**Dehaze algorithm**

The dataset we are analyzing contains thousands of satellite images of ships on the sea, among which some has smog, haze or fog, making it hard for humans and computers to identify and detect the ships. To get clearer image to analyze thus decrease the error rate of our model, we need to pick out images with heavy fog and then dehaze them.

First, we need to figure out which kind of image should be dehazed. We transform the original image in RGB color space to image in HSI color space. HSI color space is a widely used color space in computer vision which describes col-our as perceived by the human visual system. The main effect of haze is the additional air light that submerges the object light on the intensity channel, which degrades the image contrast. Further, the saturation channel changes due to the different scatter coefﬁcients dependent on the wavelength, i.e. the color distortion of the hazy images. In this paper, we use a simple criterion to decide whether this image should be dehazed: we count the ratio of the pixels whose Intensity (I) component is greater than 100, if the ratio is over 0.5, then we will implement dehazing algorithms on this image. In this step, we use the pooling technique, we will not count all pixel’s Intensity, instead, we sampled at a rate of one pixel in 100 pixels, we count the ratio of these pixel in the image whose Intensity is over 100.

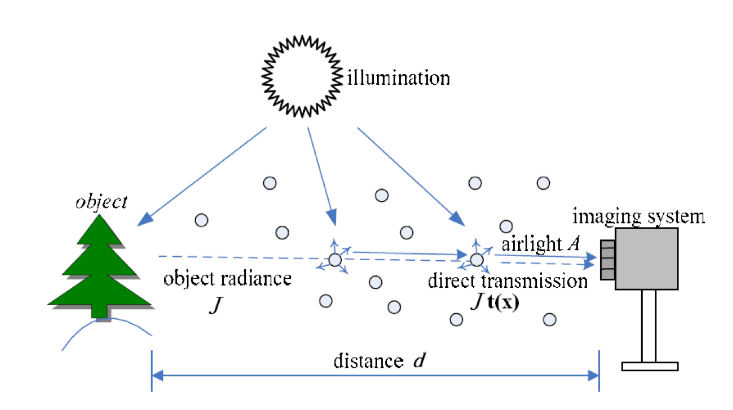
After the first step, we will then discuss the implementation of dehazing algorithms. There are two kinds of popular dehazing algorithms: algorithms based on physical model and algorithms based on supervised learning. In this project, we use Dark Channel Prior to dehaze.

First, we introduce McCartney’s model on describing the influence of fog on the original image.



，where x is the spatial coordinate of the pixel, is the image observed, is our expecting image without haze, is the atmospheric scattering coefficient, is the distance between the object and the camera, is the global constant representing the airlight. The basic idea for this physical model is deriving  based on observed image, and then solving. When the atmosphere is homogenous, the transmission can be expressed as:





He’s algorithm using dark channel is an algorithm based on physical model which perform well on many datasets. Different from most dehazing algorithms, He’s algorithm focuses on the statistical features of images without haze, it can be found that in images without haze, every local region is likely to have shadow or dark region, this statistical rule is called as dark channel prior. However, these dark regions will become gray when image has fog or haze. In addition, it is possible to use physical laws to derive the concentration of fog based on the brightness of these regions. The detailed steps of He’s dark channel dehaze algorithms are as follows:

1. Define  as the dark Channels, where is a color channel ofand  is a local patch centered at .
2. estimate the transmission simply by: ,
3. get the clean image by: 

We made some small changes to He’s algorithm so that it will perform better on our dataset. Our algorithm is required to as fast as possible and not every image in our dataset contains fog or haze, so we won’t use and check all images, instead, we decide whether images should be dehaze at first, then use He’s algorithm with , meaning these images will be completely dehazed.

**Hough Transform**

We found that lots of images in our dataset do not have ships, among which most are images of smooth and flat sea. This feature of sparsity inspired us to think up a method to identify these background images quickly so that we can increase the speed of our algorithm by reducing the size of the dataset we are going to analyze later. In this project, we use hough transform to detect the lines in the image.

The Hough transform (HT) is a feature extraction technique used in image analysis, computer vision, and digital image processing. The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in a so-called accumulator space that is explicitly constructed by the algorithm for computing the Hough transform.

The task of the project is to classify and detect ships, the most important commonalities between ships is that ships are often thin and long. We will divide all images into two groups A and B, images in group A may contain ships and they will be analyzed in the following steps while images in group B are not likely to have ships and they will be labeled as no ships. We can detect whether there are lines in the images that are longer than the threshold we set, we will attribute those images who have lines to group A while attribute images have no lines to group B. We chose adequate parameters for this preprocessing layer so that we could both decrease the error rate and removing enough background images from the original dataset.

**Future Work**

**Potential improvements**

1. Our Hough Transform (HT) used open-python library and there are few GPU packages supporting OpenCV-python, so we only ran our algorithm on CPU which may be slower than GPU. We could rewrite our program in C++, write wrapper function for these functions and run the program on GPU to improve the speed of the process.
2. The standard form of Hough transform is slow and uses too much memory especially for machine learning and computer vision application, many advanced parallel computing algorithms based on Hough Transform has been proposed, they could be executed on NVIDA GPU and could get better results. In the future, we could implement these algorithms on GPU and compare their performance to pick up the most suitable algorithm.
3. There are two kinds of widely used dehazing algorithms: Algorithms based on physical model (e.g., image dehazing with dark channel prior) and algorithms based on neural networks (e.g., All-in-One Dehazing Network (AOD-Net), Densely Connected Pyramid Dehazing Network (DCPDN), etc.), in this project, we only implement image dehazing algorithm based dark channel prior, we could test dehazing algorithms based on neural network in the future and compare their performance with algorithms based on dark channel prior