DSP Project

Distortion correction and signal restoration

This document provides an outline for a Matlab project to be completed by the end of the course. You are expected to investigate, in detail, methods related to solving the problem. There is a substantial design element to the project, and a quantitative evaluation of the performance of the proposed methods must be performed and presented. You are to write up a comprehensive report (of no more than 8 pages) describing your method and results. You should work in groups of two, although you may work alone if you really want to.

If you wish to propose a project of your own, then please come and talk to me.

Note that this document is still in preparation, and may be added to during the course of the project.

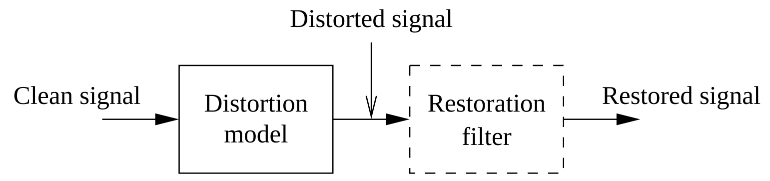
The Problem

Signals often get corrupted by unknown processes. For example, an analogue speech signal being transmitted over a communications channel often gets distorted, and consequently sounds different at the receiver than it did at the transmitter.

If we can characterise the distortion, then we should be able to restore the signal, at least partially. This project aims to explore some of the methods relating to this task.

One approach to this problem is to characterise the distortion by modelling it — the model naturally depends on the type of distortion we are expecting. A reasonable initial approach is to assume that the process causing the distortion can be represented by a linear

time-invariant filter, with unknown (or partially unknown) parameters. If we have some samples of original and distorted signals available, we can use them to estimate these parameters and thus obtain a distortion model (in a so-called training or learning stage). A restoration filter that inverts the effects of the distortion as far as possible can then be developed.



A simpler approach may be to design the restoration filter directly. That is, we can try to design a filter that takes in distorted signals and produces cleaned signals. We can again use training samples to estimate this filter, choosing filter parameters that minimise the difference between the clean signal and restored signal (at the filter output) for a given distorted signal. Thus we design the inverse filter directly.

Depending on the application, we may choose to implement the restoration filter in either the time or frequency domain. Also, in some cases it may be reasonable to process the data offline, so that causality of the filter is not a requirement.

The task

A number of examples of distorted sounds can be found on the course website (obtained from the site of Dan Ellis at Columbia). Original undistorted sounds are also available, so it should be possible to estimate the parameters of a distortion (or restoration) model.

The examples include three different types of distortion (signal filtered by unknown LTI filter, signal corrupted by additive noise, and signal corrupted by reverberation or echoing). There are also multiple samples of the same type of distortion, but with different parameters.

You are to investigate the problem of restoring distorted sounds.

Analysis

You are required to produce a quantified performance analysis for the method (or methods) developed. Admittedly this is tricky — the samples are all sounds, but you can't put sounds into a report (and I don't want a CD submitted with the report). An acceptible approach is to define some kind of performance metrics that you think are appropriate, and quantify behaviour in terms of these measures. RMS difference between the original and the reconstructed signal, for example, is a common (albeit incomplete) performance indicator.

The dataset that you are given is quite comprehensive, and it should be possible to quite effectively explore any proposed solution. In order to do this, you should keep in mind the principles of sound experimental method. The goal is to extract the maximum amount of information from the dataset, while not reaching any conclusions that are not supported by enough evidence.

For example, for each distortion type and level, you are given six samples of both the original and corrupted signals. It is not fair to use all of these samples to estimate the paramters of a reconstruction filter, and then to quantify the performance of your method by applying these same signals to your filter. In that case you would be testing the system on the same data that was used to train it. A possible approach is to train on half of the data and test on the remaining data (hold-out). A better approach is to train on half of the samples and test on the other half, then swap the sets and repeat (train on second half and test on first half), finally averaging the resulting performance (cross-validation). A still better approach is to train on five of the samples and test on the remaining sample, then to repeat over all other equivalent combinations, averaging the result (hold-one-out).

You are expected to provide insight into the problem and your method

of solution. Ideally you should be able to evaluate the range of applicability of your solution. What are the conditions under which the proposed method will work? Computational complexity is always an issue – how expensive is your algorithm in this regard?