Audio-visual localization of multiple speakers in a video teleconferencing setting

Bill Kapralos

Michael R. M. Jenkin

Evangelos Milios

Technical Report CS-2002-02

July 15, 2002

Department of Computer Science

4700 Keele Street North York, Ontario M3J 1P3 Canada

Audio-visual Localization of Multiple Speakers in a Video Teleconferencing Setting[1]

B. Kapralos[1,3], M. Jenkin[1,3] and E. Milios[2,3]

[1]Dept. of Computer Science, York University, Toronto, ON, Canada. M3J 1P3
[2]Dept. of Computer Science, Dalhousie University, Halifax, NS, Canada. B3H 1W5
[3]Centre for Vision Research, York University, Toronto, ON, Canada. M3J 1P3

{billk, jenkin}@cs.yorku.ca eem@cs.dal.ca

Abstract

Attending to multiple speakers in a video teleconferencing setting is a complex task. From a visual point of view, multiple speakers can occur at different locations and present radically different appearances. From an audio point of view, multiple speakers may be speaking at the same time, and background noise may make it difficult to localize sound sources without some a priori estimate of the sound source locations. This paper presents a novel sensor and corresponding sensing algorithms to address the task of attending simultaneously to multiple speakers for video teleconferencing. A panoramic visual sensor is used to capture a 360o view of the speakers in the environment, and from this view potential speakers are identified via a color histogram approach. A directional audio system based on beamforming is then used to confirm potential speakers and attend to them. Experimental evaluation of the sensor and its algorithms are presented including sample performance of the entire system in a teleconferencing setting.

[1]The financial support of NSERC (Natural Sciences and Engineering Research Council of Canada), CRESTech (Centre for Research in Earth and Space Technology) and CITO (Communications and Information Technology Ontario) is gratefully acknowledged.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Teleconferencing Systems

With the advent of the "Global Village", teleconferencing has found a wide range of applications. From facilitating business meetings to aiding in remote medical diagnoses, it is used by corporate, university, medical, government and military organizations. Teleconferencing enables new operational efficiencies resulting in reduced travel costs, faster business decision making, increased productivity, reduced time to market and remote classroom teaching [13].

Various commercial teleconferencing systems exist, including basic static systems for use by two participants (one at each end of the connection). For example, Microsoft's NetMeeting [17], allows for continuous video and speech transmission, document sharing and a "white-board" by the two participants (see [7, 9] for further examples). Systems intended for multiple speakers have also been developed, however, these systems typically focus on a single user and provide limited, if any, automatic speaker tracking. Development of more general teleconferenc-

ing systems, systems that can localize multiple parallel speakers, is hampered by the technology used to acquire the visual scene and the complexities involved in attending to multiple speakers in the audio domain. Existing teleconferencing systems utilize conventional cameras, thereby providing a limited number of static or manually tracked views. As a consequence, in a multiple speaker setting, either the speaker must move into the camera's view or a camera operator is used to manually locate and track and choose between speakers. This is both bothersome and inconvenient for the participants and has deterred many from using such systems. In addition to the cost overhead, the presence of a camera/equipment operator during a teleconferencing session may interfere with the group dynamics [43]. Although visual based systems capable of detecting and tracking human faces exist, they also employ conventional camera lenses, which capture only a narrow field of view.

1.2 Problems with Existing Teleconferencing Systems

Teleconferencing systems must be able to capture and transfer audio (e.g. the speaker's voice) as well as the video signal. In order to capture the speech of multiple participants, popular methods include; having each participant wear their own "tie-clip" microphone and having a human operator determine which microphone to monitor, and having speakers physically move to a location where they can speak into a shared microphone. Although sound localization systems exist,

most rely on extensive microphone arrays [5, 33, 45, 14] which require expensive specialized equipment, are computationally intensive and are non-portable. Several systems have been developed in an attempt to overcome some of these limitations. For example, the PictureTel 900 teleconferencing system [31] employs a microphone array capable of localizing a speaker over a 360^o field of view. Once the speaker has been localized, a software controllable pan-tilt-zoom camera is steered to focus on the speaker. Although this is definitely an improvement over the traditional manually operated camera systems, moving the pan-tilt-zoom camera so that it is focused on the speaker is time consuming and therefore, may negatively interfere with the meeting. Moreover, in order for the sound localization system to be most effective, it must be positioned as close as possible to the users. Finally, this system is incapable of capturing the speech and video of multiple participants simultaneously.

1.2.1 Overcoming the Problems

Because a single standard video camera needs to be fixated on the speaker, it is not an effective sensor for the multi-speaker teleconferencing task. In order to address this issue, a novel hardware device was constructed to provide the raw sensor data. Figure 1.1 illustrates the hardware components comprising the "Eyes 'n Ears" sensor. Eyes 'n Ears consists of a Cyclovision Paracamera omnidirectional video system [30] coupled with a four microphone array. The system is compact, lightweight, portable and is meant to be placed in the center

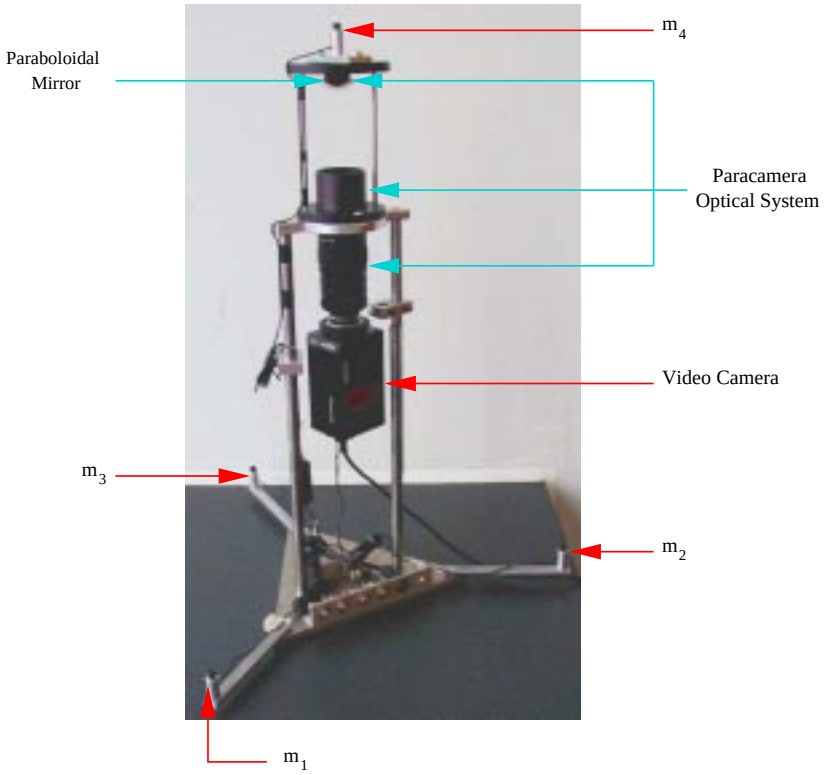

Figure 1.1: Eyes 'n Ears Sensor.

of a table with the participants of the teleconference session seated around it.

Cyclovision's Paracamera omni-directional camera system consists of a high precision paraboloidal mirror and a combination of special purpose lenses. By aiming a suitably equipped camera at the face of the paraboloidal mirror, the optics assembly permits the Paracamera to capture a 360^o hemispherical view of all potential speakers from a single viewpoint. Once the hemispherical view has been obtained, it may be easily un-warped [30] producing a panoramic view. From this panoramic view, perspective views of any size corresponding to different portions of the scene can be easily extracted (see Figure 1.2). The ability of the Paracamera to capture an image of the entire hemisphere makes it very attractive

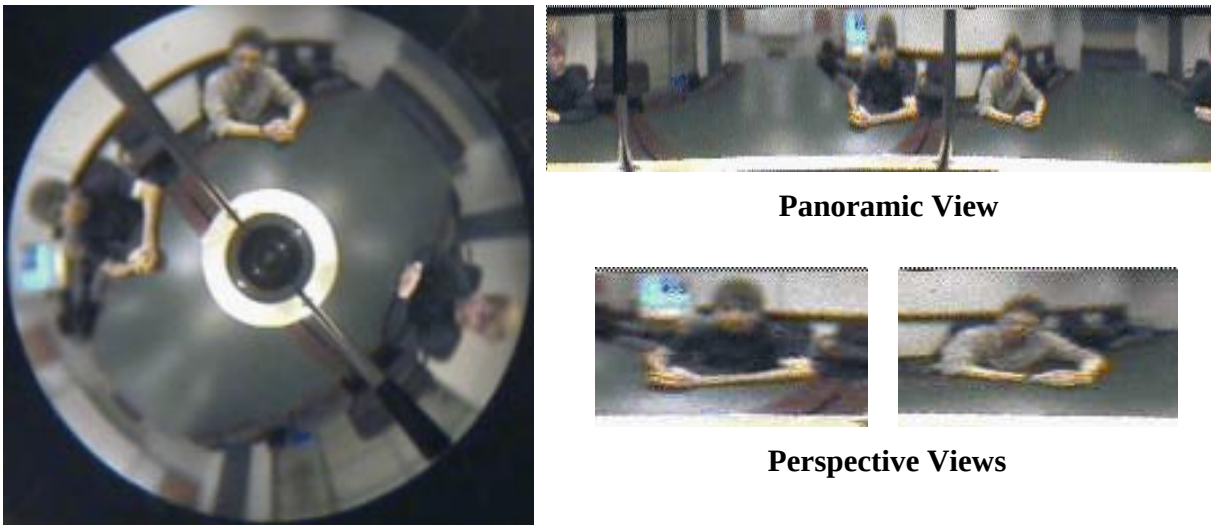

Figure 1.2: Paracamera Images. A panoramic view is easily generated by unwarping the hemispherical view obtained by the Paracamera. Perspective Images of any size may then be extracted from the panoramic.

for a number of different applications. For example, in Stiefelhagen et al. [37] and Yong et al.[43], an omni-directional camera is used to capture the simultaneous video of each participant in a small group meeting. Omnidirectional devices have also been used in many other vision based applications, including surveillance [4, 19], autonomous robot navigation [44], virtual reality [42], telepresence [42], remote view from a Dolphin [3], and pipe inspection [2].

1.3 Overview of the Eyes 'n Ears System

In the Eyes 'n Ears system, the Paracamera is used to identify potential speakers in the environment. Using a statistical color model, the video system locates regions of human skin present within the Paracamera's view. These skin regions

correspond to faces, arms and other exposed skin regions, as well as other skin-colored distracters in the image. Regions of skin in the camera image are then grouped together to form a cluster. Each cluster of skin regions is assumed to correspond to one particular person. Under the assumption that the face of a person in the Paracamera image is further away from the center of the image relative to the other skin regions, the locations within the camera image of potential speakers are identified. Once each face has been found, an estimate of its direction in the real world relative to the Paracamera is computed and provided to the audio system as the direction to a potential sound source (e.g. voice of a speaking person).

The audio system consists of four omni-directional microphones (m_1, m_2, m_3 and m_4), mounted in a static pyramidal shape about the base of the Paracamera, which provide an economical and portable acoustic array capable of localizing a sound source in 3-space [18]. Given the initial estimate of the direction to each speaker's face in the real world as identified by the visual system, using beamforming [24] and sound detection techniques, the audio system detects and validates each speaker and focuses on the speech of each individual thus reducing unwanted noise and sounds propagating from other directions.

1.4 Overview of thie Report

Chapterss two and three provide details of the vision and audio systems respectively that make up the Eyes 'n Ears sensor. Chapter four describes the system in operation and presents experimental results of the two subsystems. Chapter five discusses the approach and presents potential directions for future research. Finally, technical details regarding the Eyes 'n Ears system are provided in the appendices.

Chapter 2

Video System

2.1 Color Based Face Detection

Skin color is often proposed as an economical and efficient cue to face detection. Color is one of the simplest attributes in a set of pixels comprising the image [25]. Color does not require extensive computational processing to compute, the color of an object may be used as an identifying feature that is local to the object and color is largely independent of the view and resolution. In general, color is invariant to partial occlusion, rotation in depth, scale and resolution changes [34]. Furthermore, there are fast and simple color based human detection and tracking systems available (see [6, 20, 21, 25, 26, 40, 41]).

A major difficulty in using color information as a cue is the lack of color constancy [16]. As a result changes in lighting conditions lead to variations in the colors of an image (e.g. the color of an object in an image may change under different lighting conditions even when the object itself does not change). The lack of color constancy is most apparent in the RGB (red, green, blue) color space

where intensity is distributed equally to all three color channels [34]. In order to reduce these effects, in this work color is represented in HSV (hue, saturation, value) color space [15]. Furthermore, skin color of different people varies primarily in intensity [37]. Therefore, if value is ignored, the skin color of people regardless of race forms a tight cluster in HS (hue-saturation) space [11, 34].

2.1.1 Color Histograms

Color histograms [38] have been found to be an effective mechanism for representing colored objects in an image. A color histogram is a representation of the distribution of the discrete colors available in an image, i.e. for each possible color a pixel of an image may take on, the color histogram provides a count of the number of pixels in the image (or in an image region) with that corresponding value. Color histograms are invariant to translation, rotation about an axis perpendicular to the image and change only slightly with rotations about other axis, occlusion and change of distance to the object. Furthermore, the color histogram of different objects can differ extensively [38]. Color histograms have proven to be effective representations of two and three dimensional objects and have been used in a wide variety of computer vision applications, including image matching and retrieval [27, 38], face detection and tracking [21] and skin detection [25].

Color histogram models for skin and non-skin color classes for traditional cameras have been constructed previously and are freely available [25]. These

models are constructed using data from images obtained with normal cameras. However, the non-standard lens and curved mirror assembly of the Paracamera was found to distort the colors of skin tones and thus the Jones and Rehg model was found to be unsuitable for the Paracamera sensor. Instead, two-dimensional hue-saturation model histograms for both skin and non-skin color classes were constructed by manually classifying portions of images obtained with the Paracamera. Value was ignored, while hue and saturation values were discretized to 32 and eight discrete values respectively (e.g. each color histogram contains a total of $32 \times 8 = 256$ possible HS (hue-saturation) pairs).

Constructing Skin and Non-Skin Color Histograms

To construct color histograms for skin and non-skin regions, a total of $154,310$ skin pixels were obtained from 35 subjects of various racial groups to ensure a wide variety of skin colors. For each subject, samples of their hands, face (and in several cases legs) were obtained over multiple images. The subjects were asked to change both their pose and their distance relative to the Paracamera to ensure samples in different lighting and orientation conditions were obtained. A total of $307,546$ non-skin pixel samples were obtained by sampling portions of images obtained with the Paracamera that did not contain human skin. Once again, the samples were obtained in different locations and under a variety of lighting conditions. Figures 2.1 and 2.2 illustrate the resulting Hue-Saturation

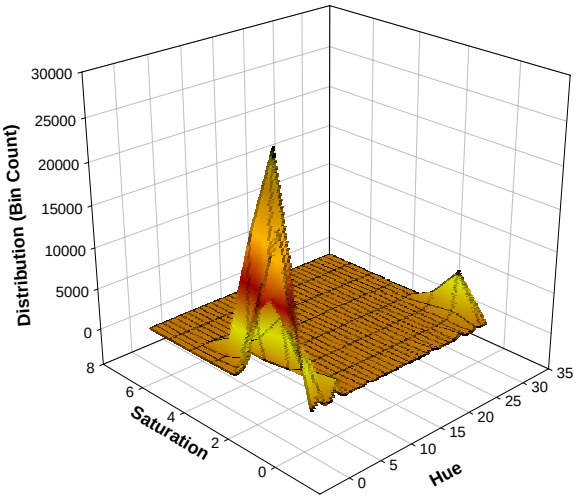

Figure 2.1: Hue-Saturation Skin Histogram. The histogram is constructed by sampling 154,310 skin color pixels in Paracamera images obtained from 35 subjects of diverse ethnic background, under various lighting conditions.

histograms for both the skin and non-skin classes. For each "bin", the axis labeled *Distribution*, records the number of pixels from the samples with corresponding hue-saturation values.

2.2 Finding the Skin Regions

Given the skin and non-skin histograms described above, the estimated probability that a particular hue and saturation pair (referred to as HS) belongs to either the skin, $P(HS|skin)$ or non-skin $P(HS|\neg skin)$ classes may be found using basic probability theory (see [32] for a review of probability theory and [25] for an earlier application of probability theory to the skin detection problem)

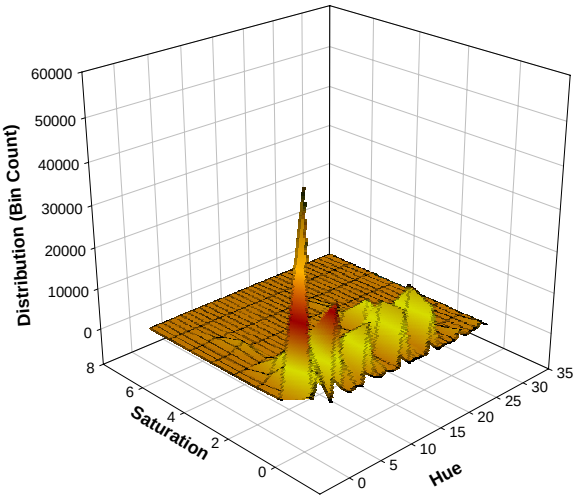

Figure 2.2: Hue-Saturation Non-Skin Histogram. The is constructed from 307,546 non-skin color pixels obtained under various lighting conditions. Note the difference in distribution with the skin color histogram plotted in Figure 2.1.

$$P(HS|skin) = \frac{skin[HS]}{T_s} \tag{2.1}$$

$$P(HS|\neg skin) = \frac{\text{non-skin}[HS]}{T_n} \tag{2.2}$$

where $skin[HS]$ and non-skin$[HS]$ is the count in bin HS of the skin and non-skin histograms respectively. T_s is the total number of pixels contained in the skin histogram while T_n is the total number of pixels contained in the non-skin histogram.

When $skin[HS]$ is greater than a pre-defined threshold value δ, the probability

that a pixel in the hue-saturation color space is skin color $P(skin|HS)$, may be determined using Bayes rule, otherwise, $P(skin|HS)$ is set to zero

$$P(skin|HS) = \begin{cases} 0 & \text{if } skin[HS] < \delta \\ \frac{P(HS|skin)P(skin)}{P(HS|skin)P(skin)+P(HS|\neg skin)P(\neg skin)} & \text{if } skin[HS]) \geq \delta \end{cases} \tag{2.3}$$

where $P(skin) = \frac{T_s}{T_s+T_n}$ and $P(\neg skin) = \frac{T_n}{T_s+T_n}$ is the probability of a pixel belonging to the skin and non-skin color classes respectively. Once the probability that a particular HS pixel value corresponds to skin has been calculated, the pixel is classified as skin if

$$P(skin|HS) \geq \Delta \tag{2.4}$$

where Δ is a pre-defined threshold.

2.3 Eliminating "Noisy" Regions

In order to enhance the efficacy of the color histogram skin detection process a number of pre- and post-filtering stages are applied. These are:

1. Prior to applying the color histogram skin detection process, an image differencing process is used to remove static (background) image locations from further processing. A time-averaged background image is computed

and portions of the image that do not differ significantly from this background are ignored.

2. Once individual pixels have been labeled as skin pixels, these pixels are combined into skin regions. Holes or concavities that may be present in these skin regions are reduced by applying dilation and erosion operations.

3. Following the dilation and erosion operations, a unique identifier is assigned to each region or connected component. Components smaller than a predefined threshold are ignored.

2.4 Locating the Face

Components in close proximity are grouped together, to form a cluster. The component in each cluster furthest from the center of the Paracamera image is identified as a potential face. Figure 2.3 shows (a) a sample image of users in a teleconferencing setting, and (b) the identified skin regions (yellow) and potential faces (red crosses). Given the center of a face or skin region as identified by the visual system, the direction to the face is computed in world co-ordinates. Since the focus of the paraboloidal mirror is also its centre of projection, it is straightforward to convert camera pixel locations to 3D direction vectors (see Appendix D for greater details). Once potential speaker directions have been determined, each direction is probed by the audio system to determine if the

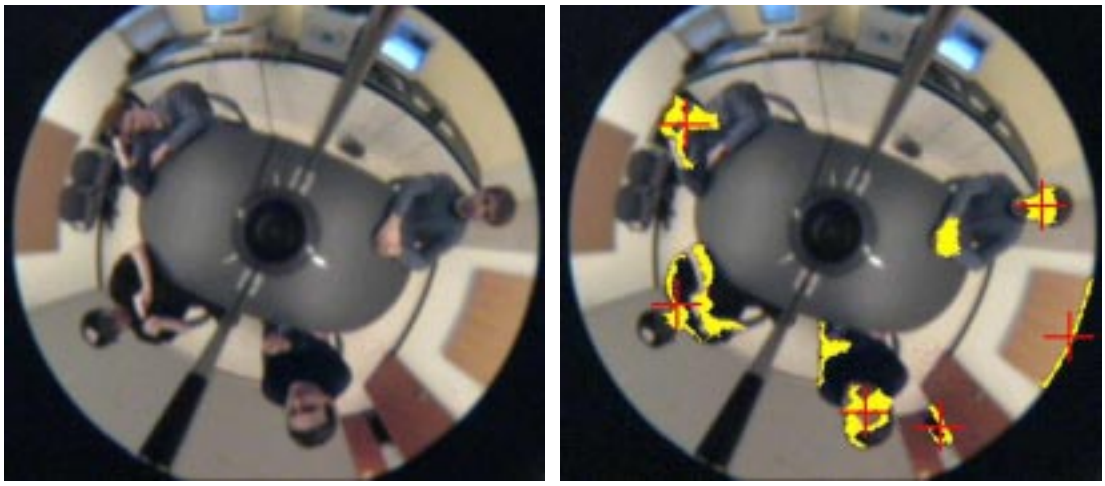

Figure 2.3: Faces identified in the input omnidirectional image. Pixels identified as skin are colored yellow and the centers of the identified faces are indicated with a red cross.

speaker direction has a corresponding signal in the audio domain.

Chapter 3

Audio System

3.1 Microphone Arrays and teleconferencing

The audio system is responsible for confirming the presence of speakers in the directions estimated by the video system. Since the participants of a teleconference session will generally be speaking, confirmation of the face is achieved by detecting the presence of a sound source within a small set of directions centered on the direction obtained by the video system. Due to various acoustical factors including reverberation, a propagating speech signal may be degraded or altered extensively before reaching a microphone. Furthermore, this degradation increases as the distance between the sound source and the microphone becomes greater [36]. As a result, the information captured by a microphone and transferred to the other parties may be incomplete and may lead to confusion amongst the participants. This has made teleconferencing a poor substitute for

person to person contact [45]. To reduce this problem the microphone should be positioned close to the person speaking ensuring the Signal-to-Noise Ratio (SNR) of the captured speech is very large (e.g. the power of any captured noise is negligible relative to the power of the speech signal) [28]. Unfortunately, this is not always practical, especially in large rooms where the distance between the participants and the microphone is large and when participants at different locations in the room need to speak and be heard. Traditional methods to overcome this problem provide each participant with their own microphone. However, an operator is then needed to determine who is speaking and turn on the speaker's microphone while turning all other microphones off [33]. Alternatively, the task of determining who is speaking and turning off all other microphones may be performed using speaker detection software (e.g. the signal of each microphone containing the greatest energy may be used to indicate the speaker). However, such an approach is limited by the number of audio input channels available and since the microphone of each participant requires its own input channel such an approach is certainly not scalable.

There are many situations in which a physical signal present in our environment is monitored by some appropriate sensor, providing us with information about the surroundings. The examples are many and include both artificial sensors (e.g. satellites, microphones, antennas etc.), as well as our own senses (e.g. eyes, ears etc.). In many situations, a single sensor may not be able provide

sufficient information. To increase the effectiveness of a sensor, rather than relying on a single sensor, an array of sensors may be used instead. A sensor array is a set of sensors placed at different locations to spatially sample a propagating wave-field (e.g. sample a wave-field emanating from a particular direction), which may be electromagnetic, acoustic, seismic etc. [24]. Using the array allows a speech signal to be captured from any location in the room while greatly attenuating background noise, reverberation and sounds from other sound sources. In addition, it eliminates the need for the participants to carry a microphone with them and the need for an operator to control the microphones. Most importantly however, it allows for the automatic speech acquisition with minimal noise and detection and tracking of an active speaker regardless of their position in the room [28]. Greater details and specifications regarding the microphone array used in this work are available in Appendix C.

3.2 Beamforming with the Microphone Array

Although various methods of beamforming have been developed, the simplest and most common is referred to as *delay and sum beamforming* [24]. Consider a sound source located at some position x_s in three dimensional space. Furthermore, consider an array of m microphones (each microphone is denoted by m_i for $i = 1 \ldots m$) where each microphone is at a unique position x_i and each is in the path of the propagating waves emitted by this sound source. In general, the time taken

for the propagating sound wave wave to reach each microphone will differ and the signal received by each microphone will not have the same phase.

The differences in the time of arrival of the propagating wave at the sensors depend on the direction from which the wave arrives, the positions of the sensors relative to one another, and the speed of sound v_{sound}. Beamforming takes advantage of these time differences between the time of arrival of a sound to each sensor [33], and allows the array to be "aligned" so as to be tuned to the particular sound source. As shown in equation 3.1, beamforming consists of applying a delay Δ_i and amplitude weighting[1] w_i to the signal received by each sensor s_i and then summing the resulting signals [24]

$$z(t) = \sum_{i=0}^{M-1} w_i s_i(t - \Delta_i) \tag{3.1}$$

where $z(t)$ is the beamformed signal at time t and M is the number of sensors.

Since the output of the beamformer will be maximized when it is steered to the location of the source, the beamformer may also be used to determine the location of a sound source(s) without any knowledge of the source's location. This may be accomplished by focusing the beamformer to every possible location and recording the location of the strongest beamformer output. The locations of maximum output will correspond to the location of a propagating source. Unfortunately, such an approach however is generally not feasible. It will take

[1]For this application, the amplitude weighting is ignored e.g. , $w_i = 1$ for all i.

far too much time and processing effort to actually focus the beamformer to every possible location. However, when the number of potential sound source locations is restricted in some manner, beamforming is an effective method of audio detection. Essentially, this is the purpose of the video system. For this application the speech of the participants is the signal of interest. By locating the face of each person present in the Paracamera's view, the video system reduces the number of potential sound sources from many thousands to several (e.g. under 10), making the audio system's task tractable.

3.2.1 Filter Delay and Sum Beamforming

In voice recognition and detection applications, the signals of interest fall within a small frequency region. Although humans are capable of perceiving sounds from $20 - 20000$Hz [1], most speech falls within a very small frequency range ($200 - 4000$Hz) [22]. By filtering the signal received by each of the microphones, frequencies which do not lie in the region of interest, can be attenuated. This reduces noise present in the original signal and leads to more accurate sound detection. Filtering is accomplished entirely in the software domain using a low and high pass digital FIR filter.

3.3 Far Field Acoustical Model

If the direction of propagation between the sound source and each microphone of the array are assumed to be the same (far-field assumption), then to focus the array in some direction $\vec{\beta}$, the delays Δ_i are computed as:

$$\Delta_i = -\frac{\overrightarrow{\beta} \cdot x_i}{v_{sound}} = -\frac{\overrightarrow{\beta} \cdot x_i}{345m/s} \tag{3.2}$$

where $\overrightarrow{\beta}$ is the unit vector denoting the direction of propagation relative to the array's origin, x_i is the vector from the array reference and the i^{th} microphone and the speed of sound v_{sound} is assumed to be constant at $345m/s$ [35].

Geometrically, the time delay is determined by projecting x_i, onto the unit vector $\overrightarrow{\beta}$. This projection gives the difference in distance the sound must travel to reach the array reference and the i^{th} microphone. Dividing this distance by the speed of sound v_{sound}, gives the appropriate time delay.

When the delays have been determined correctly (e.g. to accurately match the source location relative to the array) and applied to each microphone signal, the resulting microphone signals will be in phase. As a result, when the signals are summed together to form the beamformed signal, they will reinforce each other causing the energy of the beamformed signal to reach a maximum (see Figure 3.1). Delays that do not correspond to a true sound source direction (e.g. the

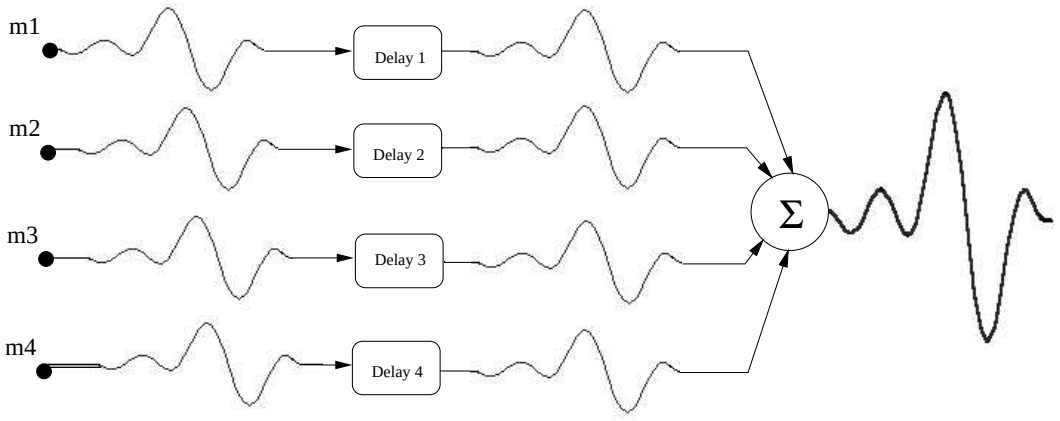

Figure 3.1: Forming the Beamformed Signal. Summing the appropriately delayed signal from each microphone will lead to a beamformed signal with maximized energy output.

beamformer is incorrectly steered to some other direction), will cause a reduction in the energy level of the beamformed signal.

3.4 Implementation Details

The delay derivation described in the previous section assumes a continuous time signal and a delay of any arbitrary time. In reality however, this luxury does not exist! The signal obtained by each of the microphones is actually a sampled version of the original propagating signal and therefore, only integer delays may be easily achieved. As a result, the beamformer may not be able to focus directly to locations corresponding to non-integer delays. For this work, the delays are calculated as described earlier, assuming a continuous time signal. After computing the time delay, as shown in equation 3.2, the discrete time delay $\Delta_{i_{discrete}}$ is obtained by "rounding" the continuous time delay to the nearest integer. Once

the discrete delays have been calculated, the beamformed signal $s_{beamform}$ can be constructed as

$$s_{beamform} = \sum_{j=0}^{N}(s_1[j] + s_2[j + \Delta_{2_{discrete}}] + s_3[j + \Delta_{3_{discrete}}] + s_4[j + \Delta_{4_{discrete}}]) \quad (3.3)$$

where $N = 2048$ is the size of the sample "time window".

3.4.1 Processing the Signal

Two measures are used to determine if the beamformed signal corresponds to a speaker, the *signal variance* and *signal difference*.

Signal Variance Criterion The variance of the beamformed signal must be substantially higher than that of the background noise. Generally, the signal variation associated with a signal emanating from sound sources such as speech, music and impulsive sounds, is greater than the variance of background noise. Following the approach of [18], the variance is computed based on 32 sample sub-windows. The variance of each sub-window is computed and finally, the variance for the entire window is estimated by the mean variance of the 52 sub-windows.

$$V_{signal} = \frac{\sum_{i=0}^{i=M-1} \sum_{j=i\times k}^{j=j+k-1} \frac{|s[j]-\bar{s}|^2}{k-1}}{M} \tag{3.4}$$

During system calibration, the variance of the background noise is computed, and this is used to establish a threshold value (V_{thresh}) for the background noise. A potential speaker is only confirmed when the variance of the beamformed signal V_{signal} is greater than V_{thresh}.

Signal Difference Criterion The difference in magnitude levels between the "correctly steered" beamformed signal and an "incorrectly steered" beamformed signal that is obtained by averaging the signal of each microphone with zero delay must exceed a threshold. An average signal s_{avg} is computed by summing the signals received by each of the microphones without introducing any delay. The energy of this signal (E_{avg}) is compared with the energy associated with the beamformed signal (E_{beam}):

$$s_{dif} = \begin{cases} \frac{E_{beam}-E_{avg}}{E_{Mean}} & \text{if } (E_{beam} - E_{avg}) > 0 \\ 0 & \text{if } (E_{beam} - E_{avg}) \leq 0 \end{cases} \tag{3.5}$$

The presence of a sound source is confirmed only when both the signal variance and signal difference exceed specific thresholds, (V_{thresh} and Dif_{thresh} respectively).

$$s_{dif} \geq Dif_{thresh} \text{ and } V_{signal} \geq V_{thresh}$$

3.5 Calibration

Beamforming with a far field acoustical model requires that the position of each microphone x_i relative to the array origin must be known. Rather than using a global coordinate system, the coordinate system defined by the Paracamera system is used. This eliminates the need to calibrate to some arbitrary global reference frame, however the audio and video systems do need to be calibrated with respect to each other. The vertical distance between the plane of the lower three microphones and the optical centre of the Paracamera system is measured. The direction to each of these three microphones is easily measured as they appear in the view of the Paracamera. Under the assumption that the plane defined by these three microphones is perpendicular to the Paracamera optical axis permits the microphones' position to be easily estimated. The fourth microphone is mounted in line with the optical axis of the Paracamera system and its position is measured directly. Greater details regarding this calibration procedure are provided in Appenix C.

Chapter 4

Experimental Results

The video system is responsible for detecting visible skin regions within the visual view of the Paracamera; grouping together the regions corresponding to each person; finding the region of each group that corresponds to the face; and finally, determining the direction to each potential face. Given this information, the audio system is focused to the direction of each potential face in order to detect sound (speech) emanating from the person when they are speaking. As described in the previous sections, both the audio and video systems can be used alone to locate potential speakers in a teleconference setting. However, by combining the two systems, each one is able to complement the other and therefore overcome some of the limitations inherent in each system when used alone. This provides a much more robust system of automatic speaker detection. This section describes two experiments that illustrate the effectiveness of the ability of the combined system to detect and focus on potential speakers in a teleconferencing session

when used together.

The first experiment is a demonstration of the combined system to detect multiple speakers in a typical group meeting scenario while the second experiment quantifies the accuracy of the system's ability to locate and focus on the face of a participant.

4.1 Experimental Room Set-Up

Both experiments were conducted in a meeting room at York University. The room is 3.7m long, 5.8m wide, with a height of 3.0m and is used for group meetings. As illustrated in Figures 4.1 and 4.2, one wall of the room contains several windows facing a hallway, while the other wall contains two windows facing outdoors. The windows facing outdoors are covered with black curtains. There is a counter running the length of one wall approximately 0.8m in height, a white-board (with a wooden cover that was closed during the duration of the experiments) mounted on one wall and a corkboard mounted on the opposite wall. The room also contains a black filing cabinet and a large table with several chars in the middle of the room. Furthermore, the room contains standard flooring tiles, concrete ceiling and normal fluorescent lighting. No effort was made to limit audio reverberation or modify the lighting conditions in any manner. This room is a good representation of the environment of a typical teleconferencing room. Both the audio and video systems rely on several pre-defined (and dynamically

Figure 4.1: The actual meeting room in which both experiments took place.

computed) threshold values. Tables 4.1 and 4.2 summarize these threshold values for both the audio and video systems. Both experiments described in this section were conducted with these threshold values.

Parameter	**Parameter Value**
Sampling Frequency	44100Hz
Sample Resolution	16 bits
Band Pass Filter Frequency Range	200-4000Hz
Filter Coefficients	128
Dif_{thresh}	0.10
V_{thresh}	Dynamically Determined

Table 4.1: Audio system parameter specifications for both experiments.

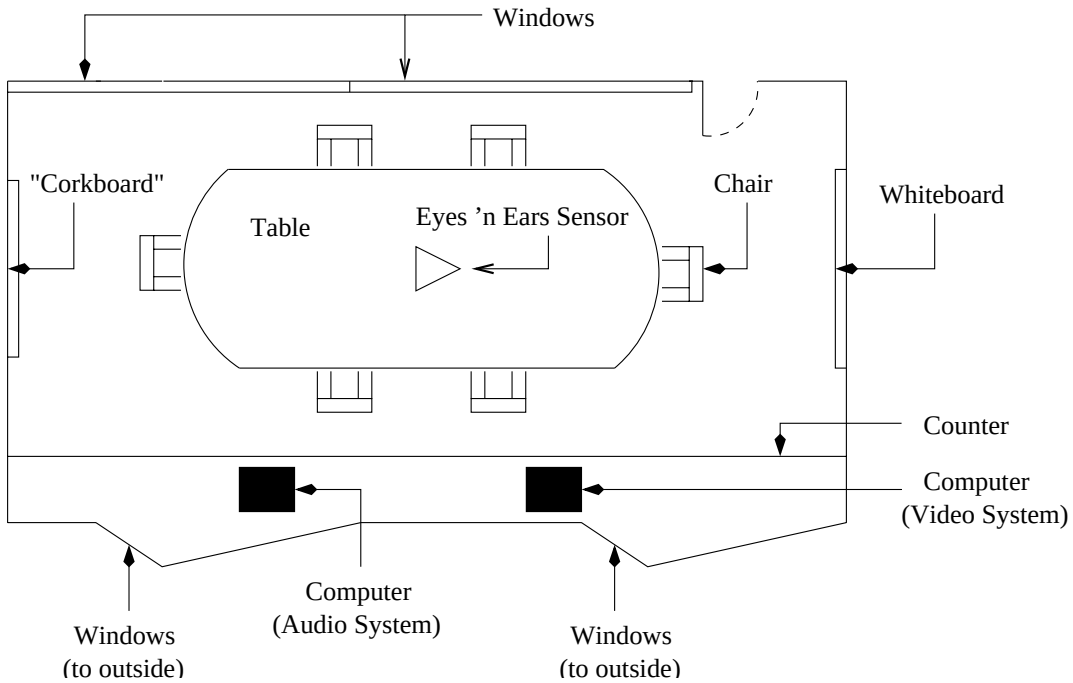

Figure 4.2: Schematic diagram of the meeting room in which both experiments took place.

Threshold	**Threshold Value**
Skin Pixel Classification (Δ)	0.8
Min. Pixels per Histogram Bin (δ)	50
Size Filter	300 (pixels)

Table 4.2: Video system threshold values used for both experiments.

4.2 Experiment One - Demonstration

The purpose of this experiment is to demonstrate the ability of the Eyes 'n Ears system to detect and focus on the participants who are speaking in a typical multiple person group meeting setting. As illustrated in Figure 4.3, four male subjects (where the i^{th} subject is denoted by s_i, for $i = 1 \ldots 4$), were seated around the sensor in the meeting room described in Section 4.1. Subjects were instructed to perform several tasks (the actual tasks are listed in Table 4.3), while the system was operating. The tasks involved different scenarios of the participants speaking in a meeting. For example, two of the scenarios involved having all subjects remain silent for the duration of the test, while in another scenario, two of the subjects would speak at the same time while the others remained silent. The subjects which did speak were free to choose their own words and phrases for the duration of the experiment and spoke in a "normal" loudness level (e.g. they did not speak purposely louder than they normally do). In addition, the subjects were free to change their pose as they wished (e.g. raise their hands, move their head etc.). In each scenario, the system was started, the appropriate subjects began speaking, and 15 seconds later, the system would locate the speakers.

In addition to the participants speaking, as shown in Figure 4.3, a radio was also placed in the room and remained "ON" during the duration of the entire experiment. The radio was tuned to an all news radio station (*CFTR* 680 *News*

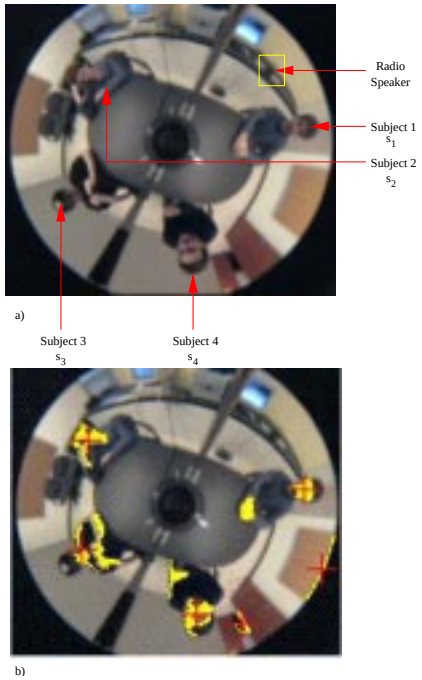

Figure 4.3: Experiment one set-up (left) and sample output (right). Four participants in a typical group meeting, seated around the Eyes 'n Ears sensor. The radio distracter is located counter clockwise of the participant on the right s_1

Toronto, Ontario) where several different announcers, both male and female, were heard. The volume of the radio was set to a moderate level (6 out of 10 possible volume settings) and was very similar, in loudness, to the level of the subject's speech. The radio loudspeaker was placed on the counter close to the sensor (and close to subject s_1) and was meant to simulate audio "distracters" (noise) which may occur in a regular meeting.

The video system detected correctly all participants in the six scenarios. Furthermore, given the real world direction to each detected face (relative to the Paracamera coordinate system), the audio system was capable of localizing participants that were actually speaking in each of the scenarios. It did incorrectly

Scenario Number	Task Description
1	Subject s_2 speaking only.
2	No subjects speaking.
3	Subjects s_2 and s_4 speaking concurrently .
4	No subjects speaking.
5	Subjects s_3 and s_4 speaking concurrently .
6	Subject s_4 speaking only.

Table 4.3: Different scenarios examined in experiment one.

determine that one participant was speaking (s_1 in scenario one and scenario six), when in fact this participant was not speaking. However, subject s_1 was close to the loudspeaker and audio from the radio was associated with the nearby silent participant. In addition to the correctly classified faces, the video system incorrectly classified several non-face skin regions as faces, however, in each of these cases, the audio system correctly determined there was no speech emanating from the direction of the incorrectly detected face.

4.3 Experiment Two - Accuracy

This experiment investigates the accuracy with which the Eyes 'n Ears system locates a speaker in the environment. In order to quantify the results of the experiments, rather than using a human subject, a test dummy speaker was used instead. Using a test dummy ensured certain parameters remained constant over the entire duration of the experiment, and permitted much longer experimental trials to be conducted. After being placed in the appropriate position, the dummy

did not move in any manner ensuring its height above the table and its pose relative to the Paracamera remained static, allowing quantitative measurements to be made.

The test face used in this experiment was a color image of a face. The picture was mounted on an "L" shaped wooden stand and the region of the image corresponding to the mouth on the picture was cut out. A small audio speaker was mounted directly behind it. The speaker was connected to a radio that was tuned to an all news radio station (CFTR 680 News, as in experiment one) for the entire duration of the experiment. This allowed the output of the speaker to emanate from the opening in the mouth simulating a person talking. The volume of the radio was set to six (out 10 possible volume levels, where 10 is the loudest setting).

A 10cm $\times$ 10cm grid was laid out on the table shown in Figures 4.4 and 4.5. The Eyes 'n ears sensor was then centered on the table. The test face was then placed at specific grid locations and face localization was performed at 18 discrete locations. For each location, the image coordinates of the face in the Paracamera were manually obtained by identifying the nose of the test face in the video image. For each test position, the test face was then located three times using the audio and video detection algorithm and these values were averaged.

Figure 4.6 shows the actual (measured) direction to each face position (denoted by $\vec{\beta}_{actual}$ vs. the computed direction to the face (denoted by $\vec{\beta}_{computed}$)

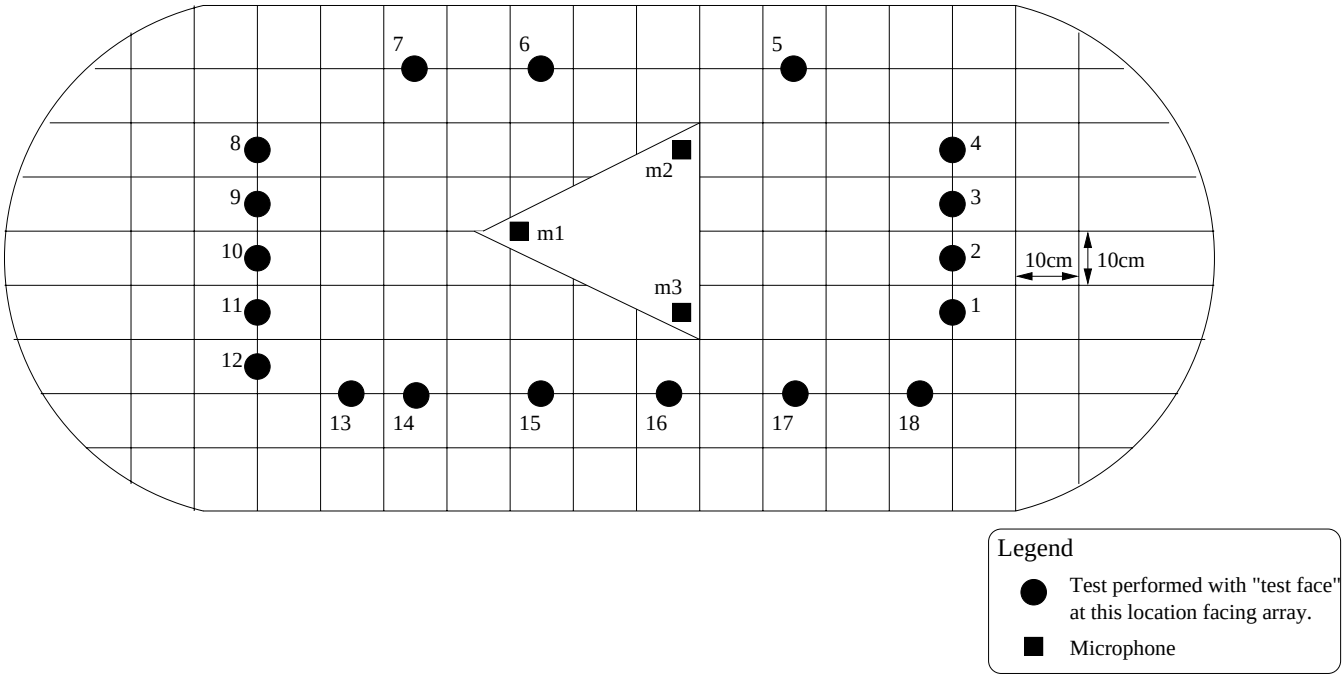

Figure 4.4: Location of test face during each of the tests.

Figure 4.5: Actual image of the experimental setup for experiment two.

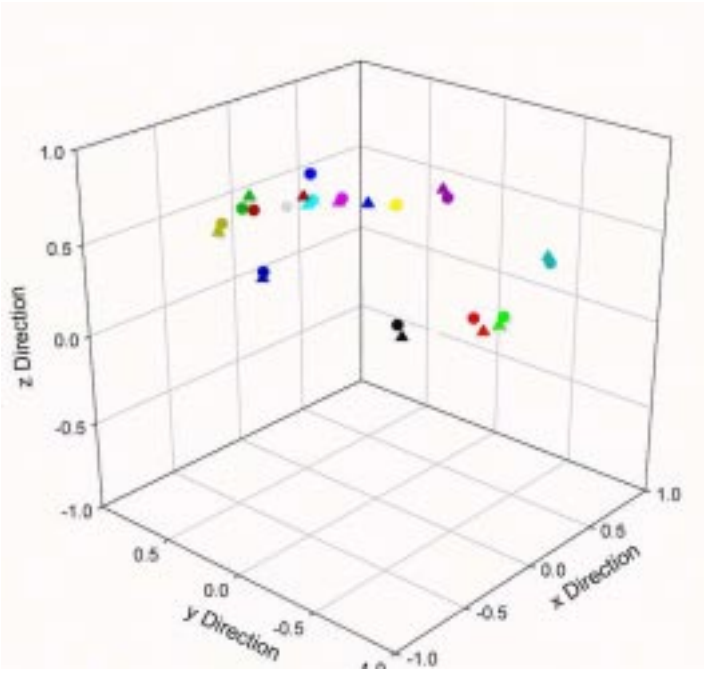

Figure 4.6: Comparison between the actual direction to the face $\vec{\beta}_{actual}$ and the computed direction to the face $\vec{\beta}_{computed}$. The tips of the direction vector, from the origin, are plotted.

in the test (provided that the face was detected). Figure 4.7 plots the difference between the actual and computed directions for each face location ($\epsilon = 1 - \vec{\beta}_{actual} \cdot \vec{\beta}_{computed}$). The video system correctly detected the skin region present in each of the ($18 \times 3 = 54$) tests (100%). A total of nine false positive potential faces were identified by the vision system. The audio system confirmed 45 of the 54 true faces (83% of the true faces were correctly identified by both the audio and video systems), and one false positive face was confirmed by the audio system. In Figures 4.6 and 4.7, only those faces which were confirmed by the audio system are plotted.

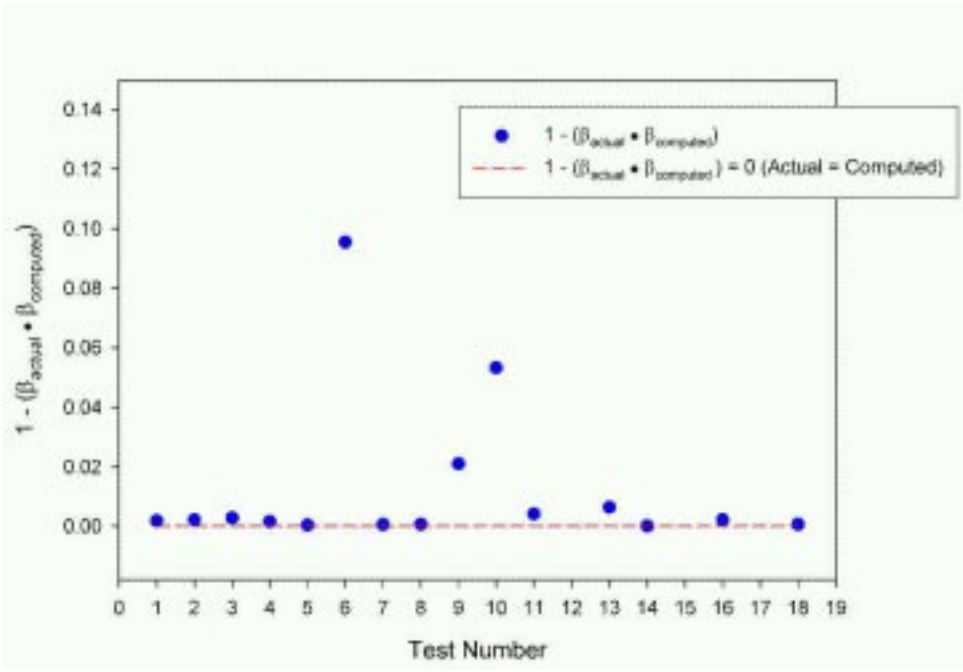

Figure 4.7: Error between the actual and computed direction to the face.

Chapter 5

Summary and Future Work

5.1 Summary

This paper describes a lightweight and portable face detection system that utilizes audio and video cues. The system is intended primarily for use in teleconferencing applications in which the sensor (consisting of the combined audio and video components) is placed on a table and the participants are seated around it. An omni-directional camera (Paracamera), allows the video component to capture dynamic views of all participants from a single viewpoint, eliminating the need for a camera operator or having each speaker move within the camera's view. Using a statistical color model constructed from 35 individuals of various ethnic backgrounds, the pixels of each incoming Paracamera image are classified as either skin or non-skin. Skin classified pixels are grouped into labeled regions. Skin labeled regions, which are spatially close, are further grouped into clusters.

Assuming there is an appropriate amount of space between the people in view, each cluster corresponds to one particular person. The region in each cluster furthest from the center of the Paracamera image is chosen as the face and an estimate of its direction in the real world, relative to the Paracamera coordinate system, is made and provided to the audio system. Beamforming and sound detection techniques with a small, compact microphone array (consisting of four omni-directional microphones), allows the audio system to be steered in the direction of potential faces, and then to focus on the speech of each participant. The detection of sound allows the audio system to confirm and therefore validate, the presence of a speaker in the direction determined by the video system.

Experiments performed in a normal meeting room environment indicate that by working together the audio and video system are capable of overcoming some of the limitations inherent in each component. Various factors may negatively affect each component, but these factors are usually specific to either the audio or video system. For example, a reverberant environment may result in the incorrect localization of a sound source, but will not affect the video system. Similarly, the color of objects in the environment has no bearing on the audio system whereas it may negatively affect the video system and lead to the incorrect classification of non-skin regions as skin (e.g. certain yellow objects, such as a cardboard box or a standard corkboard may be incorrectly classified as skin). By locating the direction to potential faces within the Paracamera's view, the

video system essentially reduces the workspace of the audio system from many thousands of directions to only a few, making the audio system's task tractable.

5.2 Future Work

Various improvements could be made to the Eyes 'n Ears system. Faster computers, and especially more sensitive microphones would enhance system speed and performance. One specific advantage of more powerful computers would be the ability to probe via audio beamforming every large skin region cluster as an audio source, rather than only probing large skin region clusters distant from the centre of the video image. Future extensions to the Eyes 'n Ears systems include incorporating the system within an audio-video tracking system to permit speakers to be attended to and then tracked as they participate in teleconferencing applications, and to apply the system to distance learning applications. Specifically, to use the sensor as a device to enable a remote instructor to interact with his or her remote class.

Appendix A

Eyes 'n Ears Hardware and Software

A.1 Overview

Figure A.1 illustrates the hardware comprising the Eyes 'n Ears system. As shown, the audio and video system communicate as two separate components, connected together through the Internet connection using a client-server model. The video system (server) continuously locates human faces present in the Para-Camera image and provides an estimate to their direction in the real world relative to the Paracamera. The direction information of each face is then provided to the audio system over the Internet connection as possible directions to sound sources (speech of a participant). Upon determining whether there is a source propogating in the direction of each potential sound source, the audio system will provide this information to the video system.

The following sections provide greater detail of the components illustrated in Figure A.1.

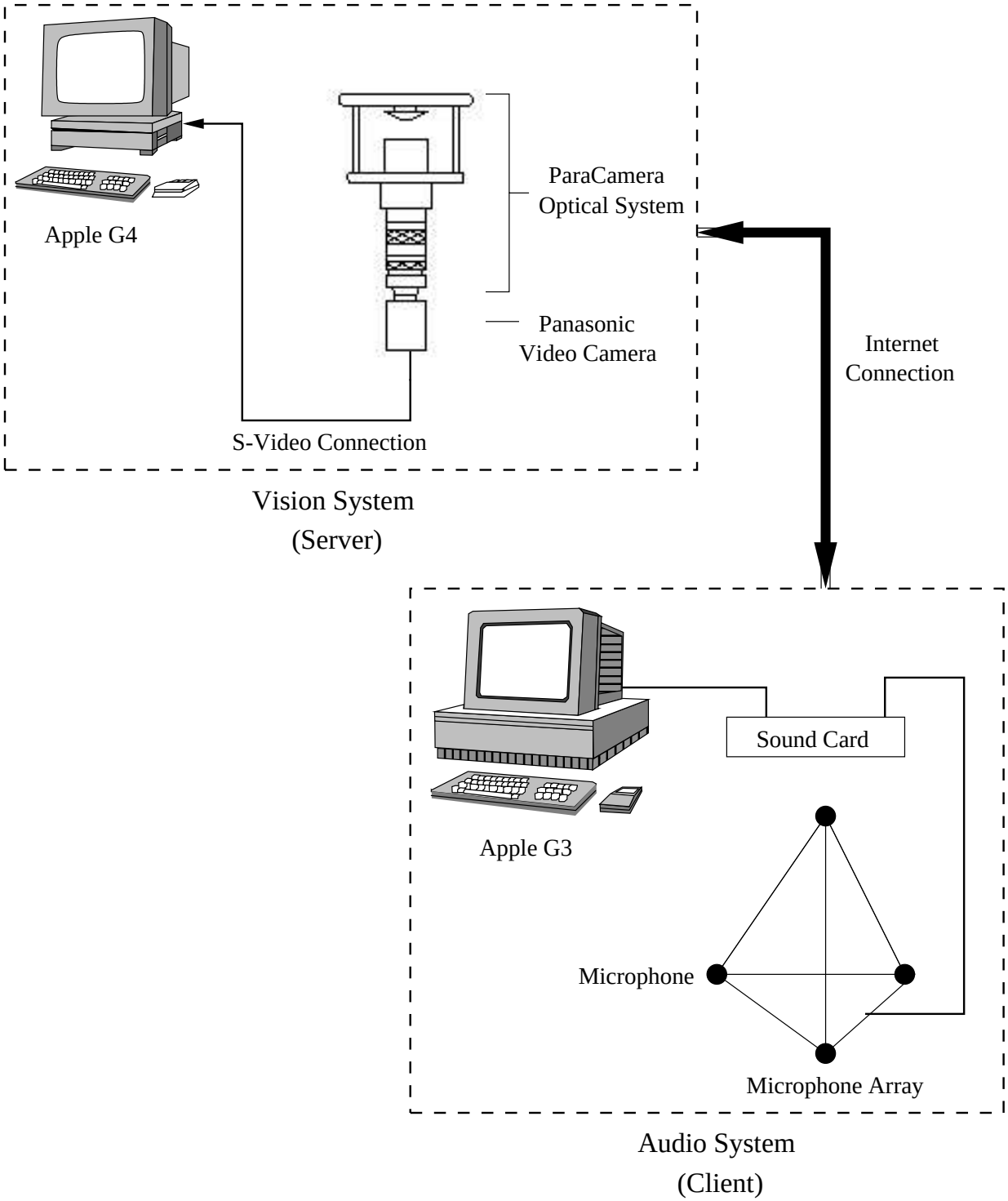

Figure A.1: Eyes 'n Ears Hardware Components.

A.1.1 Video System

Computer

Control of the entire video system is achieved with an Apple Power Macintosh G3 with the specifications given in Table A.1.

Processor	Main Memory	Operating System
Power PC G3 400MHz	128MB	Mac OS 9.0

Table A.1: Video System Computer Specifications.

Video Card

An ATI Xclaim VR 128 video card is used to obtain the images from the camera system. This video card contains the ATI 3D Rage 128-bit accelerator chip allowing for advanced 3-D graphics capabilities (e.g. alpha blending, fog effects, z-buffer, Gouraud shading etc.). It is capable of providing 30 frames per second at a resolution of 320×240 and can support a maximum display resolution of 1600×1200. Finally, the card supports S-Video or Composite Video input, however, for this application, only S-Video input is used.

Video Camera

A Panasonic model GP-KR222 color CCD Camera is used. The camera contains a single $\frac{1}{2}$" CCD with a 640×480 resolution and is predominantly used for

surveillance purposes.

ParaCamera Lens System

The ParaCamera optical system [29] consists of a high precision paraboloidal mirror and a combination of special purpose lenses (see Figure A.2). By aiming a camera to the face of the paraboloidal mirror, the combination of these optics permit the ParaCamera to capture the entire hemisphere surrounding it from a single viewpoint[1]. The single viewpoint feature allows for fast and simple generation of perspective views from the original hemispherical image. As described in Appendix B, it also allows for simple and problem free calibration [39].

Software

The entire software portion of the video system has been developed with Java SDK 2.1 for the Macintosh (equivalent to the Java 1.6 on the Windows/Sun platforms) using Code Warrior IDE version Pro 4.0 programming environment for the Macintosh. The QuickTime 4 for Java API is used to obtain images from the camera. This API is fairly simple to use and certainly abstracts much of the low level details involved with such a process. Further information regarding QuickTime 4 for JAVA may be found on the Internet at:

http://www.apple.com/quicktime/qtjava/index.html

[1] Except for a small "polar cap".

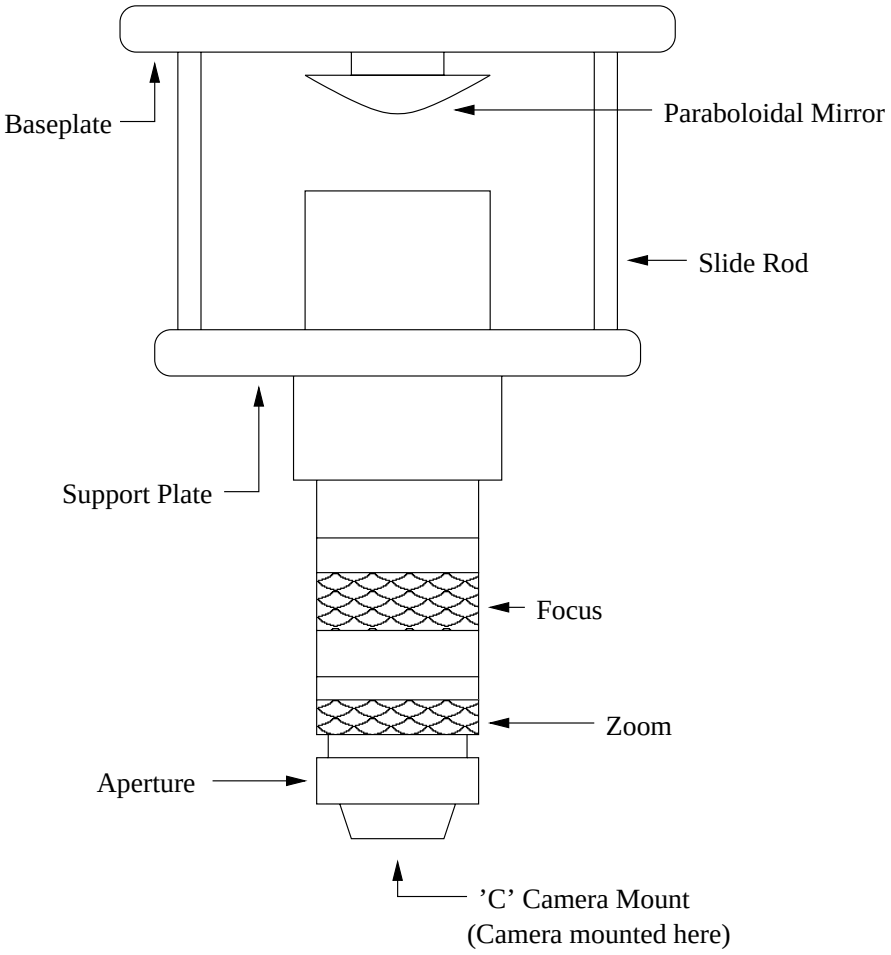

Figure A.2: ParaCamera Optical System.

A.1.2 Audio System

Computer

Control of the audio system is achieved with an Apple Power Macintosh G3 with the specifications listed in Table A.2:

Processor	Main Memory	Operating System
Power PC G3 266MHz	96MB	Mac OS 8.6

Table A.2: Audio System Computer Specifications.

Omni-directional Tie Clip Microphone

Four Genexxa 33-3003 tie-clip electret condenser microphones are used. They are manufactured by Genexxa and purchased from a commercial electronics store for approximately $60.00 (CDN) each. Each microphone contains an acoustic 'wind screen' which according to the manufacturer, minimizes "voice popping" and background noise. Further specifications are provided in Table A.3.

Frequency Response	Sensitivity	Impedance	Dimensions	Power Supply
70-16000Hz	65dB	1000Ω	17.5mm long × 8mm dia.	1.5V

Table A.3: Omni-directional Microphone Specifications.

Figure A.3: Layla Multi-Track Recorder.

Sound Card (Layla)

Since the Power G3 supports stereo audio input only, an external sound card is required to allow for the simultaneous input of data from the four microphones. The 'Layla' (see Figure A.3), manufactured by Echo [8] is used. Layla is a professional digital multi-track recording system for both PCs and Macintosh platforms. It allows for 20-bit multi-track audio recording simultaneously from eight input channels and for the simultaneous audio output from 10 output channels. Software control of the Layla is accomplished using the ASIO API. The following section describes the API in greater detail.

ASIO API

Personal computers currently available are very restricted in terms of their audio capabilities as they allow for stereo input and stereo output only. Although the number of input and output channels can be increased, using for example various DSP boards, this is a complex task with the potential for major problems. In particular, it is very difficult to synchronize the various input and output

channels. To overcome these limitations, Steinberg[2] has developed the Audio Stream Input Output (ASIO) API which allows for efficient audio processing, high data throughput, synchronization and extendibility on the audio hardware side [10]. The ASIO API does not place a limit on the number of input/output channels, sample rate ($32KHz$ to $96KHz$ or greater) or sample format (16, 32, 64 bit or 16/32 floating point), thereby allowing the 10 input and 12 output channels available on the Layla audio card to be accessed simultaneously without concern for synchronization issues etc. The API C++ source code is in the Public Domain and free for download from the following Internet site:

http://www.steinberg.net/developers/ASIO2SDKAbout.phtml

It includes routines to initialize the channels (e.g. set the number of channels, sampling rate, sample format etc.); allow simultaneous input/output from the channels, and routines to terminate the session and return any allocated resource (e.g. memory) to the system. Since all software for both the audio and video systems in this thesis is written entirely in Java, the Java Native Interface [23] is used to incorporate the ASIO API C++ source code into this application.

The Layla must be powered on before the G3. In addition, before using the Layla, the application "Echo Console" must be executed (this application allows one to control the audio Input/Output and clocking functions of Layla). Sound

[2]Steinberg is a trademark of Steinberg Soft-und Hardware GmbH.

will not be recorded unless this is done! Further details regarding Layla, including documentation may be found from the manufacturer's web site:

http://www.event1.com/download/MacLayla.html

Software

Similar to the video system, the software portion of the audio system was also developed with Java SDK 2.1 for the Macintosh using Code Warrior version 3.2. However, as discussed in Section A.1.2 above, the ASIO API obtained from Steinberg, is written in C++ and was modified slightly and incorporated into the Java applications using the Java Native Interface.

Sound System Specifications

Although the ASIO API supports a variety of sampling frequencies, ranging from $32kHz - -96kHz$, the Layla is much more restrictive and supports a sampling frequency of $44.1kHz$ only. Similarly, the Layla is also restrictive with respect to the sample type and number of samples recorded from each channel ("window size"). Each sample is represented by a double precision value requiring 32 bits, while the window size is restricted to 2048[3] (e.g. 2048 samples are obtained after requesting data from a channel).

[3]Although limiting the window size to 2048 samples may seem restrictive, it could be overcome by recording and combining consecutive windows to obtain a window size of any arbitrary length.

According to the Nyquist theorem [22], when a signal is sampled at a rate at least twice the highest frequency it contains, the original signal can be reconstructed. A sampling rate of 44.1kHz allows signals containing frequencies up to $22,050Hz$ to be sampled and recovered with ideally, no distortion. As a result, since the frequency range of human speech is in the range of 20Hz—20kHz[4] [1], any speech signals may be recovered. Such a high frequency rate however results in a large number of samples the system has to process. For example, each second of a sampled signal will generate 44100 samples whereas using half the sampling frequency ($22050Hz$) will result in half the samples to be processed (22050 samples). With the simultaneous recording of four channels, and the large size of each sample, it results in a substantial amount of data processing.

To keep the data processing performed on the sampled signals simple, the 32 bit double precision sample values are converted to a 16 bit integer representation. This may be accomplished by simply casting the double precision value to an integer (truncating and ignoring any decimal portion) and then shifting the bits of this newly formed value 16 places to the left.

[4]The majority of speech signals occur in a much smaller range. For example, the bandwidth of most telephone lines is $300Hz - -3.4kHz$ [22], however, we are clearly capable of communicating with each other using telephones.

Appendix B

ParaCamera Image Calibration

B.1 Description

Given the ParaCamera's single viewpoint constraint, calibration is simple and straightforward and involves determining the size (in pixels) of the following four parameters (see figure B.1):

r_{outer} The radius of the outermost circle of the ParaCamera image.

r_{inner} The radius of the innermost circle of the ParaCamera image.

c_i, c_j The image coordinates of the center of the circular portion of the ParaCamera image.

The values of the above parameters must be determined in order to generate a correct panoramic or perspective view. The following section describes how these values are obtained.

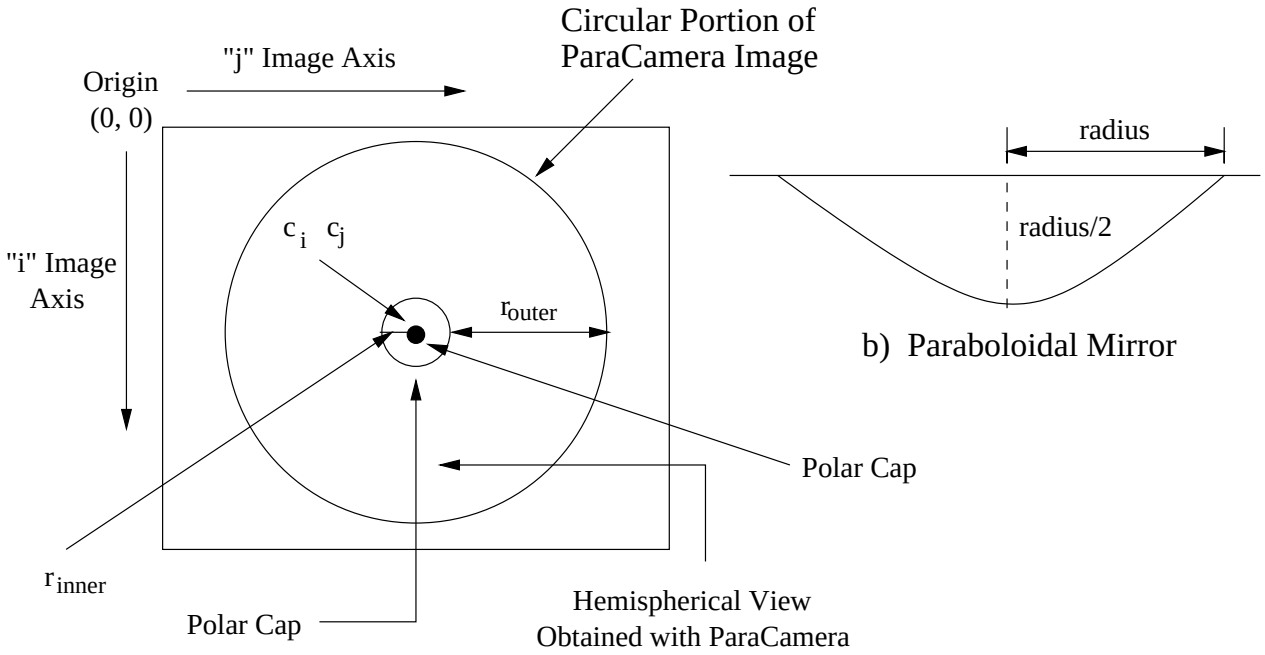

Figure B.1: ParaCamera Image and Paraboloidal Mirror Geometry.

B.1.1 Performing the Calibration

Referring to Figure B.2, determining c_i and c_j is simply a matter of moving (dragging) the "cross hairs" until they are the center of the circular portion of the ParaCamera. After the these two parameters have been determined, r_{inner} and r_{outer} are found. Each value is determined by "sizing" an expandable circle until it is slightly larger than the inner or outer circle respectively.

A simple software application was written allowing the values of the four parameters to be easily determined. A ParaCamera image is obtained and both the cross hair and expandable circle are overlayed on top of it. Using the mouse, the cross hair as well as the expandable circle may dragged to the appropriate

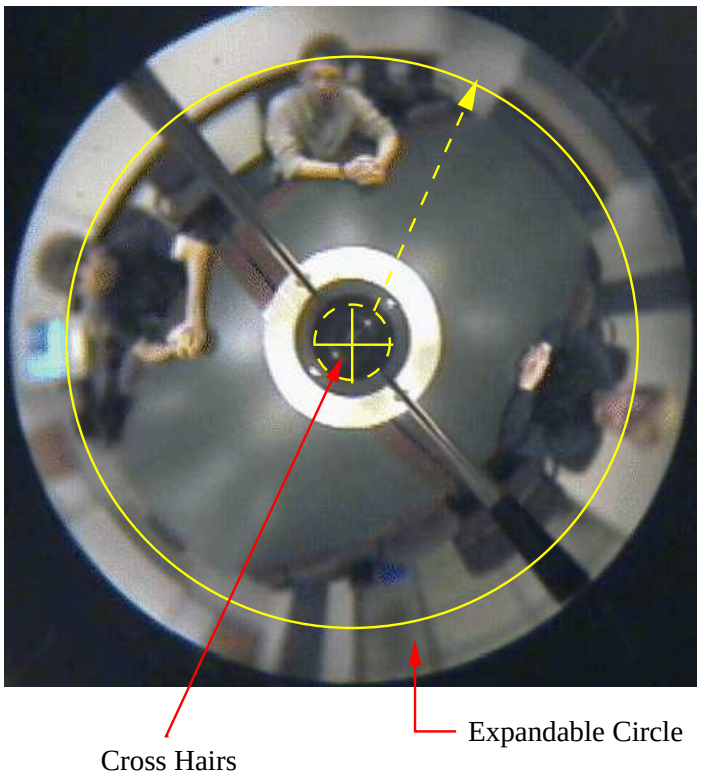

Figure B.2: ParaCamera Image Calibration.

position. Once they are in place, the user may choose to save the corresponding parameter values.

Appendix C

Eyes 'n Ears Microphone Array

C.1 The Array Used in this Work

Figures C.1 and C.2 provide details regarding the microphone array used in this work. As with the video system, the origin of the array coordinate system is located at the vertex of the paraboloidal mirror[1]. In addition, to avoid a complicated calibration procedure and the problems associated with it, the position of each microphone in the array is defined relative to the video system (Paracamera) coordinate system. The position of the microphones will remain fixed in place (e.g. the microphones are not physically moved in any manner). However, their position in the image (i, j coordinates) may change depending on the orientation (e.g. rotation about the z-axis of the Eyes 'n Ears Sensor. This will lead to a change in their real world position (x, y and z coordinates) relative to the Paracamera coordinate system and must therefore be determined once the

[1] A sensor (microphone) does not have to be present at the origin of the array however when it is, it usually simplifies the mathematics required to focus the array to some particular location.

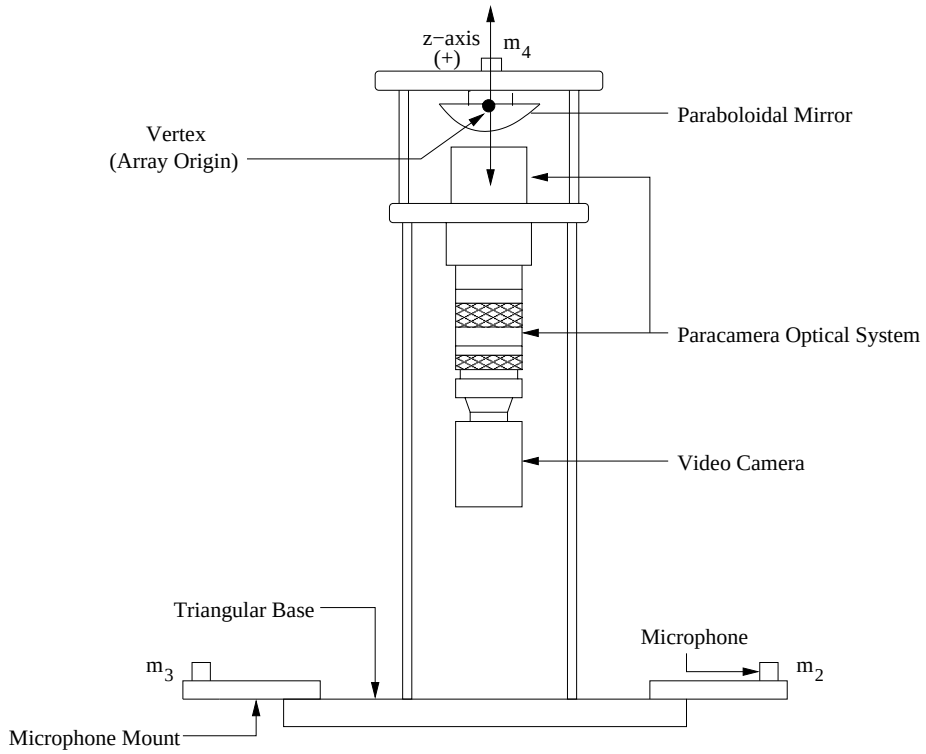

Figure C.1: Overview of the Audio System Microphone Array. The vertex of the paraboloidal mirror defines the origin of the microphone array. The position of microphones m_1, m_2 and m_3 (microphone m_3 is not shown in the figure but lies behind and parallel to m_2) are determined using a simple "point and click" method. The position of microphone m_4 is a simple translation about the z-axis.

sensor has been placed in its desired location and orientation. The position of the microphones can then be determined using the simple initialization procedure described in in the following sections.

C.2 Microphone Array Coordinates

As Equation 3.2 illustrates, when beamforming with a far field acoustical model, the position of each microphone (x_i) relative to the array origin must be known. In this work, the direction to each potential face in the real world (relative to the

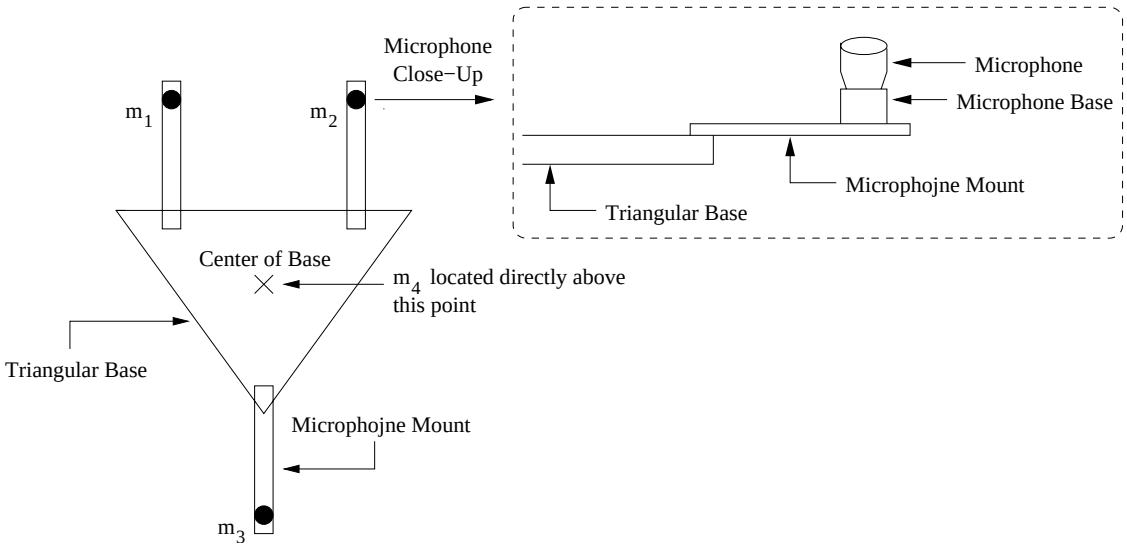

Figure C.2: Microphone Array Base and Microphone Close-Up.

Paracamera) will be determined by the video system. Since a direction will be given relative to the Paracamera coordinate system, rather than defining separate coordinate systems, beamforming will be performed using the same coordinate system. Therefore, the position of each microphone (x, y and z coordinates) must be given relative to the array origin (the vertex of the paraboloidal mirror).

Using the same coordinate system eliminates the need for complex calibration between the audio and video systems required to ensure their coordinate systems are "aligned". However, determining the coordinates of each microphone relative to the Paracamera may also require a complex calibration procedure. Fortunately for this application this is not the case. Rather, a simple "point and click" procedure has been developed allowing one to easily determine the position of the microphones during system initialization.

C.2.1 Determining the Position of Each Microphone

As previously mentioned, the microphones of the array will remain fixed (e.g. they will not be moved in any manner), and therefore, their height relative to the origin (vertex of the paraboloidal mirror) will also remain stationary. As described in Appendix D, the real world position of an object in the Paracamera image can be determined by assuming the point of interest in the object lies on a groundplane perpendicular to the optical axis of the Paracamera at some specific height. Since the height of the microphones remains static and can be easily determined (by physically measuring it), the method described in Appendix D can be used to determine the position of the microphones relative to the coordinate system of the video system. This is precisely the approach taken in this work in order to determine the position of microphones m_1, m_2 and m_3. As shown in Figure C.3, these three microphones are visible in an image obtained with the video system. During system initialization, an image obtained by the video system (see the left image in Figure C.3) is presented and the coordinates of the microphones in the image must be specified by manually "pointing and clicking" (e.g. pointing the mouse and clicking on the position of the microphone in the image). Once the image coordinates have been specified, the method described in Appendix D is used to determine the position (x, y and z coordinates) of the three microphones in the real world relative to the video system coordinate system.

Since microphone four is not visible in an image obtained by the video system,

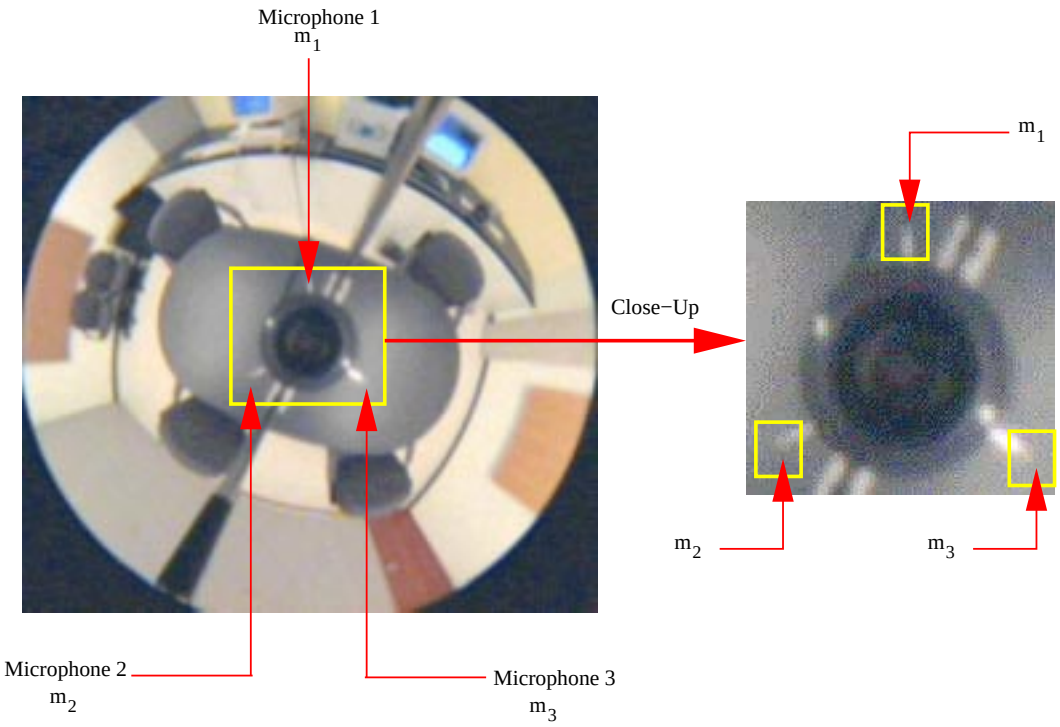

Figure C.3: Determining the Position of the Microphones. Image coordinates of the three microphones are obtained by clicking to the center of each individual microphone in the image. The position of the microphones in the real world can then be determined using the procedure described Section D.

this procedure cannot be used to determine its position. However, as shown in Figure C.1, microphone four is directly above the vertex of the paraboloidal mirror and therefore corresponds to a simple translation (which is easily measured) on the z-axis of the video system coordinate system. Therefore, its real world position can also be determined.

C.2.2 Advantages of the "Point and Click" Method

In addition to eliminating the need for complex calibration ensuring the coordinate systems of both the audio and video systems are aligned, the "point and click" method previously described allows the Eyes 'n Ears sensor to be placed on a table (or any surface) in any orientation (rotation about the z-axis of the video system coordinate system). Regardless of the orientation of the sensor, determining the the direction to a face in the real world relative to the Paracamera can be easily determined. This allows beamforming relative to the Paracamera coordinate system to be performed without any transformation (translations and rotations) of the direction information provided by the video system, thereby reducing the potential for errors associated with such transformations and simplifying the procedure.

C.2.3 Drawbacks (Problems) with the "Point and Click" Procedure

As illustrated in Figure C.3, the main drawback of the "point and click" method is actually determining the position of the microphones in the image during initialization. As shown, the microphones in the image are rather small (but nevertheless consist of several pixels in the image) and appear blurry, making the task of locating the exact center of the microphone in the image a difficult and error prone[2]. As a result, the actual image coordinate may not necessarily correspond to the actual center of the microphone leading to errors in the position estimate.

Despite the potential for the errors, this does not pose any significant problems. The microphone itself occupies a very small portion of the image and as long as the center of the microphone is chosen to be somewhere in the image region corresponding to it, the error will be rather small. Furthermore, informal lab surveys indicate that after repeating the "point and click" operation several times, the difference in the image coordinates of the microphone center is actually small, usually between one or two pixels in each of the i, j image coordinates. Finally, the exact center of the microphone in the image may not actually be required. the microphone itself is certainly not a "point receiver" and as a result, the center of the microphone may not necessarily define its position in the real world correctly. Therefore, it is sufficient to chose the image coordinates

[2]Even if a higher resolution and sharper image was provided, clicking at the exact center of the microphone may still be a difficult task given the microphone consists of several pixels and their small size.

anywhere in the region occupied by the microphone in the Paracamera image.

Appendix D

Converting Image Coordinates to Directions in the "Real World"

Given the coordinates (i, j) of an object in a single Paracamera image, its position (x, y and z) in the real world can be determined only if a "ground-plane" perpendicular to the optical axis of the Paracamera is assumed [12]. Given this assumption, real world coordinates can be determined using the method described in [19], where tracking of multiple people using a Paracamera is achieved by detecting changes between an adaptive background model and foreground objects. In [19], the Paracamera was mounted on the ceiling and the floor directly beneath the Paracamera was perpendicular to the optical axis of the camera. World coordinates of the point at which each person being tracked touch the floor ("floor pixels") were then determined by extending the line from the point the person was touching the floor in the Paracamera image so that it intersected the floor in the real world.

For this application, the Paracamera is placed on a table with the participants

seated around it. It is the real world direction of the face of each participant being sought in order to supply the audio system with potential directions to sound sources (speech of the participants). When the actual height of the face of each seated participant is known, the method described in [19] can be used to determine the real world position (x, y and z coordinates) of each participants face (given the position, the direction can easily be found). However, determining the height of persons face in the real world by physically making this measurement, is clearly impractical. In addition, even if the height measurements were obtained, the seated participants will surely not remain motionless and will undoubtedly move their head, change their posture or adjust the height of the chair, making the initial height measurement invalid. Finally, rather than physically measuring the height of each seated participant's face, an average height may be assumed for each participant. Once again, this assumption is impractical and will lead to errors, as the seated participants will not remain motionless causing their height to vary due to the considerations listed above. Furthermore, the height of the participants may vary substantially making such a height assumption invalid.

The ground-plane assumption may be relaxed thereby eliminating the need to make any height measurements of the participant's face or removing any height assumptions in place, by determining a direction (unit vector) to the face of each participant as opposed to an actual (x, y and z) position. This corresponds to the *far* field acoustical model as opposed to the *near field* model which requires the

position of the sound source as opposed to a direction only. The following section provides greater details regarding the calculation of both the position (with the ground-plane perpendicular assumed to be at some height g) of and the direction to a face in the real world.

D.1 Determining the Position of a Face in the Real World

As illustrated in Figure D.1 and described in [19], since the focus of the paraboloidal mirror is also the center of projection, the line r_{ref} (*reflection line*), between the focus of the paraboloidal mirror and the point of the reflection of the head (face) on the mirror's surface, will intersect the head's location in the real world. Furthermore, the azimuth angle in this ground-plane is the same as the azimuth angle in the image. By extending this line so that intersects the ground-plane, the world coordinates at this point of intersection may be found, as described below.

World coordinates of a pixel in the Paracamera image with coordinates (i, j) are obtained by first converting Paracamera image coordinates (i, j) into polar coordinates (r, θ).

$$r = \sqrt{(i - c_i)^2 + (j - c_j)^2} \tag{D.1}$$

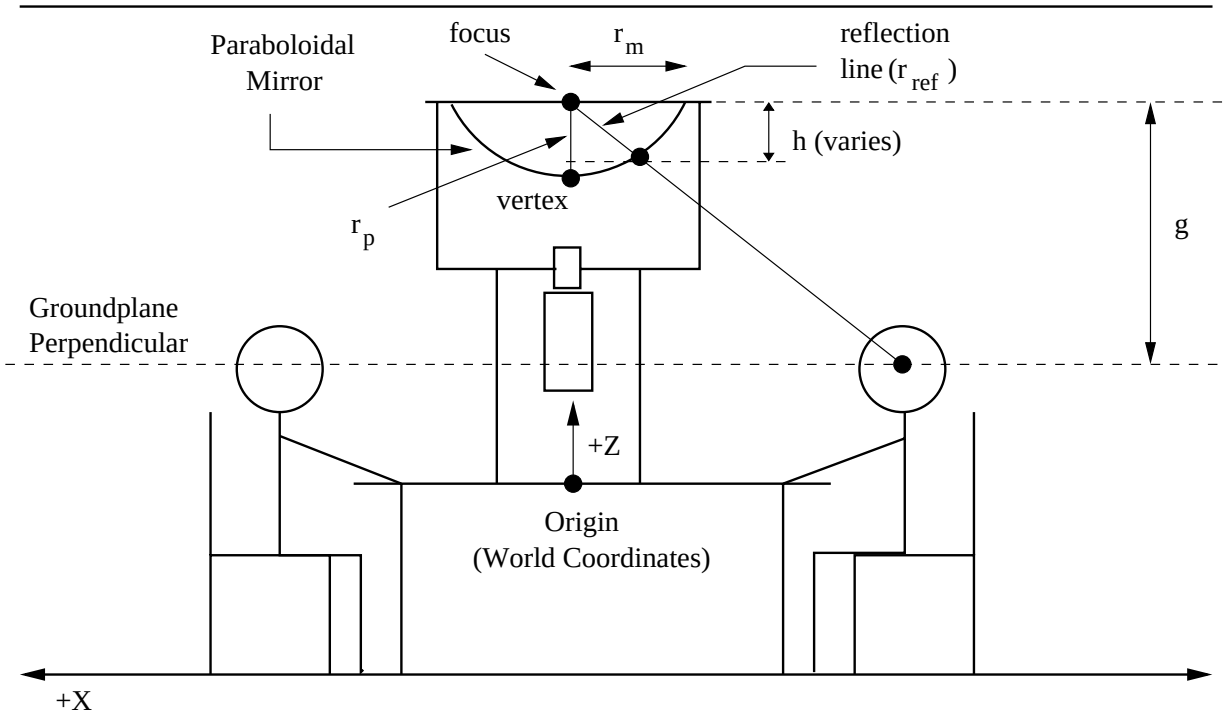

Figure D.1: Geometry to Convert Paracamera Image Coordinates to Positions in the Real World.

$$\theta = \sin^{-1}(i, j) \tag{D.2}$$

where c_i and c_j refer to the center of the Paracamera image. Once the polar coordinates have been determined, the x and y world coordinates of the point of intersection obtained by extending line r_{ref} until it hits the ground-plane are found using

$$z(r) = \frac{r_m^2 - r^2}{2 \times r_m} \tag{D.3}$$

$$x_{\text{world}} = \frac{g \times r \sin\theta}{z} \tag{D.4}$$

$$y_{\text{world}} = \frac{g \times r \cos\theta}{z} \tag{D.5}$$

$$z_{\text{world}} = g \tag{D.6}$$

where $z(r)$ represents the surface equation of the Paracamera's paraboloidal mirror, r_m is the radius (in pixels) of the Paracamera image, $r_p = \frac{r_m}{2}$, x_{world} and y_{world} are the x and y world (room) coordinates respectively. In addition, g is the assumed height of the ground-plane perpendicular and h is the is the height (in meters) of the focus of the paraboloidal mirror and the point corresponding to $z(r)$.

D.2 Determining the Direction of a Face in the Real World

As shown in Figure D.1 and as previously described, the line r_{ref} (*reflection line*), between the focus of the paraboloidal mirror and the point of the reflection of the head (face) on the mirror's surface, will intersect the head's location in the real world. In other words, the line r_{ref} "points" in the direction of the head in the real world. Given the image coordinates i, j of the object of interest in the Paracamera image, the Paracamera coordinates ($x_{paracam}$, $y_{paracam}$ and $z_{paracam}$)

of the "tip" of this line (e.g. the point the line intersects the paraboloidal mirror), can be determined as follows:

$$\begin{aligned} x_{paracam} &= i - c_i \\ y_{paracam} &= j - c_j \frac{y}{s} \\ z_{paracam} &= z(r) \end{aligned} \tag{D.7}$$

where, as previously described, $z(r)$ represents the surface equation of the Paracamera's paraboloidal mirror and c_i and c_j are the coordinates of the center of the Paracamera image. Finally, the real world direction vector (unit vector), with individual components x_{dir}, y_{dir} and z_{dir} is determined by normalizing the components listed in Equation D.7:

$$\begin{aligned} s &= \sqrt{(x_{paracam}^2) + (y_{paracam}^2) + (z_{paracam}^2)} \\ x_{dir} &= \frac{x_{paracam}}{s} \\ y_{dir} &= \frac{y_{paracam}}{s} \\ z_{dir} &= \frac{z_{paracam}}{s} \end{aligned} \tag{D.8}$$

Bibliography

[1] J. Adler. Virtual audio: Three-dimensional audio in virtual environments. Technical Report T96-03, Swedish Institute of Computer Science, 1996.

[2] A. Basu and D. Southwell. Omni-directional sensors for pipe inspection. In *IEEE Trans. Syst. Man Cybern*, volume 25, pages 3107–3112, 1995.

[3] T. Boult. Dove: Dolphin omni-directional video equipment. In *Proc. IASTED Int. Conf. On Robotics and Automation*, Honolulu Hawaii, 2000.

[4] T. Boult, R. Michaels, P. Gao, C. Lewis, W. Yin, and A. Erkan. Frame rate omnidirectional surveillance and tracking of camouflaged and occluded targets, 1998. http://www.eecs.lehigh.edu/ tboult/TRACK/LOTS.html.

[5] M. Brandstein, M. Adcock, and H. Silverman. A practical time-delay estimator for localizing sound sources with a microphone array. *Comput. Speech. Lang.*, 9:153–169, 1995.

[6] D. Chai and K. Ngan. Face segmentation using skin-color map in videophone applications. *IEEE Trans. Circuits Syst. Video Technol.*, 9(4):551–564, June 1999.

[7] US Robotics Company. Teleconferencing systems: Conference link cs1000 and cs1050.

[8] Echo Corp. *Layla by Echo.* 6460 Via Real, Carpenteria, CA USA, 1999.

[9] Panasonic Corp. KXC-AP150 video communication terminal with detachable hand held color camera unit.

[10] Steinberg Corp. *ASIO.* Hamburg, Germany, 2000.

[11] J. Crowley and F. Berard. Multi-modal tracking of faces for video communication. In *Proc. Conf. Comput. Vis. Pattern Recogn.*, Puerto Rico, 1997.

[12] K. Danilidis. Personal communication.

[13] A. Davis. *Integrated Collaboration: Driving Business Efficiency into the Next Millennium.* Forward Concepts, 1999.

[14] J. Flanagan, D. Johnston, R. Zhan, and G. Elko. Computer steered microphone arrays for sound transduction in large rooms. *J. Acoust. Soc. Am.*, 78(2):1508–1518, 1985.

[15] J. Foley and A. VanDam. *Computer Graphics Principles and Practice.* Addison-Wesley Publishing Company, USA, 1996.

[16] D. Forsyth. A novel algorithm for color constancy. *Int. J. Comput. Vis.*, 5(1):5–36, 1990.

[17] L. Freed. Microsoft netmeeting 3.0. *PC Magazine*, May 2000.

[18] K. Guentchev. Learning based three dimensional sound localization using a compact non-coplanar array of microphones. Master's thesis, Department of Computer Science, Michigan State University, MI, USA, 1997.

[19] D. Gutchess, A. Jain, and S. Cheng. Automatic surveillance using omni-directional and active cameras. In *Proc. Asian Conf. Comput. Vis.*, 2000.

[20] S. Hans, H. Anderson, and E. Granum. Skin color detection under changing lighting conditions. In *Proc. 7th Symp. Int. Rob. Sys.*, pages 187–195, Columbia, Portugal, July 1999.

[21] R. Herpers, K. Derpanis, D. Topalovic, and J. Tsotsos. Detection and tracking of faces in real environments. In *Proc. Int. Workshop on Recognition, Analysis and Tracking of Faces in Real-Time Systems*, pages 96–104, Korfu, Greece, September 1999.

[22] W. Hioki. *Telecommunications.* Prentice Hall, Englewood Cliffs, NJ USA, 1990.

[23] C. Horstmann and G. Cornell. *Core JAVA: Advanced Features*, volume 2. Sun MicroSystems Press, Palo Alto, CA, USA, 1998.

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

[25] M. Jones and J. Rehg. Statistical color models with applications to skin detection. Technical Report CRL 98/11, Compaq Computer Corp., Cambridge, MA USA, 1998.

[26] J. Mckenna, S. Gong, and Y. Raja. Modeling facial color and identity with gaussian mixtures. *Pattern Rec.*, 31(12):1883–1892, 1998.

[27] B. Mehtre, M. Kankanhalli, A. Marasimhalu, and G. Man. Color matching for image retrieval. *Pattern Recogn. Lett.*, 16:325–331, 1995.

[28] R. De Mori, editor. *Spoken Dialogues with Computers*, chapter 2, pages 23—67. Academic Press Limited, London UK, 1998.

[29] S. Nayar. Omnidirectional video camera. In *Proc. DARPA Image Understanding Workshop*, pages 235–241, New Orleans, LA, 1997.

[30] V. Peri and S. Nayar. Generation of perspective and panoramic video from omnidirectional video. In *Proc. DARPA Image Understanding Workshop*, pages 243–245, New Orleans, LA USA, 1997.

[31] PictureTel. http://www.picturetel.com.

[32] J. Pitman. *Probability.* Springer Verlag, New York, NY USA, 1993.

[33] D. Rabinkin. Digital hardware and control for a beam-forming microphone array. Master's thesis, Department of Electrical Engineering, Rutgers University, New Brunswick NJ USA, January 1994.

[34] Y. Raja, J. McKenna, and S. Gong. Segmentation and tracking using color mixture models. In *Proc. Third Asian Conf. Comput. Vis.*, January 1998.

[35] G. Reid. Active binaural sound localization: Techniques, experiments and comparisons. Master's thesis, Department of Computer Science, York University, Toronto, Ontario, Canada, April 1999.

[36] J. Renomeron. Spatially selective sound capture for teleconferencing systems. Master's thesis, Department of Electrical and Computer Engineering, Michigan State University, New Brunswick, NJ, USA, October 1997.

[37] R. Stiefelhagen, J. Yang, and A. Waibel. Modeling focus of attention for meeting indexing. In *Proc. ACM Mult. '99*, pages 3–10, Orlando, FL USA, October 1999.

[38] M. Swain and D. Ballard. Color indexing. *Int. J. Comput. Vis.*, 7:11–32, 1991.

[39] Cyclovision Technologies. *ParaServer 1.0: User's Guide.* 295 Madison Ave., New York NY USA, April 1998.

[40] J. Terrilion and S. Akamatsu. Automatic detection of human faces in natural scene images by use of a skin color model and of invariant moments. In *Proc. Third Int. Conf. Automatic Face and Gesture Recognition*, pages 112–117, Nara, Japan, April 1998.

[41] J. Yang and A. Waibel. A real time face tracker. In *Proc. WACV '96*, Sarasota, FL, USA, 1996.

[42] Y. Yasushi. Omni-directional sensing and its applications. *IEEE Trans. Inf. & Syst.*, E82-3, March 1999.

[43] R. Yong, A. Gupta, and J. Cadiz. Viewing meetings captured by an omni-directional camera. In *ACM Trans. Comput.-Hum. Interact.*, March 2001.

[44] J. Zheng and S. Tsuji. Representation for route recognition by a mobile robot. *Int. Conf. Comput. Vis.*, 9(1):55–76, 1992.

[45] D. Zotkin, R. Duraiswami, V. Philomin, and L. Davis. Smart videoconferencing. In *Proc. Int. Conf. Mult. Expo.*, pages 3107–3112, New York City, NY USA, August 2000.