

1

Introduction

1.1 The Need for, and Consumers of, Sound Capture and Audio Processing Algorithms

The need for capturing sound and converting it to electric signals came with the first telephones. This is why the first microphones were designed. For a long time telecommunications was the only user of captured sound. The radio broadcasting and music recording industries increased the demand for high-quality microphones, good amplifiers, and systems for sound reproduction.

Sound capture and audio processing in general stayed in the analog signal processing domain until after World War II. At that time the first programmable digital computers were designed and researchers started to work on digital signal processing algorithms. Initially communications were the major consumer of signal processing algorithms, such as echo cancellation and digital speech compression. In the meantime, digital computers become more powerful, with more memory and faster processors. They invaded offices and homes and far exceeded their initial role as a tool for increased productivity for information workers. Modern computers are communication and entertainment centers, many of them having attached or integrated loudspeakers, microphone, and web camera. They are used for storing and playing music and videos. Programs for audio and video chat are widely used. Sound capture and audio processing algorithms today are an integral part of every personal computer. Mobile phones for the first time took the phone out of quiet rooms and exposed the microphones to substantially higher noise levels. This increased the demand for real-time implementations of noise suppression and speech enhancement algorithms running on inexpensive processors.

Automatic speech recognition had an initial task of speech dictation as the primary scenario in an office or home environment. The microphone was placed in its best

position, close to the mouth, and provided good-quality sound. With the advancement of the underlying technology, speech recognition gradually became an integral part of the human–machine interface. Speech-enabled dialog systems are deployed in mobile phones and cars and they even greet us when we dial the phone lines of many companies. They are used in a wide range of tasks: from song selection to booking hotels and plane tickets. Speech recognition emerges as the other large consumer of sound capture and audio processing algorithms.

Humans do not like to wear headsets and close-talk microphones, which drives the demand for hands-free sound capture and processing algorithms. Acoustic echo cancellation and microphone array processing are engaged to provide comfortable communication. Some emerging scenarios are speech-enabled dialog systems and voice-controlled devices – from mobile phones to the multimedia equipment in the family room.

Increasing the speed and the memory of computing devices and reducing the power consumption leads to the creation of personal devices with a small size and rich functionality. Features include voice and text communications, media player, navigation, and information access via the Internet. Owing to their small size and light weight, people carry these devices with them everywhere. Small size means a small number of keys or any other means for a rich user interface. This increases the role of speech recognition in interaction with these devices because it can provide a more convenient and natural interface. Increasing the bandwidth will allow most of the phones to have video communication features. This is the moment when the microphone is removed from its best position close to the mouth and placed an arm’s length away. Such devices will be used in practically every place, which means an increased demand for better microphones, devices, and algorithms for sound capture and processing for the needs of real-time communications and speech recognition.

1.2 Typical Sound Capture System

Capturing sounds starts with converting the acoustic wave in the air to an electrical signal by one or more microphones. These microphones can have very different characteristics and – if properly designed, selected, and positioned – can provide a substantially better sound for the next processing steps. The microphone signals are amplified, filtered, and pre-processed. An analog-to-digital convertor performs discretization and quantization and converts the sound into a stream of numbers. This is where the digital signal processing starts.

One of the first stages of the sound capturing system is the *acoustic echo reduction system*. It removes from the captured signal the sound coming from the loudspeakers – the voice from the other side in a telecommunication session, or the sound track from a CD or DVD. What is left at the end of this type of processing is the local sound – the voice we want to capture, plus some noise and reverberation.

If we have multiple microphones arranged in a device called a *microphone array*, we can combine the signals from the microphones such that the microphone array will listen in the direction of the desired sound source, suppressing the sounds coming from other directions. This process is referred to as *beamforming*. The microphone array is electronically capable of changing the listening direction and following the movements of a human speaker. To do this it employs algorithms for *beamsteering* and *sound source localization*.

The microphone array output still contains some residual noise and reverberation. The audio quality is further improved using *speech enhancement techniques* such as *noise suppression* and *de-reverberation*.

At the end of the sound capture system and audio processing chain we should have a speech signal with good enough quality for the final consumer – telecommunication or speech recognition.

1.3 The Goal of this Book and its Target Audience

The book provides a reference for most of the audio signal processing algorithms in the areas of speech enhancement, microphone array processing, sound source localization, acoustic echo reduction systems, and dereverberation. It contains information about various types of sound capturing devices. The exercises provide sample MATLAB[®] scripts and audio files to play with the algorithms, improve them, and create new ones. Most of the presented algorithms have been implemented and evaluated by the author. They are illustrated with real-life audio recordings, in scenarios close to the major application of these algorithms.

The material in the book allows for a quick building of a complete end-to-end sound capture and audio processing system with relatively good quality. This can be a good baseline for further work on improvement of some of them. Software engineers and designers working on sound capture and processing systems can benefit from this audio processing toolbox to build the initial prototype of the system with off-the-shelf algorithms, to evaluate end to end, and to focus on the processing blocks that need improvement.

The target audience includes graduate and undergraduate students. If this book can spark interest in digital signal processing and influence the decision to further study audio and signal processing it will have achieved one of its goals. It can help a grad school student quickly acquire the necessary information for the available algorithms and what they can achieve. The book provides comparisons of the existing approaches, evaluation parameters, and methodology on how to measure them.

University students, researchers, or academic staff, working on projects where design of audio processing algorithms is not the primary goal, can benefit from the book as well. They can use the presented algorithms for projects like ‘we just need a noise suppression algorithm to clean up the sound we capture during our user studies’. Many of the algorithms described in this book are directly applicable or can become a

good starting point for modification by researchers working in neighboring areas – processing of biological signals (EEG, ECG, EMG), for example.

A separate goal of this book is to shorten the distance between the audio research community and the industry. On the one hand the industry needs education and information about the state of the art in sound processing algorithms. The information should be from their own perspective: what works well in real practice and adds value to the designed product and what is a very cool research algorithm, which may open the door for further research and the creation of better algorithms, but at this point it brings little value or requires an unattainable amount of resources such as memory or CPU time. The research community needs feedback about the issues the industry faces, which will stimulate the creation of robust and practically applicable algorithms.

We want to close the gap between ‘algorithms for sound capture and processing’ and ‘algorithms for manufacturable systems for sound capture and processing’. Under the conditions of increased demand of such algorithms this will happen sooner or later, but this book will help to speed up this process.

1.4 Prerequisites

While the book starts with the basics of audio processing, an understanding of the foundations of digital signal processing (sampling theory, conversion to the frequency domain, filtering, and adaptive filters) is necessary to fully benefit from the book. Knowledge of MATLAB is required for the execution, modification, or porting of the provided sample code to another programming language. A general mathematical background (integrals, differentials, matrix operations, probability, and statistics) is needed to understand the mathematical equations and follow the algorithm evaluations. Some basic knowledge about sound as a mechanical wave in the air and how it propagates will be handy to understand the sound capturing devices part.

1.5 Book Structure

The book covers digital signal processing algorithms and devices for capturing sounds, mostly human speech, and enhancing these signals to achieve a sufficient quality for the needs of real-time communication and speech recognition. It starts with the digital sound processing basics in Chapter 2, which includes defining the noise and speech properties, definition of the basic terminology, the overlap-add processing chain for processing in the frequency domain, and some aspects of sound quality evaluation. Chapter 3 covers the sound capturing sensors: microphones of various types. The parameters of these microphones are described and models for differential and unidirectional microphones are provided. The chapter gives some ideas about what can be achieved for capturing better sound using acoustical means and decreasing the noise and reverberation at the entrance of the system. Chapter 4 is dedicated to single-channel speech enhancement. Various algorithms for gain-based

noise suppression are described and the overall architecture of a noise suppressor provided. Other speech enhancement techniques and approaches are discussed. Chapter 5 covers the microphone arrays as sound capture devices and the corresponding processing algorithms, while Chapter 6 is dedicated to using microphone arrays for sound source localization. Chapters 7 and 8 cover acoustic echo reduction systems and de-reverberation algorithms – important parts of the sound capturing system. Each chapter ends with a list of reference materials for further reading. Most of them are commented on briefly in the chapter.

Out of the scope of the book are sound processing algorithms and approaches that are computationally expensive, iterative, and in general not suitable for implementation in real-time working sound processing systems.

1.6 Exercises

Each chapter contains several exercises, which consist of writing or modifying MATLAB implementations of the discussed algorithms. A computer with MATLAB installed on it is required to execute the sample code, modify it, and evaluate the presented techniques. We chose MATLAB as one of the most common programming systems used in the signal processing community for modeling and experimenting with digital signal processing algorithms. It has a good set of mathematical functions and well-developed graphical output, which makes for ease of implementation, debugging, and illustration of waveforms and signals.

Every signal processing algorithm is an abstraction before it is implemented and evaluated with real audio signals. Implementing and experimenting with processing algorithms is the best way to learn more and understand better. We understand that for practical usage these algorithms have to be ported to a compliable programming language, such as C or C++, which is specific to each particular project.

All sample sound files are provided in a non-compressed WAV format. It is common and MATLAB reads and writes it well. The absence of compression does not introduce or mask distortions in the signals. Good headsets or high-quality loudspeakers are recommended for listening. Readers are encouraged to record and process their own audio files. To do this an additional microphone should be connected to the personal computer.

