ShipNet: A Fishing Vessel Dataset Challenges and Opportunities

Gregory Weaver
Department of Computer Science
Stanford University
Stanford, CA 94305
gregweav@stanford.edu

Abstract

Opportunities in capturing global ship movements in real time through the use of satellites, alongside publicly available images of ships, open up the possibility of creating new data sets with precise labels giving information about the type of gear that the ship uses while fishing. We detail the process of creating a data set that maps publically sourced images to precise labels about that ship, pre-training on existing data sets with ResNet convolutional architecture, training with these models on our data set, and understanding network performance through unsupervised learning techniques. Lastly, we propose future directions for research with this data set.

1 Introduction

1.1 Background / Importance of Task

One of the largest threats to humans is a lack of understanding of how activities that are needed for our survival also contribute to future biosphere and human welfare insecurity.Through agriculture, forestry, and fishing, humans exert influence and reshape ecosystems, and will do so at larger scales as the total population of the world increases. In each of these mediums, large-scale human influence threatens organism populations on all trophic levels, and negatively affects key abiotic variables such as air, water quality, and CO2 concentration.

However, recent advances in satellite-based observation methods give us solace in understanding our impact on the Earth in real-time for land-based activities such as forestry and agriculture. However, until recently, understanding the specifics of our global fishing footprint was daunting by virtue of the high seas' size (90 percent of all habitable space on earth) and remoteness.

However, interception of signals that much of the world's shipping and fishing fleet relies on to avoid collisions has given us many time series about a ship's coordinates, velocity, and course. Specifically, a recent expansion of the automatic identification system (AIS) presents an opportunity to quantify the behavior of global fleets down to individual vessels. Although AIS was originally designed to help prevent ship collisions by broadcasting to nearby vessels a ship's identity, position, speed, and turning angle every few seconds, these messages can also be recorded by satellite- or land-based receivers. Whereas its usefulness as a tracking tool has been established, work by Kroodsma et al. (2018) use these signals, captured over a period of time, as a signature of the ships' movements across the oceanscape.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

1.2 Dataset and Task Definition

The authors of this paper processed 22 billion AIS positions via satellite between 2012 to 2016 and extracted the features from AIS as described above. For each ship, 12,000 consecutive time points were collected. For each time point, 12 features indicating direction, velocity, and other temporal qualities were extracted. Using this 12,000 * 12 = 144,000 dimensional representation of a ship's movement patterns as input for their supervised classifier. They used labels from cross-referenced shipping registeries to produce labels of gear type for the neural network to train on. The specific architecture that they used was a 1-d convolutional neural network, as is common for time series classification of non-variable length inputs. The labels indicated what the gear type of the ship was, and the researchers reported an accuracy of over 90 percent.

The lab headed by Dr Barbara Block at Hopkins Marine Station of Stanford University thought that this data set would be valuable because of the relevance of the gear type (in the case of a fishing vessel) to what type of fish that the ship is targeting. For example, long-liners (ships with very long fishing lines with large hooks trailing at the stern) have equipment that is different than purse-seiners (ships that use large nets underneath to capture any life that may be swimming directly underneath the boat at a depth less than 100 meters), and each ship will have a few target species of fish that it is directing its fishing effort towards.

The lab saw an opportunity to augment this labelled data set by using each sample's Maritime Mobile Service Identity (MMSI) number to search through publically available databases that contain ship images. We used the confidential data set given to us by Global Fish Watch (GFW) to scrape the web using the MMSI's contained therein. After using a multi-threaded web scraper, we were able to get hit rate of 2 percent (25,000 images out of more than 1.2 possible ships) out of the database.

Using the images scraped from the web, we tested the claim about the accuracy of the classifier that generated the labels. Using a ship identification expert, we found that the confidence score outputted by the classifier accurately reflected the probability that the label was correct. To our knowledge, this is the first time that this GFW data set has been externally validated through visual means.

After scraping was done, the data set had images from 30 types of ships, with drastically different quantities for each ship type. Surprisingly, we found that the label distribution reflected the empirical distribution of ships on the open sea right now: dominated by cargo and passenger ships. As one can see in 1 below, the main ship types that fall under the category of fishing are drifting long lines and trawlers.

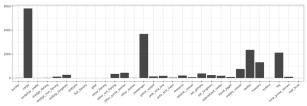


Figure 1: Distribution of classes in the ShipNet data set

This data set, for the rest of the paper, will be called ShipNet.

1.3 Related Work

As the workhorse for our transfer learning scheme, we utilized the work done by He et al. (2015) by using their weights both for our baseline and pre-trained model. The reason for this was He et al. (2015)'s ability to combine accurate models and high-quality feature extraction from images with a

(a) Research Vessel

(b) Cargo Ship

Figure 2: Samples from the ShipNet data set

small number of weights. This particularly useful because of the limitations that we found with GPU compute time.

A data set that will be used to augment ShipNet will be the MARVEL data set. This data set, created by Gundogdu E. (2016), was created by scraping publically available ship image databases, similar to the one that we scraped. However, we did diverge in the source of the labels to create the data set. In the case of Gundogdu E. (2016), they used the information on the website, which is self-reported and may have biases towards vessel types that are not commonly associated with illicit fishing efforts. Constrastingly, the process through which candidate samples for the data set were created was through the remote satellite observation of fishing activity. We find that this source is of higher quality and to the benefit of our planned use in the wild. To see this, we have included a histogram of Gundogdu E. (2016)'s histogram of class labels in 3. First, we note that our data set is more relevant for an application with an eye towards fisheries management by virtue of the content of the tail-end of our data set, which is more populated with ship types such as squid-jiggers, purse seines, set long lines, gill-nets vs yachts, crude oil tanker, and sea dredgers as seen in Gundogdu E. (2016).

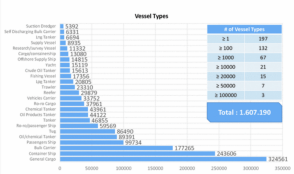


Figure 3: Distribution of classes in the Marvel data set

After running the scraper as prescribed by Gundogdu E. (2016), we arrived at a secondary data set with an additional 38,000 images.

2 Methods and Results

2.1 Training

Our first approach was to simply use the pre-trained weights from He et al. (2015) up to the layer that contains outputs relevant to the class structure in Deng et al. (2009). In this training regime, we trained on 22,000 images and used the remaining 3,000 as validation. We used a batch size of 32, as is the street-fighting technique used by Bengio (2012). Also, for purposes discussed in Leclerc et al. (2018), we used a learning rate of .0001 in combination with the ADAM optimizer. Furthermore, for our loss scheme, we used in separate experiments cross-entropy and weighted cross-entropy loss. The use of weighted cross-entropy was in an attempt to mitigate the class imbalance present in the data set. In both regimes of weighted cross-entropy loss and standard cross-entropy, the training accuracy was 75 percent and the training accuracy was 65 percent. The classifier did not discriminate less-abundant classes from the common ones; there were 17 classes out of 30 that had 0 percent accuracy. Further inspection of the confusion matrix indicated that there were zero false positives for these 17 classes as well. This to us indicates that the network is settling on a strategy that is just outputting the most common classes, with the further observation that this is not mitigated by label-aware losses. Later geometric analysis will attempt to explain this.

Another approach that was tried was using first pre-training on the MARVEL data set, and still starting with the weights from He et al. (2015). In this case, the training procedure was the same, in which 37,000 samples were used in the training set, and a small test set of 1,000 images was kept to measure performance on this auxiliary data set. Within the Marvel data set, there was a validation accuracy of nearly 95 percent. After taking these trained weights and training further on ShipNet, it was found that training accuracy was 75 percent and test accuracy was 53 percent. However, the number of classes with 0 percent accuracy was reduced to 14. It is hopeful in the future to train further with this pre-training approach.

2.2 Unsupervised Learning and Data Set Geometry

After going through the training schemes described above, we used extensive unsupervised learning techniques to understand the behavior of our best- trained model. We felt that, given our resource

(a) Correlation distance matrix, row + columns ordered by ship label

(b) Correlation distance matrix with no ordering by label

(c) Euclidean distance matrix, row + columns ordered by ship label

(d) Euclidean distance matrix with no ordering by label

Figure 4: Samples from the ShipNet data set

constraints, this approach would provide for quicker iteration cycles that would lead to more fruitful analysis for the end aim of providing an intuition for the ShipNet data set. To do this, we took the latent representations for each sample from our best model (per test set accuracy) and used this new, condensed representation as features for more streamlined unsupervised learning algorithms. The first algorithm that we used was spectral clustering

2.3 Spectral Clustering

We chose this method due to its slim coding profile on Pedregosa et al. (2011) and its ability to not eat up gpu time. Spectral clustering is described in the algorithm below.

Algorithm 1: Spectral Clustering

Input: Similarity matrix $S \in \mathbb{R}^{n \times n}$, number k of clusters to construct. ;
Let W be its weighted adjacency matrix. ;
Compute the unnormalized Laplacian L. ;
Compute the first k eigenvectors $v_1 ... v_k$ of the generalized eigenproblem Lv = $\lambda D v$.
Let $V \in \mathbb{R}^{n \times k}$ be the matrix containing the vectors $v_1 ... v_k$ as columns;
For i = 1...n, let y_i $S \in \mathbb{R}^k$ be the vector corresponding to the i-th row of V. ;
Cluster the points $(y_i)_{i=1,...,n}$ in $S\mathbb{R}^k$ with the k-means algorithm into clusters $C_1 ... C_k$

Note that we constructed a distance matrix, visualized both randomly and with row-ordering induced by labels and different distance metrics, in 4. In order to construct the similarity matrix from a distance matrix, we simply used the Gaussian Kernel as discussed in class: $s(i,j) = e^{-1*d(i,j)}$. Intuitively, this expression drives higher distance values towards zero similarity values and has an upper bound of 1. This similarity matrix can be viewed as a matrix construction of a graph, albeit with continuous values for the weights of the edges of the graph. In order to construct the graph laplacian of the similarity matrix, we subtract D, a diagonal matrix whose entries at index i along the diagonal (coordinate (i,i)) represent the ith row-sum of S, the similarity matrix that we just computed, from S: $L = D - S$.

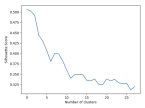
To first show that the differences in representations of the images were statistically significant between in-label and out-of-label distance, we did a simple independent sample t-test. To do this, we sampled distances in the block diagonal of our distance matrix, whose sub-matrix boundaries were defined by the abundancy of each label, and compared them to all values off of our collection of sub-matrices. Firstly, we report that all mean distance value between within-label distances are statistically significantly lower than mean inter-label distance, with a p-value of roughly 1e-15. This can be intuitively seen by the fact that having rows organized by label introduces block structure within the distance matrix that is not present with shuffling. This means that there is coherency in the latent space separation of the classes, a positive sign for the network extracting meaningful features along the borders of the labels.

After running some statistical analysis of the distance matrix, we used our similarity matrix, constructed via the Gaussian kernel, for spectral clustering. Since spectral clustering relies on a hyper parameter k, we tuned this parameter by settling on the k that maximized the silhouette score of the labels returned by algorithm. The silhouette score is a measure of how similar (in this case, the euclidean distance between samples in the latent space)a sample is to data points in its own cluster

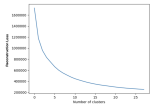
compared to data points in other clusters. Note that the clusters in this case are derived independently of labels. It was outside the scope of time for a one-man team to identify the coherency of the spectral labels to the hueristic labels that denote the gear type of the ship from ShipNet. The silhouette scores range from -1 to +1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters. The value of a single data point i is defined as:

$$s(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}}, a(i) = \frac{1}{|C_i|-1}\sum_{j\in C_i, i\neq j} d(i,j)\ , b(i) = \min_{i\neq j} \frac{1}{|C_j|}\sum_{j\in C_j} d(i,j)$$

We take the silhouette score to be the average of the silhouette values as defined above across all samples in the data set. In Figure 5a below, we find that the silhouette score evens out at around k = 12 clusters. While a higher silhouette score indicates better clustering, we also validated this roughly 12 cluster figure by looking at the reconstruction loss for the k-means algorithm, which ran not through the similarity matrix but through the original distance matrix. As we can see in 6b, we see an elbow point around 8 - 12 cluster.



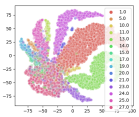
(a) Silhouette scores as we vary hyperparameter k, spectral clustering



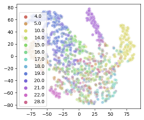
(b) Correlation distance matrix with no ordering by label

Figure 5: Hyperparameter tuning in the unsupervised setting

Finally, to potentially understand the empirical behavior of the best trained network, which had a large bias towards predicting well-represented classes that was not alleviated by label-aware losses, I looked at the latent representations of the top-7 represented classes vs the rest with t-SNE, which is well-known embedding technique that maps data points to a low-dimensional (typically 2-dimensional) subspace that preserves local orientation of data points. Here, note that the cut-off of 7 comes from 3 sources of intuition: silhouette score, reconstruction loss, and also the fact that it makes up 85 percent of the data set. As we can see in 6, the classes in the bottom-23 classes become more entangled, indicating that the features extracted by the neural network do not sufficiently differentiate between these classes. Future work points to a possible future direction for this data set.



(a) t-SNE embeddings for top-7 represented classes



(b) t-SNE embeddings for bottom-23 represented classes

Figure 6: t-SNE embeddings

3 Conclusion and Future Work

In this paper, we explained the process through which we created a new supervised learning data set, ShipNet, and documented a well-known convolutional neural architecture's (resNet18He et al. (2015)) behavior in a transfer learning scenario under multiple training/loss regimes. Specifically, label-aware loss functions, the first approach typically taken in trying to alleviate class-imbalanced data sets, was ineffective. Lastly, we used unsupervised learning techniques to understand the class structure of the data set as represented in the latent space of the best model that was created. These analysis indicated that the network was really only differentiating between the top-7 most prevalent classes, and had a tangled embedding space for the bottom-23. A future protocol that addresses empirical training concerns with a theoretical basis is to outfit the convolutional architecture with attentional module. This system could potentially extract fine-grained aspects of gear types that are in the image, but maybe not picked up by standard convolutional architectures used here.

4 Acknowledgements

All code is available at `https://github.com/gregweav/229Project`. I would like to acknowledge that Figure 1 came courtesy of Emma Gee. Furthermore, the broad idea of scraping this data set for a supervised learning data set was tried by myself in Spring of 2018 for CS 231n, of which the repo from github will have a copy of the write-up. Note that the code base for this project, with the exception of a few lines from the web-scraper, was done completely from scratch.

References

References

Yoshua Bengio. 2012. Practical recommendations for gradient-based training of deep architectures. *Neural Networks: Tricks of the Trade*, page 437–478.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Yucesoy V. Koc A. Gundogdu E., Solmaz B. 2016. Marvel: A large-scale image dataset for maritime vessels. Asian Conference on Computer Vision (ACCV).

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

David A. Kroodsma, Juan Mayorga, Timothy Hochberg, Nathan A. Miller, Kristina Boerder, Francesco Ferretti, Alex Wilson, Bjorn Bergman, Timothy D. White, Barbara A. Block, Paul Woods, Brian Sullivan, Christopher Costello, and Boris Worm. 2018. Tracking the global footprint of fisheries. *Science*, 359(6378):904–908.

M. Leclerc, R. Tharmarasa, M. C. Florea, A. Boury-Brisset, T. Kirubarajan, and N. Duclos-Hindié. 2018. Ship classification using deep learning techniques for maritime target tracking. In *2018 21st International Conference on Information Fusion (FUSION)*, pages 737–744.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.