Readme 2D-CFAR

Implementation steps for the 2D CFAR process

What is CFAR?

Technically speaking, the most often utilized CFAR detection method is the Cell Averaging CFAR (CA-CFAR) approach. Based on the surroundings of the vehicle, CFAR adjusts the detection threshold. The CFAR method calculates the amount of interference in the "Training Cells" (also known as the radar range and doppler cells) on either or both sides of the "Cell Under Test". The target's location in the Cell Under Test (CUT) is then determined using the estimate. The procedure iterates through each range cell and determines if a target is present based on an estimate of the amount of noise. The basis of the process is that when noise is present, the cells around the cell of interest will contain a good estimate of the noise, i.e., it assumes that the noise or interference is spatially or temporarily homogeneous. Theoretically it will produce a constant false alarm rate, which is independent of the noise or clutter level.

Implementation Steps

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 2D CA-CFAR implementation involves the training cells occupying the cells surrounding the cell under test with a guard grid in between to prevent the impact of a target signal on the noise estimate.

1. Determine the number of Training cells for each dimension Tr and Td. Similarly, pick the number of guard cells Gr and Gd.
   - Tr: Number of Training Cells = 10
   - Td: Number of doppler cells = 8
   - Gr: Number of Guard Cells = 4
   - Gd: Number of Guard doppler cells = 4
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 in signal strength in dB) to the threshold to keep the false alarm to the minimum.
   - offset=1.2: Adding room above noise threshold for desired SNR
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.

suppress the non-thresholded cells at the edges.

2D CFAR suppress the noise and separate the target signal: 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.

```
RDM (RDM~=0 & RDM~=1) = 0;
```

Reference

- FMCW Radar
- FAR
- Radar Sliding Window Detector
- 2D FFT
- 2D CFAR
- https://github.com/tooth2/2D-CFAR/blob/main/radar-target-generation-and-detection.m
- https://github.com/ken-power/SensorFusionND-Radar/blob/main/src/radar_target_generation_and_detection.m
- https://github.com/tooth2/2D-CFAR/blob/main/radar-target-generation-and-detection.m