Interfacing Speech Recognition and Vision Guided Microphone Array Technologies

by

Vibhav Shyam Rangarajan

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2003

Author . . . . . .
Department of Electrical Engineering and Computer Science
May 9, 2003

Certified by . . . . . .
Trevor Darrell
Associate Professor
Thesis Supervisor

Accepted by . . . .
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Interfacing Speech Recognition and Vision Guided Microphone Array Technologies

by

Vibhav Shyam Rangarajan

Submitted to the Department of Electrical Engineering and Computer Science
on May 9, 2003, in partial fulfillment of the
requirements for the degree of
Master of Engineering

Abstract

One goal of a pervasive computing environment is to allow the user to interact with the environment in an easy and natural manner. The use of spoken commands, as inputs to a speech recognition system, is one such way to naturally interact with the environment. In challenging acoustic environments, microphone arrays can improve the quality of the input audio signal by beamforming, or steering, to the location of the speaker of interest. The existence of multiple speakers, large interfering signals and/or reverberations or reflections in the audio signal(s) requires the use of advanced beamforming techniques which attempt to separate the target audio from the mixed signal received at the microphone array. In this thesis I present and evaluate a method of modeling reverberations as separate anechoic interfering sources emanating from fixed locations. This acoustic modelling technique allows for tracking of acoustic changes in the environment, such as those caused by speaker motion.

Thesis Supervisor: Trevor Darrell
Title: Associate Professor

Acknowledgments

I would like to thank Professor Trevor Darrell for giving me the opportunity to work on this project and for all the advice he has offered to me since I began working in the Vision Interfaces group.

I would like to thank Kevin Wilson for all his help and guidance and the wisdom he has imparted on me over the past year and a half. I surely would not have been able to do this project without his tutelage.

I would like to thank Neal Checka for his support along the way and for helping me feel welcome in the group.

I would like to thank my parents and my brother for their constant love and support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

One goal of a pervasive computing environment is to allow the user to interact with the environment in an easy and natural manner. In a pervasive computing environment, appliances and systems in the environment are completely controlled by computers. In such an environment, various types of technologies can be used to achieve such a natural interaction with the environment. The use of spoken commands is a natural way for humans to interact with each other. Voice commands, as inputs to a speech recognition system, can also be a natural way to interact with the environment. Many systems require each user to be equipped with a close-talking microphone to acquire the commands. A more compelling alternative to this problem is the use of a microphone array. Placing a microphone array in the environment allows for users to walk around and command the system without the use of any extra equipment [16].

The audio signal received at each microphone in the array is a mixture of the spoken signal(s), undesired multipath propagation phenomena such as echoes and reverberation, and background noise (e.g. fan or air-conditioning noise). Thus, the microphone array must be steered, or beamformed, in the direction of the source. Simple techniques, such as delay-and-sum beamforming, do not work well in the presence of strong interfering signals or reverberations which often arise in the real world. Thus, advanced beamforming techniques, which attempt to both steer toward the source signal and cancel out the interfering signals, are required to achieve source separation [11].

This thesis presents four different techniques for achieving source separation. The first, delay-and-sum beamforming is the simplest technique which attempts to place the source signal received at each microphone in phase with one another, thereby amplifying the source in the summed output. The second technique is adaptive beamforming, which uses knowledge of the of the signal correlations between microphones to more accurately cancel out the interfering signal. The third technique is geometric beamforming, which uses spatial information to steer towards the source location while attenuating sounds from other locations. The fourth technique, which is the focus of this thesis, is a variation of adaptive beamforming, which uses acoustic modelling to model the reverberations in the audio signal as separate, anechoic interfering sources emanating from fixed locations, which can then be attenuated.

The acoustic modelling technique is evaluated against the delay-and-sum beamformer. We examine signal-to-noise ratios and view the beam patterns created by the acoustic modelling technique and the delay-and-sum beamformer in order to evaluate the performance of the system.

Chapter two provides background information on microphone array technology and person tracking.

Chapter three describes the details and implementation of the acoustic modelling technique that I use to model the reverberations in the audio signals along with the tracking system that I have implemented.

Chapter four describes the experiments that were run to evaluate the effectiveness of both the acoustic modelling technique and the tracking subsystem, and presents the results that were gathered from these experiments.

Chapter five presents ideas for future work in this field of research and concludes.

Chapter 2

Background

The use of various techniques in array signal processing, combined with a vision-based person tracking system, enables the system to selectively amplify a speaker's voice as he/she moves throughout the room. The background relevant to these areas in discussed below.

2.1 Microphone Array Technology

There are several ways in which audio can be acquired from the scene. One existing solution is the use of a wireless, close-talking microphone. This solution produces very high quality audio, but has the disadvantage of each user needing to wear extra equipment. Another solution is to use a shotgun microphone. The shotgun microphone, however, can focus in a single direction and has the unfortunate property of amplifying both the source and any interfering noise coming from that direction. The use of a microphone array eliminates the need for extra equipment for each user and also allows for flexibility in focusing on different locations in the room through the use of beamforming techniques, which are discussed in section 2.3.

Microphone array processing is a particular type of sensor array processing, which has been studied extensively in radar and sonar applications [13]. The *Huge Microphone Array Project* involves the study of very large arrays containing hundreds of microphones [12]. This project explores the possibility of acquiring high-quality audio

from a moving source in a noisy environment, however utilizes an audio-only solution to the problem. Additionally, although increasing the number of microphones certainly increases the flexibility and accuracy of the system, developing systems with hundreds of microphones is often impractical.

Another related project involves the use of an audio-guided active camera. Target localization is achieved using the audio from a microphone array, and this information is used to steer a camera on a pan/tilt base [15].

2.1.1 Vision-guided Microphone Array Processing

There are a number of projects that use vision to produce the source localization information necessary to steer a microphone array. One such project tracks the face of the speaker of interest using a standard camcorder on a pan/tilt base [2]. Another system similarly use a single camera to capture the image of the user and track various features of the user [3]. A third system, which also employs the use of a single, moveable camera, couples the detection of moving regions of skin color in the scene with a face detector to produce the localization information [5].

2.2 Multimodal Person Tracking

In order to focus the microphone array, the location(s) of the speaker(s) of interest must be known. There are a number of techniques that exist which use only acoustic cues [14], but these techniques can perform poorly in the presence of reverberation and/or multiple sound sources [17]. The addition of a video-based tracker reduces these localization errors that result when using audio-only trackers. Although a vision-based tracker is not used in the experiments I have performed, the use of such a tracker would help improve the initialization procedure described in section 3.1.2.

2.2.1 Vision-based Person Tracking

Several person tracking algorithms have recently been developed to detect the number of people in a particular environment, and to track the 3D positions of these people over time. The algorithms used in these systems combine foreground/background classification, clustering of novel points, and trajectory estimation over time in one or more camera views [7, 9].

The vision tracker used in our test environment applies a probabilistic framework that combines audio and video to achieve a more robust and accurate tracking system for multiple objects. A particle filter is applied to track multiple people using a combination of audio and video observations. The video observations are applied to plan-view images of detected foreground points. These points are detected using a range-based background model as described in [6]. Details of the vision tracking system are presented in [4].

If the speaker of interest moves out of the field of view of the cameras or is occluded by objects in the room, or if the tracker mistakes the background for an object, the vision tracker may fail. In these cases, audio clues are used in conjunction with the vision estimate to achieve a more accurate location estimate.

Time Delay of Arrival

The time delay of arrival (TDOA) of audio signals detected at different microphones is also used to estimate the location of the speaker. This information is included in the state-space model used in the particle filtering framework. TDOA determines the location of a speaker by finding the difference in arrival time of the audio signal at different microphones in the array.

Using only the audio information for tracking may fail if the target speaker stops talking for a moment or if two speakers are talking at once, thus the video tracker is still needed to extract a good localization estimate [4].

2.3 Beamforming

The use of an array of microphones has a number of advantages over the use of a single microphone. The most important advantage is that filters can be applied at each microphone thereby allowing the system to not only steer toward the speaker of interest, but also to cancel out interfering noise sources. The group of array processing algorithms which focus the array to enhance sound from a particular direction are called beamforming techniques [8].

2.3.1 Delay-and-Sum Beamforming

The oldest and simplest beamforming technique, delay-and-sum beamforming, involves simply delaying the audio signal heard at each microphone by appropriate amounts and adding the resulting signals together. Accordingly, the signal of interest can be amplified with respect to interfering noise or other waves in the environment [8].

$$y(t, \mathbf{r}_{target}) = \sum_{i=1}^{N} a_i x_i(t + d_i(\mathbf{r})) \tag{2.1}$$

$$d_i(\mathbf{r}) = \frac{\|\mathbf{r}_{target} - \mathbf{r}_i\|}{v_s} \tag{2.2}$$

$\mathbf{r}_{target}$: *target position*

$\mathbf{r}_i$: *position of* i^{th} *microphone*

v_s : *speed of sound*

The varying distance between each microphone and the sources results in a slightly different audio signal being detected at each microphone in the array. If the input audio signal is a mixture of two sources, the portion of the signal due to one of the sources will be delayed by a different amount than the portion of the signal due to the other source, assuming that the sources are located at different positions. Thus, each signal can be delayed such that the portion of each signal that is of interest is in phase with one another. Adding these signals together results in the amplification

of the audio source of interest.

One problem with delay-and-sum beamforming is that portions of the mixed audio signal that are from the interferer are added rather incoherently instead of being attenuated. Thus, the output signal usually contains quite a bit of noise in addition to the amplified target signal. This technique also assumes that no distortion or filtering occurs at each of the microphones and does not have any way to deal with reverberations or reflections in the signal. Other, more advanced, beamforming techniques take advantage of the statistical information in the signal, knowledge of the interferer locations, and the environment geometry.

2.3.2 Adaptive Beamforming

In adaptive beamforming, knowledge of the signal correlations between microphones is used to more accurately cancel out the interfering signal. Minimum variance distortionless response (MVDR) beamforming, as described in [8], is used to minimize the total power in the output signal, with the constraint that the array response is unity in the direction of the source. Solving this constrained optimization problem results in the frequency response shown in equation 2.3. This weight vector is then applied to the observation vector to obtain the desired result: the signal at the source location is passed through, while parts of the signal originating from other locations in the room are attenuated as much as possible in order to keep the overall signal power low [11].

$$\begin{aligned} \min_{\mathbf{H}} \mathbf{H}'\mathbf{R}\mathbf{H} \text{ subject to } Re\{\mathbf{e}'\mathbf{H}\} = 1 \\ \mathbf{H}(\omega) = \frac{\mathbf{R}(\omega)^{-1}\mathbf{e}(\omega)}{\mathbf{e}(\omega)'\mathbf{R}(\omega)^{-1}\mathbf{e}(\omega)} \\ \mathbf{H}(\omega) : \textit{frequency response} \\ \mathbf{R}(\omega) : \textit{covariance matrix} \\ \mathbf{e}(\omega) : \textit{source steering vector} \end{aligned} \tag{2.3}$$

If only one of the users is speaking for a portion of the overall sample, MVDR can generate filters that will completely cancel almost all interfering noise and/or speakers. For instance, if the data contains only the interferer's speech and the source location is known, MVDR can be used to optimally cancel the interferer. The filters generated by MVDR are unity in the direction of the source and try to completely cancel the signal at all other locations in the room. These filters are then applied to the original signal and are effective in passing the source and attenuating any interfering signals. This technique works very well because MVDR can use all of its degrees of freedom to cancel out interfering signals since the interferer-only signal contains no signal power coming from the source direction.

2.3.3 Geometric Beamforming

Another technique, geometric beamforming, uses spatial information to steer towards the source location while attenuating sounds from other locations. A particular type of geometric beamforming, cell-based beamforming, is described in [1]. In cell-based beamforming, the visual tracking system supplies the beamformer with the source and interferer locations. The experimental environment is then divided into a number of non-overlapping regions, and then classified as either source regions or interferer regions as shown in Figure 2-1. The source location(s) is(are) given a weight of 0, indicating that no attenuation should occur in these regions. All other regions are given non-negative weights corresponding to the audio signal power in that region. The larger the signal power in a given region, the greater the weight assigned to that region. The filters at each channel are then chosen so as to minimize the overall response, with the constraint that the total response from the source location(s) is unity. This results in the solution to the optimization problem given in equation 2.3.

A limitation of geometric beamforming is that only positional information, and no statistical information, is used to specify the filters. Thus, the beamformer has trouble separating the source if the propagation model is not accurate [11].

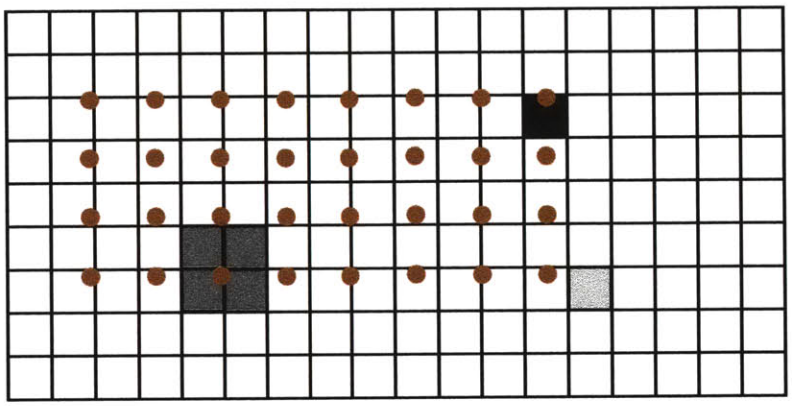

Figure 2-1: The room is divided into a number of non-overlapping regions as shown. The desired source is colored black. The interferers are colored gray, with the lighter gray square corresponding to a weaker interfering signal. The microphone array is represented by the set of red circles [1].

Chapter 3

Implementation

In practice, the signal received by the microphone array is comprised of not only the signal propagating along the direct path from the speaker, but also several copies of this direct path signal which arise from reverberations and reflections in the environment. These reflections and reverberations tend to diminish performance because they are often unaccounted for when beamforming occurs. The acoustic modelling technique described in this chapter tries to model these reverberations and reflections as separate, anechoic interfering sources emanating from fixed locations. Nulls can then be steered toward these interfering sources, which I will refer to as "image" sources, thereby effectively cancelling the reverberations and reflections in the signal.

This chapter provides a description of the implementation of the acoustic modelling technique and the overall system. Additionally, a discussion of the various considerations that went into the design of this system follows.

3.1 Implementation Details

This section describes in detail the implementation of the system. First, the process of acquiring speech using a microphone array is discussed. Next, a description of the initialization procedure including the acoustic modelling step is given. Finally, the tracking procedure for moving speakers is described.

3.1.1 Acquiring Speech Using a Microphone Array

A linear microphone array is used to detect the audio signal propagating from the source and from the interferer. Assume the array has N sensors. The signal received at the microphone array is in general a mixture of speech from a target speaker and speech from an interfering speaker. We denote this mixed signal as:

$$\mathbf{y}(t) = \mathbf{h}_{tar}(t) * x_{tar}(t) + \mathbf{h}_{int}(t) * x_{int}(t) + \mathbf{n}(t) \tag{3.1}$$

where $x_{tar}(t)$ and $x_{int}(t)$ are the unfiltered target speech and interferer signals, $\mathbf{h}_{tar}(t)$ and $\mathbf{h}_{int}(t)$ are filters that, when applied to the target and speech signals, describe how the signals propagate to the array, and $\mathbf{n}(t)$ represents noise in the room. The goal of the initialization process is to find a way to model the filters $\mathbf{h}_{tar}(t)$ and $\mathbf{h}_{int}(t)$ using the acoustic modelling techniques described in section 3.1.2.

The next step in the process is to estimate the noise covariance matrix of the interfering signal, $x_{int}(t)$. Since the target is quiet during a portion of the initialization phase, it can be seen that

$$\mathbf{y}(t) = \mathbf{h}_{int}(t) * x_{int}(t) + \mathbf{n}(t). \tag{3.2}$$

The interfering signal at each microphone is then divided into K windows, with a DFT applied to each window. For the k^{th} window, the interfering signal's DFT is given by $\mathbf{Y}_\omega(k)$, where ω is the frequency bin index. The noise covariance matrix, $\mathbf{R}_\omega$ is then calculated by taking the outer product of the interfering signal with itself at each frequency bin of interest and summed over the windows:

$$\mathbf{R}_\omega = \sum_{k \in K} \mathbf{Y}_\omega(k)^H \mathbf{Y}_\omega(k) \tag{3.3}$$

The noise covariance matrix is then used to find a set of model image sources that best represent the room acoustics.

3.1.2 Initialization with Acoustic Modelling

The system is initialized during periods of time where the target and interferer are speaking alone. Good estimates of the noise covariance matrices can be acquired during these periods of time. The acoustic modelling technique is then used to represent the noise covariance matrix as a weighted sum of a small number of anechoic coherent sources emanating from different locations in the environment. This representation is crucial to the system as it allows for generalizations to other locations in the room for moving speakers as discussed in section 3.1.3. The acoustic modelling technique is comprised of three parts: using eigenanalysis methods to create a vector representation of the covariance matrix, creating a set of image sources to choose from, and choosing the appropriate image sources whose weighted sum is a close match to the vector representation of the covariance matrix.

Eigenanalysis Methods

Finding a vector representation of the covariance matrix is important, as this allows for a linear error minimization function when choosing the appropriate image sources. One way to do this is to look at the eigenvectors of the covariance matrix. For a covariance matrix that models M propagating signals, the M eigenvectors associated with the largest M eigenvalues constitute the signal subspace of the covariance matrix. The remaining eigenvectors constitute the noise subspace of the covariance matrix.

When initializing the system, we are only concerned with the covariance matrix that describes the interfering signal. Thus, we try to match the eigenvector associated with the largest eigenvalue of the interfering signal's covariance matrix. The eigenvector that we are trying to match is denoted as the N element vector $\mathbf{v}_\omega$. Further discussion on eigenanalysis methods can be found in [8].

Image Source Configuration

The image sources are set up in a semicircle configuration as shown in Figure 3-1. In our experiments, 25 image sources are set up in a semicircle arrangement around the

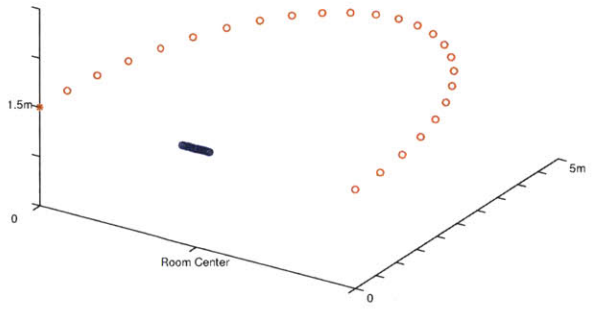

Figure 3-1: The actual location of the image sources depend on the relative delays that are calculated. The red circles represent the image sources and the blue circles represent the microphone array.

microphone array. A small number of image sources from this set are then chosen to represent the signal of interest (see subsection "Choosing Image Sources" below). The frequency response for the set of image sources is given by the N element vector $\hat{\mathbf{v}}_\omega$.

Choosing Image Sources

The basic idea is to choose a set of image sources and their weights, such that their weighted sum closely matches the eigenvector associated with the largest eigenvalue of the covariance matrix, $\mathbf{R}_\omega$. An iterative method is used based on matching pursuits (see subsection "Matching Pursuits" below) to find this set of image sources. The closest match is found by minimizing the ℓ_2-norm of the difference between the eigenvector and the weighted sum of image sources. Additionally, we would like to limit the number of image sources chosen to a small number and have found that we reach a point of diminishing returns after only a few image sources have been selected.

Ideally, there would be some set of image sources whose weighted sum would be

equal to the eigenvector of interest. Thus, we would like

$$\mathbf{v}_\omega = \sum_{\ell=1}^{L} \beta_\ell \hat{\mathbf{v}}_{\omega\ell} \tag{3.4}$$

where β_ℓ is a vector of weights assigned to the image sources and L image sources have been chosen.

In practice, it will not be possible to match the eigenvector exactly, so we find the image sources that best match the eigenvector according to the following error criteria:

$$\text{Let } \mathbf{z}_\omega = \sum_{\ell=1}^{L} \beta_\ell \hat{\mathbf{v}}_{\omega\ell} \tag{3.5}$$

$$\text{Then, } error = \sum_{\omega} \sum_{n} |\mathbf{z}_\omega - \mathbf{v}_\omega|^2 \tag{3.6}$$

where the error is summed over all frequency bins of interest and all microphones.

One issue with finding the appropriate image source is that the eigenvectors are all normalized and thus the phase information is missing. To compensate for the missing phase information, we rotate the eigenvector so that it is in phase with the selected weighted image sources and recalculate the error according to:

$$error = \sum_{\omega} \sum_{n} |\mathbf{z}_\omega - e^{-i\theta_\omega} \mathbf{v}_\omega|^2 \tag{3.7}$$

where θ_ω is the appropriate phase at each frequency.

We alternate between picking the weights and the phases for a few iterations. This procedure of alternating between picking weights and phases allows for more accurate choices of weights and phases because the phases are picked with the weight information accounted for, then the weights are re-chosen with the phase information accounted for.

Matching Pursuits

A matching pursuits algorithm, as described in [10], is used to try to match the eigenvector. After each image source is chosen, the residual error is calculated as follows:

$$\mathbf{q}_\omega = e^{-i\theta_\omega}\mathbf{v}_\omega - \beta\hat{\mathbf{v}}_\omega \tag{3.8}$$

where $\mathbf{q}_\omega$ is the residual and $\beta\hat{\mathbf{v}}_\omega$ is the weighted sum of all the image sources chosen so far.

The eigenvector is then set to be equal to the residual, $\mathbf{q}_i$, which we then try to match with another image source from the set.

Relative Distances

At first all the image sources that we are choosing are equidistant from the microphone array. To more accurately represent the eigenvector, the appropriate relative distances between the chosen image sources must be found. These relative distances are very closely related to the phases that were chosen earlier. After each image source is chosen, we take the IDFT of the phases that were chosen to rotate the eigenvector. Since a rotation in frequency corresponds to a delay in time, the IDFT is a delayed impulse that tells us how much to delay each image source in order to get the correct relative distances. Thus, we find the location of the peak of the IDFT, and this peak corresponds to the necessary delay.

Creating the Beamformer

The last step in the initialization is to create a beamformer using the weighted combination of image sources. The estimated noise covariance, $\hat{\mathbf{R}}_\omega$ is then given by the outer product:

$$\hat{\mathbf{R}}_\omega = (\beta\hat{\mathbf{v}}_\omega)^H(\beta\hat{\mathbf{v}}_\omega) \tag{3.9}$$

This estimated covariance matrix is then used to create an adaptive beamformer based on MVDR (see section 2.3.2). Before computing the frequency response of the

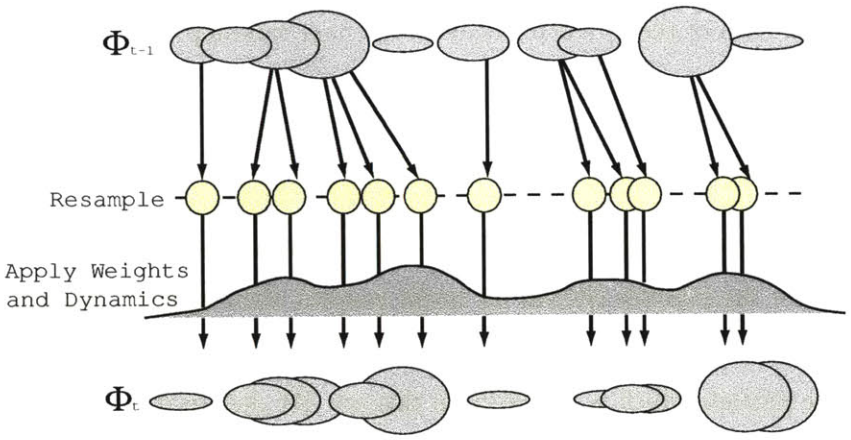

Figure 3-2: An illustration of the particle filtering model. First, the particles from time $t-1$, Φ_{t-1}, are resampled. Next, weights are applied according to the likelihood function shown. Finally, dynamics are applied to spread the particles out to get the set of particles at time t, Φ_t.

beamformer, the covariance matrix is altered by adding a scaled version of the identity matrix to each frequency bin. The amount of identity matrix added to each frequency bin depends on the total power of the covariance matrix at a given frequency.

3.1.3 Tracking Moving Speakers

Once initialization is complete, a particle filtering framework is employed to allow for generalizations to other locations in the room for a moving speaker. An illustration of the particle filtering model is shown in Figure 3-2. All experiments were done with just two speakers, a target and an interferer; however, this approach can be generalized to more than two speakers.

The first step in the tracking portion of the system is to obtain the noise covariance matrix and its eigenvectors for the overall mixed signal received at the microphone array. This step is very similar to the corresponding step described in section 3.1.2, with the N eigenvectors corresponding to the N largest eigenvalues chosen if there are N speakers in the room. Next, the particles are defined and weights are assigned to each particle. The particles are then resampled according to the weights assigned to them. Finally, the new particles are slightly perturbed so as to explore the space

surrounding each of these new particles in the next timestep.

Particle Filter Model

The tracking problem can by formulated in a state-space estimation framework. We can associate the possible choices for image sources, their weights, and their relative distances at time t with a set of particles, Φ_t. A state of the environment is described by a single particle which is chosen from this set. We denote this set of particles as:

$$\Phi_t = (\phi_t^1, ..., \phi_t^n) \tag{3.10}$$

where $\phi_t^i = [\beta_{\ell,tar}, \gamma_{\ell,tar}, \mathbf{d}_{\ell,tar}, \beta_{\ell,int}, \gamma_{\ell,int}, \mathbf{d}_{\ell,int}]$ describes a particular particle. We assume that L image sources have been chosen. β_{tar} and β_{int} are the weights for the image sources chosen for the target and the interferer, γ_{tar} and γ_{int} are the indices into the original set of image sources which correspond to the target image sources and the interferer image sources that have been chosen, and $\mathbf{d}_{tar}$ and $\mathbf{d}_{int}$ are the relative distances for the target image sources and the interferer image sources.

Initially, there is just one particle with the parameters above set equal to the values calculated in the initialization phase. More particles arise after the resampling step occurs, which is described in the "Resampling" section below.

Assigning Weights to Each Particle

Once the particles have been created, weights must be assigned to each particle so that when resampling occurs, the better particles are chosen with higher frequency. First, the weighted sum of the image sources is calculated for the source and interferer:

$$\mathbf{z}_{\omega,targ} = \sum_{\ell=1}^{L} \beta_{\ell,targ} \hat{\mathbf{v}}_{\omega\ell,targ} \tag{3.11}$$

$$\mathbf{z}_{\omega,int} = \sum_{\ell=1}^{L} \beta_{\ell,int} \hat{\mathbf{v}}_{\omega\ell,int} \tag{3.12}$$

where $\hat{\mathbf{v}}_{\omega\ell,targ}$ and $\hat{\mathbf{v}}_{\omega\ell,int}$ are the frequency responses for the target and interferer image sources.

Next, an orthonormal basis is found for the space spanned by the vectors $\mathbf{z}_{\omega,targ}$ and $\mathbf{z}_{\omega,int}$. Let this orthonormal basis be μ. The orthonormal basis μ is then projected onto the space spanned by the eigenvectors corresponding to the largest N eigenvalues, assuming there are N speakers. The squared difference between this projection and the space spanned by the vectors $\mathbf{z}_{\omega,targ}$ and $\mathbf{z}_{\omega,int}$ is then calculated per frequency. Let the error per frequency be ϵ_ω. The log of the weight to be applied to each particle is then:

$$\psi_t^i = \frac{\sum_\omega \epsilon_\omega^i}{\rho} \tag{3.13}$$

where ρ is a scaling factor that determines how much error we are willing to tolerate. A large ρ would mean that we expect a large error when we attempt to match the signal subspaces.

We then normalize these "log weights" to sum to 1 and also exponentiate them to get the actual weights, ξ_t^i. The weights now represent the probability of a particular particle being chosen in the resampling phase of the algorithm.

Resampling

In the resampling phase, particles are chosen from the current set of particles to be used in the set of particles for the next timestep. If this is the first iteration of the algorithm, then all S particles will be the same as the initial particle, where S is the number of particles desired. However, after the dynamics are applied, these particles will no longer be exactly the same.

The weight assigned to each particle, ξ_t^i, determines the probability that a particular particle is chosen for the next set of particles. For example, if there are 4 particles with weights $\frac{1}{2}$, $\frac{1}{4}$, $\frac{1}{4}$, and 0, then one would expect the first particle to be chosen twice, the second particle to be chosen once, the third particle to be chosen once, and the fourth particle to not be chosen at all. Once the particles have been chosen, we apply dynamics to each particle so as to explore the space surrounding

each of the particles.

Applying A Dynamic Model

The final step of the tracking subsystem is to apply a dynamic model. The basic idea is to apply a bit of randomness to each of the parameters that make up a particle. A zeroth order model with a random excitation force applied to each of the particles is used. The dynamics can be written as the following:

$$\beta(t+\delta t) = \beta(t) + F\delta t \tag{3.14}$$

$$\gamma(t+\delta t) = \gamma(t) + G\delta t \tag{3.15}$$

$$\mathbf{d}(t+\delta t) = \mathbf{d}(t) + P\delta t \tag{3.16}$$

where $\beta(t)$ are the image source weights for both the source and interferer, $\gamma(t)$ are the image source indices for both the source and interferer, and $\mathbf{d}(t)$ are the relative distances for both the source and interferer image sources. F, G, and P are independent random excitation forces that are distributed as Gaussian random variables with zero mean and variances σ_β^2, σ_γ^2, and σ_d^2 respectively.

Thus, after dynamics have been applied, the particles are slightly perturbed in various directions. We then iterate through the tracking procedure again, beginning with assigning weights to each of the particles in this new set of particles. As time progresses, we would expect the particles to begin to converge on a particular state that best describes the room acoustics. Using the image sources that are chosen at each timestep, we can also create a beamformer as described in section 3.1.2 and try to achieve source separation.

3.2 Design Considerations

This section highlights the design considerations and decisions with regards to the system whose implementation was described in section 3.1.

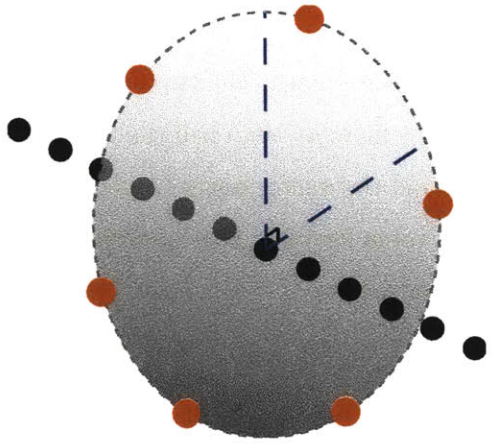

Figure 3-3: The symmetry of a linear microphone array is illustrated above. All locations on the circle perpendicular to the microphone array are indistinguishable. The red circles represent image source locations and the collinear set of black circles represent the microphone array.

3.2.1 Experimental Setup

In designing the experimental setup, decisions had to be made with regards to the size and geometry of the microphone array and with regards to the initialization procedures.

The linear arrangement of the microphone array was chosen due to the symmetry that such an array exhibits. As shown in Figure 3-3, all locations on any circle centered on the array and in the plane perpendicular to the array are indistinguishable by the array. This property arises from the fact that all locations on such a circle are exactly the same distance away from the array with the same relative phase to each of the microphones. Thus, many locations in the room can be cancelled at once due to the symmetric nature of the array. If the source and interferer are both located on one of these perpendicular circles around the array, however, the array symmetry can cause a problem since the beamformer will try to both achieve unity gain and steer a null in the same direction, which is impossible.

The size of the microphone array (i.e. number of microphones) and the spacing between microphones were chosen so as to optimize performance in the room the experiments were performed in. The choice of 15 microphones was due to the use

of a 16-channel data acquisition card, where one of the channels was reserved for a reference (close-talking) microphone. The maximum spacing between any two microphones in the array can be no more than half the wavelength of any of the frequencies in the audio signal. Should the microphones be placed farther than half a wavelength apart, directions in the room become indistinguishable from one another, as grating lobes appear in the beam pattern. On the other hand, if the spacing between microphones is too small, all directions in the room look very similar to the microphone array as the difference in the relative phase to each of the microphones approaches zero. Thus, the spacing between microphones was chosen to be small enough to avoid grating lobes in the beam pattern, but large enough such that the relative phase to each of the microphones can be exploited.

The initial source location information is provided using the acoustic modelling technique as well. This procedure was only possible because the system is initialized by first having only the target speak and then having only the interferer speak. Thus, the initial location of both the source and the interferer are found before the speakers begin to talk at the same time. Additionally, the vision tracker could have been employed to provide the initial source and interferer locations for the initialization step. The locations for the initialization may be slightly more accurate if the vision tracker was used.

3.2.2 Image Sources

There were four main areas of consideration that went into the design of the image sources used in the acoustic modelling step: the configuration of the image sources, the variable distances of the image sources, the number of image sources chosen, and the issue of whether the image sources should be correlated or uncorrelated.

The configuration of the image sources was done in such a way so as to exploit the symmetry of the linear microphone array (see figure 3-1). A semicircle configuration was chosen, since each of the image sources are an equal distance away from the array and only locations on one side of the array need to be chosen since locations on the other side of the array are indistinguishable to the array. Since the linear array

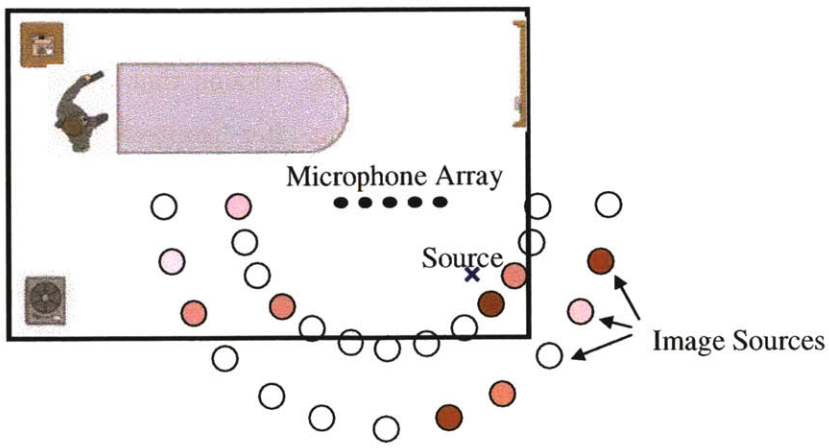

Figure 3-4: Illustration of the use of multiple arcs of image sources. It is evident that if multiple arcs are used, we will be searching over a much larger set of image sources, thereby making the algorithm much slower, and there is less flexibility as we are limited to the distances set by the arcs.

exhibits finer resolution on the broadside of the array, it may have been advantageous to place more image sources on the broadside. However, the advantage of placing more image sources on just the broadside, as opposed to evenly spacing the image sources but trying more image source locations, only minimally sped up the process. The number of image sources used to choose from depends on the resolution of the array in the frequency band of interest. The finer the resolution of the array, the more image sources needed to model various locations in the room. 25 image sources seemed to work well for the linear array configuration. More image sources could have been used, but increasing the number of image sources would slow down the overall system.

The use of variable distance image sources has many advantages. The alternative, which is to have multiple arcs of image sources (as shown in Figure 3-4, all of which are equally spaced, is very easy to implement, but makes the system very slow due to the addition of so many image sources, all of which need to be searched in order to find the best fit. Additionally, the use of variable distances is much more flexible, as it allows the image sources to be located at any distance away from the microphone array, whereas the use of multiple arcs allows for only a finite number of distances.

The number of image sources chosen depends on how many reflections (i.e. how much reverberation) the audio signal contains. I found that the error steadily decreases after each image source is chosen, but after five iterations, the decrease in error is minimal. Thus, choosing five image sources to model the audio signal seemed to work well.

The audio signal is represented by a set of correlated image sources. The image sources were chosen to all be perfectly correlated with one another because all early reflections are just delayed copies of the direct path signal and thus are perfectly correlated with the direct path. If the image sources were chosen to be completely uncorrelated, they would be able to better explain the random noise and later reflections, which tend to be mostly garbled sound. A greater portion of the signal is represented by the direct path and the early reflections, and thus the correlated noise sources better represent the signal.

3.2.3 Error Criteria

There are several error metrics that I could have used for determining how well the image sources that are chosen model the room acoustics. One such metric is looking at the resulting beam patterns, or frequency response plots, that form after a beamformer is created using MVDR with acoustic modelling and after a beamformer is created using MVDR without acoustic modelling (see Figure 3-5 for a typical beam pattern). In this approach, one would look to see if the beam patterns look similar in that the nulls are in the same places in both patterns and that the source signal is passed through in both patterns. Additionally, one would look at the range between the peak of the response and the largest null in the response. If the null is steered in the right location, you would expect a larger range to produce a signal in which the interferer was cancelled more effectively and better source separation was achieved. A problem with this approach is that it is very difficult to determine whether or not the reverberations in the interfering signal are being effectively cancelled. This is because we are not sure of the exact locations of where the reverberations are emanating from, and thus we cannot be sure that the nulls that are formed in the beam pattern

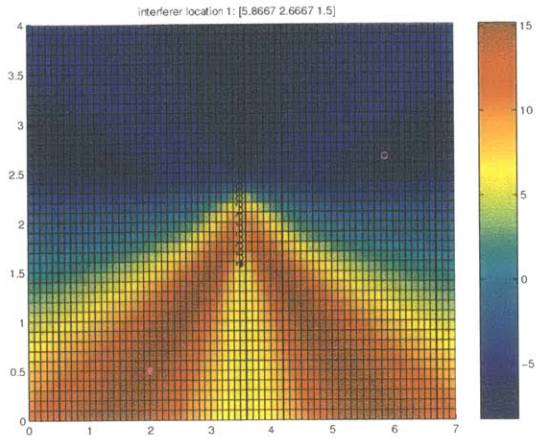

Figure 3-5: A typical beam pattern. The blue areas correspond to regions in the environment where audio is attenuated. The red areas correspond to regions in the environment where audio is amplified. The line of small black circles in the center of the plot represent the microphone array. The magenta circle is where the interferer is located and the magenta star is where the target is located. Notice that a dip in the response occurs in the interferer location and a peak occurs in the source location. Thus, this beam pattern is highly desirable.

actually correspond to locations of reflections.

A more analytic error metric would be to compare the noise covariance matrices that result from using MVDR alone and from using MVDR with the acoustic modelling technique. Since the noise covariance matrices are all that differs in creation of the beamformers for each of these techniques (see equation 2.3), it is clear that if the noise covariance matrix is matched perfectly, then the acoustic modelling beamformer will work just as well as the MVDR beamformer, which as described in section 2.3.2, can create filters that can completely cancel out the interferer if only the interferer is talking for a portion of the input signal. One problem with this approach is that matching the noise covariance matrices is a non-linear problem. Since solving non-linear equations can often be time consuming and more difficult than solving a linear problem, we would like to find a way to linearize the problem.

One way to linearize this problem of matching covariance matrices, is to not try and match the matrices themselves, but instead to try and match the vector whose outer product with itself results in each of these covariance matrices. For the noise

covariance matrix from MVDR, we use the eigenanalysis techniques described in section 3.1.2 to come up with a single vector whose outer product with itself contains most of the relevant information needed to create a beamformer that can effectively cancel out the interfering signal. We then try to find the weighted combination of image sources that best matches this eigenvector and try to minimize the difference according to equation 3.1.2. I found that although matching the eigenvector perfectly does not result in perfectly matching the covariance matrix, a small error using this approach does indeed correspond to desirable results in the output signal (i.e. the interferer is greatly attenuated).

Another important error criteria is signal-to-noise (SNR) ratio. By comparing the increase in SNR in the target from the input signal to the output of the MVDR beamformer and the acoustic modelling beamformer, we can see if the the acoustic modelling beamformer is performing as well as the MVDR beamformer. Additionally, we can compare the decrease in SNR in the interferer from the input signal to the output of the MVDR beamformer and the acoustic modelling beamformer. Clearly, we would like the SNR of the target to increase from the input to the output and the SNR of the interferer to decrease from the input to the output.

3.2.4 Processing in the Frequency Domain Versus the Time Domain

All the processing in this algorithm takes place in the frequency domain. The main reason for processing in the frequency domain is that we can extract the eigenvectors using the eigenalysis technique discussed in section 3.1.2 and try to match the eigenvector instead of the entire noise covariance matrix, which as discussed earlier is a non-linear problem. A problem with working in the frequency domain is that the phase information of the eigenvectors is lost and we need to find the correct phase at each iteration of the algorithm. Had the processing been done in the time domain, the phase information would have been preserved, but the eigenanalysis techniques could not have been used to approximate the noise covariance matrix.

3.2.5 Beamforming

As described in section 3.1.2, a varying amount of identity matrix is added to each frequency bin of the covariance matrix. This step is performed for a couple of reasons. For one, in some instances, the values on the diagonal are so small we end up with a singular matrix that has no inverse. Since, we need to take the inverse of the noise covariance matrix when creating the beamformer (as shown in equation 2.3), it is important that the covariance matrix be non-singular. Another reason for adding in the identity matrix is that is smooths out the response at each frequency bin. Thus, it allows for the algorithm to compensate for small errors in the noise covariance matrix. One drawback of this approach is that if the covariance matrix estimate is perfect, adding the identity matrix will make the frequency response slightly inaccurate.

3.2.6 Particle Filtering

A particle filter was used to track moving speakers in the environment. Although other probabilistic filters, such as the Kalman filter and its many variations, could have been used, a particle filter worked well for a couple of reasons. First, the particle filter is very easy to implement. One needs to come up with just a state-space model, apply a probabilistic weighting to each particle, resample, apply dynamics, and iterate. Basically, finding the proper likelihood model for weighting the particles and applying the correct dynamics are the important design considerations. The flexibility of being able to choose the likelihood model and the dynamic model also makes the particle filter a more attractive choice than other probabilistic filters.

Number of Particles

The number of particles to use in each iteration is dependent on a couple of factors. If speed is a critical performance issue, using less particles will make the algorithm run faster. However, if very accurate tracking is the most critical performance issue, then the use of more particles will generally produce more accurate results. Thus, it is important to find the balance between speed and accuracy when choosing the

number of particles.

Likelihood Model

The likelihood model that is used to assign weights to each of the particles is dependent on how well the signal subspace of the particles matches the signal subspace of the observed data. If there is a very small error between these subspaces, one would expect that particle to be given a larger weight. The normalizing factor, ρ, is used to reflect how much error we are willing to tolerate. For instance, if there is a large amount of noise in the system and we expect the error to be high, we would set ρ to a larger value. Another consideration would be to set a different ρ for each frequency of interest. Doing so may be advantageous because the error may vary widely over different frequencies, and we can account for this by setting ρ to different values at different frequencies.

Dynamic Model

An important design consideration for the dynamic model is the selecting the variances used for the random excitation forces F, G, and P, which are applied to the parameters in each particle. The variance depends on the typical change in each of the parameters from one timestep to the next. We choose a variance such that the typical change is less than the square root of the variance. Thus, the variances, σ_β^2, σ_γ^2, and σ_d^2, were chosen such that their square roots are larger than the typical change in the image source weights, the image source indices, and the relative distances respectively.

Another design decision revolved around what probabilistic model to choose for the random excitation forces. This decision depends mainly on how people move around in the room. If people move around fairly slowly, a Gaussian model would work well. On the other hand, if people tend to move quickly, an exponential model may work better. These models shouldn't make much of a difference, however, if we use enough particles and if our likelihood function does a good job at applying large weights to the particles we are most interested in.

Chapter 4

Results and Discussion

This chapter describes the experiments that were performed to evaluate the performance of the system and the discusses the results. First a description of the experimental setup is given, followed by a presentation of the results. Finally, an analysis of the initialization and tracking results is given.

4.1 Experimental Setup

The experimental setup that I am using is depicted in Figure 4-1. The microphone array consists of 16 microphones arranged in a linear configuration. The microphones were placed in holes in blocks of wood as shown to ensure the linearity of the array. Each microphone is spaced three inches apart from the next microphone. We use a 16 channel data acquisition card to record the audio signals detected at the microphones. Since we only have 16 channels available for recording, we only use 15 of the microphones depicted, and connect one of the channels to a reference microphone, which is a close-talking, clip-on microphone.

The results shown in the following sections were performed on simulated data. We used the signals detected by the reference microphone and applied filters to the signals which add reverberant effects and appropriately delay the signals depending on where they originate from.

Figure 4-1: The linear microphone array configuration used to run all the experiments.

4.2 Initialization Results

I ran 150 trials of the initialization procedure using simulated target and interferer signals in random locations. The filter shown in Figure 4-2 is an example of a reverberant filter that is applied to each channel of the interferer signal to simulate the effects of reverberation. A similar filter is applied to each channel of the target signal as well. Each of the peaks in the filter corresponds to a scaled and delayed copy of the original signal that will arise after the filter is applied. Thus, the overall effect is to create several reverberations in the signal.

The average signal-to-noise ratio (SNR) of the input of both the target and interferer signals was compared with the output of a delay-and-sum beamformer and the output of our acoustic modelling technique. Looking at the SNR results in Table 4.2, we find that the SNR of the target increased from the input signal to the outputs, with a greater increase exhibited by the acoustic modelling technique. Additionally, the SNR of the interferer decreased from the input to the outputs, with a greater decrease exhibited by the acoustic modelling technique. These results show that the beamformer resulting from the acoustic modelling technique performs better than a delay-and-sum beamformer as it has a larger SNR for the signal that we are trying to steer towards and a lower SNR for the signal we are trying to cancel.

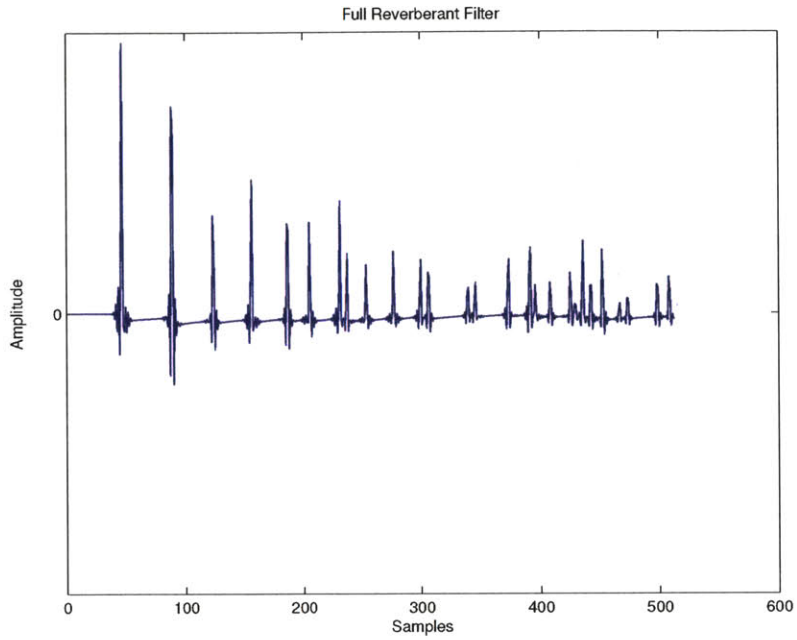

Figure 4-2: An example of the impulse response of a filter applied to one channel of the interferer signal. It is evident that this signal will be highly reverberant, since each peak in the filter corresponds to a delayed copy of the input signal.

Table 4.1: Initialization Results: Average SNR (dB) over 150 Simulated Trials

	Input Signal	Delay-and-Sum	Acoustic Modelling
Target	-6.1385	0.0883	0.5972
Interferer	-1.9593	-4.7950	-7.5039

Sample beam patterns for both the delay-and-sum beamformer and the beamformer created with the acoustic modelling procedure are shown in Figure 4-3. The acoustic modelling beamformer plot contains a distinct null in the direction of the interferer. The delay-and-sum beamformer, on the other hand, does not contain this distinct null. This is because the delay-and-sum beamformer depends only on the source location and thus completely concentrates on trying to pass the source as well as possible. We see that the delay-and-sum beamformer has nulled about half the room and clearly does not have the same precision that the acoustic modelling beamformer displays.

4.3 Tracking Results

In our scenario, the initialization technique described above is used to initialize a model of each of two speakers, which are then tracked while they are simultaneously talking. In the acoustic model that we used to synthesize our data, each speech signal comes from three coherent point sources that move independently through the room. This approximates an environment in which the array receives a direct path signal and two strong reflections.

Over 50 simulated tracking trials, we tracked pairs of speakers and created an MVDR beamformer toward one of the two at the end of their trajectories. We assume that the location of the target speaker is known, and we use our acoustic model of the interfering speaker to create the model "noise covariance matrix." Because the speakers typically move a distance that is greater than a wavelength for the frequencies of interest, the noise covariance matrix that was observed at the beginning of the trajectory would be completely inappropriate for creating a beamformer when the speaker is at the end of the trajectory. By using information across a range of frequencies, however, we are able to track the changing acoustics of each source and create an improved beamformer. Over the 50 trials, our acoustic model-based tracking allowed for the creation of beamformers that passed the target signal with an average SNR improvement of 2.2794 decibels over a reference delay-and-sum beam-

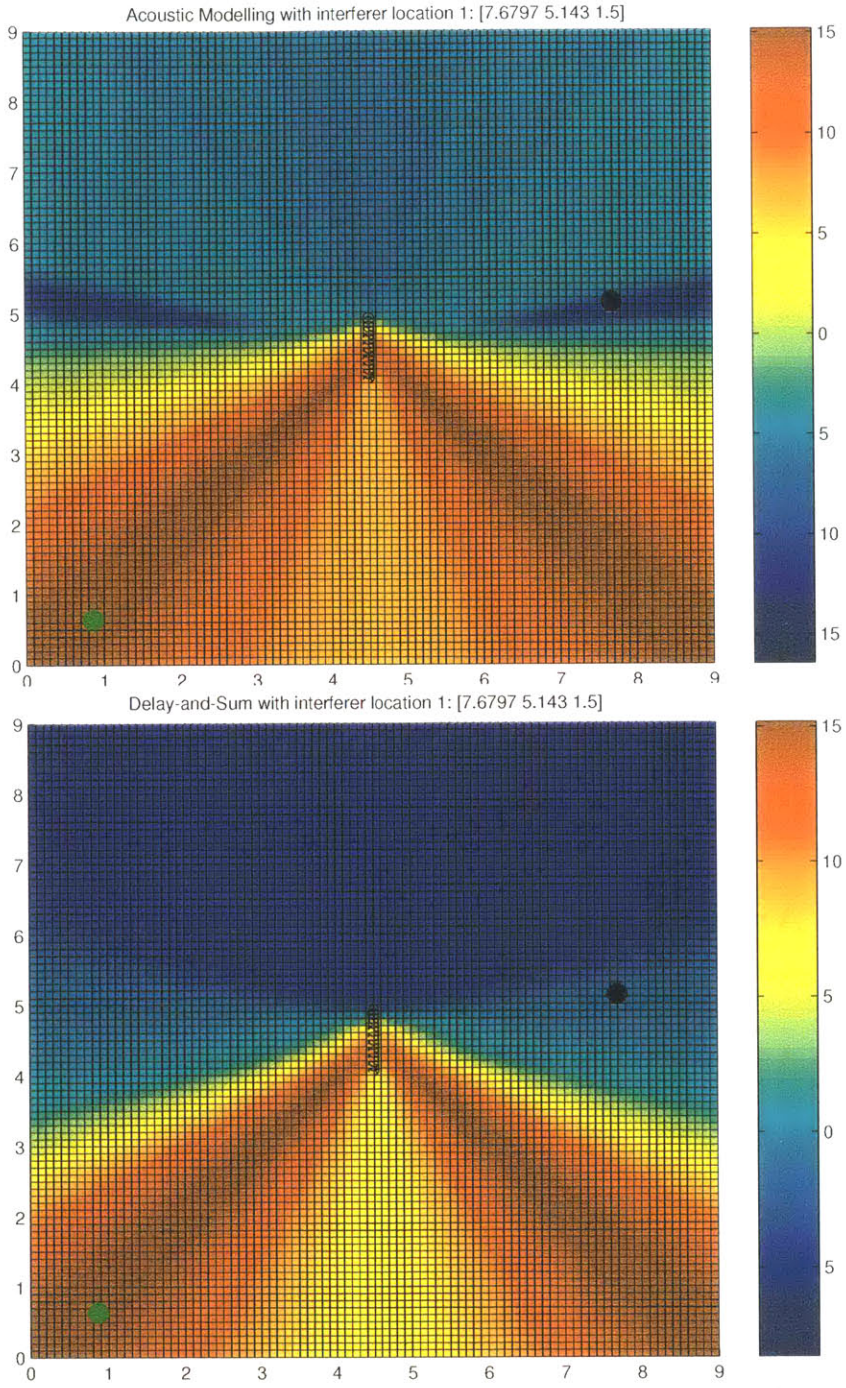

Figure 4-3: A sample beam pattern resulting from the acoustic modelling technique (top plot) and from delay-and-sum beamforming (bottom plot) are shown. The green circle corresponds to the source location and the black circle corresponds to the interferer location. The acoustic modelling technique creates a much more well-defined null in the direction of the interfering signal.

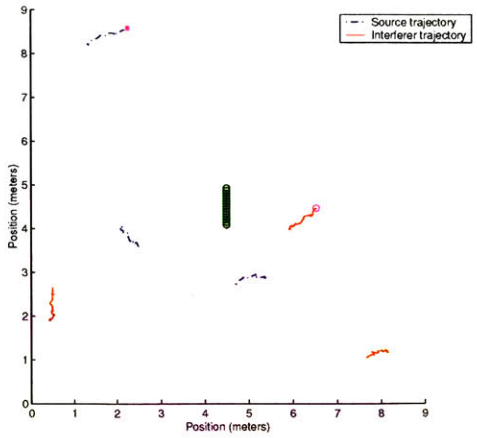

Figure 4-4: A typical run from our tracking experiment consists of 3 coherent copies of each speech source moving through a 9m x 9m. The blue tracks and red tracks represent the trajectories of the sources associated with each speaker. The green circles represent the locations of the microphones in the microphone array. There is a magenta star on the location of the start of the direct path of the source trajectory and there is a magenta circle on the location of the start of the direct path of the interferer trajectory.

former steered toward the target source. Figure 4-4 shows a typical scenario from our experiments.

4.4 Analysis and Discussion

Various factors can affect the performance of both the initialization and tracking subsystems. A discussion of the results with regards to the factors that limit performance follows.

4.4.1 Initialization

There are several aspects of the initialization procedure that could lead to very different results in the output. The location of the source and interferer, the amount of reverberation in the signal, and characteristics in the speaker's voice all play a role in the performance of the system.

The source and interferer locations are extremely important factors in the system's performance. One problem with all beamformers is that if the source and interferer are

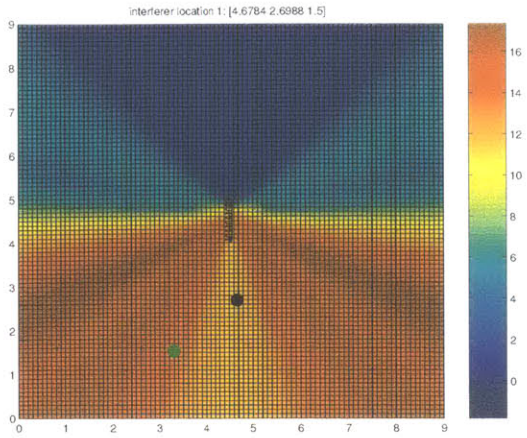

Figure 4-5: A beam pattern for the situation where the source and interferer are too close together for the beamformer to discern them. The black circle represents the interferer and the green circle represents the source. It is clear that the interferer is not well cancelled in this case.

very close to one another, we expect that the beamformer will have trouble achieving source separation. This is because the system is trying to steer a null in the direction of the interferer, while at the same time being constrained to pass the signal coming from the direction of the source. Since the source and interferer positions are very close to one another, this creates a problem for the beamformer. Figure 4-5 shows a beam pattern for the case where the source and interferer are too close to one another. As shown in the plot, the beamformer has tried to amplify the signal coming from the source and since the interferer is located nearby, it also gets amplified. In such a situation, even a low error does not guarantee good performance. The best that the system can do is reproduce the eigenvector associated with the largest eigenvalue of the covariance matrix. If this eigenvector does not do a good job of representing the interfering signal, which is the case when the source and interferer are too close together, then even if you are able to match the covariance matrix perfectly the output will still not be very desirable. Thus, despite the error (according to equation 3.1.2) being very low, the interfering signal will not be cancelled well in the output.

To study the effects on the output of various interferer locations in the environment, 150 trials were run with simulated data using a fixed source location and

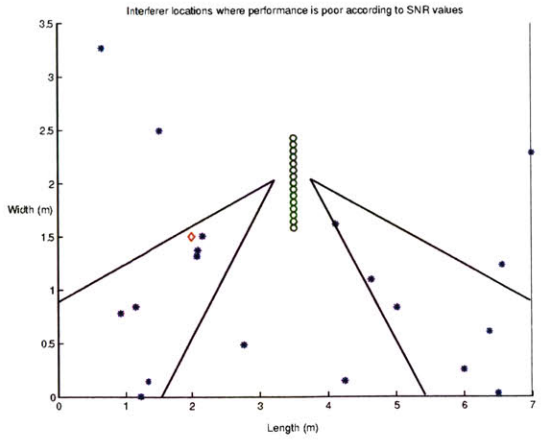

Figure 4-6: The figure above shows interferer locations where the performance of the system is poor according to SNR values. The blue stars correspond to interferer locations and the red diamond corresponds to the source location. The green circles represent the microphone array. The black lines shown mark off the area where there would be a peak in the response of a beamformer created with the acoustic modelling technique with the shown source location. It is evident from this plot that most of the locations for which the system's performance is poor are located in the direction of the source.

random interferer locations. Out of these 150 trials, there were 19 interferer locations that produced both a less than 3dB increase in the target and a less than 3dB decrease in the interferer using the acoustic modelling technique. Figure 4-6 shows these locations. As shown in the figure, the locations at which performance was poor were mostly in the direction of the source, and the corresponding direction on the other side of the array (since these two directions are indistinguishable to the linear array as discussed in section 3.2.1). There are a few outliers, however the majority of the poor locations are in the direction of the source.

It is important to note is that the true test of whether the system is performing well or not depends mainly on how good the output sounds. There are instances in which the error can be low and the system still performs poorly, as in Figure 4-5. Similarly, there are times when the beam pattern looks as if it is not producing the desired result, when in fact the output sounds very good. This is the case that is shown in Figure 4-7. In this plot, it seems as though the interfering signal is being passed through without attenuation, since the plot does not show a null in the direction of the direct

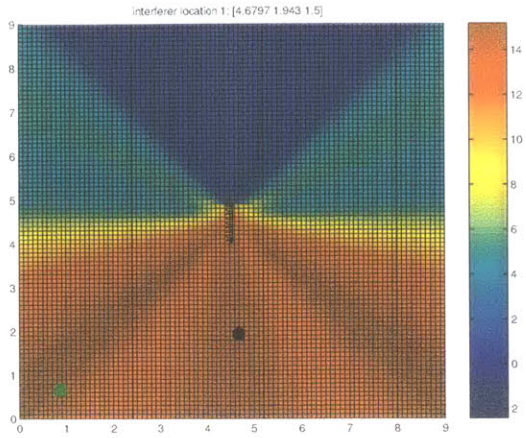

Figure 4-7: The beam pattern shown seems as though it is doing a poor job at cancelling the interferer. On the contrary, the SNR values claim that the interferer is being cancelled quite effectively. It is possible that the null in the top half of the plot is working in conjunction with the peak in the direction of the interferer to effectively cancel out the overall effect of the direct path signal and its reflections. The interferer signal is represented by the black circle and the source signal is represented by the green circle.

path of the interfering signal. However, the actual result of this trial showed that the target in the acoustic modelling output had an SNR increase of approximately 11dB over the target in the delay-and-sum output. Additionally, the interferer in the acoustic modelling output had an SNR decrease of approximately 16dB less than the interferer in the delay-and-sum output. A possible explanation for the discrepancy in the beam pattern is that there are several ways in which a signal and its reflections can be cancelled. For instance suppose an interfering signal is composed of the direct path and one reflection. We assume that the reflection is weaker than the direct path. This interfering signal can be cancelled by either steering a null in the direction of the direct path and in the direction of the reflection, or equivalently can be cancelled by steering unity response in the direction of the direct path and then steering a compensating negative amplitude response in the direction of the reflection. Thus, we effectively sum up a positive copy of the interferer and a negative copy of the interferer, which should cancel each other out and create the desired effect in the output.

To analyze the effects of varying amounts of reverberation in the signal, I applied a

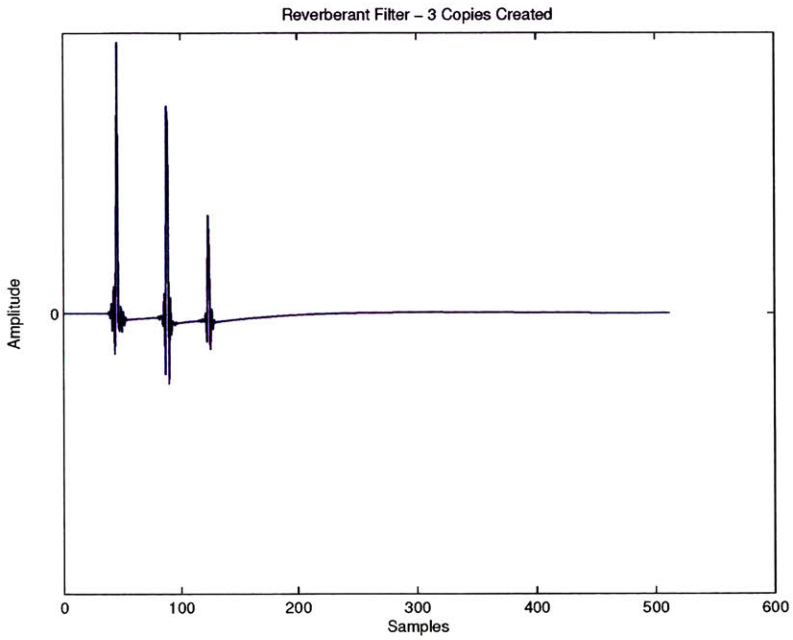

Figure 4-8: An example of a "short" filter applied to one channel of the interferer signal. This filter only has three peaks in its response and thus will only produce three copies of the original signal. The tail is flat and will not cause nearly as many reverberations in the signal as the filter in Figure 4-2.

new set of "short" filters to the target and interferer signals. These short filters contain only three copies of the original signals, thereby limiting the amount of reverberation in the signal. The three reflections in the short filter correspond exactly to choosing three image sources to model these reflections. An example of one of these short filters is shown in Figure 4-8. The tail of the short filter is flat and therefore will not produce several extra copies of the original signal. This is in contrast to the reverberant filter shown in Figure 4-2, which has many oscillations in the tail and will thus produce many copies of the original signal.

The difference in SNR values between the use of the fully reverberant filter and the use of the short filter is summarized in Table 4.4.1. It is evident that the signals generated with the short filter produce much better results than the signal generated with the fully reverberant filter. This is expected because there are fewer copies of

Table 4.2: Effects of Reverberation: Average SNR (dB) over 150 Simulated Trials

	Input Signal	Fully Reverberant Filter	Short Filter
Target	-6.1385	0.5972	1.7814
Interferer	-1.9593	-7.5039	-12.2168

the original signal to deal with, and thus the noise covariance matrix is more easily modelled with a sparse set of image sources. Hence, it is evident that a major factor in the performance of the overall system is the amount of reverberation in the original signals. The fewer copies of the original signals that the system has to deal with, the better it will perform.

4.4.2 Tracking

The tracking results show that the acoustic modelling technique does indeed aid in the tracking of room acoustics for a moving speaker. There is definitely an improvement in SNR in both the target (an increase in SNR) and the interferer (a decrease in SNR). However, some trajectories seemed to produce better results than others.

Out of the 50 simulated trials that I ran to evaluate the effectiveness of the tracking subsystem, 18 of the trials produced unfavorable results. In these trials, the beamformer created after using the acoustic modelling technique to track the changes in the room acoustics produced a lower SNR for the source signal in the output than a reference delay-and-sum beamformer applied to the final source location. Of these 18 trials, 5 produced results that were worse than the SNR of the source signal in the input. These 5 locations corresponded to places where no type of beamforming with a linear array would be effective. This is because the direct path of the source and interferer signals were very close to each other in each of these trials, and thus the system was unable to both amplify this direction and steer a null towards this direction. An example of such a scenario is shown in Figure 4-9. As shown in the figure, the direct path of the source and interferer are located on opposite sides of the array. However, due to the symmetry of the array, these locations are indistinguishable, as discussed in section 3.2.1. Thus, proper source separation is much more difficult in

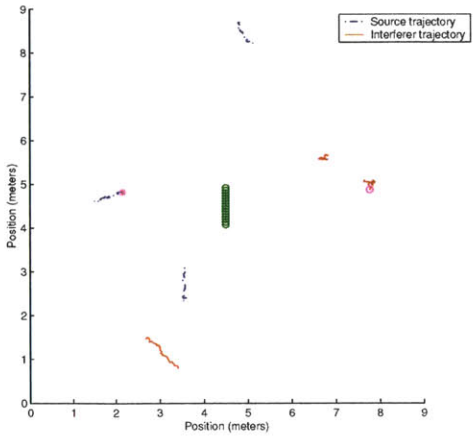

Figure 4-9: An example of where the tracking procedure fails to produce a good beamformed result. The direct path of the source and interferer signals are on opposite sides of the array. However, these locations are indistinguishable due to the symmetry of the array. Thus, proper source separation is more difficult as discussed in section 3.2.1.

this scenario. It is possible, however, that if these trajectories were continued, that a better result could be obtained. This is because the source and interferer could move away from one another to locations where proper beamforming can occur. Since the room acoustics are still being tracked, it is possible that at this new set of locations proper source separation could occur.

Of the remaining 13 trajectories that produced unfavorable results, in three of the scenarios delay-and-sum produced only marginally better results than the acoustic modelling technique. Thus, it can be argued that acoustic modelling worked just as well in these situations. In the remaining trials the source and interferer were very close to the microphone array. This can cause problems because our model represents the source and interferer as a combination of farfield anechoic sources.

Chapter 5

Conclusion

Achieving good source separation from a mixed audio signal is highly desirable for several reasons. For instance, if two people are speaking at once in a pervasive computing environment and both would like to command the system at once, it is important to be able to separate their commands from the mixed signal received at the microphone array. The ability to achieve source separation in situations where the speakers are not moving is a problem that has been explored quite heavily. A more interesting problem is trying to find a way to achieve source separation when the speakers are moving throughout the environment.

This thesis suggests a sparse representation for the room acoustics using a linear combination of a set of coherent broadband sources to allow for tracking as the speaker moves throughout the environment. Tracking the room acoustics is important because the noise covariance matrix changes rapidly as the speaker moves from location to location. The acoustic modelling technique described in this thesis allows for tracking of the room acoustics.

The results presented show that the acoustic modelling technique provides an improvement over delay-and-sum beamforming by several dB in the output SNR. Additionally, after tracking the room acoustics over a period of time and applying the beamformer based on the room acoustics signal received at the end of the speakers' trajectories, the output's SNR is still an improvement over just using delay-and-sum beamforming at the final source and interferer locations.

5.1 Future Work

Future work on this project includes the possibility of adding some speech modelling techniques to the system. The speech modelling techniques, which take advantage of knowledge of what typical speech signals should look like and how they behave, would allow for a more accurate representation of the noise covariance matrices. Additionally, we would like to use this system in conjunction with a speech recognition system, such as the SLS SUMMIT Speech Recognition system, to allow multiple users to command a pervasive computing environment, such as the Intelligent Room at the Artificial Intelligence Laboratory at the Massachusetts Institute of Technology.

Bibliography

[1] M. Brandstein and D. Ward. Cell-based beamforming (ce-babe) for speech acquisition with microphone arrays. *IEEE Transactions on Speech and Audio Processing*, 8(6):738–743, November 2000.

[2] U. Bub, M. Hunke, and A. Waibel. Knowing who to listen to in speech recognition: Visually guided beamforming. In *1995 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1995.

[3] M. Casey, W. Gardner, and S. Basu. Vision steered beam-forming and transaural rendering for the artificial life interactive video environment, (alive). In *99th Convention of the Audio Engineering Society*, 1995.

[4] N. Checka, K. Wilson, V. Rangarajan, and T. Darrell. Probabilistic framework for multi-modal multi-person tracking. In *Workshop on Multiple Object Tracking*, 2003.

[5] M. Collobert, R. Feraud, G. LeTourneur, O. Bernier, J. E. Viallet, Y. Mahieux, and D. Collobert. Listen: a system for locating and tracking individual speakers. In *2nd International Conference on Face and Gesture Recognition*, 1996.

[6] T. Darrell, D. Demirdjian, N. Checka, and P. Felzenszwalb. Plan-view trajectory estimation with dense stereo background models. In *International Conference on Computer Vision*, 2001.

[7] T. Darrell, G. G. Gordon, M. Harville, and J. Woodfill. Integrated person tracking using stereo, color, and pattern detection. *IJCV*, (37(2)):199–207, June 2000.

[8] D. Johnson and D. Dudgeon. *Array Signal Processing: Concepts and Techniques.* Prentice Hall Signal Processing Series, 1993.

[9] J. Krumm, S. Harris, B. Meyers, B. Brummit, M. Hale, and S. Shafer. Multi-camera multi-person tracking for easyliving. In *3rd IEEE Workshop on Visual Surveillance*, 2000.

[10] S.G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, December 1993.

[11] V. Rangarajan, K. Wilson, and N. Checka. Source separation using audio-video sensor fusion. In *Special Interest Group on Computer Science Education: Student Research Contest*, 2003.

[12] H. F. Silverman, W. R. Patterson, and J. L. Flanagan. The huge microphone array. *IEEE Concurrency*, pages 36–46, October 1998.

[13] B. D. Van Veen and K. M. Buckley. Beamforming: A versatile approach to spatial filtering. *IEEE ASSP Magazine*, April 1988.

[14] M. Viberg and H. Krim. Two decades of statistical array processing. In *31st Asilomar Conference on Signals, Systems, and Computers*, 1997.

[15] C. Wang and M. Brandstein. Multi-source face tracking with audio and visual data. In *IEEE International Workshop on Multimedia Signal Processing*, 1999.

[16] K. Wilson, N. Checka, D. Demirdjian, and T. Darrell. Audio-video array source separation for perceptual user interfaces. In *Workshop on Perceptive User Interfaces*, 2001.

[17] K. Wilson, V. Rangarajan, N. Checka, and T. Darrell. Audiovisual arrays for untethered spoken interfaces. In *International Conference on Multimodal Interfaces*, 2002.