**DETECTING SHIPS IN SATELLITE IMAGERY**

**Introduction**

Team Members:1. Aditya Pujari (11491374)  
2. Brinda Potluri (11526591)

3. Praveen Kumar Somara (11525451)

Goals and Objectives:

**Motivation:**

The purpose of this study is to learn more about image processing and object detection techniques. Object detection is quite common these days, as most of them employ pre-trained models such as YOLO, but we want to explore and do everything from scratch. As a result, we chose a project with images (satellite photos) of high dimensionality and objects that must be detected are very small in size, making pre-trained models ineffective.

**Idea Description:**

The main objective is to detect the ships in satellite images. We'll utilize faster Region Convolutional Neural Network (R-CNN). To balance the dataset, data augmentation is employed and for item detection selective search.

The dataset that we are going to use for this project is ‘Ships in Satellite Imagery’ from Kaggle. It's made up of 4000 80x80 RGB photos labeled with "ship" or "no-ship" classifications and saved as a 19200-item list. These images were extracted from ‘Planet satellite imagery collected over the San Francisco Bay and San Pedro Bay areas of California’.

**Goals:**

Developing a model for categorizing ROIs into two categories: ship and no-ship. Obtaining ROIs and testing the model on geospatial data using Selective Search. To obtain the Region of Interests, we used Regional Proposal Networks, which cut down on computing time.

**Significance:**

This project can help address the issue of detecting massive ships' whereabouts via satellite pictures also in analysis of the supply chain and monitor the levels of port activity. It also offers significant insights on a variety of markets, including energy, defense and intelligence, finance, and agriculture.

**Literature Survey:** This work focuses on producing land cover classification mapping based on satellite data using a convolutional machine learning model with a modified U-Net structure. Two datasets were used one for classification task and the other for segmentation. ResNet50 model and the pre-trained ResNet50 classification model were the two architectures employed. For the evaluation, precision and recall metrics were used. Class-based analysis and accuracy measurement were used since the land cover was highly uneven.

**Objective:**

Main milestones in our project are:

1. Dataset selection based on the availability of quality of images.
2. Data augmentation to increase the dataset by applying rotation, shearing and cropping to images. And also balance the dataset for classification model (ship/no ship).
3. Image preprocessing – making all the images of same size and applying feature extraction filters for better training of model.
4. Training Convolutional Neural Network for classifying if the image has ship or no ship
5. Training R – CNN model for detecting ships in image on ship images dataset
6. Evaluating the model on test dataset.
7. Making a main pipeline for production ready.

**Features:**

1. **Images – balancing dataset using data augmentation :**

Done in order to expand the dataset and balance the categorization models for ships and no ships. Image augmentation is a process in which images are processed to a variety of image manipulations such as rotating, shearing, and cropping.

1. **Image Feature Extracted using convolutional kernels :**

CNNs can learn filters that detect edges and blob-like structures in lower-level layers of the network using convolutional filters, nonlinear activation functions, pooling, and back propagation, and then use the edges and structures as building blocks to detect higher-level objects in deeper layers of the network.

1. **Selective Search for Object Detection :**

Selective Search begins by over-segmenting the image based on pixel intensity using Felzenszwalb and Huttenlocher's graph-based segmentation approach. Selective Search is a strategy for item identification and area proposal. It is designed to be quick and have a high recall. It functions by calculating a hierarchical grouping of similar regions based on color, texture, size, and form compatibility.

**Related Work (Background)**

Detecting an object, drawing a boundary and assigning them a class label is called object detection. It is important as its applications include self-driving cars, anomaly detection, and video surveillance. The R-CNN Model Family and the YOLO Model Family are the top two deep learning models for object detection. Faster R-CNN has the shortest prediction and computation time, whereas R-CNN has the longest, with fast R-CNN in the middle. YOLO uses a single neural network that is trained from end to end, taking a photograph as input and simply predicting bounding boxes and class labels for each bounding box. Because only two anchor boxes in a grid forecast only one class of object, YOLO has trouble recognizing objects that are small and close together.

**Dataset**

The dataset contains 4000 80x80 RGB photos that have been classified as "ship" or "no-ship." Within the data list, the pixel value data for each 80x80 RGB image is saved as a list of 19200 integers. The red channel values are found in the first 6400 entries, the green channel values in the following 6400, and the blue channel values in the last 6400. The image is stored in row-major order, with the first 80 items of the array representing the first row's red channel values.

**Detail design of Features**

* This is done to increase the dataset and balance the categorization models for ships and no ships. Image augmentation is a technique that involves rotating, shearing, and cropping photographs.
* CNNs can use convolutional filters, nonlinear activation functions, pooling, and back propagation to learn filters that detect edges and blob-like structures in lower-level layers of the network, and then use the edges and structures as building blocks to detect higher-level objects in deeper layers of the network.
* Selective Search begins by over-segmenting the image based on pixel intensity using Felzenszwalb and Huttenlocher's graph-based segmentation approach. Selective Search is a strategy for item identification and area proposal. It is designed to be quick and have a high recall. It functions by calculating a hierarchical grouping of similar regions based on color, texture, size, and form compatibility.

**Analysis**

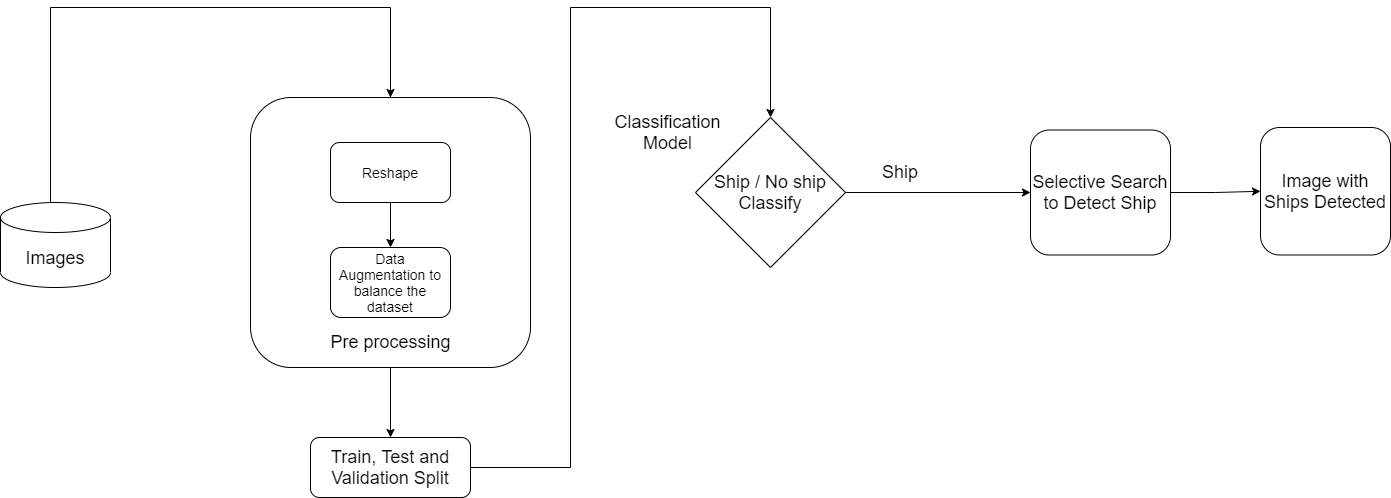
Exploratory Data Analysis is performed to understand the data more clearly. The dataset is unbalanced with 3000 no-ship images and 1000 ship images. If this data is not cleaned or balanced the result will be biased. Augmentation is performed. The purpose of image augmentation is to generate modified copies of the original images in order to artificially expand the size of the training image dataset. Now our data is balanced with 50-50 ratio.

**Expected Outcome**

The initial classification model used to determine whether or not there is a ship in the image. And the training classification accuracy should be as close to 98 percent as possible. The pipeline's trained classification model is then connected to an object detection model, which recognizes the ships in the image. We anticipate that the bounding box in which a ship is spotted from a picture will be almost perfect.

**Implementation**

Github Link: <https://github.com/adityapujari98/Detecting-Ships-in-Satellite-Imagery>

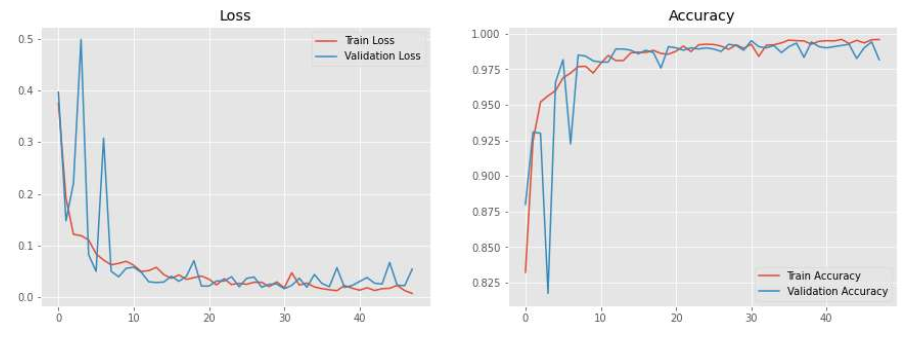
****

Firstly, the dataset was unbalanced, therefore we pre-processed it. Images have been reshaped. It is a crucial pre-processing step because it equalizes the size of all the pictures. Data augmentation is done as part of the pre-processing. By creating new and varied samples to train datasets, it is possible to improve the performance and results of machine learning models. A machine learning model performs better and is more accurate when the dataset is rich and sufficient.

Training and testing are done after pre-processing the data. Confusion matrix is used for validation. It demonstrates how any classification model can be confused when making predictions. The confusion matrix not only reveals the errors that your classifier is making, but also the types of errors that are being made. The images are classified using a classification model depending on whether or not they contain a ship. If a ship is found, selective search is utilized to locate the Region Of Interest and construct a boundary around it. Finally, the image with the detected ships will be displayed.

**Preliminary Results**

**Ship / No Ship Image Classifier**

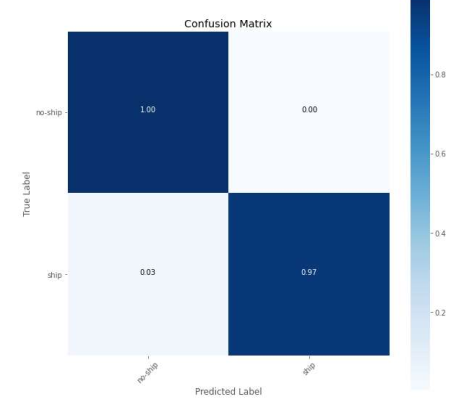
****

Training Accuracy: **99.59%**

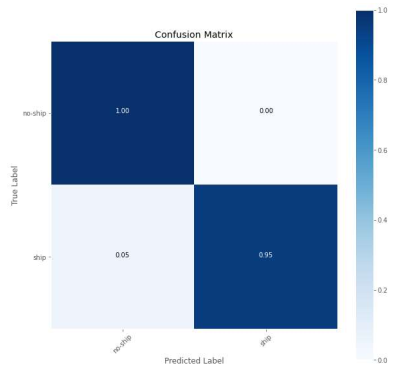
Validation Accuracy: **98.17%**

**Confusion Matrix**

**Training Dataset**

****

**Testing Dataset**

****

**Project Management**

**Implementation status report**

**Work completed**

**Description:** Completed collecting dataset from Kaggle, Analyzing Data and preprocessing (balancing dataset), building classification model.

**Responsibility (Task, Person):**

Aditya Pujari:

* Finding the suitable dataset for performing the ship or no ship classification task (Part 1)
* Implemented Data Augmentation on ship images to create a balanced dataset.
* Working on ship detection (Part 2) based on Selective Search (R-CNN)

Brinda Potluri:

* Performed Exploratory Data Analysis on Dataset and found that dataset is imbalanced.
* Build an CNN Image Classifier model to classify whether there is ship or no ship in image
* Finding best all possible methods for small object detection in large image

Praveen Somara:

* Hyper tuned image classifier model to get training accuracy of 99.59% and validation accuracy of 98.17%
* Applying Selective Search for ship detection in scene images where multiple ships are present and analyzing the result.

**Contributions (members/percentage)**

Aditya Pujari 35%

Brinda Potluri 35%

Praveen Somara 30%

**Work to be completed**

**Description:** Building Object Detection method using Selective Search, making production ready pipeline, deploying it on Heroku instance

**Responsibility (Task, Person):**

Aditya Pujari:

* Finding the best method for ship detection (object detection)
* Making production ready pipeline to deploy on website
* Building frontend for detecting ships website

Brinda Potluri:

* Implementing Fast R-CNN method for Object Detection

E.g. Selective Search

* Building APIs to request image and give response whether ship is present in image or not. If present display detected ships in image

Praveen Somara:

* Making Heroku instance ready with all required dependencies
* Deploying the site on Heroku platform.

**Issues/Concerns**

* The ship detection method may be time consuming and require more computation resource
* While deploying on Heroku instance the code files should not exceed 500mb as free instance has limit of 500mb
* The images in production should be of large size more than 2048 pixels for proper ship detection (need to add a validation)
* As in validation accuracy the image classifier (ship/no ship) is giving 98.17% , in production the image classifier should work properly as the ship detection is dependent on it

**Expected Outcome:**

The initial classification model used to determine whether or not there is a ship in the image. And the training classification accuracy should be as close to 98 percent as possible. The pipeline's trained classification model is then connected to an object detection model, which recognizes the ships in the image. We anticipate that the bounding box in which a ship is spotted from a picture will be almost perfect.

**References**

* PRIIT ULMAS1, Tallinn University of Technology, Akadeemia Tee. “Segmentation of Satellite Imagery using U-Net Models for Land Cover Classification”
  + [**https://arxiv.org/pdf/2003.02899.pdf**](https://arxiv.org/pdf/2003.02899.pdf)
* Roger Xu Jiang, “Neural network for satellite image segmentation”
  + [**https://towardsdatascience.com/dstl-satellite-imagery-contest-on-kaggle-2f3ef7b8ac40**](https://towardsdatascience.com/dstl-satellite-imagery-contest-on-kaggle-2f3ef7b8ac40)
* J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, and A.W.M. Smeulders, University of Trento, Italy, University of Amsterdam, the Netherlands. “Selective Search for Object Recognition”
  + [**http://www.huppelen.nl/publications/selectiveSearchDraft.pdf**](http://www.huppelen.nl/publications/selectiveSearchDraft.pdf)
* Voodan, “Satellite Image Segmentation: a Workflow with U-Net”
  + [**https://medium.com/vooban-ai/satellite-image-segmentation-a-workflow-with-u-net-7ff992b2a56e**](https://medium.com/vooban-ai/satellite-image-segmentation-a-workflow-with-u-net-7ff992b2a56e)
* Hannah Peterson, “A Beginner’s Guide to Segmentation in Satellite Images”
  + [**https://medium.com/gsi-technology/a-beginners-guide-to-segmentation-in-satellite-images-9c00d2028d52**](https://medium.com/gsi-technology/a-beginners-guide-to-segmentation-in-satellite-images-9c00d2028d52)