extraction techniques like Histogram of Oriented Gradients to achieve this goal.

2. Related Work

Previous works on Synthetic aperture radar (SAR) images, include using multivariate approaches, support vector machines (SVMs), and Convoluted Neural Networks (CNNs) for classification. A two-class maximum likelihood model can classify ships with an accuracy of 93% [7], by maximizing the posteriori probability obtained from bayes rule. A study conducted by the German Aerospace Centre show that CNNs outperform SVMs, by achieving an F1 score of 97%.

Spatial FCM clustering method was used to detect coastlines in FCM images [11]. Although our data set is smaller, the idea can be borrowed to detect the edge of the object in the center of the image. A filter can be applied to further remove the background noise. Moreover, shape properties can be exploited to build the model. The combination of HV/HH bands, can be used for feature extraction as shown by [11]. This conforms with our theory that ships are more likely to generate reflections on the edges in the HH bands.

Convoluted Neural Networks are leading the way in image classification problems. The AlexNet algorithm [10] , won the 2012 ILSVRC (ImageNet Large-Scale Visual Recognition Challenge), and the publication is regarded as one of the most influential publications in deep learning. In this project, we experiment with different CNN architectures, particularly residual networks [8]. SAR images are associated with speckle noise, that affect their quality. Spatial filters reduce the speckle noise at the expense of image details and edges. Various methods have been developed to suppress speckle including multi-look processing [12], filtering methods [5],wavelet-based despecking methods [15], block-matching 3D (BM3D) algorithm [3] and Total Variation (TV) methods [2]. We augment the data, use despeckling algorithms to reduce the speckle noise, and compare different optimizers to classify SAR images using a 50 layer Residual Network.

3. Dataset and Features

Data Visualization

We make a collage of the data with corresponding band maps shown side by side. We first zero centre the data by translating it such that the lowest element in the array is 0. We now extract some critical statistics and plot histograms using the arrays.

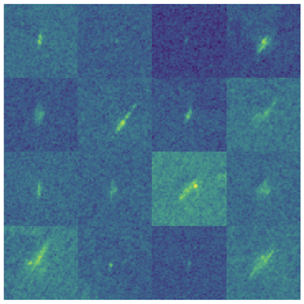


Figure 1: Iceberg HH

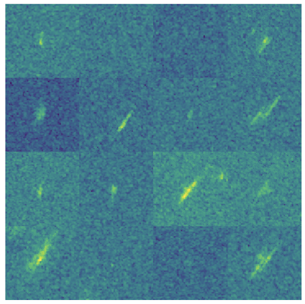


Figure 2: Iceberg HV

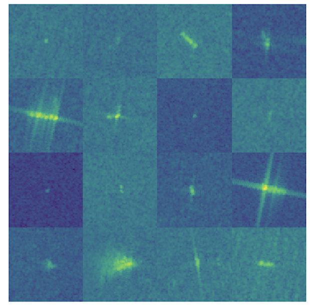


Figure 3: Ships HH

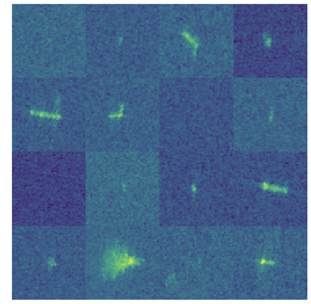
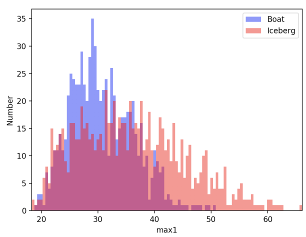
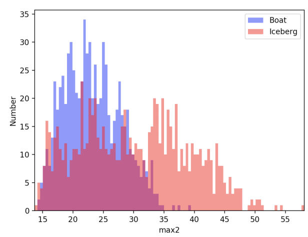


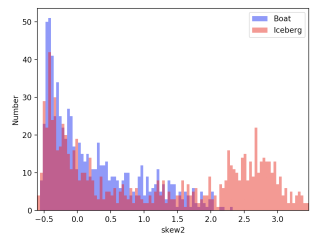
Figure 4: Ships HV

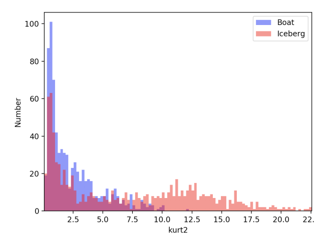
The following statistics have been used to compute histograms to visualize how the data separation occurs.

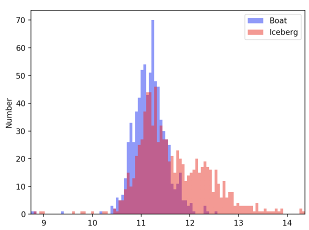
- Maximum activation element in band HH and band HV
- Skewedness and Kurtosis of elements of HV
- Mean, Standard deviation, Kurtosis and Skewdness of the HOG features in HV

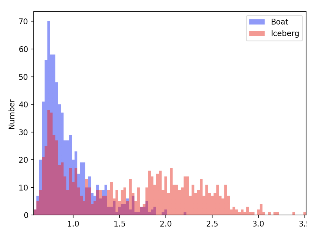


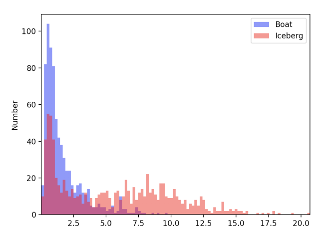


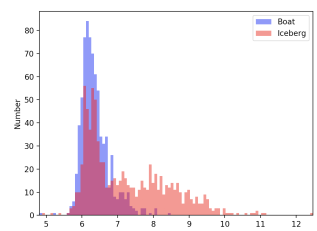












Data Augmentation

We perform rotations in multiples of 30 degrees on each image resulting in 12 images. We then flip the images leading to 24 augmented images from a single image. This increases the amount of data that we can train our deep models on to 38496 training images. Before augmenting however we split the data into training set and validation set in a 7.5:2.5 split respectively.

4. Methods

Reducing Speckle Noise

SAR images are contaminated by multiplicative noise called speckle that makes their processing and interpretation difficult for computer vision systems as well as human interpreters[6]. The SAR image speckle is fully developed and can be represented as a multiplicative noise model as follows:

$$I(t) = R(t) \times v(t) \tag{1}$$

where $I(t)$ is the noise affected signal, $R(t)$ represents the radar back-scatter property, and v(t) is the speckle noise. We explore the following despeckling techniques:
Lee Filter Model: We apply a spatial filter to pixels, which replaces the center pixel value with the value calculated using local statistics of neighboring pixels in a square window. The despeckled value is given by:

$$P_C = L_M + K\dot{(}P_C - L_M) \tag{2}$$

$$K = L_V/(L_V + AV)$$

L_M = local mean, L_V = local var, AV = additive noise var, and P_C = central pixel

Since Lee filter assumes that speckle noise is uniform over the image, it causes blurring of the edges and image details. Thus, to account for the heterogeneous parts of the image, we use the enhanced Lee filter proposed by Lopes. This is achieved by dividing the image into 3 regions and processing them independently. In the homogeneous region, the centre pixel value is replaced by the mean window value. In the heterogeneous regions, where the speckle noise has developed completely, the pixel value is determined by an adaptive filtering algorithm given below.

$$P_C = L_M\dot{K} + (1-K)\dot{(}P_C - L_M) \tag{3}$$

$$K = e^{-D\dot{(}C_i - C_u)/(C_max - X_u)}$$

L_M = local mean, C_i = image variation coefficient, C_u = noise variation coefficient additive noise var, C_{max} = Maximum noise variation coefficient, D = damping coefficient, and P_C = central pixel

Support Vector machines with HOG

We used weighted histogram of gradients to count occurrences of gradient orientation in center and edge areas. After some trial and error, we decide to divide the original image into 5 blocks, and for each block we calculate the gradient angles and magnitudes at each pixel, then find the histogram of the gradient angles with 16 bins using the magnitudes as weights. Thus each image gives us (16 bins)$\times$ (5 blocks) = (a feature vector of length 80). For the SVM model, we used regularization = 100 with a radial basis function as the loss function.

$$f = \arg\min_{f\in\mathcal{H}} \left\{ \frac{1}{\lambda n} \sum_{i=1}^{n} (1 - yf(x))_+ + \frac{1}{2}||f||^2_{\mathcal{H}} \right\} \tag{4}$$

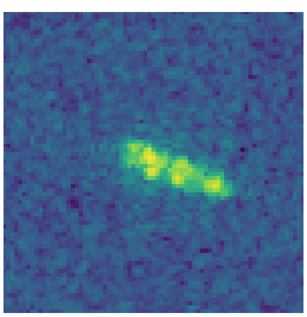


Figure 5: Original

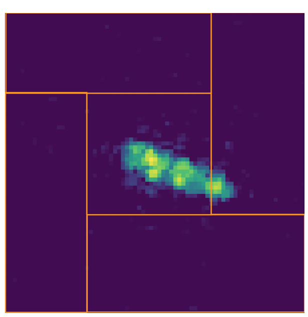


Figure 6: Processed

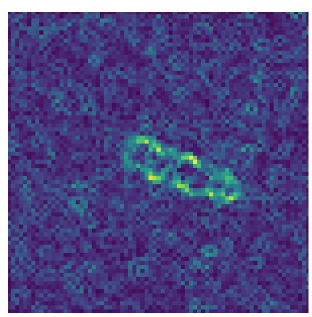


Figure 7: g - Magnitude

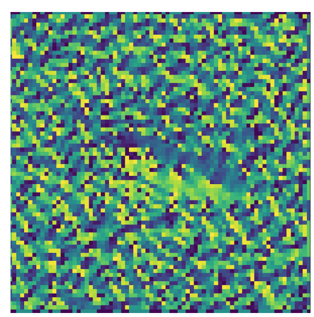


Figure 8: θ - Angle

Convolution Neural Nets

We perform an experiment by running the following deep model on the data. We use a 0.2 dropout layer and a 2×2 maxpool layer after every convolutional layer. The dropout layer is to reduce over-fitting. The neural network has a fully connected layer at the end.

Layer (type)	Filter / Neurons	Size (Depth)	Parameters	Output
2D Convolution	$3 \times 3 \times 3$	64	1,792	$73 \times 73 \times 64$
2D Convolution	$3\times 3 \times 64$	128	73,856	$34 \times 34 \times 128$
2D Convolution	$3\times3 \times 128$	128	1,47,584	$15 \times 15 \times 128$
2D Convolution	$3\times3 \times 128$	64	73,792	$5\times5 \times 64$
Flatten	256	0	0	256
Dense	512	0	1,31,584	512
Dense	256	0	1,31,328	256
Dense (Sigmoid)	1	0	257	1
Total			560,193	

We then train the augmented and denoised data on a 50 layer Residual Network (ResNet). ResNets implement residual learning in the form of skip connections that allow the gradient to be directly propagated to the earlier layers. Very deep neural networks suffer from vanishing gradients (i.e. gradient descent is extremely slow). The identity block of the ResNets provide an identity mapping, and hence the deeper model has a training error no more than its shallower counterpart [8]. This coupled with the ease of learning of identity blocks account for the remarkable performance of ResNets over conventional CNNs of the same depth and number of parameters. The output of a layer with residual learning is:

$$y = F(x, W_i) + x \tag{5}$$

where y and x are the layer outputs and inputs respectively, and $F(x, W_i)$ represents the residual mapping to be learned.

5. Results and Discussion

We tabulate the results of the models we have tried and also plot the necessary evaluation metrics (precision,recall,score) for the denoised SVM model as well as the Resnet model with noise.

Model	Data	Noise	Train	Val
SVM	1604	Yes	98.35	67.19
SVM	1604	No	77.20	69.52
CNN	1604	Yes	89.45	88.33
CNN	1604	No	84.62	88.28
ResNet	38496	Yes	99.44	96.63
ResNet	38496	No	97.42	94.84

Model	Class	Prec	Recall	Score
SVM	1	72.21	63.81	67.75
SVM	2	67.58	75.44	71.29
SVM	T	69.89	69.62	69.52
ResNet	1	97	95	96
ResNet	2	94	96	95
ResNet	T	95	95	95

For our CNN and Resnet models, we compared 2 optimizers- RMSprop [13] and ADAM [9]. RMS prop, for every iteration divides the learning rate by average momentum of squared gradients and a constant to prevent diminishing gradients. ADAM reduces the noise fluctuation of RMSprop, by replacing the gradients by the gradients average. ADAM gives us a steadier result after 10 epochs. It is an algorithm for first-order gradient-based optimization of stochastic objective functions, based on adaptive estimates of lower-order moments. The hyper parameters used for this optimizer consist of a learning rate of 0.001, β_1 equalling 0.9, β_2 equalling 0.999, an ϵ of e^{-08} and a decay rate of zero. We use batch gradient descent with a batch size of 24 and a train to test split of 0.75.
We infer the following from this exercise:

- Denoising significantly improves the SVM performance from 67% to 70%.
- ResNet performs better than baseline and CNN.
- The performance of CNNs drop as we denoise the data using Lee filter. Lee filter suppresses speckle noise well in the homogeneous regions. In heterogeneous regions of the image, some important image features are possibly lost.

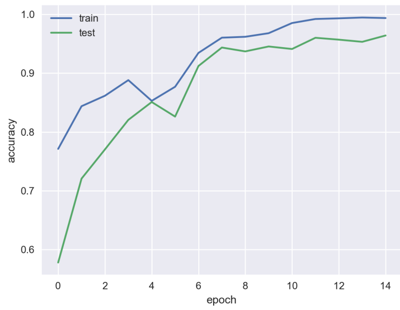


Figure 9: Resnet

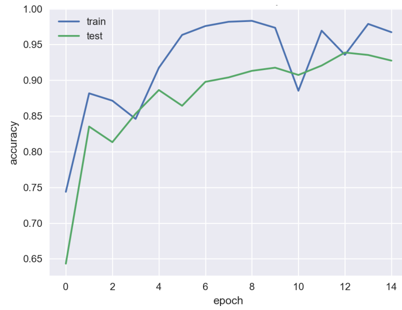


Figure 10: ResNet (Lee)

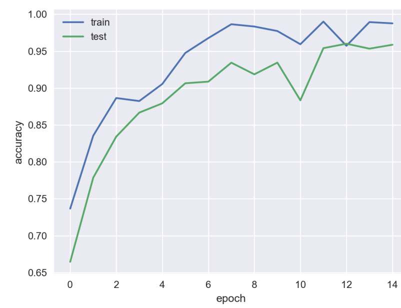


Figure 11: ResNet (En Lee)

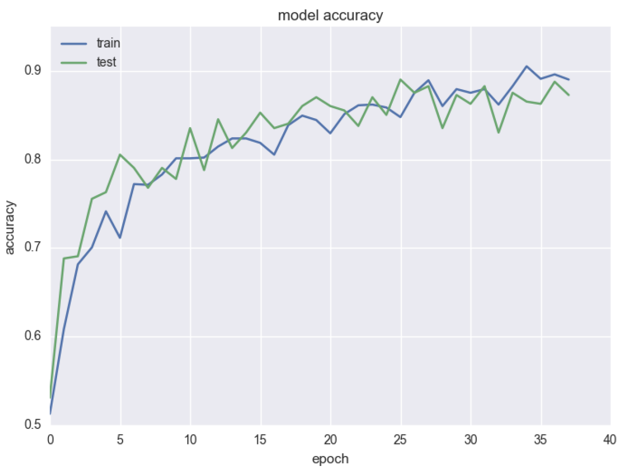


Figure 12: CNN

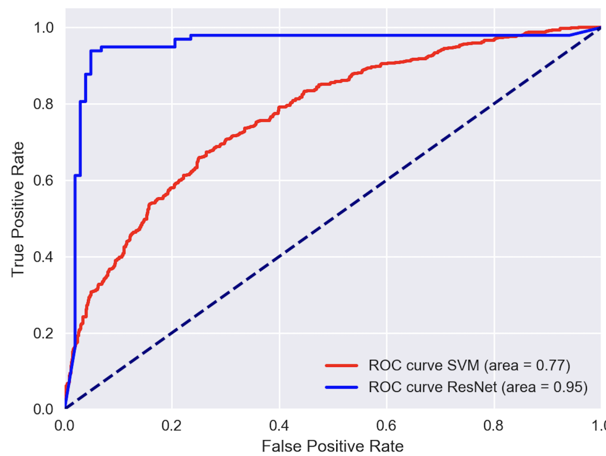


Figure 13: ROC graphs

- The Enhanced Lee Filter works well over the entire image and gives a better accuracy.
- Models on denoised data generalize well and rarely overfit the data.
- The model trains faster after denoising (using enhanced Lee filter) and achieves a 95% accuracy in 6 epochs.

6. Future Work

Our final model will be an ensemble of deep neural nets which work in tandem and are trained separately on the augmented data. We then average out predictions to calculate the final validation.
From the results we saw that denoising using Lee filter reduces the accuracy and the enhanced lee filter is a better option. However, the enhanced Lee filter has difficult reducing the coherent noise, while maintaining the image edges and features. Thus, we would like to look at advanced filtering models that combine enhanced Lee filter with other filters like mean and median filters. Furthermore, we can use CNNs for learning a discriminative model, to extract the speckle component and subtract it from the noisy data [4].
For our future models, we should also keep in mind the different consequence between a false positive and a false negative. A false negative results in ships being unnecessarily rerouted, causing delays and financial loss, while a false positive (missed detection) can be catastrophic[1]. In our future models we will take these two cases into consideration independently.
Combining enhanced Lee Filter, with multistage median filter, perform better at despeckling, smoothing and preserving edges and texture of SAR images.

7. Contribution

1. Atharva Parulekar: Data Visualisation, SVM, CNN models.
2. Dhruv Samant: CNN models, deeper models like residual networks.
3. Jun Li: Data Augmentation, Data Visualisation, SVM models.

References

[1] A.Krezelb A.K.Mazurab A.K.Wahlina. "An object-based SAR image iceberg detection algorithm applied to the Amundsen Sea". In: *IEEE* (2016).

[2] Jose M Bioucas-Dias and Mario A T Figueiredo. "Multiplicative Noise Removal Using Variable Splitting and Constrained Optimization". In: *IEEE Transactions on Image Processing* 19.7 (2010), pp. 1720–1730. DOI: 10.1109/tip.2010.2045029.

[3] Guangyi Chen et al. "Adaptive video denoising using block matching 3-D filtering". In: *2012 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)* (2012). DOI: 10.1109/ccece.2012.6335062.

[4] G. Chierchia et al. "SAR image despeckling through convolutional neural networks". In: *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (2017). DOI: 10.1109/igarss.2017.8128234.

[5] Victor S. Frost et al. "A Model for Radar Images and Its Application to Adaptive Digital Filtering of Multiplicative Noise". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-4.2 (1982), pp. 157–166. DOI: 10.1109/tpami.1982.4767223.

[6] J. W. Goodman. "Some fundamental properties of speckle*". In: *Journal of the Optical Society of America* 66.11 (Jan. 1976), p. 1145. DOI: 10.1364/josa.66.001145.

[7] C. Howell et al. "A Multivariate Approach to Iceberg and Ship Classification in HH/HV ASAR Data". In: *2006 IEEE International Symposium on Geoscience and Remote Sensing* (2006). DOI: 10.1109/igarss.2006.918.

[8] Kaiming Xiangyu Shaoqing Jian. "Deep Residual Learning for Image Recognition". In: *IEEE* (2015).

[9] Diederik P. Kingma and Jimmy Ba. In: ().

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. "ImageNet classification with deep convolutional neural networks". In: *Communications of the ACM* 60.6 (2017), 84fffdfffdfffd90. DOI: 10.1145/3065386.

[11] Gholamreza M. Mohammad A. "A level set based method for coastline detection of SAR images". In: *IEEE* (2017).

[12] Jean-Marie Nicolas and Frdric Adragna. "The Principles of Synthetic Aperture Radar". In: *Processing of Synthetic Aperture Radar Images* (2010), pp. 25–55. DOI: 10.1002/9780470611111.ch2.

[13] Sebastian Ruder. *An overliew of gradient descent optimization algorithms*.

[14] Statoil/C–CORE. *Statoil/C–CORE Iceberg Classifier Challenge*. URL: https://www.kaggle.com/c/statoil-iceberg-classifier-challenge.

[15] Hua Xie, L.e. Pierce, and F.t. Ulaby. "SAR speckle reduction using wavelet denoising and Markov random field modeling". In: *IEEE Transactions on Geoscience and Remote Sensing* 40.10 (2002), pp. 2196–2212. DOI: 10.1109/tgrs.2002.802473.