ECE278C Imaging Systems

Lab 6

March 18, 2021

Student:	Ivan Arevalo
Perm Number:	5613567
Email:	ifa@ucsb.com

Department of Electrical and Computer Engineering, UCSB

0 Introduction

The main objective of this report is to reconstruct the image from GPR gathered data. We will focus on reconstructing by summing up the range profiles at each tranceiver location.

1 Procedure

In order to reconstruct our image, we'll generate the range profile from each transceiver location. To do so, we take the N-point FFT of the 128-point wave-field sample sequence g(n) at each transceiver position. In order to get better resolution, I decided to take the 1012 FFT.

Next, we need to generate the sub-image that's spanned from the range profile at each transceiver location. For each pixel of the sub-image array, calculate the distance r to each transceiver location and determine the value of each pixel by looking for the value of the corresponding FFT spectral bin. For locations where the distance was not an integer (in order to index the FFT vector), I interpolated the FFT value by averaging the FFT values of the 2 closest indices. Finally, I multiplied the value of each pixel by $exp(-j2fo(2r)/v)$ to finalize the sub-image array.

Last step is to iteratively generate and superimpose each recreated sub-image from each of the 200 transceiver locations.

2 Results

We will reconstruct our image by range estimation. The data set was taken over the walkway pavement in front of the Broida Hall at UCSB. The ground-penetrating radar imaging unit scanned along a linear path and took data at 200 spatial positions. The spatial spacing between the data-collection positions is 0.0213 m (2.13 cm).

At each data-collection position, the system illuminates the walkway pavement with microwaves in the step-frequency mode, stepping through 128 frequencies with a constant increment, from 0.976 GHz to 2.00 GHz. The relative permittivity ϵ_r is approximately 6.0.

For simplicity, we'll use the phase only version of green's theorem. Furthermore, we need to adjust the velocity of wave propagation used in the calculation with the relative permittivity provided.

$$v = \frac{C_0}{\sqrt{\epsilon_r}}$$

Where C_0 is the speed of light. Now we just need to iteratively reconstruct and add the images from each calculated range profile at each receiver location. Figure 3 shows the final reconstructed image.

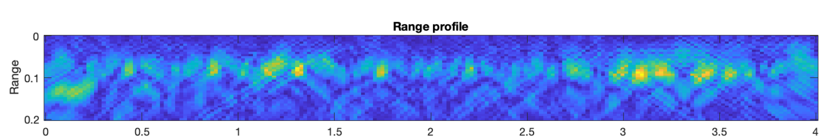

Figure 1: Reconstructed Image

3 Summary

Observing the reconstruction video https://youtu.be/QeEPC1cF8Js, we can observe how the reconstructed image gets generated from left to right as we add the sub-images from each transceiver location.

4 Code

```
close all;
quadrature_data = load('/Users/ivanarevalomac/Documents/MATLAB/Range
    Estimation/gpr_data.mat');
B = 1.024e9;
N = 128;
v = 3e8/sqrt(6);
n = linspace(1, 128, 128);
R = n.*v/(2*B)';


subim = zeros(40, 200);

freq = linspace(quadrature_data.f(1), quadrature_data.f(end), 1012);

vid = VideoWriter('newfile.avi');
open(vid)

for i = 1:size(quadrature_data.F,2)
    fft_func = (fft(conj(quadrature_data.F(:,i)), 1012));

    for y = 1:size(subim,1)
        for x = 1:size(subim,2)
            r = sqrt((i-x)^2 + y^2);
%             r = (r * 0.0213)+1;

            pxl_val = 0.5 * (fft_func(floor(r)) + fft_func(ceil(r)));

            f0 = 0.5 * (freq(floor(r)) + freq(ceil(r)) );

            pxl_val = pxl_val * exp(-1i*2*pi*f0*(2*r*0.0213)/v);
            subim(y,x) = subim(y,x) + pxl_val;


        end
    end

    frame = abs(subim) / max(abs(subim),[],'all');
    writeVideo(vid,frame);

%     if mod(i,25) == 0
%         image = imagesc(abs(subim));
%         title('Range profile');
%         xlabel('Position');
%         ylabel('Range');
```

```
%         pause(2)
%     end

end

close(vid)

image = imagesc([0,4], [0,0.2],abs(subim));
title('Range profile');
xlabel('Position');
ylabel('Range');
pause(2)
```