1. PROBLEM DESCRIPTION

PC Game industry, especially Online PC Game, is no doubt one of the fastest growing industries in the 21st Century. Among all kinds of different PC Games, World of WarCraft, a 3D massively multiplayer online role-playing game (MMORPG) by Blizzard Entertainment is for sure the most popular one. With 10.2 million subscribers as of December 2011, World of WarCraft is currently the world's most-subscribed MMORPG and holds the Guinness World Record for the most popular MMORPG by subscribers. (Glenday, 2009)

The game have several different playing modes which including different scenarios. One popular mode is called Arena. In the Arena, two small teams (2, 3 or 5 players) fight each other in a particular field until one team is eliminated. (Figure 1) Unlike the normal Boss Combat, Arena is fully based on players’ interactivity and it is a real-time battle. Actually, the most attractive thing may be the unpredictability of the rivals as well as the teammates. Thus, quickly reaction to the opponent's action and teamwork become the key to win the battle.

![Figure 1 the Arena game interface.](image-url)
During the battle, each player needs to use their own ‘Actions’ to either kill the opponents or help the teammates. Since every ‘Action’ has an action range (figure 2), the relative positions of each player become extremely important here.

*Figure 2 an example between two players (Rogue and Mage). In the left case, rogue is out of mage’s action range, thus do not affect by the mage’s action. In the right case, rogue is inside the action range, thus the mage gives damage on the rogue.*

How to quickly figure out other players’ position? Normally, game players will hold the right click of their mouse and control the characters’ view to turn around, try to find others by their eyes. But in the Arena battle, it is not efficient enough, especially when other players hiding behind barriers or using some special actions. (Figure 3)

*Figure 3 Rogue using action ‘stealth’. Although the mage is near the rogue, it is hard for him to see the rogue.*

In real life, human beings can localization objects quite quickly. It is mainly because they can combine the visual cues with the auditory cues together. Therefore, if we can provide auditory cues to those Arena players, it will create a novel game experience.

2. **SPECIFICATION**

To give complete auditory cues, we need to analyze the scenario first. As for one player, the sound he receive from the game should including the following components a) his own actions’ sound b) the virtual sound of position-fixed objects in the game. For example, in one Arena field, there are fire-walls which contain fire sound. C) The actions sound from other players (both the opponents and the teammates). D) Background music. A to C are the main objects we want to locate.
At the same time, most of the Arena players tend to use either the internal communication system provided in the game (figure 4) or the assistant communication software (figure 5) to talk to teammates in the real-time in order to keep a good teamwork. Assume that we can position those teammates’ voices into a place which will make them sounds like as if come from the their relative virtual positions to the main character in the game, then the players of Arena will be able to predict their teammates’ virtual position by just listen to their conversations. It will not only increase the efficiency of localization, but also bring immersive to the game player.

Therefore, we will design a plug-in particular for players of Arena, which can provide a 3D sound source localization function. The plug-in should have an interface which provides customized options. Players can choose which sound source they want to have localization function and at the same time they can adjust these sounds’ volume. (Figure 6)

3. IMPLEMENTATION

3.1. Individual Coordinate

To achieve the 3D sound source localization function, first we want introduce a concept which we called ‘individual coordinate’.
In typical virtual audio applications the user’s head is the central reference for audio objects rendering. (Michele, Simone, & Federico, 2011) In our case, Arena players will control the actions of the main character in the game as if they are the main character, so the main character’s head is the central reference. In principle the location of this head establishes a virtual coordinate system and builds a map of the virtual auditory scene. In the case of a mixed environment sound objects are placed in the virtual world around the main character and hence, conceptually, positioned consistently with a virtual coordinate system. Locating virtual audio objects into a mixed environment requires the superimposition of one coordinate system onto another as shown in Figure 7.

![Figure 7](image)

Figure 7 a typical 2v2 (2 players vs. 2 players) scenario with virtual individual coordinates

Here, the main character (Rogue)’s head establish a master coordinate, which is the central reference of the entire virtual sound source that the player of rogue will receive. The other two opponents and the teammates will have their own individual coordinates as well. The relationship between those coordinates and the master coordinate can be described by distance $d$ and azimuth angle $\theta$ (here, in order to simplify the figure, assume all the characters are being at same horizontal plane, in practical, it should also including elevation angle $\varphi$)

Particularly, most ranged dealer’s actions (Mage, Priest etc.) are not act on themselves but can be assigned to a particular place (as shown in figure 2), thus those place also establish their coordinate of sound.
3.2. 3D localization of sound source in headphone

Within the data of each individual coordinate, we can calculate their distance, azimuth angle and elevation angle towards the master coordinate. Using these 3 values, theoretically we can simulate the virtual localization of sound source.

3.2.1. Theory

Researches have shown that there are different cues which affect the localization of sound source. Two most important cues are the interaural time differences (ITD) and interaural intensity differences (IID). In general, as IID and ITD increased, the position of the virtual sound will move to the ear. Secondly, the head motions can obtain great localization cues. Third, in the situation shown in figure 8, if we assume that our head is spherical, then the sound source from point A and B will theoretically produce same ITD and IID information and similarly for point x and y. The cone shown in the figure is called the cone of confusion. ITD and IID cues will be weaken inside this cone area. Instead, the pinna-head-torso system can provide more useful cues here. (Begault, 2003) Thus, a function called head-related transfer function (HRTF) has been introduced to describe the effects of this system.

![Figure 8 the cone of confusion (Begault, 2003)](image)

Within digital signal processors (DSP), it is now possible to create 3D directional sound cues for headphones within a reasonable cost. In general, if we can design a system simulate all these directional cues and combine them corresponding to the sound sources, we will be able to solve the problem described in section 1.

3.2.2. Prototype of the plug-in

In order to achieve the functions described in Figure 6, first we will collect all the individual coordinate data (C/D) as well as all the action signals (A/S) which generated when the players press their action keys. Here, the player’s own C/D and A/S will generate at his own computer, while those data of the position-fixed objects in the game is also stored in his own computer’s memory. The C/D and A/S of other players are sent from the remote game server. Actually, once the player begin the game his computer will automatically connect to the remote game server then begin to upload his own data and download other player’s data at the same time.

All the A/S will select and trigger the proper virtual sound from the sound source in the computer. Here, background music does not need localization function. At the same time, all the C/D will be sent to a Position calculator. This calculator will take the player’s own coordinate as central reference then output values for producing Binaural Cues which including the individual coordinates’ distance, azimuth angle and elevation angle towards the master coordinate. After that, all these values will combine with those triggered sounds and the teammates’ conversation voice then be sent into the
Binaural Processor. Finally, the processor will produce binaural sound to a mixer, which corresponds to the interface shown in Figure 6, where the players can adjust the entire sound as they like.

A structure of the plug-in is shown below in Figure 9.

![Diagram](image)

*Figure 9 the structure of the 3D sound source localization plug-in*

Here, in order to make the figure small, there are several abbreviations.

A/S = Action Signal, when the players press the action keys, it generate an Action Signal on the computer.

B/G = Background Music

B/C = Values for producing Binaural Cues which including the distance, azimuth angle and elevation angle towards the master coordinate

B/S = Binaural Sound that generated from the binaural processor.

C/D = Individual Coordinate Data.

O = opponents  T = teammates  L = left output  R = right output

### 3.2.3. Binaural Processor

As for the binaural processor, researchers have developed different methods to achieving the binaural simulation. One possible solution can be found in the author’s Lab Report 2 for Digital Audio System class (Boey 2012) which is a MATLAB function. The function simulates a Structural model of the pinna-head-torso system which comes from C.P. Brown and R.O. Duda. (C.P. & R.O., 1998) (Udo Zölzer, 2002).
The Whole structure of the process is shown below in figure 10. For different distance of sound sources, we can add a gain controller into the processor, which accept the distance data and make corresponding gain adjustment to the input signal.

For different distance of sound sources, we can add a gain controller into the processor, which accept the distance data and make corresponding gain adjustment to the input signal.

In the function shown above, the user needs to provide azimuth angle and elevation angle of the sound source as well as a radius of their head. Because the radius of the head do affect the results (Udo Zölzer, 2002), in the customer interface, we can provide a function for the players to enter their own head’s radius value. (Figure 11) The other values needed in the function will be automatically generated in the position calculator in our case.

First, we should notice that there is an important assumption in our case. Here, we assume that all the players will use headphones for the game. Actually, this assumption is quite reasonable. As mentioned before, most Arena players like to communicate during the game. This feature indicates that a large amount of players may prefer to use headphones since the loudspeaker can easily bring annoying feedback when the players turn on their microphones.

Secondly, in practical, for a complete simulation of all the binaural cues, besides designing the filter for HRTFs and make proper delay for ITD, people also monitor the head movements of listeners. In some
applications this information can be used to modify the binaural cues that are reproduced so that the head movements give rise to realistic changes in the signals sent to the ears. (Rumsey, 2001) However, a head-tracking is quite expensive and usually require the listeners wearing specially equipment. Thus it is not so practical for real life applications.

One important feature of role-play game is that the players will control the character to do things in the game instead of doing it themselves. Thus, when the players want to look around, what they do is not shank his own head, but holds the right click and move the mouse. This can bring great benefit to achieve sound source localization. As mentioned before, we will use the data of individual coordinate to simulate the 3D sound field. Actually, if consider the virtual head of the main character as the central reference, the coordinate also provides information of head movements. Thus, using individual coordinate data to localize sound source in Arena becomes practical.

Besides, as the most popular game in the world, World of WarCraft has open protocols for users to design their own plug-ins. Thus, there is no obstacle for a player to have his own 3D sound source localization plug-in in his computer. Actually, players can share this plug-in through some specific website like http://www.curse.com/ and for all the trial users we can design a survey which will be shown after their trials. (Figure 12)

![Survey Image]

Figure 12 a possible 5-item scaled survey which will collect the users’ feedback.

After analyze the data, we may make some improvement to the plug-in. At the same time, the data of users’ preferred setting may also suggest some interesting case in virtual sound source localization. Thus, the results can be meaningful for a further study.

5.REFERENCE


