Using Neural Networks and Object Recognition to Classify Ice Objects

Mingyang “Sky” Sun

Author Bio

Mingyang Sun is a senior at Brentwood College School, who participated in the world-renowned Pioneer Academics Research Program. Hoping to accelerate humanity’s transition from an extractive relationship with nature to a harmonious one, he and his mentor, Professor Susan Fox from Macalester College, implemented machine learning algorithms to categorize arctic ice images. He hopes to positively impact his community through research endeavors. After graduation, he plans to explore the intersection of environmental engineering and data science.

Abstract

Manufacturing processes evolved at an unprecedented rate in the European continent and the United States between 1760 and 1820. They led to population growth as well as an abrupt increase in carbon dioxide and other greenhouse gas emissions. Those greenhouse gases have raised atmospheric temperatures remarkably, melting glaciers and arctic ice in high latitudes. The phenomenon inevitably brings about the most notorious disasters in polar regions, gradually removing animals from their natural habitats. Therefore, the detection of sea ice, especially the task of classification, is influential in supporting environmental scientists with assessing threats. This paper applies a new classification procedure — integrating two-dimensional convolutional neural networks (CNN) with support vector machines (SVM) — to classify arctic ice imagery acquired from various sources, including public domain images and satellite images from Sentinel-2. Because snow and ice are highly reflective and vision conditions of images vary greatly, the images are preprocessed by masking and contouring prior to inputting them into CNN and SVM classifiers in sequence. We conclude that the performance metrics of the CNN-SVM method are 19 percent higher than single feature machine learning algorithms, such as CNN and SVM, because there are fewer training samples and a shorter training time. The classification system can be effectively applied to real-life situations to raise environmental awareness among younger generations.

Keywords: Arctic ice classification; CNN; SVM; Deep learning; Machine learning.

1. Introduction

Climate change is a concerning global issue. There is much more evaporation on a warmer Earth, even by a half-degree Celsius, which puts our physical health, agriculture, and water supply at risk [1].

Sea ice and glaciers store about three-quarters of Earth's freshwater and are presently melting at an unprecedented rate worldwide [2]. For instance, in the period between 1993 and 2019, an average of 279 billion tons of ice per year was liquified in the region surrounding Greenland (according to satellite data), contributing to global sea level rise [3]. Furthermore, the National Snow and Ice Data Center of the United States has approximated the coverage of arctic sea ice in August to drop from 8 million of square kilometers (1981-2010 median) to 6 million of square kilometers (2021). Staggeringly, the area lost within such a short span of time can fit into 0.28 billion regular soccer fields. Scientists claim that if there is no human intervention to decelerate the trend, there will not be enough freshwater available to meet global energy needs by 2040 [4] and, consequently, all living species on Earth could go extinct.

To prevent arctic ice collisions from further damaging private and public properties and forecasting natural disasters, assessing ice conditions via classification has significance. However, analyzing images manually is laborious and expensive for environmental engineers and captains of surface vessels.

As an application of artificial intelligence (AI), machine learning (ML) algorithms are programs that have the capability to improve automatically through experience. Classical classification methods such as the minimum distance classifier (MDC), maximum likelihood classification (MLC) and K-means clustering method have relatively lower accuracy and are solely based on spectral statistical features. On the other hand, deep learning (DL) techniques, which are commonly implemented in, but not limited to autonomous cars, facial recognition systems, natural language processing and speech recognition, can effectively solve the classification task at a faster speed and higher accuracy [5]. Therefore, we applied an existing method, combining a simple 2-Convolutional Layer with Max Pooling Neural Network and a linear Support Vector Machine (CNN-SVM) to classify arctic ice images into either brash ice or iceberg. While CNN-SVM is extracting the spatial information of arctic ice images, it can exploit the spectral characteristics hidden in images as well.

To testify, we compare the performance indicators, which include accuracy, recall, precision, and F1 score, against CNN and SVM algorithms. As expected, the CNN-SVM method achieves a whopping test accuracy of ≈88.82% using our dataset, which is significantly higher than other methods.

2. Related Work

Ice classification has enormous influences in modern society as it has the invisible power to save hundreds of lives that are lost in unforeseen collisions. Consequently, it is not surprising that there have been several ways to approach this task developed over the past 30 years.

2.1 Close-range Imagery

The standards for analyzing ice imagery continue to be underdeveloped, despite fast expansion in machine learning technologies. Even though computer scientists attempt to address this issue, unfortunately none of the existing algorithms can distinguish between arctic ice types utilizing optical images as the majority of methods use low-resolution airborne images.

2.2 Synthetic Minority Oversampling Technique (SMOTE) Algorithm

Pederson et al. conducted a similar experiment of arctic ice classification; but instead, they divide images into nine categories, a large portion of which overlaps with each other and, therefore, require experts' assistance [6]. Moreover, the dataset is imbalanced due to human preferences and environmental conditions, which will make the trained network biased towards the majority classes (brash ice, broken ice, deformed ice, etc.), and very seldomly predict the minority classes (pancake ice, etc.). To solve this problem, the SMOTE algorithm is implemented to automatically synthesize new samples of minority classes. However, this algorithm does not consider neighboring examples possibly from other classes, which can introduce additional noise and increase the overlapping of classes. Noticing these drawbacks, Zheng proposes DSMOTE and ESMOTE based on the original SMOTE algorithm [7]. Instead, the sample weight of an underrepresented sample is dependent on the density of minority and majority samples nearby in DSMOTE. Whereas, in ESMOTE algorithm, the weight is determined by information entropy.

2.3 Squeeze-and-Excitation (SE) Networks

Han et al. propose a novel remote sensing sea ice image classification system, implementing squeeze-and-excitation (SE) network, three-dimensional convolutional neural network (3D-CNN), and support vector machines (SVMs) [8]. Through SENet, the system can rank the importance of an individual feature channel: features "persuasive" for the classification are promoted and those with less effectiveness are suppressed. Weighted features can efficaciously enhance the classification performance of arctic ice pictures.

In spectral feature-based 1D-CNN, kernel slides along one direction. In spatial feature-based 2D-CNN, kernel moves in two dimensions. Similarly, kernel proceeds in three directions in 3D-CNN. The technique takes advantage of both, and is frequently utilized on 3D image data, such as video recordings and medical scans.

3. Method

The implementation framework of our research can be divided into three sections: CNN-SVM, SVM and CNN, as demonstrated in Figure 1. The CNN-SVM part consists of two components: 2D-CNN and the SVM classifier; CNN method is composed of two convolutional layers, two max pooling layers, a fully connected layer and the SoftMax classifier; for SVM, we extract HOG features from preprocessed images prior to feeding them into the SVM classifier. We then evaluate the confusion matrix and compute performance metrics. All methods above will be thoroughly discussed in following chapters.

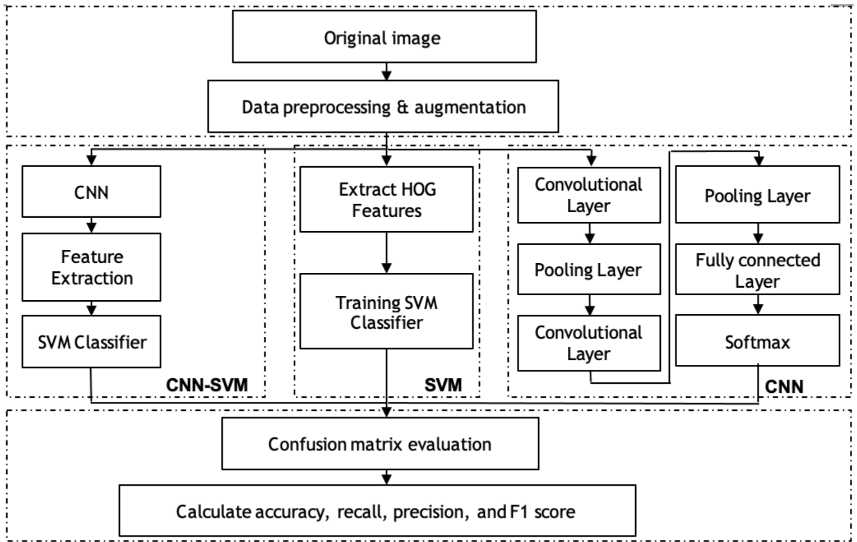


Figure 1. General framework

3.1 Dataset

It is impossible to train a deep learning model without training and testing the dataset. The dataset consists of 112 images altogether, ranging from satellite images to open images found on Google and Yandex Images. Due to the poor weather conditions in the Arctic region, we select high-resolution images without watermarks to maximize the accuracy. A shortage of labeled images online results in a relatively small dataset, since it is extraordinarily arduous and time-consuming to manually split pictures. To generate more data, we implement data augmentation techniques, which will be thoroughly discussed in the following sections.

3.2 Arctic Ice Categories

For the sake of simplicity, we have chosen to segment arctic ice images into two categories based on their characteristics. Underneath, in Table 1, are descriptions of iceberg and brash ice, two of the most accessible out of 220 ice terms defined by the WMO [11].

Table 1. Definition of ice classes (WMO, 2019)

Class	WMO Sea Ice Nomenclature
Iceberg	A massive piece of ice of greatly varying shape, protruding more than 5 m above sea level, that has broken away from a glacier and may be afloat or aground.
Brash Ice	Accumulations of floating ice made up of fragments not more than 2 m across, the wreckage of other forms of ice.

Not recognizable

3.3 Data Augmentation

As shown in Figure 2, we expand the size of the dataset by a factor of 3 by flipping all images both horizontally and vertically (rotating 180-degrees + horizontal flip) to improve the overall performance [12].

(a) Original (b) Flipped horizontally (c) Flipped vertically

Figure 2. An example of data augmentation

3.4 Image Preprocessing

Public images downloaded from the Internet vary widely in size (width and height) due to settings on each individual's device. Here, we implement OpenCV's Resize function to scale a picture up or down and to stretch an image to fixed values of $128px \times 128px$. Following that, thresholding/masking is used to limit the data our computer vision program works on to speed up the process. Gamma correction, whose main objective is to reproduce recordings of Closed-Circuit Television (CCTV) cameras, is applied to several images to control the overall brightness of an image by adjusting the contrast ratio — the ratio between the maximum and minimum brightness.

3.5 Support Vector Machine (SVM)

Linear support vector machine (SVM) is a supervised-learning model and was initially developed by Vladimir Vapnik for binary classification. It is preferable when there is a clear margin of separation between classes, and its primary purpose is to find the optimal hyperplane $f(w, x) = w \cdot x + b$ to maximize the margin while separating two classes in a given dataset [13]. Figure 3 illustrates the idea.

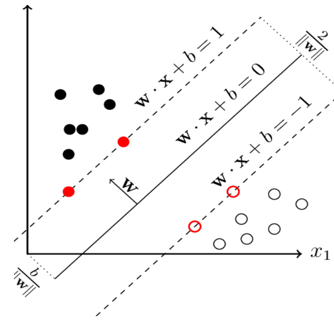


Figure 3. An SVM separating two classes by hyperplane $w \cdot x + b = 0$ (Suárez-Paniagua, 2019)

The equation below is known as L1-SVM, with the standard hinge loss. Explanations of parameters in both Equation 1 and Equation 2 are written out in Table 2.

$$min \frac{1}{p} w^T w \; + \; C \sum_{i=1}^{p} max(0, 1 - y_i'(w^T x_i + b))$$

(1)

Its differentiable equivalent, L2-SVM (Equation 2), produces more stable outcomes with the squared hinge loss [13] — a square of the output of the hinge's *max* function. Figure 4 shows the loss function it generates.

$$min \frac{1}{p} ||w||_2^2 \; + \; C \sum_{i=1}^{p} max(0, 1 - y_i'(w^T x_i + b))^2$$

(2)

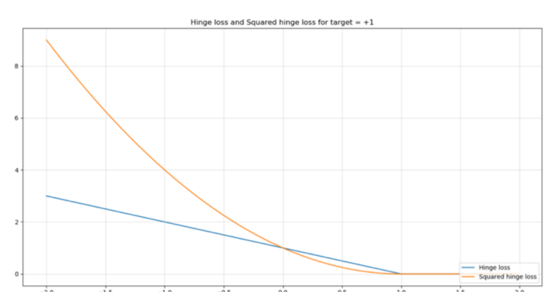


Figure 4. Squared hinge loss compared to regular hinge loss (Chris, 2019)

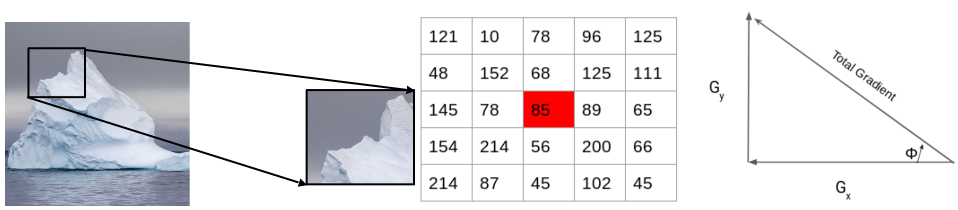
Table 2. Parameters in L1-SVM and L2-SVM

Parameter	Explanation
$w^T w$	Manhattan norm (aka L1 norm)
C	Penalty parameter
y'	Actual label
$\|\|w\|\|_2$	Euclidean norm (aka L2 norm)

3.5.1 Histogram of Oriented Gradient (HOG)

We perform a Histogram of Oriented Gradients (HOG) feature extraction on a labeled training set of images to generate features for SVM to work on. It focuses on the shape or structure of an object and is widely used in computer vision tasks for object detection.

First, we compute the gradient — the small change in the x and y directions — for every pixel in the image, as illustrated in Figure 5a below [14].



(a) (b) (c)

Figure 5. Steps of calculating HOG. (a) A small patch is taken from the iceberg image; (b) An example matrix (Applied Machine Learning Course, 2020); (c) Calculation of total gradient from gradient in x-direction and y-direction

Then, we generate the above pixel matrix (Figure 5b) for the given patch. To determine the change in x-direction of the box highlighted in red, we subtract the pixel value on the left from the value to the right. In similar fashion, we subtract the pixel value below from the value to the top. Thus, the gradient in x-direction for the highlighted pixel is $89 - 78 = 11$; the gradient in y-direction for the highlighted pixel is $68 - 56 = 12$. From the values we calculated, we can conclude that there is a slightly sharper change in intensity in the y-direction at that specific point. The same process is then repeated for other pixels. Next, we need to find the magnitude and direction of each pixel using the Pythagorean theorem (see Figure 5c above). For more information, please visit [14].

3.6 Convolutional Neural Network (CNN)

Different from SVM, CNN is an iterative DL algorithm that can categorize, process, and identify images. Even though training a CNN is computationally expensive, it generally has high accuracy in image recognition problems. The model is composed of multiple layers, including varying numbers (from six to 200) of convolutional layers and pooling layers, a fully connected layer, and a SoftMax classification layer (as demonstrated in Figure 6). Each of these layers is essential as it supports the machine process and classifies pictures.

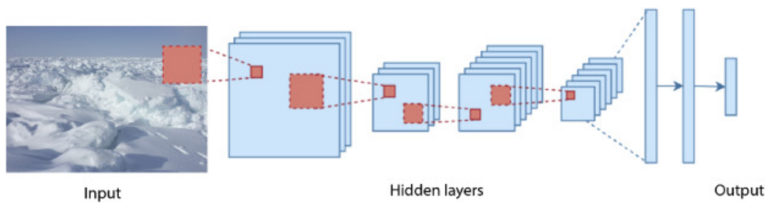


Figure 6. Typical CNN model (Kim, 2019)

3.7 CNN-SVM

We replace the SoftMax classifier in CNN with SVM to create the CNN-SVM model. Hyperparameters for the deep learning models in Table 3 are manually assigned.

Table 3. Hyperparameters used for CNN-SVM and CNN models.

Hyperparameters	CNN-SVM	CNN
Batch Size	128	128
Dropout Rate	0.5	0.5
Learning Rate	0.001	0.001
Steps	10000	10000

SVM Classifier	1	N/A

Here is the overall architecture of CNN-SVM method we applied in this paper [13]:

$$\text{INPUT: } 32 \times 32 \times 1 \tag{3}$$

$$\text{CONV5: } 5 \times 5 \text{ size, } 32 \text{ filters, } 1 \text{ stride} \tag{4}$$

$$\text{ReLU: } max(0, h_\theta(x)) \tag{5}$$

$$\text{POOL: } 2 \times 2 \text{ size, } 1 \text{ stride} \tag{6}$$

$$\text{CONV5: } 5 \times 5 \text{ size, } 64 \text{ filters, } 1 \text{ stride} \tag{7}$$

$$\text{ReLU: } max(0, h_\theta(x)) \tag{8}$$

$$\text{POOL: } 2 \times 2 \text{ size, } 1 \text{ stride} \tag{9}$$

$$\text{FC: 1024 Hidden Neurons} \tag{10}$$

$$\text{DROPOUT: } p = 0.5 \tag{11}$$

$$\text{FC: 1024 Hidden Neurons} \tag{12}$$

The traditional Softmax classifier with the cross-entropy function is replaced by the L2-SVM equation to compute hinge loss at the 10th layer of CNN-SVM. Then, the weight parameters are learned making use of Adam, an optimizer which will be discussed in the following section.

3.8 Adam Optimizer

Adam is superior to other optimization techniques, especially the classical stochastic gradient descent. As opposed to the fixed learning rates, Adam takes advantage of the average of the first and second moments of the gradients. It combines advantages of AdaGrad and Root Mean Square Propagation algorithms and can solve deep learning problems efficiently. As a result, we incorporate the Adam optimizer into our research on arctic ice classification. Adam's unique update rules are listed below [15]:

First, Adam calculates moving averages of gradient and squared gradient. (m_0 $v_0$are initialized to zero)

$$g_t = \nabla_\theta J\left(\theta_{t-1}\right), \tag{13}$$

$$m_t = \beta_1 \mathrm{m}_{t-1} + (1 - \beta_1) g_t,$$

(14)

$$v_t = \beta_2 \mathrm{v}_{t-1} + \left(1 - \beta_2\right) g_t^2 \, .$$

(15)

Then, bias correction is applied to gradient mean (m_t) and gradient variance (v_t), so that the expected value is the one we want.

$$\hat{m}_t = \frac{m_t}{(1 - \beta_1^t)},$$

(16)

$$\hat{v}_t = \frac{v_t}{(1 - \beta_2^t)} \, .$$

(17)

Lastly, Adam optimizer uses previously calculated moving averages to scale the learning rate separately for every single parameter.

$$\Theta_t = \theta_{t-1} - \alpha * \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \varepsilon}}.$$

(18)

Explanations and default values of hyperparameters (appeared above) are as follows:

Table 4. Hyperparameters in Adam Optimizer

Hyperparameter	**Representation**	**Purpose**	**Default Value**
β_1	Exponential decay rate	Controls the weight assignment	0.9
β_2		Weights the mean of the gradient squares	0.99
ε	N/A	Prevents the denominator from being 0	10^{-8}

4. Experiment

Results for the experiment are generated on a computer with a 2.3GHz 8-core 9th-generation Intel Core i9 processor and 16GB 2666MHz DDR4 memory. The proposed method is implemented using the Keras deep learning framework on Google Collaboratory.

4.1 Dataset Description

We employed satellite images captured by Sentinel-2 and 112 public domain images to verify the performance of CNN-SVM method. Of those images, 76 are labeled as icebergs and the remaining 36 are identified as brash ice.

Furthermore, 80% (90) of which are used for training and validation; 20% (22) for testing (as illustrated in Table 5). This split percentage is optimal as computational costs in training and evaluating the model are minimized.

Table 5. Number of Datasets of Images

	Train	**Test**
Iceberg	61	15
Brash Ice	29	7

4.2 Results

Table 6 shows the confusion matrix for the CNN-SVM image classification system. 13 out of 15 pictures the algorithm accurately predicts iceberg as iceberg, whereas the other 13.33% are mistakenly classified. For brash ice, although a larger percentage of images (14.29%) are misclassified, 6 out of 7 pictures are still correctly predicted by the system.

4.2.1 Performance Indicators

If a model flawlessly predicts the positive class, we refer to it as a true positive result. Likewise, a true negative outcome is achieved whenever the algorithm correctly forecasts the negative class.

On the other hand, when the positive class is incorrectly predicted, the model can result in a false positive situation. Similarly, a false negative outcome can be resulted when the algorithm mistakenly predicts the negative class. [16] The most intuitive visualization for interpreting the performance of a statistical classification model is a confusion matrix. See Table 6 below.

Table 6. Table of Confusion

		Predicted	
		Negative	**Positive**
Actual	Negative	True Negative	False Positive
	Positive	False Negative	True Positive

The accuracy formula is a performance metric typically used to measure the percentage of images in a dataset that are correctly predicted. However, it does not consider severe class imbalance.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} = \frac{TP + TN}{TP + FP + TN + FN}$$

(19)

Where TP stands for True Positive, TN represents True Negatives, FP means False Positives and FN is False Negatives. (Same as below)

The precision equation calculates the percentage of images that are actually positive among predicted positive ones [16].

$$Precision = \frac{TP}{Predicted\ results} = \frac{TP}{TP + FP}$$

(20)

Precision is a terrific measurement to consider whenever the cost of FP is high. For example, in email spam detection, a false positive implies that a true negative (non-spam) email has been incorrectly recognized as spam. Consequently, email users might miss business emails if the precision is low.

Recall computes the proportion of true positives the technique captures through labeling it as positive. Applying a similar understanding, we know that Recall is an appropriate indicator while there is a high cost corresponding to false negative [17].

$$Recall = \frac{TP}{TP + FN}$$

(21)

The F1 Score might be a suitable measurement if there is a non-negligible imbalance (large number of actual negatives) [16]. It is a combination of Precision and Recall.

$$\frac{2}{F_1} = \frac{1}{P} + \frac{1}{R} \Rightarrow F_1 = \frac{2PR}{P + R} = \frac{2TP}{2TP + FP + TN}$$

(22)

Performance metrics, including accuracy, recall, precision and F1 score, for CNN-SVM, CNN and SVM are listed in Table 7.

***Table** 7*. Performance Indicators

Method	Accuracy	Recall	Precision	F1 Score
CNN-SVM	0.9050	0.9175	0.8900	0.8403
CNN	0.8152	0.8140	0.7955	0.5882
SVM	0.7273	0.8571	0.5455	0.6667

4.3 Evaluation

Not surprisingly, as a shallow learning technique, the SVM method's classification accuracy is generally low (69.92%). Performance metrics for CNN with its SoftMax classifier are relatively higher. In contrast with previous methods, CNN-SVM can acquire better results than CNN overall since SVM classifier excels in small dataset and nonlinear high-dimensional feature classification tasks. However, according to the experimental outcome, we unexpectedly discover that the recall indicator of the SVM is almost as high as CNN-SVM's. Experts in related fields may need to research to determine the cause.

5. Conclusion

There are multiple ways to classify ice images, but these methods have been left out due to limitations on time and computational resources. In this research paper, we evaluate the performance metrics of classification through different methods, such as CNN-SVM, CNN and SVM. Based on the results we achieve, the novel hybrid CNN-SVM (a combination of simple 2-Convolutional Layer with Max Pooling Neural Network and a linear Support Vector Machine) model has the highest accuracy, recall, precision, and F1-score of 90.50%, 91.75%, 89.00% and 84.03%, respectively. Exceeding 90%, the results can be almost blindly trusted by captains of vessels travelling within the arctic region to assist with ice navigation tasks and alleviate the workload of environmental engineers — for instance, interpretation of ice conditions and locating near-collapsing ice. Hopefully, the development of this deep learning-based classification system can reduce the risk of accidents and collisions. Furthermore, the decreasing number of icebergs identified annually may potentially warn the younger generation of the imminent threats of climate change.

The quantity of optical ice images accessible on the Internet for training and testing is extremely limited and was the biggest problem I encountered while conducting the experiment, as it affects the overall performance. To effectively resolve the issue, I'm planning on integrating the system with drones or IoT equipment in the future. Additionally, the arctic-ice classification system would be convenient and practical if it can segment images into multiple categories by breaking down the task into smaller subproblems, all of which are binary classification problems. We are looking forward to seeing further development with the system and being put into practical use in the upcoming years.

References

[1] UCAR Center for Science Education. (n.d.). *The Water Cycle and Climate Change*. The Water Cycle and Climate Change | UCAR Center for Science Education. https://scied.ucar.edu/learning-zone/climate-change-impacts/water-cycle-climate-change.

[2] USGS. (n.d.). How much of the Earth's water is stored in glaciers? https://www.usgs.gov/faqs/how-much-earths-water-stored-glaciers?qt news_science_products=0#qt-news_science_products.

[3] National Snow and Ice Data Center. NSIDC Arctic News and Analysis RSS. (2021, June 8). http://nsidc.org/arcticseaicenews/.

[4] NASA. (2021, May 10). Climate Change Evidence: How Do We Know? NASA. https://climate.nasa.gov/evidence/.

[5] IBM Cloud Education. (2020, July 15). What is Machine Learning? https://www.ibm.com/se-en/cloud/learn/machine-learning.

[6] Pedersen, O.-M., Kim, E. (2020). Arctic Vision: Using Neural Networks for Ice Object Classification, and Controlling How They Fail. *Journal of Marine Science and Engineering, 8*(10). https://doi.org/10.3390/jmse8100770

[7] Zheng, H. (2020). Improved SMOTE algorithm for imbalanced dataset. *2020 Chinese Automation Congress (CAC).* https://doi.org/10.1109/cac51589.2020.9326603

[8] Han, Y., Wei, C., Zhou, R., Hong, Z., Zhang, Y., Yang, S. (2020). Combining 3D-CNN and Squeeze-and-Excitation Networks for Remote Sensing Sea Ice Image Classification. *Mathematical Problems in Engineering, 2020,* 1–15. https://doi.org/10.1155/2020/8065396

[9] Ling, Z., Li, X., Zou, W., Guo, S. (2018). Semi-Supervised Learning via Convolutional Neural Network for Hyperspectral Image Classification. *2018 24th International Conference on Pattern Recognition (ICPR).* https://doi.org/10.1109/icpr.2018.8545709

[10] Khaleghian, S., Ullah, H., Kræmer, T., Hughes, N., Eltoft, T., Marinoni, A. (2021). Sea Ice Classification of SAR Imagery Based on Convolution Neural Networks. *Remote Sensing*, *13*(9). https://doi.org/10.3390/rs13091734

[11] WMO. (1971). WMO sea-ice nomenclature. *Journal of Hydrology, 14(3-4),* 354. https://doi.org/10.1016/0022-1694(71)90048-5

[12] Gandhi, A. (2021, May 20). *Data Augmentation: How to use Deep Learning when you have Limited Data.* AI & Machine Learning Blog. https://nanonets.com/blog/data-augmentation-how-to-use-deep-learning-when-you-have-limited-data-part-2/.

[13] An Architecture Combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for Image Classification. (2019, February 7). https://arxiv.org/pdf/1712.03541.pdf

[14] Singh, A. (2020, May 10). *Feature Engineering for Images: A Valuable Introduction to the HOG Feature Descriptor*. Feature Descriptor | Hog Descriptor Tutorial. https://www.analyticsvidhya.com/blog/2019/09/feature-engineering-images-introduction-hog-feature-descriptor/.

[15] Brownlee, J. (2021, January 12). *Gentle Introduction to the Adam Optimization Algorithm for Deep Learning*. Machine Learning Mastery.

[16] Shung, K. P. (2020, April 10). *Accuracy, Precision, Recall or F1?* Medium. https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9.