

Research Article

Audio Interaction in Computer Mediated Games

J. R. Parker and John Heerema

Sport Technology Laboratory, Faculty of Kinesiology, University of Calgary, Calgary, AB, Canada T2N 1N4

Correspondence should be addressed to J. R. Parker, jparker@ucalgary.ca

Received 26 April 2007; Revised 12 August 2007; Accepted 13 October 2007

Recommended by Michael J. Katchabaw

The use of sound in an interactive media environment has not been advanced, as a technology, as far as graphics or artificial intelligence. This discussion will explore the use of sound as a way to influence the player of a computer game, will show ways that a game can use sound as input, and will describe ways that the player can influence sound in a game. The role of sound in computer games will be explored some practical design ideas that can be used to improve the current state of the art will be given.

Copyright © 2008 J. R. Parker and J. Heerema. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

The term *video game* implies a natural bias towards visual interaction in the typical computer game. It is true that most people gain about 70–80% of their perceptual input from vision [1–3]. Yet a large number of people function very well with a significant degree of visual impairment. In addition, video games that lack sound have never been popular, in spite of excellent graphics and other features. Sound is something that is assumed to be there, and that can influence game sales negatively when it is badly done; it is rarely if ever the cause of high sales when well done and developers do not expect sound and music to be the major selling point.

Sound is a key aspect of a modern video game [4]. The development team either contains musicians or some are hired to create a sound track, usually at great expense. Foley operators, those persons responsible for making sound effects and placing them in the game (or movie) are also key to a development team. A clear example is the motion picture *Cast Away* in which a lot of the sound that the audience thinks is normal background is actually inserted after the fact. In the sequence where the main character escapes from the island, almost all of the sound is generated by a foley artist, and was not recorded on location. Frequently the real sounds do not impress the audience as being realistic!

Video games also have the equivalent of foley artists; although many effects are now recorded on CDs and sold as

sets, there is still a need for sound design and certainly for music composition. Many games have musical themes for each character, and segues for each character between each activity; composition is a key part of the overall design and “feel”.

A knowledge of how audio works, why it is important, how musical assets fit into the scheme of a game, and how they are manipulated is crucial to the game development process, although it is rare for sound to be the actual primary feature of a game. Acceptable audio is merely expected, and so games with poor audio do not sell very well; games with excellent audio may or may not sell depending on other content.

2. WHY IS AUDIO IMPORTANT?

At the lowest level, sound carries with it the sense of another presence and of activity. For example, as a character is made to walk down a street, the activities of other characters can be inferred from the sounds they are making: blacksmiths are pounding hammers on anvils, passing cars generate positional engine noise, and so on. There is a sense of things going on even though they cannot be seen. This is also one definition of background music: it is there to make the player feel less alone (i.e., connected with something else, or continuity) and to remind them that the game is going on, even if they are drawn away for the moment. It is also about activity

in the simple sense that fast music is associated with a great deal of activity, and slow music is associated with little activity.

More importantly, sound has been seen to carry with it more emotional content than any other part of the game, in that sounds trigger feelings and memories. It is said by psychologists that humans learn first by seeing, next by hearing. However, the sense of hearing connects to the limbic system, where memories can be recalled using sounds like music and voices, and loud and sudden sounds cause an immediate startle reflex in fear [5, 6]. The mechanism of this is not widely agreed upon.

However, to see why this might be, imagine the life of a primitive human living in Africa some hundreds of thousands of years ago. They are a small, slow, and relatively weak primate with no sense of smell. They have a good brain, but no technology yet, and are still vulnerable to many predators. Now, predators of various types tend to make similar sounds—low-pitched growling, rumbling sounds. It should be no surprise that this kind of sound still makes humans fearful—many generations of being a prey animal could easily have done that.

In fact, it should be understood that hearing can be used to generate fear in a multimedia narrative than can vision for other reasons. If a predator can be seen, then it is probably less of a threat than if it can only be heard. A prey animal that can only hear the predator is in an unknown amount of trouble, and it pays to believe the worst.

There are other sorts of limbic affecting noises in specific categories. The sudden, sharp noise is a surprise, and it makes sense to react instinctively. It could be a branch breaking or a rock falling. Humans make certain kinds of noise when in pain or fear—screaming, usually high-pitched, ululating, or descending. As humans we tend to react with emotional similarity when we hear such sound, not in sympathy so much as in fear of whatever is inflicting pain or fear on the other. Loud noises are generally disturbing because anything that could make such a noise is big enough (and close enough!) to be a real danger [6].

So there is a good reason why some sounds might affect us at a profound emotional level, and this can be used by designers to provide an interesting emotional experience in multimedia. There are still more reasons why sound is interesting: it cannot be ignored as easily as vision (cannot “close your ears” as it were); humans use it directly for communication to a greater extent; people have the ability to isolate one sound out of many simultaneous sounds and pay attention to it; and so on.

3. AUDIO IN VIDEO GAMES

Current video games use audio in four distinct aspects: music, speech, effects, and input.

Music carries with it a continuity aspect (background), an emotion aspect (spooky VS comical), and a tempo aspect (fast VS slow). This is pretty well known in both music and game development quarters. The use of character themes is standard practice (e.g., a good example being *The Simpson's*

Hit and Run [7]) as is composing a segue between activities (walking to running). Automating more of this might be a valuable exercise, not to eliminate musicians but to provide them with more possibilities.

Speech is used in games that involve human-like characters to communicate information to player in a natural way. One can overhear conversations that give valuable information, or one can be given a briefing or command from non-player character (or NPC, controlled by the artificial intelligence part of the game software). Sometimes the speech is almost purely for entertainment or mood. An example is when opponents make threats or congratulatory noises, or when the speech is by a voice known to the player (James Bond or Bart Simpson).

Sound effects serve multiple roles. They can serve as confirmation that a requested activity has taken place: as in, “yes, the missile was launched.” They serve as a warning that something has happened—a branch broke—or is about to happen—footsteps are approaching, look out! Effects need to be representative of the sounds that things make in the world, insofar as the objects in the game are real. As such, another function of effects is to add to a sense of reality and presence in an environment. Rainfall is both seen and heard, and nearness to the ocean or a river is often heard before it is seen.

Audio input to a game is almost always as a human voice, as in *Nintendogs* [8]. This will be discussed in more depth very soon, but it is enough for now that audio input is quite rare in computer games at this time.

Speech is easy to record, and if specific voice talent is not required this can be very inexpensive. Sound effects can be purchased as sets of CDs, and once bought can be used in many games. Rarely a designer might need to record some new effects, but these can be reused and in some cases can even be dumped into a collection and sold. Special purpose music is farmed out, and is composed and recorded by studios for a fee. This is not cheap, but is less expensive than maintaining a staff of musicians and composers and building and operating a recording studio. A lot of music is purchased and royalties are paid; well-known bands and their music are used for many games, good examples being *Grand Theft Auto* [9] and *1080° Avalanche* [10]. Usage fees allow the original recordings to be used, which sound correct to the player, as opposed to remixes or cover versions.

Games use MP3, WAV, or other standard format to store sounds on the CD or DVD, and simply play them back when they are needed. There is a limited degree of mixing going on, meaning that few sounds need to be played at the same time and their relative volume is specific by the game developer and is played back at that level. Overall levels are relative, so if someone starts speaking while the music plays, the software does something simple to maintain intelligibility levels while still playing the music.

So, in summary: how do games use sound adaptively to create an interesting interface? The theme music and segue is one, as is the use of music for emotion and tempo. Alerting the player to actions using sound effects, and speech as a narrative component is another. However, so far the use of sound is mostly reactive rather than interactive.

4. AUDIO GAMES

An audio game is one that uses sound rather than pictures as the main game state display modality [11]. It is not known what the first audio game was, and it may not be possible to find out. It was quite possibly *Bear Hunt*, a variation on *Hunt the Wumpus*, an old text-based Unix game. *Bear Hunt* was devised in the early 1980's by a company named *QSound* (Calgary, Alberta, Canada). The idea was simple—the player would stand in the field of some stereo speakers with a gun (a toy). Once in while a bear would move towards the player from a particular direction and the player would be expected to shoot it. They could only hear the bear, of course, and there was no visual display. The game was in support of their 3D sound product.

There are relatively few audio games, many intended for play by the vision impaired. It is very likely that the full potential of audio as a display mode is not even near being reached. An examination of some audio games may be useful so as to see what aspects of sound are being used.

4.1. Karaoke revolution

Karaoke is two Japanese words: *Kara* = missing and *Oke* = orchestra. Most people know what it is about, and the appeal of the game is that it can be social, among friends, or it can be used for practice for the shy types.

In this game, the player has access to a selection of songs [12]. Choose one and it starts to play along with the printed lyrics moving on the screen. It is therefore not completely audio (nor are most games completely video!) Players are expected to sing along, and the computer scores according to how well (accurately) the melody is followed. The system extracts the main frequency in the player's singing and matches it to that of the original song.

This is actually a lot of fun, and the display is really not needed, since most folks know the words for the song they are going to sing. One problem—ornamentation (see Section 4.7.2). If the song is not sung exactly the way the original singer does, a penalty will result. Ornamentation and artistic license is punished.

4.2. Shades of Doom

Shades of Doom' is, like the original *Doom* game [13], a first-person shooter, but it is audio only. *Shades of Doom* follows the same basic plot of *Doom* as well—a research facility has suffered an accident, and the player is expected to fix it. The player guides their character through tunnels, hallways, and chambers of the laboratory and tries to shut the experiment down.

There are no graphics in this game. All cues are audio. Players can collect items and encounter (and kill) creatures by using the arrow keys to travel the maze of rooms. They can get navigational assistance if needed. The positional sound is good, but requires a large degree of experience to make use of, and many beginning players get killed early.

4.3. Mueckenjagd (Mÿckenjag, or "Mosquito Hunt")

There are Flash animations with this game, but again, it is played using sound primarily. The player hears a mosquito buzzing about. Orientation is controlled by keys, and the space bar smacks the player's hands together, with the hope of crushing the insect between them.

This game is self-voicing and only available in the German language. The entertainment value does not depend on the language, though—information provided by speech is not required.

4.4. Bobby's revenge

A "shooter", but an odd one. Santa Claus flies past, and the player can hear him using the position of the sound of bells and such. Then an effort is made to shoot Santa with a paintball gun. Santa will return if he is shot with an electrical bolt (using a key). This is a bit perverse, but fun—again, the positional sound is pretty good. Again, the game has a significant learning period.

4.5. Top speed 2

A racing game, a very popular genre that uses sound only to guide the car. Players can hear the cars whizz past in 3D sound, and the sound of the side of the road allows steering to be done. It has a vast number of tracks, many cars, up to 7 opponents, 4 or 5 different cars, and the ability to customize. It is free for downloading.

4.6. Ten pin alley

With a name like *Ten Pin Alley* this game can only be about bowling. The background sounds are quite convincing, but the announcer is irritating. Simple tones guide the player in aiming the ball. The demo version is freely downloadable, and it permits two frames to be played.

These games and about 195 more can be found at the web site <http://www.audiogames.net> [14]. The sheer number of audio games available is impressive, but many games cannot be downloaded, some are not available anymore, and a good number are just poorly implemented. Still, this site is a great resource.

4.7. Sound as input

4.7.1. Voice

Voice recognition is being used in some games right now. *Nintendogs* [8], for example, has a speech recognition system that is trained to the player's voice, and it executes on a Nintendo DS, a portable game platform having limited processing power. The *Game Commander* software is available for converting speech into keystroke sequences for PC games, so using speech with existing games is easily possible (if sometimes frustrating).

Voice recognition has achieved an effective level of utility, although it is still weak in the general sense, and requires training to be really useful [15]. Most applications of voice recognition take advantage of a restricted domain of discourse; that is, they involve a small domain of human endeavor that has a small vocabulary associated with it. There are two reasons for this. First, the restricted vocabulary makes it easier to match the input audio signals against the known speech examples.

The second issue is more difficult to deal with in general. It turns out that computer algorithms have had little success in parsing general human speech. That is, even if the voice recognizer succeeds in converting a sentence into text, the correct interpretation of that sentence is still impossible in the general case. Restricting the domain helps here, too.

When giving the computer an oral command, how can it be structured so as to provide a context assist, or a tight syntax that allows the domain to be restricted? Assistance will be obtained from two unlikely places: the military, and the *Star Trek* television program [16]. The author's experience with the Canadian Air Force taught that a command on a parade ground has four important parts: the identifier, precautionary, cautionary, and executive parts. This was reduced to a simple identifier/acknowledge/command protocol in *Star Trek*. This syntax can be used when giving spoken commands to a computer.

The important parts of the command, and the only two that will always occur, are the identifier and executive. The identifier states who is to execute and gives a warning of a command to come. The executive allows a detailed command and indicates that it is to be done now. The military command "squad stand-at ease" has squad as the identifier, indicating who is receiving the command, stand-at as the cautionary, informing the squad what the command will be, and ease as the executive, indicating exactly when the command must be executed (in Canada, at least). Precautionary commands are less informational and more formal and timing related—move to the right in column of route, as an example, or to the right in file.

The *Star Trek* version is directly relevant to this situation. The command is normally a dialog, as follows:

Spock: Computer.

Computer: Working.

Spock: Calculate, to the last digit, the value of PI.

The idea is to alert the computer that it is to pay attention to the next spoken sequence and execute it as a command. The alternative is that the computer listens to everything, and may execute commands that are spoken but not intended. For example, if a dog starts to chew on a chair leg while the owner's computer video system is recording a favorite TV program onto the hard drive and someone shouts "Hey, stop!" the computer may interpret this as a command to stop recording. Precautionary and cautionary commands can be used to further restrict the domain; after speaking the identifier/precautionary command "computer," the user may speak the command "music," meaning to search the music library/database function of the system for the next command and then execute it.

4.7.2. Non-voice (visual music)

Although there has been a little research and development work done on the use of voice for nonvoice audio control of a computer application (e.g., [15]) almost none has been done of the input of nonspeech sounds in games. Partly because of this, the Digital Media Lab at the University of Calgary (Calgary, Alberta, Canada) is developing a game for teaching music, and this will be used as an example of the use of nonvoice input. Some detail will be given so as to show how much useful information can be extracted from what seems to be straightforward audio input.

The game, *Visual Music* is designed to make practice time more effective by making the player more aware of how well they are playing or singing. This is done by engaging both the auditory and visual centers while music is being played. The system listens to the performance of the player/student and assesses it on the basis of rhythmic quality, melodic accuracy, and in advanced instances the intonation and expression.

In coaching mode, the score is displayed on the screen along with the notes that are being played by the student:



Red notes do not always mean mistakes, of course. They could be quite reasonable interpretations of the score, what is called *ornamentation*. To accommodate this, there are two lesson types: *learn the music* mode and *ornamentation* mode, the latter mode is for more advanced players.

An example of a visual display of intonation is provided in Figure 1. The lower part of the screen shows the intonation, and the exact timing of each note played. In Figure 1, *Visual Music* has been set to display a heavy vertical line every bar, and a light line every quarter note. If every note was played perfectly, and had no vibrato at all, everything would be drawn along the middle "0 cent" line—the line that represents the exact pitch of the note. If the instrument is out of tune, then every note would be same amount sharp or flat (the light horizontal lines are 5 cents apart 3). One of the primary things that keep pitch lines from being a straight horizontal line on a sustained note is vibrato.

If the player's intonation (pitch) is perfect, a wavy burgundy line marching across the screen will always be right over the center "on pitch" line. Sustained tones should be centered over the big horizontal "on pitch" line.

Note transitions are another aspect of musical performance that can be evaluated by this system. If the player is a singer, the transition between notes is apt to be a continuous change in pitch from one note to the next. With other instruments, the transition may be more abrupt. Regardless of the instrument, with the exception of keyboard instruments, the player may find that their intonation is not perfect during the transition from one note to another. This fact can also be easily seen when the music is visualized.

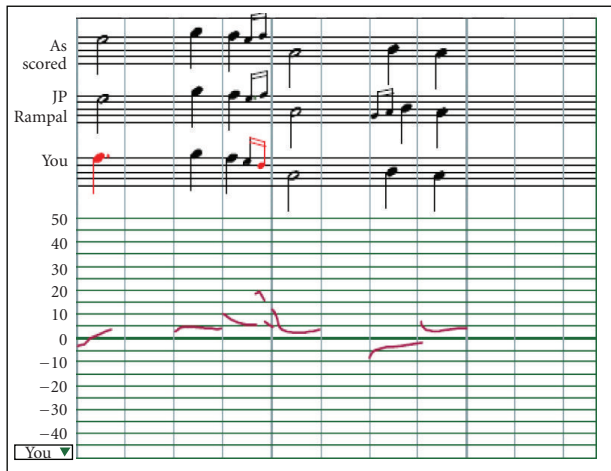


FIGURE 1: Intonation visualization section of the *Visual Music* game/toy.

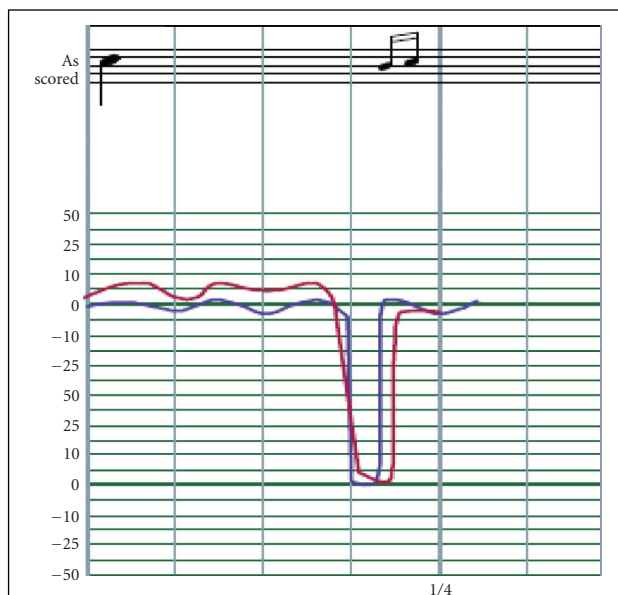


FIGURE 2: Comparing ornamentation and intonation between the player and that of Jean-Pierre Rampal.

Finally, the system permits various multimedia assists to be used, such as videos of professional musicians playing the piece in various ways. One can also compare a performance against those of virtuoso players. In the example in Figure 2, the intonation display is set to show Jean-Pierre Rampal's performance (in blue) of the overture to the *Spring* movement of Vivaldi's *The Four Seasons* over the player's performance (in red) of an ornament: the intonation region shows exactly a semitone. In this example, it is seen that the player's performance of the ornament matches the notated interval more closely than Mr. Rampal's. When he played the ornament, the first note took only about a third of the 16th note, while the player's was closer to half. How-

ever, Mr. Rampal's transitions are faster and cleaner. Looking more closely, it can be seen that Mr. Rampal managed to play the ornament so that it matched the timing of his vibrato.

The game play aspects of this software are in development, but will involve scores relating to constancy in melody and rhythm, vibrato, and the ability to read and play melodies in real time. Players will be able to play along with famous musicians and have their performances evaluated, and will ultimately be able to play along with other players at distant locations on the Internet.

The details of the game play aspects have not been described, and will only be sketched here. There are really a set of minigames, each contribution to a whole. Rather like racing games with distinct tracks, one aspect is to compete with well-known musicians at certain performance aspects. The player selects a piece to be played, and plays it while the game "listens" and computes a score. Errors, including tonal variations, are scored negatively, while success, the correct playing of the piece, is rewarded. This is not unlike the kind of play seen in *Dance, Dance Revolution*. Errors accumulate, intonation and expression is evaluated, and the player has the chance to improve their score.

Players can play along with prerecorded musical tracks, and again will be scored according to their ability to maintain a degree of fidelity with the original. Ornamentation is still the subject of research, but a goal of the developers is to accurately detect ornamentation and to award good scores to successful ornamentation.

At beginning levels simply completing the piece is worth points, but as skill levels increase it gets more difficult to score. Quality and accuracy are scored, and are related to the history or the player. A player that does not improve and progress to the maximum score is not learning. The game is, quite naturally, a piece of educational multimedia as well, and can be played casually just for fun or as a single-piece tutor.

The key novel aspects of the *Visual Music* system lie in the ability to simply play music and have the computer and software accept that as input. This is an example of a natural interface. An interface to a game (or any software, really) is natural if the activity used in the realworld situation is recognized and used by the interface, and means the same thing to the software system or game. A non-natural interface causes an interruption in the flow [1] of the activity being performed so as to perform the interaction, and this often results in a splitting of attention that is not productive or amenable to the effective completion of the task being performed. Clear, audio input can be used in inappropriate and nonnatural ways, too, which must be avoided.

5. GAMES INVOLVING THE CONTROLLING OF SOUND

So far video game audio has been briefly examined, both games that use audio as the main output, and games that use audio as input, both speech and nonspeech. What remains is to look at games that manipulate sound as a major goal or game play feature.

As usual, there are not many of these. A simple example would be the *Amplitude* [17] video game for the PS2. In these games players can remix songs given a set of audio tracks. An attractive feature is that the songs are from well-known bands like Blink 182 instead of a set of anonymous loops. A problem lies in the assessment of goals—music has an aesthetic component that is difficult to analyze by computer.

5.1. PC conductor

The Digital Media Laboratory at Calgary is developing a game that uses special devices [15] along with gestures and motions to allow a player to conduct an orchestra and to “compose” music. This game will never be complete; the idea is to use it as a platform for evaluating interface technologies in the context of games in general, and audio games in particular.

Conducting is done by selecting score and having the computer play it. Hand motions can be used to control aspects of the performance—cues to instruments, volume, and tempo. If desired, foot motions can control tempo as well, through pressure sensors in the player’s shoes.

5.1.1. Pressure sensors

A computer keyboard is a primitive sort of pressure sensor, wherein an impact on a specific part of the sensor is recognized and coded for transmission to a computer. The switch is either *on* or *off*, with no degree of pressure associated with it. They are really simple switches.

The degree of pressure can be measured using piezoelectric sensors that respond to pressure with an electrical voltage. This voltage is sampled and converted into pressure (*pounds* or *pascals*) by the computer [2]. In the situations being described here, a single piezoelectric device is placed in each shoe, and simple motions such as tapping are converted into signals that are entered to the computer through the sound card, which is really a simple analog-to-digital conversion that works at low frequencies. The tapping frequency can be used as a simple metronome. The amount of pressure is reflected in the strength of the signal, and could also be used.

If one now creates a small grid of these pressure sensors and places them inside of the shoes of a player then the amount and location of pressures on the foot can be identified [18]. The entire grid of pressures is sent to a PC many times each second and is converted into a pressure image. A machine learning algorithm can be used to match the pattern against those for that player that were used for training.

The player can keep rhythm by tapping their foot, a quite natural activity. However, the PC Conductor game allows players to choose what each sensor does in terms of audio control, and will remember.

5.1.2. Computer vision

Kinesiology researchers have been using machine vision technology for quantitative work on human performance for many years. Machine vision is weak in a practical sense

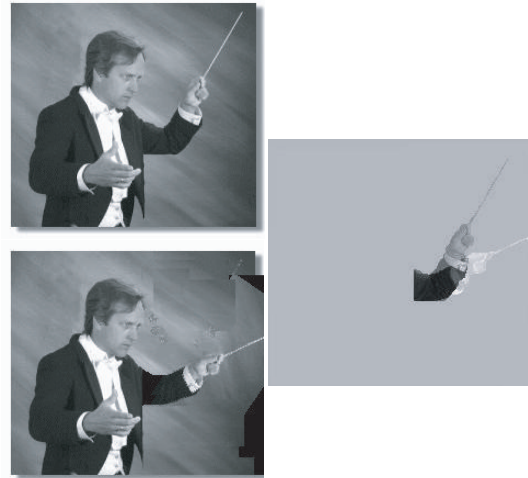


FIGURE 3: (left) Two frames in a video of a conductor. (Right) The result of a pixel by pixel subtraction of the two images. Subtracting one frame from the other in a video shows only what has moved, and thus gives essential information about the nature of the gesture, and to the action that the game is to perform as a result. (Dr. Steven Errante, Conductor, Wilmington Symphony Orchestra, from <http://www.wilmingtonsym-phony.org/AboutUs/Conductor.asp>).

as a general input mechanism. Many of the algorithms that work well either require too much real time to perform, or only function correctly in very specific circumstances. Fortunately, a set of specific circumstances can be generated for the PC Conductor game.

Most of the difficult problems in vision concern three-dimensional problems; inferring depth from 2D views is difficult, even with multiple cameras. Recognizing objects in any orientation is likewise difficult. Avoiding the hard vision problems is a key to workable vision-based game interfaces, and is what is seen in current vision oriented games. In PC Conductor the player’s 2D image drawn on the monitor, largely for feedback purposes. The player can select the meaning of body parts and gestures, but for the sake of discussion assume that the left hand controls volume and the right hand controls tempo. The player’s image is sampled many times each second and a volume level is determined from the height (or size) of the left hand. The down stroke of the right hand will be made to correspond with the beat of the music (Figure 3).

A trick is to recognize certain simple objects that permit easy classification. A ball, for instance, is a circle from any viewing point; being the only circular moving object, it is easy to recognize and to follow. Another trick is to use specific colors. Experiments are being performed with the use of a conducting baton that has a small brightly colored sphere on the end, and also with an infrared LED. This would be used instead of the hand, and since the color makes it easy to track, it should be possible to use it for everything for which a real conductor would use their baton. However, in the long term, general motions of hands and arms need to be recognized in a more general context [19].

Game play starts out simply: a simple orchestral piece is chosen and played through the computer's audio system. The player conducts the orchestra; volume control is easy to do, and the speed (tempo) of the music can be altered in real time using one of a variety of temporal shifting algorithms. The goal again is to match a benchmark performance that has been predefined as part of the game. The game presumes, for instance, that the benchmark performance of *Eine Kleine Nachtmusik* (Serenade, K. 525; 1st.) is the one conducted by the Academy Of St. Martin-In-The-Fields under Neville Marriner. That being the case, the goal of the player is to conduct the invisible orchestra so as to imitate that performance most closely.

Every time someone plays the game, the system records data for later analysis, data on preferences on the interfaces, training times, and time to achieve skill levels using specific interfaces. These data will be evaluated after enough has been collected. While this seems obvious, there are many people working on serious games for teaching and training who are not taking advantage of the fact that the computer can record everything that the player does, and that this can be used for assessment of the player/student.

Conducting is an art, and computer science is not. The analysis of data involves using a ground truth, or data for which the correct answer is known. For this, musical pieces that have been analyzed by a conductor and that can be used in a moment-by-moment comparison are needed. Getting this done is currently the major difficulty in the work.

6. DESIGN AND IMPLEMENTATION ISSUES

In a typical video game there are between 25 and 50 hours of game play connected with a typical CD. There cannot, of course, be this much novel audio stored on a CD, even if it were compressed. A CD can hold no more than a few hours of sound. Without reusing sound sequences, the games could not present a consistent audio presence, so it is clear that the sounds have to be reused somehow. Games usually solve the problem by repeating the same sounds over and over again. Music is played in a loop, and the same one or two sound effect files are used for all instances of the event that cues them. Overall sound quality suffers for this, but it has worked so far. Imagine how well it would have worked for graphics, though: if the same visuals were used for damage to vehicles all of the time, and the same explosion animation was played for all explosions. This would not be acceptable in a high-quality game.

It is possible to generate realistic sounds using software, and so create as much of any given sound as is needed for a game. Sound synthesis from first principles is not sufficiently realistic yet for most purposes; computer created wind or surf sounds often sound artificial. Ambient sounds and sound effects possess distinctive, identifying characteristics, while also having a natural variation from moment to moment. One way to solve the problem is to use small samples of a desired sound and to reconstitute them for a new, longer, and nonrepeating sample. This was done a few years ago to great effect [20, 21].

Of course, this method will not work for all sounds. Human speech would be an example where a person would quickly determine whether the sound made "sense" or not. Sounds that are too complex may also present problems. Still, the synthesis of new sounds from existing samples has the potential to reduce repetition and improve the audio presentation of games.

The audio generation algorithm has an interesting secondary application: it permits the audio to be slowed down and speeded up without concomitant changes in the sound frequency. This is the technology that permits PC Conductor to modify the tempo of the music according to the player's motions.

7. CONCLUSIONS

When looking carefully at the design of the existing audio games, it is clear that they use sound in only a few, banal ways. Not to offend the designers, who almost certainly had limited funding and restricted goals, but the variety of ways in which sound is used in games at this time is limited, partly by the effectiveness of the available technology, partly by cost concerns, and partly by a too-visual way of thinking.

The audio modes of operation of the games examined include the following:

- (1) Forcing the player to recall specific sound patterns, either tonal or rhythmic. The old *Simon* [22] game is an example of this.
- (2) The use of positional audio to place important objects in 3D space, for navigation or interaction. Scoring is by collisions with the objects or by shooting the objects.
- (3) The use of simple (tactical) spoken language input. The game is designed to recognize certain specific responses, but cannot deal with the general case.
- (4) Synchronizing rhythm to other activities in a game. (*Dance, Dance Revolution*).
- (5) The use of tonal qualities of sound input as a match for an existing musical piece. *Karaoke Revolution* [12] and *Visual Music* are examples.
- (6) The use of specific mouse or touch pad gestures to create musical vignettes and interesting sounds (*Electroplankton* [23]).
- (7) The use of gesture and human motion to control sounds and music (PC Conductor).

Most of the existing audio games are really retreaded video games, and specifically use positional audio to replace vision. That is ok for providing ways to share computer games with visually impaired players. It leaves novel aspects unexplored, though. Perhaps a measure of success would be if sighted players choose to play.

Two specific examples of mixed video/audio games have been described: *Video Music* and PC Conductor. Both could be made more accessible to visually impaired players. The point was to mix modes, including physical gestures and motions, and use more natural interfaces to the game software. Experiments with the PC Conductor platform should expose further possibilities.

REFERENCES

- [1] R. Boon, "Sensory-Motor Integration and Learning," <http://home.iprimus.com.au/rboon/SensoryIntegration.htm>.
- [2] D. Dickinson, "A quarterly newsletter for families and professionals on visual impairments and deafblindness. A collaborative effort of the Texas school for the blind and visually impaired and department of assistive and rehabilitative services, division for blind services," *See/Hear Newsletter*, vol. 7, no. 1, 2002.
- [3] K. Kreeger, "Penn researchers calculate how much the eye tells the brain," July 2006, http://www.eurekalert.org/pub_releases/2006-07/uops-prc072606.php.
- [4] J. R. Parker, *Start Your Engines: Developing Driving and Racing Games*, Paraglyph Press, Scottsdale, Ariz, USA, 2005.
- [5] S. Brown, M. J. Martinez, and L. M. Parsons, "Passive music listening spontaneously engages limbic and paralimbic systems," *NeuroReport*, vol. 15, no. 13, pp. 2033–2037, 2004.
- [6] J. A. Winer, "Decoding the auditory corticofugal systems," *Hearing Research*, vol. 207, no. 1–2, pp. 1–9, 2005.
- [7] Radical Entertainment, "Simpson's hit and run," Vivendi Universal, 2003.
- [8] Nintendo, "Nintendogs," Nintendo, 2005.
- [9] R. North, "Grand theft auto," Rockstar Games, 1998.
- [10] Nintendo Software Technology, "1080° Avalanche," Nintendo, 2003.
- [11] L. Williams, *Music Videogames: The Inception, Progression and Future of the Music Videogame*, Audio Mostly, Pitea, Sweden, 2006.
- [12] Harmonix, "Karaoke revolution," Konami, 2003.
- [13] J. Romero, "Doom," id Software, GT Interactive, Activision, 1993.
- [14] R. van Tol and S. Huiberts, "AudioGames.net," June 2006, <http://www.audiogames.net/>.
- [15] T. Igarashi and J. F. Hughes, "Voice as sound: using non-verbal voice input for interactive control," in *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology (UIST '01)*, pp. 155–156, Orlando, Fla, USA, November 2001.
- [16] J. R. Parker, "Human Motion as Input and Control in Kinetic Games," FuturePlay, London, Ont, Canada, 2006.
- [17] Harmonix Music, "Amplitude (PS2)," Sony, 2003.
- [18] J. Munk-Stander, "Evaluation of piezoelectric film sensors for in-shoe pressure measurement," Tech. Rep. 06/04, Department of Computer Science, University of Copenhagen, Copenhagen, Denmark, 2004, (joint technical report with University of Cambridge).
- [19] J. R. Parker and M. Baumbach, "Creating an enhanced reality user interface—ERSolitaire," in *Proceedings of the Conference on Human Factors in Computing Systems (CHI '03)*, pp. 958–959, Ft. Lauderdale, Fla, USA, April 2003.
- [20] J. R. Parker, S. Chan, and B. Behm, "Generating audio textures by example," 2007, to appear in *Journal of Game Development*.
- [21] N. N. Saint-Arnaud and K. Papat, "Analysis and synthesis of sound textures," in *Proceedings of the IJCAI Workshop on Computational Auditory Scene Analysis*, pp. 125–131, Montreal, Canada, August 1995.
- [22] Milton-Bradley, "Simon," Milton-Bradley, 1978.
- [23] T. Iwai, "Electroplankton," Nintendo, 2005.