University of Moratuwa
Faculty of Engineering
Department of Electronic & Telecommunication Engineering
EN4353 Radar and Navigation

Assignment 1

B.Sc. Eng., Semester 8 2016 Batch

1. Cloud detection is a vital component in civilian radars.Rain clouds can have a downcraft which can be dangerous as they can generate an electrical charge. Therefore, they must be avoided by pilots. As an EN4353 student, your task is to help develop an algorithm to differentiate and visualize clouds from a raw video signal.

Description

You are given three files (***Test_ case1.csv, Test_ case2.csv, and Test_ case3.csv***), which contain the input for first three test cases. Each test case contains 18000 samples of the output present after demodulation. These samples are taken at a very high sampling rate (1080 KHz) to give you the essence of a continuous signal so you are expected to down-sample the signal to fit into **30** range slots.Your task is to use this demodulated signal to identify clouds and targets present using threshold detection and post detection integration. For threshold detection use a window size of 21 on the original samples(samples taken @ 1080KHz) to determine the average noise voltage for a particular range slot. A guard band of 4 would be appropriate in this case. Output of a few example cases are given in the proceeding sections to illustrate the procedure. In the presence of clouds identify the rough contour depicting the cloud. In the presence of a target indicate the range and the azimuth angle assuming the initial azimuth angle is zero.
(Hint: use $V_{TH} = (V_n)_{Avg} + k\sigma_n$ *; where* $k = 1$ *)*

Radar Description:

Pulse Repetition Frequency = 1800Hz
Horizontal Beam Width = 2^{o}
Number of Revs per minute = 12
$\sigma_n = 1$

Submission Details:

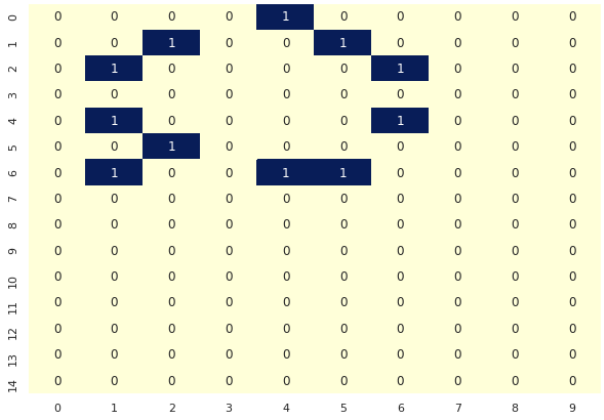
You are expected to use ***Python*** language to implement your algorithm. The details of the algorithm must be properly explained in your report along with the necessary calculations and visualization. Additionally, you are expected to submit all your codes preferably as a ***.ipynb*** with comments and figures. Include simple instructions to run your code as we may use an undisclosed set of test cases to evaluate your algorithm. A clean and a tidy code is appreciated.

Nomenclature:
We use the following encoding to classify objects,

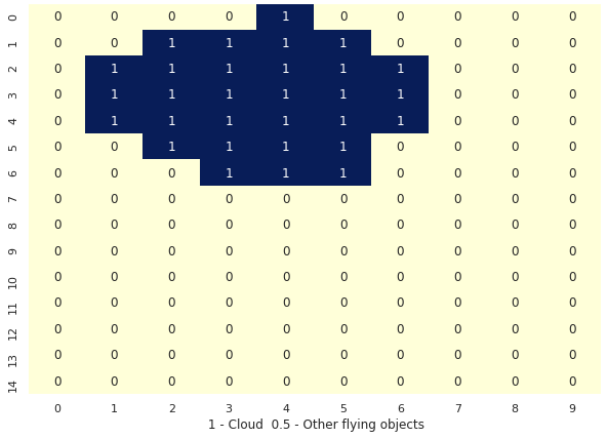
1 - Cloud
0.5 - Other objects
0 - No object

Examples:
In this section, we illustrate the expected output after processing the raw video. Please note that these examples are used to illustrate the process. The expected output for the provided test cases is different from these examples.

(a) **Example - I**
Input: ***Test_case_illustrate1.csv***;
Expected Output:



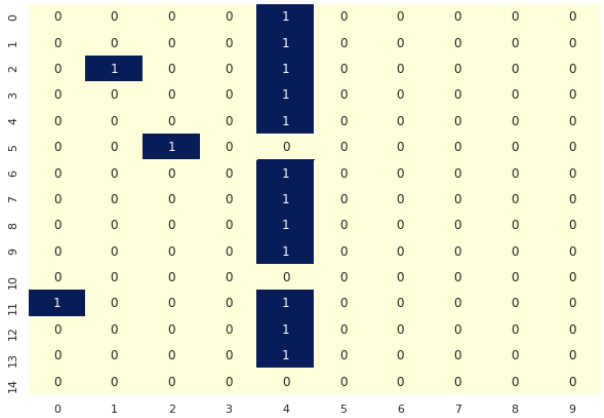
(a) After Processing the Raw Video



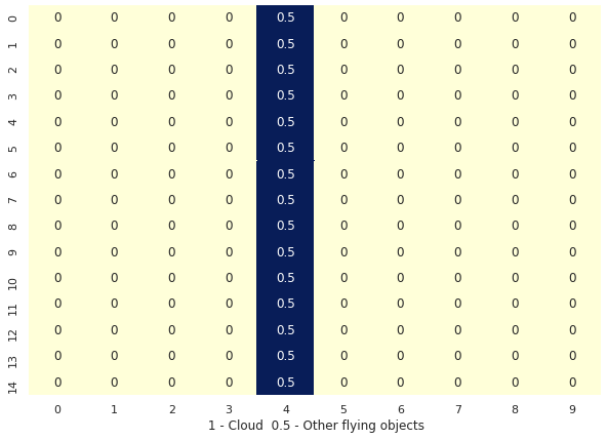
(b) After encoding (1 - cloud, 0.5 - other objects) and Noise Filtering

Figure 1: Expected Output for Test Case - I

(b) **Example - II**
Input: ***Test_case_illustrate2.csv***;
Expected Output:



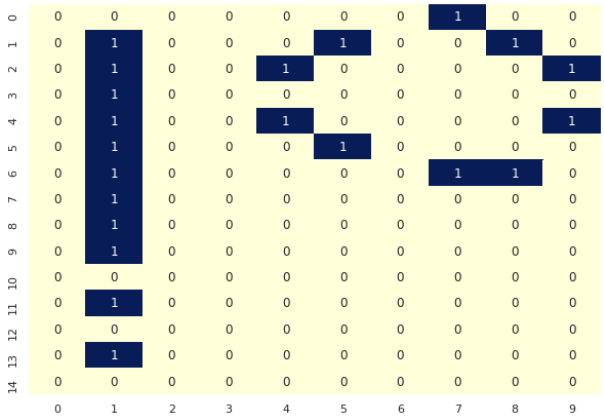
(a) After Processing the Raw Video



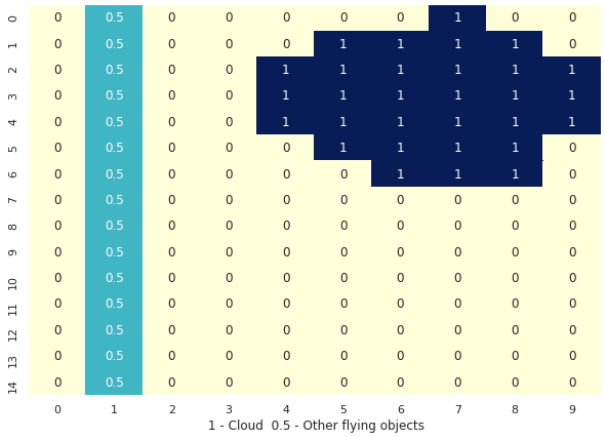
(b) After encoding (1 - cloud, 0.5 - other objects) and Noise Filtering

Figure 2: Expected Output for Test Case - II

(c) **Example - III**
Input: ***Test_case_illustrate3.csv***;
Expected Output:



(a) After Processing the Raw Video



(b) After encoding (1 - cloud, 0.5 - other objects) and Noise Filtering

Figure 3: Expected Output for Test Case - III

References

1. You can use Colab online note book.
2. You can use Numpy to handle matrices.
3. To visualize the output you can use Seaborn and Matplotlib.