Evaluation of various techniques, and execution times, of different filtering systems

Alexander O’Callaghan

Acknowledgements

# Contents Page

Contents

[Contents Page 3](#_Toc39350264)

[1 Introduction (10%) (800 – 1000) 4](#_Toc39350265)

[1.1 Background 4](#_Toc39350266)

[1.2 Focus 4](#_Toc39350267)

[1.3 Value 4](#_Toc39350268)

[2 Literature review (30%) (2400 – 3000) 5](#_Toc39350269)

[2.1 Introduction 5](#_Toc39350270)

[3 Methodology (15%) (1200 – 1500) 6](#_Toc39350271)

[3.1 Introduction 6](#_Toc39350272)

[3.2 Code Breakdown 6](#_Toc39350273)

[4 Findings/Results (5%) (400 – 500) 7](#_Toc39350274)

[5 Discussion (30%) (2400 – 3000) 8](#_Toc39350275)

[5.1 Introduction 8](#_Toc39350276)

[6 Conclusion (10%) (800 – 1000) 9](#_Toc39350277)

[6.1 Wrap up 9](#_Toc39350278)

[6.2 Future work 9](#_Toc39350279)

[References 10](#_Toc39350280)

# 1 Introduction (10%) (800 – 1000)

## 1.1 Background

Digital signal processing (DSP) is a way of using computer code to make a series of operations on signal data with the intention of creating data that is more useful in some manner, to the user. This can range from <<CITE SOMETHING>>. The main advantage of using software to do this is that designs can be created very easily, cheaply, and more effectively than creating similar designs in hardware. <<compare hardware to software reference>>. This is because modifying design in software can be done very quickly and tested on the fly. Hardware components must be physically altered to change processing aspects and rely on each part working correctly. This introduces more points of failure to the system.

## 1.2 Focus

The focus of this project is to make comparisons between different filtering techniques that can be used to increase correlation between audio samples and their <<dirty>> counterparts. The <<Dirty>> samples will be filtered using different methods and the performance will be examined. While it may be difficult to directly compare the results should indicate certain advantages and disadvantages in certain aspects.

QUESTIONS: Which filtering technique can produce the best results, which is faster etc.

## 1.3 Value

The scope of this project focuses on examples where audio files are used. Though that alone could provide useful, the scope for filtering techniques can extend to a large range of different signal mediums. The results found here can contribute to the wider field of Digital Signal Processing in which the benefits of which can have far reaching effects.

# 2 Literature review (30%) (2400 – 3000)

Basics of dsp (high level overview), why filtering, various filtering techniques (FIR vs IIR), various fir filters

Analogue vs digital

DSP

## 2.1 Introduction

# 3 Methodology (15%) (1200 – 1500)

Experimental, get signal from public repository, apply filter, convolution, analyse results,

Briefly explain code that compares and justify work so far. Then talk about implementation of filters and how I assessed worth.

## 3.1 Introduction

For the project, Matlab was used to process the audio files and filter them. 5 audio files were picked from <<Website>> to use.

The source control on the project was Git which was hosted on Github.com and was used throughout. <<(git/source control is good reference?)>>

Audio files (https://www.bensound.com/royalty-free-music - Accessed 18/02/2020)

Matlab (flow chart?)

Github

Computer/matlab setup

Correlation of file to file

Correlation before filtering

## 3.2 Code Breakdown

The work of the project was completed entirely in MatLab and the following is an overview of the code itself.

The original audio files were cut down to a length of 30 seconds to reduce computational times while in the prototyping phase.

Both the cut down, and recorded files, were imported into the program on a loop to keep them in separate matrixes. During the import, the two audio channels we combined into a singular value of left and right to make an average. The sampling frequencies were also changed to make both sets the same.

Next a loop ran through each recorded file and found the correlation of it against each of the original files <<using this MatLab resource>>. The method used found at which point the two files were most similar and compared the recorded file against the original at that point. On some files that did not match, this point was at a moment in time that did not allow for a full comparison. As this only happened on files that did not match, marking these comparisons as Null was acceptable and could be dismissed in the analysis.

The matrix was then converted into a table for easier viewing and the file names were added in the columns to further aid analysis.

While the code was only taking two sets of five audio files, the code could easily run with any amount of files to compare against.

This was how the baseline was created and these figures could then be used when comparing against the filtered results.

For the filtering, <<matlab link>> was a useful resource in designing the filters.

<<designing filters>>

Loop through different filters, probably use methods?

# 4 Findings/Results (5%) (400 – 500)

Pictures of result tables

# 5 Discussion (30%) (2400 – 3000)

## 5.1 Introduction

The findings previously exhibited can

# 6 Conclusion (10%) (800 – 1000)

## 6.1 Wrap up

As is the case in much work, the accuracy of these results can be improved by increasing the number of samples tested against.

## 6.2 Future work

Future work and extensions of this project should look to change the type of data being used. A suitable example would be the medical field of study. Implementing filtering techniques to data such as MRI, ultrasound or other similar signal-based areas could create beneficial tools to medical professionals. By processing MRI data, for example, with filters that specifically determine tumour sizes in a patient could aid a doctor’s decision making prosses.

# References