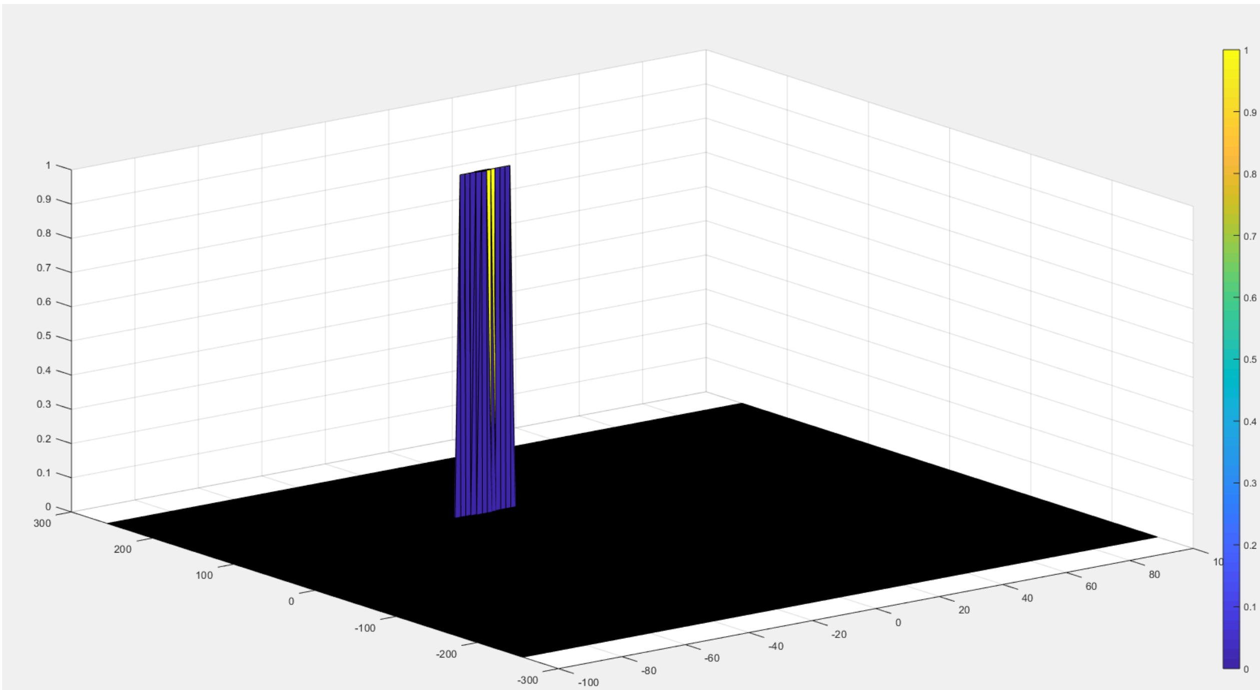
Radar Target Generation and Detection

Sensor Fusion

Gaurav Borgaonkar

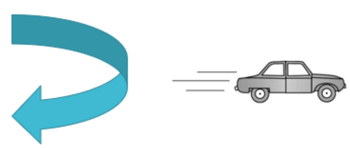


The aim of this project/code is to simulate signal generation for a Frequency Modulated Continuous Wave (FMCW) radar and using the signal to detect a target by its range and velocity using MATLAB.

Modeling Signal Propagation for the Moving Target scenario

Slope (α) = Bandwidth / Tchirp

Transmit signal (Tx) $\cos(2*\pi*(f\_c*t + \alpha*t^2/2))$



Range , time_delay = τ

Receive signal (Rx)
(time delayed version of the transmit signal)

$$\cos(2*\pi*(f\_c*(t-\tau) + \alpha*(t-\tau)^2/2))$$

Subtracting (Mixing or Dechirping) the receive signal with the transmitter signal gives the frequency shift

Tx.*Rx
(element wise matrix multiplication)

$$\cos\left(2*\pi*\left(2*\alpha*\frac{R}{c}*t + 2*f\_c*v/c*t\right)\right)$$

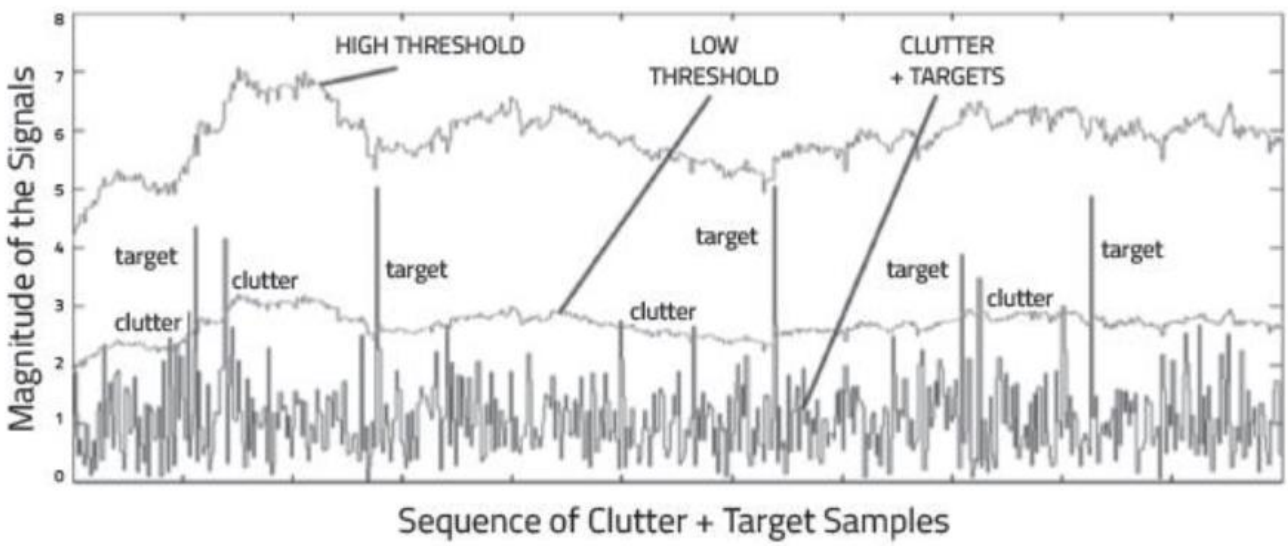
Range Doppler

False Alarm Rate

The false alarm rate is the rate of erroneous radar detections by noise or other interfering signals. It is a measure of the presence of detected radar targets when there is no valid target.

Clutter

Radar receives data not just from the vehicles or pedestrians, but also from other objects like buildings, trees, road or water and other objects. These reflections of signal are called as clutter. We need to filter these unwanted reflections from our signal in order to detect the actual target.



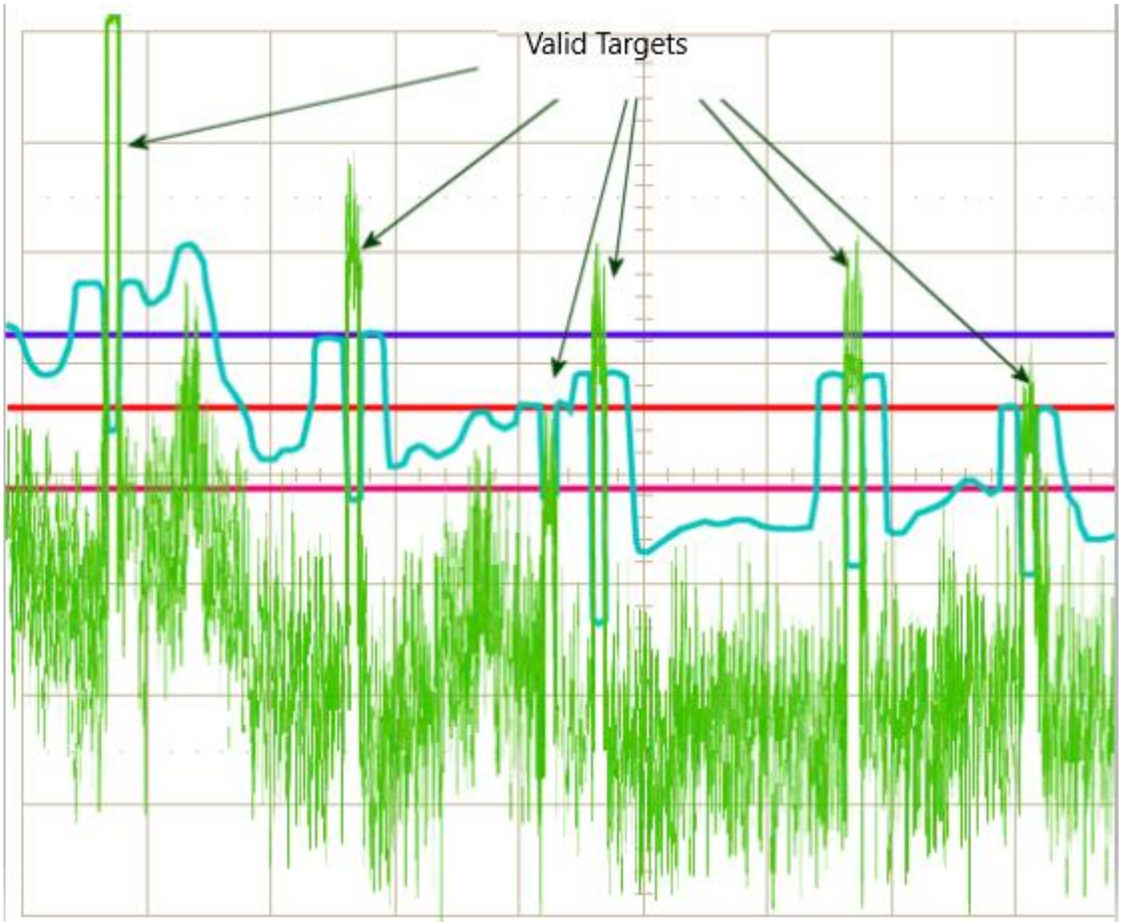
To remove the clutter from the signal, we can use thresholding. The problem with fixed thresholding is that if the detection threshold is set too high, then there will be very few false alarms, but it can also mask the real valid targets. If the threshold is set too low, it would lead to too many false alarms.

Hence, we use dynamic thresholding. It involves varying the threshold level to reduce the false alarm rate. With this technique, the noise at every level is compared to the local noise level. This comparison is used to create a threshold which holds the false alarm rate constant.

Constant False Alarm Rate

CFAR varies the detection threshold based on the vehicle surroundings. The CFAR technique estimates the level of interference in radar range and doppler cells “Training Cells” on wither or both sides of the “Cell Under Test”. The estimate is then used to decide if the target is in the Cell Under Test (CUT). Basically, it assumes that the noise is spatially or temporarily homogeneous.

In this code, we use Cell Averaging CFAR technique. (CA-CFAR)



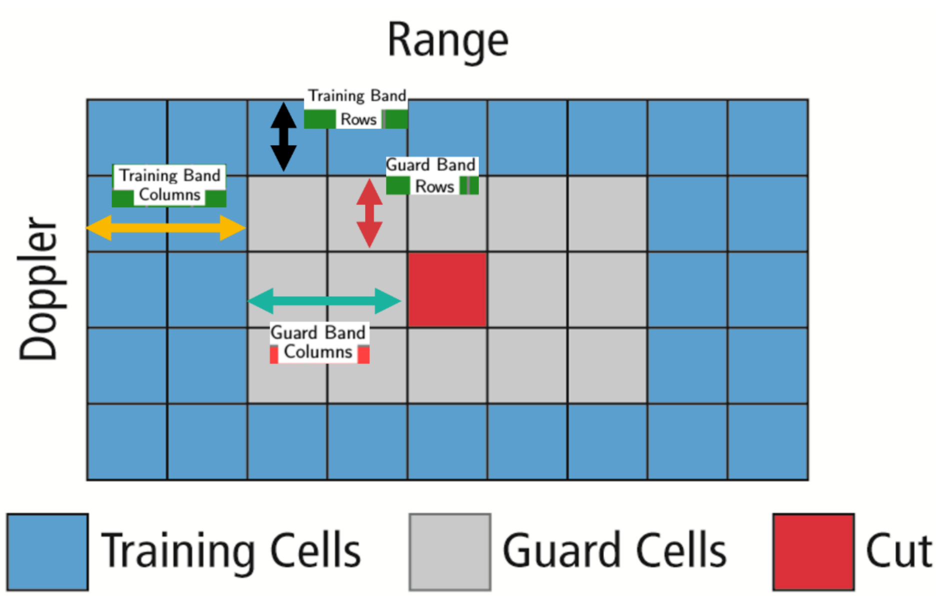
CA-CFAR

Source: http://www.radartutorial.eu

CFAR Steps

The steps for implementing CA-CFAR are as follows:

1. Determine the number of Training cells for each dimension Tr and Td. Similarly, pick the number of guard cells Gr and Gd.
2. Slide the Cell Under Test (CUT) across the complete cell matrix
3. Select the grid that includes the training, guard and test cells. Grid Size = (2Tr+2Gr+1) (2Td+2Gd+1).
4. The total number of cells in the guard region and cell under test. (2Gr+1) (2Gd+1).
5. This gives the Training Cells: (2Tr+2Gr+1) (2Td+2Gd+1) - (2Gr+1) (2Gd+1)
6. Measure and average the noise across all the training cells. This gives the threshold
7. Add the offset (if signal strength in dB) to the threshold to keep the false alarm to the minimum.
8. Determine the signal level at the Cell Under Test.
9. If the CUT signal level is greater than the Threshold, assign a value of 1, else equate it to zero.
10. Since the cell under test are not located at the edges, due to the training cells occupying the edges, we suppress the edges to zero. Any cell value that is neither 1 nor a 0, assign it a zero.



We tried using several different values for training and guard cells for detecting the signal and suppressing noise. Finally, we came up with the following parameters –

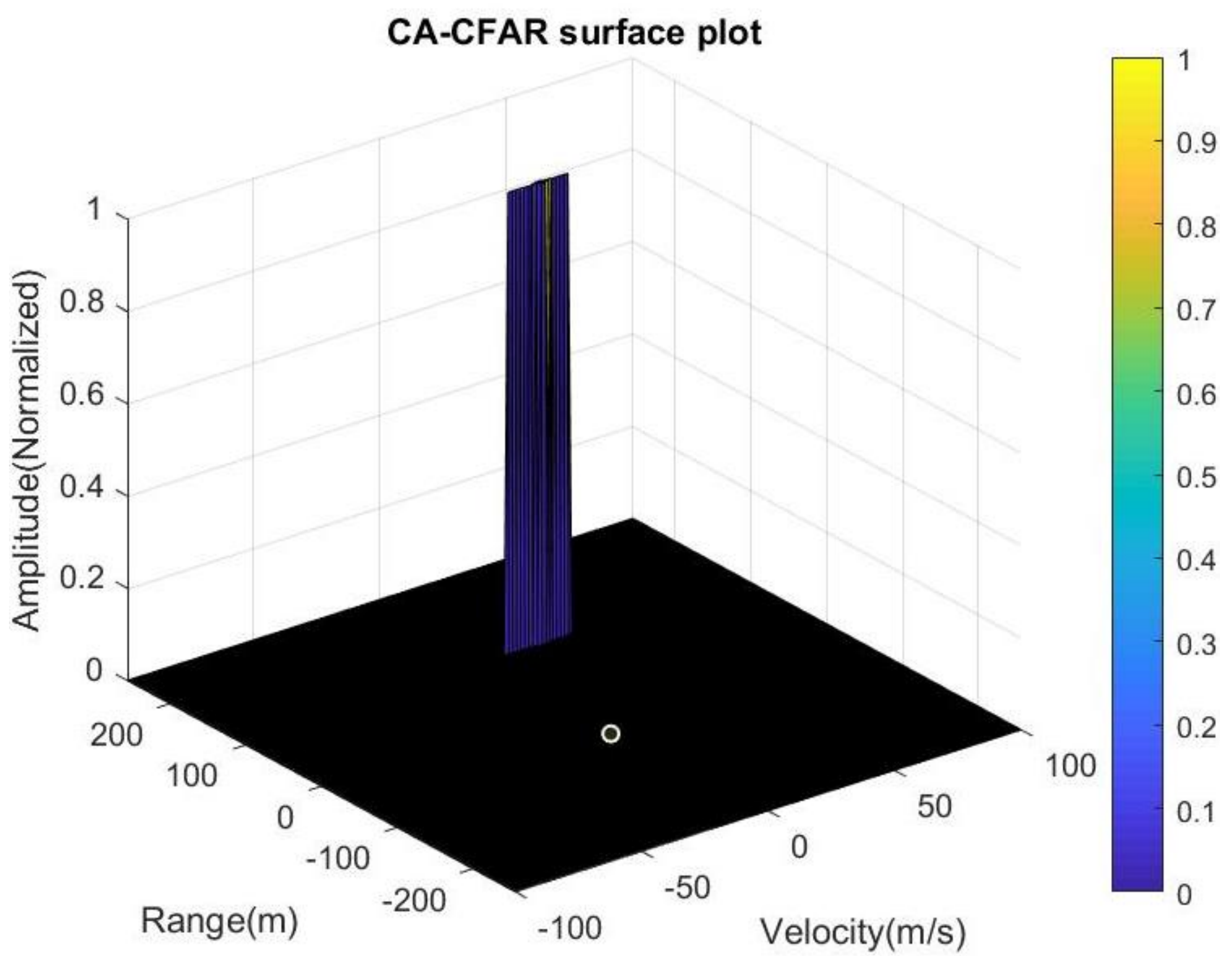
Training cells in Range dimension	Tr	50
Training Cells in Doppler dimension	Td	25
Guard Cells in Range dimension	Gr	5
Guard Cells in Doppler dimension	Gd	5
Offset	Offset	10

Code implementation –

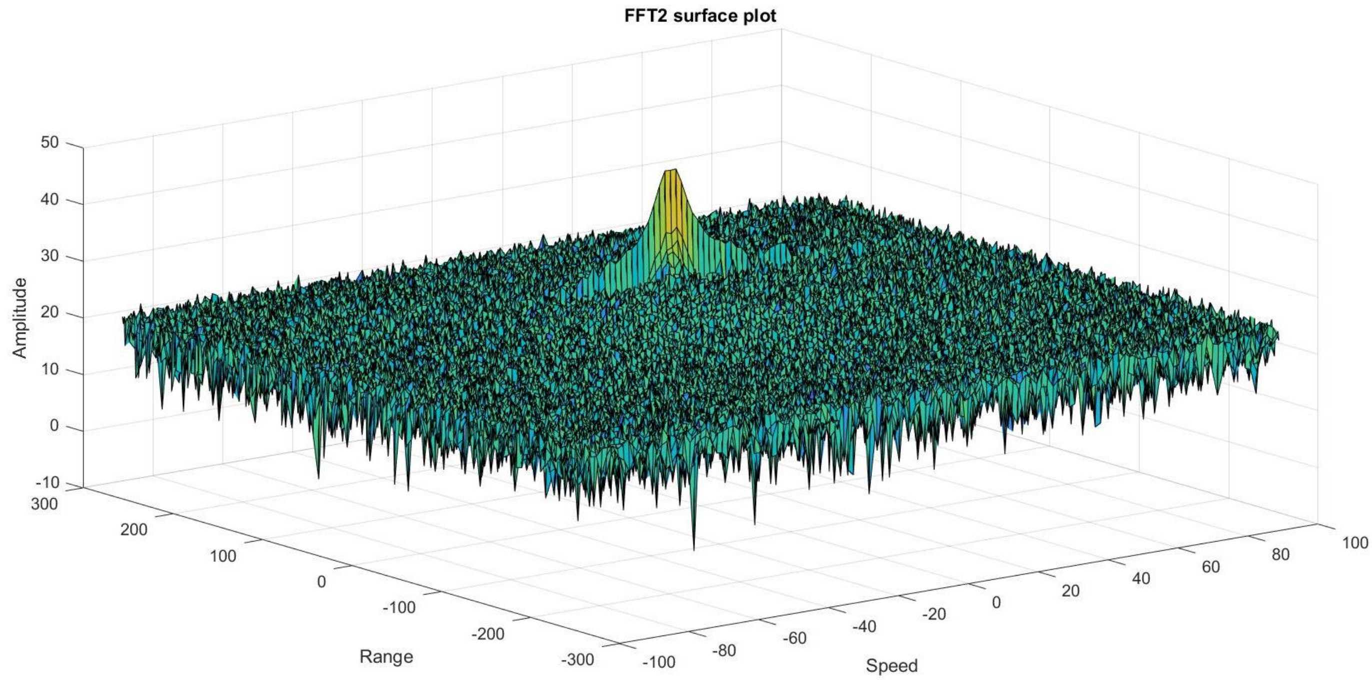
- We use for loops for iterating through the cells by selecting a block of cells, first across the range dimension and then across the doppler dimension
- For every loop, we initialize total noise level to zero and add the noise level of all training cells together to calculate average noise using the formulae given above, this gives us the average
- Then we add offset value to the threshold to keep false alarm rate minimum
- We check if the cell under test has a value higher than the threshold. If yes, we assign it a value of 1, else value of zero.
- Finally, along the edges, we suppress all the cells to zero if they do not have a value that is either 1 or zero. This step leaves us with only the target.

The plots are obtained for a velocity of 20m/s and range of 100 m. Similar plots can be obtained for different velocities and ranges. Here are the plots obtained-

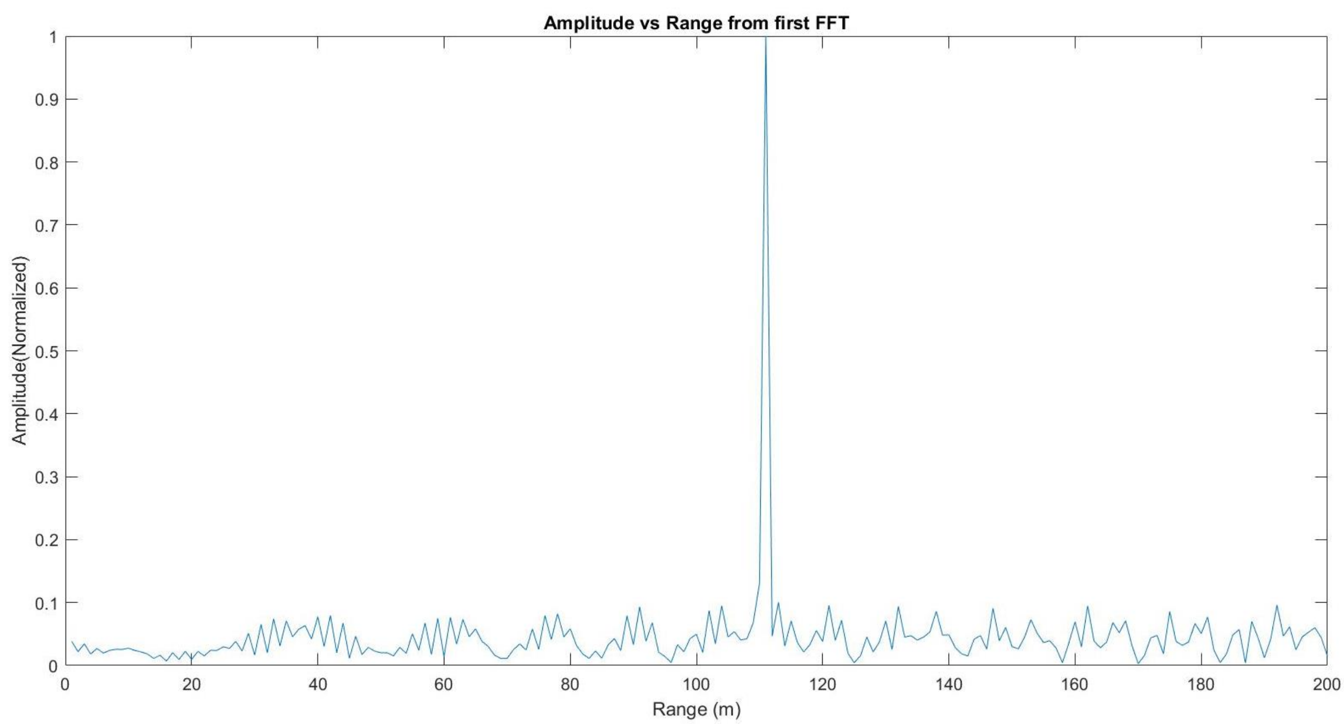
1. **CA-CFAR plot**



2. FFT2 2d plot



3. Range plot (Amplitude vs Range)



Rubric points:

Task No	Task	Specifications and Explanation
FMCW Waveform Design	Using the given system requirements, design a FMCW waveform. Find its Bandwidth (B), chirp time (Tchirp) and slope of the chirp.	For given system requirements the calculated slope should be around 2e13. **The calculated slope is 2.0455e+13 and is printed on the screen as output.**
Simulation Loop	Simulate Target movement and calculate the beat or mixed signal for every timestamp.	A beat signal should be generated such that once range FFT implemented, it gives the correct range i.e. the initial position of target assigned with an error margin of +/- 10 meters. **The code correctly predicts the range of the target within margin of +/- 2 meters.**
Range FFT	Implement the Range FFT on the Beat or Mixed Signal and plot the result.	A correct implementation should generate a peak at the correct range, i.e. the initial position of target assigned with an error margin of +/- 10 meters. **The code generates a peak at the correct range value of the target within margin of +/- 5 meters.**
Doppler FFT	Implement the 2D CFAR process on the output of 2D FFT operation, i.e. the Range Doppler Map.	The 2D CFAR processing should be able to suppress the noise and separate the target signal. The output should match the image shared in walkthrough. **See images**
2D Constant False Alarm Rate (CFAR)	Create a CFAR README File	In a README file, write brief explanations for the following: • Implementation steps for the 2D CFAR process. • Selection of Training, Guard cells and offset. • Steps taken to suppress the non-threshold cells at the edges. **Images plotted**