# Karaoke Speech Signal Isolation EE392A

Ramyata Pate, 180592 Shivam Tulsyan, 180723
ramyata@iitk.ac.in shivtuls@iitk.ac.in

IIT Kanpur

***Abstract*—The separation of two overlapping audio signals can be carried out in quite a few ways. Here, we have worked on the separation of a voice signal from a music signal. The input music signal is considerably different from the music signal which adds to the voice signal and is recorded. This is the effect of factors like time shift, amplitude attenuation.This paper discusses methods using which we can calculate the music signal which needs to be subtracted from the combined recording to obtain the voice signal.**

***Index Terms*—Cross correlation, Attenuation, Normalization, Transfer Function, Stationary Waves**

## I. Introduction

THE aim, here, is to develop an efficient algorithm for a cross-platform application which can extract voice signals from its combination with music signals. For this we have worked on the platform- MATLAB. As we are working on a real-life problem of signal processing, the model can be concluded to be causal. We consider the factors- time-shift, amplitude attenuation and external noise addition.

### A. Problem statement

*1) Unknown signals:*

Speech signal- This input signal adds to the music signal

$$S[w], s[n]$$

Noise- A environment and machine dependent factor

$$N_o[w], n_o[n]$$

Recorded music signal

$$M_1[w], m_1[n]$$

*2) Known signals:*

Recorded audio signal- Combined music and speech signal along with external noise

$$A[w], a[n]$$

Music signal- This is the input signal stored in our system

$$M[w], m[n]$$

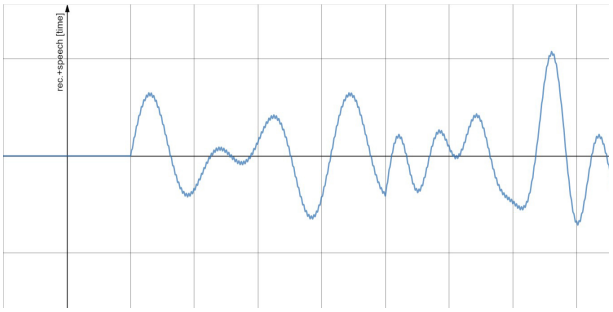


Fig. 1. Recorded audio signal in time domain

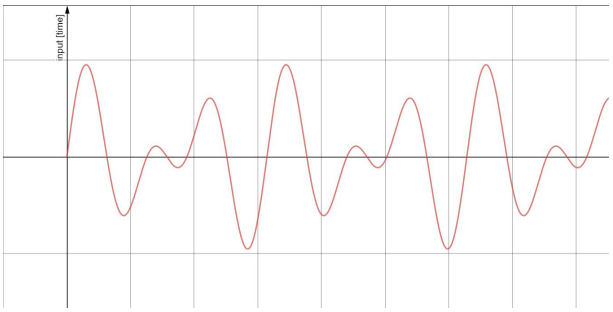


Fig. 2. Music signal in time domain

*3) Relation:* (1)

$$A[w] = S[w] + M_1[w] + N_o[w]$$

$$a[n] = s[n] + m_1[n] + n_o[n]$$

The relation between the music signal and the recorded music signal has been explored in the later section, Approaches.

### B. Already existing work

There are implementations of this feature available in Windows. But currently, there is no **cross-platform** application available which can isolate the voice signal in a robust manner. We have cross-platform applications which use frequency clipping. The audible range of human voice signal frequency is extracted using high band and low band pass filters.This serves as an inefficient method as the quality of sound signal extracted gets degraded to quite a large degree in the process.

## II. Approaches

The following approaches discuss ways to find what the recorded music signal would have been, without the addition of speech signals, given the input music signal is known. 3 approaches have been discussed below. The first approach deals with signals in frequency domain whereas the second and third approach deal with signals in time domain.

### A. Transfer Function

We first calculate transfer function of the system between the original music signal and the recorded music signal. This function calculated for a given setup(machine and environment) and frequency must be constant and is calculated before starting the experiment using a test signal. Since it is more convenient to work with the transfer function in this case as compared to the impulse response, we prefer to work in the frequency domain. The transfer function is obtained by dividing the recorded test signal by the input test signal.
Input test signal -

$$T[w]$$

Recorded test signal -

$$T_1[w]$$

Transfer function - [a]

$$H[w] = T_1[w]/T[w]$$

$$M_1[w] = H[w].M[w]$$

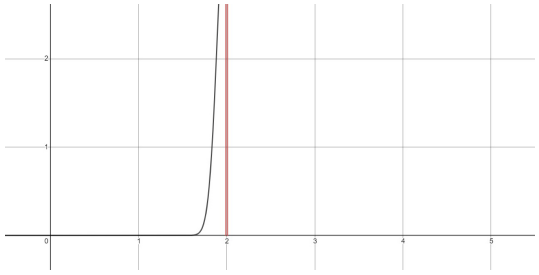

Fig. 3. Calculated Impulse-Response

### B. Cross Correlation and Normalization

The recorded music signal is time shifted and attenuated as compared to the input music signal. For this method, it is required that we play the music in absence of speech signal for a small amount of time at the beginning.

- Time Shift: To find the time shift we cross-correlate the input music signal with the music signal recorded at the beginning. The point where F(lag) has a global maximum, is the time shift of the system.
  $F(lag) = \sum_{n=-\infty}^{\infty} f_1[n] \times f_2[n + lag]$
- Attenuation: We calculate the attenuation factor by taking the ratio of maximas of the input music signal and the recorded music signal during that small time window.

Shifting the input music signal by the time shift calculated and then multiplying it with the ratio found gives us the recorded music signal

$$m_1[n] = ratio * m[n - lag]$$

### C. Assuming Stationary Waves

The assumption here is that the original input signal is stationary, i.e. the signal is periodic. We record the input music signal, for one time period, before starting the experiment. This recorded music signal is then looped and subtracted from the final audio recording.

All the 3 approaches more or less produce similar output, which can be seen in the figure below-

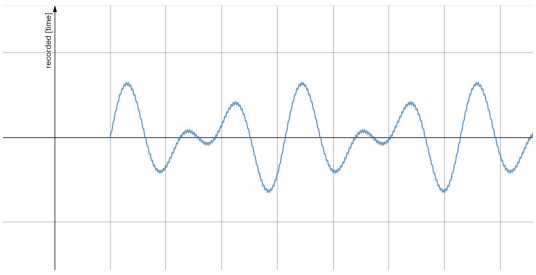


Fig. 4. Recorded music

## III. Evaluation of the application

We replace the human speech signal with some other speech signal(stored in our device) and play that in the experiment. Thus, we can compare the output of the application with the input speech signal. This way, we can test the efficiencies of the different approaches.The final output, i.e. the speech signal obtained is of the form-

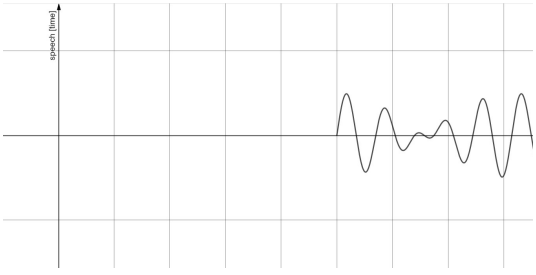


Fig. 5. Speech signal- final output

## IV. Conclusion

Using any one of the above methods, we can find the recorded music signal which is used in the relation (1). All the above discussion caters to solving this problem for recordings, i.e., post-processing these signals. The next problem is to alter these approaches to tackle the problem in real time, i.e., extracting out the required speech as soon as it gets recorded. This task must be done using the concept of STFT (Short-Time Fourier Transform). This involves making short time windows of the incoming signal and applying the same algorithm on these windows.

## Acknowledgment

We would like to thank Prof. Vipul Arora for providing us with the opportunity and necessary guidance to study and implement various aspects of digital signal processing in a real-life model.

## References

[1] Relation between input, recorded signal and noise- https://www.sciencedirect.com/science/article/pii/B9781904275268500014
[2] EE301A- Digital Signal Processing
[3] Real-time Digital Signal Processing Applications and Implementations by Sen M. Kuo, Bob H. Lee and Wenshun Tian