Locating Ships on the High Seas and in Ports

Lee Arthurs (larthurs@Stanford.edu), CS230 Project

Motivation

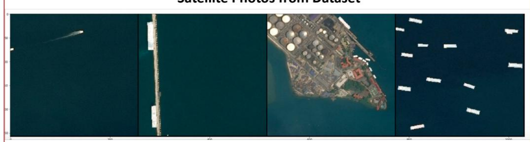
- Kaggle "Airbus Ship Detection Challenge", (https://www.kaggle.com/c/airbus-ship-detection) to located ships in satellite photographs
- Interesting problem with real world application and available data due to the Kaggle Competition
- No specific papers published on this dataset due to ongoing Kaggle competition

Dataset

The Kaggle dataset has 192,555 of images that were labeled with Masks for Ships if they are present. The dataset is very unbalanced with 78% of the images with no ships. As you can small and some have many ships on one image. Here some examples of the data:

Images	Number	Percentage
No Ships	150,000	78%
Ships	42,555	22%

Satellite Photos from Dataset

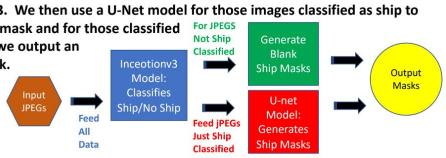


Same Satellite Photos with Masks Overlaid

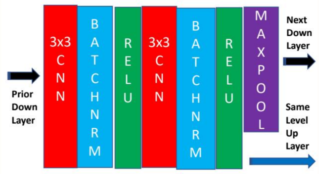
The data was randomly divided into 172,555 training images, 10,000 development images and 10,000 test images.

Model Structure: Divide Into Two Parts

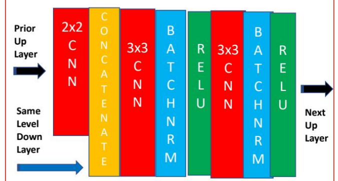
As you can see in the chart below we divide the problem into two parts because of the unbalanced data. We first classify images as ship or no ship using Inceptionv3. We then use a U-Net model for those images classified as ship to generate a mask and for those classified as no ship we output an empty mask.



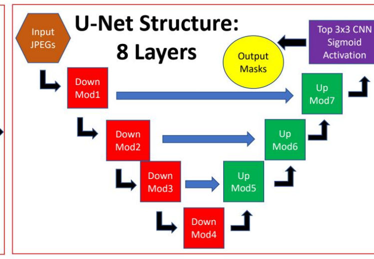
U-Net Down Module



U-Net Up Module



U-Net Structure: 8 Layers



Inception V3

For the first classification part of the model we used the TF-Slim implementation of inception_v3 from Tensorflow HUB (https://tfhub.dev/google/imagenet/inception_v3/feature_vector/1) with the pretrained weights. The top level of the model was trained on the training set to distinguish between images with and without ships. We explored using different learning rates to come up with best results.

Inception V3 Results

LR	Train %	Val %	Test %
.005	94.1%	94.1%	93.8%
.00875	94.1%	94.0%	93.8%
.01	94.1%	93.9%	93.9%
.02	93.8%	93.6%	93.6Z%
.05	93.7%	93.3%	94.3%

LR = Model Learning Rate

Contributions from Others

I had originally partnered with Suren Talla, another SCPD student in CS230. We worked together through the Milestone. Since the Milestone he had significant family issues and ultimately had to drop from the project. He contributed in choosing this project, the initial research and ideas discussed in the Milestone. All the modeling done here is all my work and has been done since the Milestone.

Link to YouTube Video Presentation

https://youtu.be/uHAMBrHxMDk

Dice Coefficient

I used the Dice Coefficient to evaluate the mask results.

$$DC = \frac{2\sum P*T+1}{\sum P+\sum T+1}$$

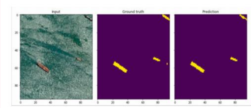
Where P is the predicted mask and T is the true mask. Values range from 0 to 1, with 1 being the best match. I used the negative of the Dice Coefficient as the lost to train the U-net Model

U-Net Best Model

The Best U-Net Model was trained just on training data that had ships and evaluated on training data that just had ships. We found was the 8 Layer Structure shown above (more layers did not improve performance) with Batch Norm included had these results:

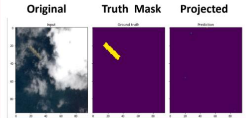
Epochs	Train Dice Coef	Val Dice Coef
50	91.5%	89.9%
100	95.6%	94.1%
150	97.2%	95.2%

U-Net Example Results: Val Data



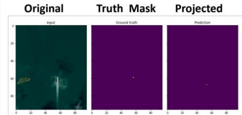
Dice Coefficient for Projected Mask: .968 Missing Tiny Ship Center Far Right

Original Truth Mask Projected



Dice Coefficient for Projected Mask: .007 Ship Not Picked up by Projected Mask Probably Due to Clouds

Original Truth Mask Projected



Dice Coefficient for Projected Mass: .50 Very Small Ship with large wake. Projected mask had small ship in Wake instead of where It actually is

Original Truth Mask Projected

Combined Results

I did not have time to run combined results of these best two scenarios for this poster but I will Include these combined results in the final paper.

Future Work

- Try Different Classification Models for the first model other than Inception V3 to see if we get better results
- Explore more U-Net structures
- Try other pretrained mask models such as MASK RCNN as an alternative to U-Net
- Test if data augmentation will improve the results of either model..