UNIVERSITY OF BATH
Department of Electronic
& Electrical Engineering

EE40054 Digital Image Processing

Assignment 2024

Noise Reduction of Synthetic Aperture Radar (SAR) and Ultrasound Images

Introduction

In remote sensing applications, scientists and geologists often want to identify features such as land/sea boundaries or forest borders for land-use monitoring or scientific purposes. SAR imaging systems provide a high spatial resolution and are unaffected by adverse weather conditions.

Similarly, in medical imaging radiographers and physicians often wish to identify the boundaries of anatomical features within images. Once this has been achieved measurements such as the size a foetus' head or the cross-sectional area of the left ventricle in the heart can be determined for diagnostic or monitoring purposes. Ultrasound is often used as the imaging medium as it is non-invasive, safe and cheap.

Both ultrasound and SAR images are formed using a coherent source and, as such, are characterised by multiplicative speckle noise that reduces the image contrast and obscures fine details. They therefore present a challenge to those digital image processing techniques whose aim is to automatically identify edges and features within the images.

Aims and Objectives

The aim of this assignment is to develop your understanding and appreciation of some of the techniques introduced during the unit through their application to real world problems, drawn from the areas of remote sensing and medical imaging. In addition, image processing is by nature a practical subject and the augmentation of the underlying theory by some hands-on experience should provide a deeper insight into the subject and help develop skills/experience that will be of benefit when you leave university.

Getting Going

The images for this assignment, nzjers1 (a jers1 SAR image of a headland in the Marlborough Sounds, New Zealand) and foetus (an ultrasound image of a feotus), in jpg or pgm formats can be downloaded from the Moodle course for the unit, which can be accessed at

https://moodle.bath.ac.uk/course/view.php?id=460

The page also contains c and java utilities for opening and writing Portable Graymap (.pgm) format images and links to additional files and information (Matlab also has inbuilt function for reading and writing images: imread and imwrite).

It is up to you to decide what software language you use to develop your algorithms. However, the preferred/recommended options are c, python and Matlab. Other languages will not receive the same level of lecturer support!

The Assignment

The overall aim is to reduce the noise in the images to (1) improve their subjective appearance and (2) make them more suitable for a subsequent edge detection algorithm, which is the first step of an automatic image interpretation scheme to automatically identify the boundaries within the images. Further details are provided below.

Image filtering, to reduce the speckle noise
Both linear and nonlinear filters should be implemented and evaluated. The simplest linear filter is the mean filter but other, more advanced linear filters can also be tried. Likewise, for nonlinear filters the best starting point is the median filter. For more advanced nonlinear filters, you could start by writing a median filter kernel and then extend its performance by the inclusion of some additional structure. This may be one of the filters introduced in class or, extra mark-worthily, one from the literature. Mathematical Morphological filters can also be tried. Note that as you hopefully have access to a human visual system you can readily evaluate the results produced and people aiming for a top mark may wish to compare and contrast several different filters. The implementation of the filters will also be assessed; for example an efficient median algorithm will receive more credit than a straightforward implementation. (Matlab users please note that as Matlab is an interpreted language your efficient implementation may actually be slower!). For the filters you choose, investigate the effects of using different mask sizes, and compare the best results achieved by each of your linear and non-linear filters.

Evaluation using edge detectors (optional extra)
Another way of evaluating the performance of the filters is to apply an edge detector to the filtered images. You are not expected to code the edge detectors yourself, for example Matlab has an "edge" command that implements a variety of edge detectors (Sobel, Prewitt, Roberts, Laplacian of Gaussian and Canny) that can be used. The benefit of the filtering stage can be assessed by applying the edge detectors to the original and your best linear and nonlinear filter results.

What Am I Expected to Give In?

A report containing brief details of the techniques you have implemented, the results obtained and a discussion of their performances. Please also include a fully annotated copy of all your code, explaining how it works, what the commands used do and what the thinking behind your code was. Also provide references where necessary. No marks will be given for uncommented or plagiarised code.

When marking, I will give credit for the complexity of the methods attempted, the quality of the implementation (including efficient of the coding), the results achieved and discussion.

The work is individual and your submission must consist of a single pdf file with all text and code in a computer-readable format.

The assignment is categorised as Generative AI Type A Use of GenAI is not permitted. The rationale for this is that one of the main aims of the coursework is to give you an understanding of the operation of various linear and nonlinear filters and this can be achieved by programming them at a low level. This understanding can then be used as a basis for implementing more advanced filters or efficient implementations (which GenAI cannot do). Similarly, many of the standard filters are available as functions in Matlab or OpenCV and while you are welcome to use Matlab or OpenCV functions as a comparison to verify that your implementations are performing correctly, just using Matlab/OpenCV functions by themselves will receive little or no credit.

Deadline and Feedback

This assignment should be submitted to the Moodle portal by 4:00pm on Thursday 28 November 2024. Provisional marks and feedback will be available on or before 9 Jan 2025. The feedback on this assignment will consist of an individual feedback paragraph, comments on your report, in addition to generic feedback. The feedback be available through Moodle and will potentially be useful for your final examination.

I hope you enjoy the exercise
Adrian Evans, October 2024

EE40054 Digital Image Processing Assignment Grade Descriptors

General Guidelines

A formal technical report is not required. Just briefly mention what the filters you have implemented are supposed to do (no need to reproduce lots of theory from my notes/books), how you have implemented them (extracts of well commented code from the appendix could be included in the main report for this), show results over a range of mask sizes and say which you think is best (and why). Final conclusions should compare the best results from different filters. Your convolution only needs to be described once, as it is the same for all filters. Appendices containing your code should be included but no need to repeat code for convolution, image display etc.

The most popular two grade descriptors are provided below: First Class (What do I need to do to get a high mark?) and Third Class (What is the bare minimum I need to do to pass?). Please note that the assignments will be positively marked, meaning that marks will be awarded for the work you have done rather than being subtracted for things not done and so the First Class Descriptors provide guidelines rather than an exhaustive list – any student who does something relevant to the aims of the assignment will receive credit for it.

First Class >=70%

A good range (minimum of 2/3 linear and 2/3+ nonlinear) of filters implemented using Matlab function files or functions/classes/methods in another language. At least one of the nonlinear filters should be more advanced, for example the truncated median or adaptive weighted median. Some attempt at an efficient implementation of a median filter, or an implementation of additional non-linear filters from the literature included. A thorough evaluation using appropriate parameters for both the nzjers1 and the ultrasound test image presented and conclusions drawn about the performance of each filter. Final discussion and conclusions as to which filter performs best provided. Code clearly presented and generally correct and well commented.

Third Class 40% - 50%

Mean and median filters implemented using Matlab script or other simple language. The convolution should be implemented using loops but the mean and median operations could use Matlab's inbuilt commands. Results evaluated on the jers1 test image using a range of mask sizes, for example 3×3, 5×5, 7×7 and 9×9, and conclusions about the best performing techniques and mask sizes drawn. Code mostly correct with some comments.