

“Implementación de un Prototipo de una Batería Musical Virtual”

Laura Marianella Zapata Aspiazu ⁽¹⁾ Diego Andrés Herrera Samaniego ⁽²⁾ Ph.D. Sixto Ernesto García Aguilar ⁽³⁾

Facultad de Ingeniería en Electricidad y Computación.

Escuela Superior Politécnica del Litoral.

Campus Gustavo Galindo, Km. 30.5 vía Perimetral.

Tel.: 593.4.2269151, Apartado 09-01-5863, Guayaquil-Ecuador.

lzapata@fiec.espol.edu.ec; dherrera@ezone.com.ec; sgarcia@espol.edu.ec;

Resumen

Este documento presenta la implementación de un prototipo de una Batería Musical, utilizando la tecnología de Realidad Virtual y la Multimedia. Este prototipo crea un ambiente virtual 3D adecuado, que es interactivo e inmersivo, que permite ejecutar un instrumento musical sencillo, con una adecuada sincronización entre las acciones del usuario, la retroalimentación visual y el sonido generado.

Mediante el uso de un dispositivo sensor de posiciones (tracker), cuyos sensores se adjuntan a un par de palillos reales (baquetas), se capturan los movimientos motrices que el usuario realiza, luego se generan las imágenes que corresponden a estos movimientos, mostrándolas en una pantalla Head Mounted Display (HMD) que permite ver en tres dimensiones (3D).

Cada instrumento tiene asociado una imagen y un sonido en el escenario virtual, este sonido se reproduce mediante golpes que genera el usuario a través de las baquetas. Esta reproducción va asociada con diferentes niveles de volumen, de tal forma que a mayor fuerza de golpe se eleva el volumen de reproducción.

Palabras Claves: Realidad Virtual, Multimedia, 3D, tracker, Head Mounted Display.

Abstract

This document presents the implementation of a prototype for a drum kit, through the use of Virtual Reality Technology and Multimedia. The prototype creates a proper 3D virtual environment, which is interactive and immersive, this lets you to play a simple musical instrument, with proper synchronization between user actions, visual feedback and the sound generated.

By using a position sensing device (tracker), which sensors are attached to a pair of real drumsticks, the motion movements that the user makes are caught, then the images that correspond to these movements are generated, showing these on a Head Mounted Display (HMD) that allows you to have a three-dimensional view (3D). Each instrument is associated with an image and a virtual sound stage, the sound is generated through the hit made by the user. This reproduction is associated with different volume levels, so that a larger strike force will increase the volume of reproduction.

Key Words: Virtual Reality, Multimedia, 3D, tracker, Head Mounted Display.

1. Introducción

A lo largo de los tiempos, conforme la ciencia y la tecnología avanzan, se han desarrollado nuevas formas de expresarse musicalmente, desde los rústicos instrumentos de percusión y viento que usaban troncos y conchas para generar sonidos, hasta las avanzadas interfaces eléctricas y electrónicas utilizadas en la actualidad. Muchos artistas precisan de nuevas interfaces y controles que, aprovechando los avances tecnológicos, les permitan mayor libertad y les abran paso a nuevas expresiones musicales en su continuo afán de innovar en este arte. De la misma manera en que aparecieron los instrumentos eléctricos y revolucionaron la música con el rock, hoy se piensa

que con los avances tecnológicos, los controles musicales podrían romper barreras que permitan al músico sumergirse en un ambiente donde no existan necesariamente las limitaciones de interacción con instrumentos físicos, y pueda aprovechar de mejor manera sus sentidos y movimientos para generar una nueva forma de expresión musical [2].

Algunas interfaces y controles musicales actuales, sobre todo aquellos de percusión, generan algunos problemas, porque no son ligeros y portables, su configuración y calibración en algunos casos puede resultar costosa y consumir tiempo al momento de adaptarse a la comodidad y gustos específicos del músico. Otros ocupan mucho espacio o generan mucho ruido cuando se realizan prácticas. Finalmente

podemos añadir que, para aquellas personas con alguna discapacidad, hay pocas alternativas de accesibilidad para tocarlos.

La implementación de instrumentos musicales con tecnología de realidad virtual no es nueva, ya existen baterías, guitarras, pianos [3]. Este proyecto crea un ambiente de realidad virtual que permite al usuario interactuar con una Batería Musical, en el cual se puede experimentar con conceptos como inmersión, presencia e interacción que son propios de la tecnología de Realidad Virtual.

La razón por la que se eligió a la Batería, es debido a que goza de nuestra preferencia, además de que es un instrumento de fácil reproducción en el ambiente virtual.

A continuación se presenta la introducción, los requerimientos del sistema, se explica la arquitectura, las herramientas de desarrollo, se detalla la etapa de implementación y se describen las pruebas. Finalmente se presentan las conclusiones y recomendaciones del trabajo realizado.

2. Requerimientos del Sistema

El sistema virtual cumple con los siguientes requerimientos:

2.1. Requerimientos Funcionales

Se requiere que el prototipo simule por medio de dispositivos de realidad virtual el funcionamiento de una batería real.

A continuación se detalla las funcionalidades necesarias de una batería:



Figura 1. Batería musical

- 1) Platos Ride y Crash.
- 2) Tom de piso.
- 3) Toms izquierdo y derecho.
- 4) Bombo.
- 5) Redoblante o caja.
- 6) Hi Hat.

Escena virtual: Se debe crear un escenario virtual en tres dimensiones -3D- en el cual deben constar todos los instrumentos musicales que conforman la batería. Estos instrumentos se agrupan en dos tipos: tambores y platillos, esto se lo puede observar en la figura 1.

Movimiento de las Baquetas: Esta funcionalidad permite que el usuario mueva las baquetas en la escena virtual mientras mueve las baquetas reales, esto se lo hace con un dispositivo sensor de movimientos (Tracker) y con sensores adjuntos a las baquetas reales.

Movimiento de la Cabeza: Esta funcionalidad permite al usuario tener una visión panorámica de la escena virtual, la relación entre el movimiento de la cabeza y el movimiento de la escena es de 1 a 1, es decir, si el usuario mueve su cabeza 1 grado, la escena se mueve 1 grado.

Reproducción de sonidos musicales: Al producirse un golpe con las baquetas sobre la batería, se reproduce el sonido asociado al instrumento golpeado y se muestra una retroalimentación visual del golpe generado.

En la tabla 1 se presenta el rango de frecuencia de los instrumentos musicales que son parte de la batería, estos sonidos son grabaciones de una batería real y no sonidos sintetizados o generados por la computadora.

Tabla 1. Rangos de frecuencias de los instrumentos

Instrumento	Fundamental (Hz)	Armónicos (KHz)
Tambor	100-200	1-20
Bombo	30-147	1-6
Platillos	300-587	1-15

Reproducción de Bombo: El bombo es el único instrumento de la Batería que se toca con un pedal, su funcionamiento se implementa utilizando un pedal de los que se usan para juegos de computadora.

2.2. Requerimientos No Funcionales

A continuación se detallan los requerimientos no funcionales que el prototipo debe cumplir:

Escalabilidad: El sistema debe ser escalable con respecto a las partes instrumentales de la batería, de tal forma que se puedan incorporar nuevos elementos de percusión a la escena.

Eficiencia en la reproducción de los sonidos musicales: Se debe garantizar un buen desempeño de la reproducción del audio, especialmente cuando dos o

más instrumentos se golpean simultáneamente, por ejemplo: si se golpea el redoblante y el esterbil al mismo tiempo, ambos sonidos deben ser reproducidos simultáneamente y no se debe esperar que uno finalice para empezar el otro.

Eficiencia: el manejo de la visualización de la aplicación debe ahorrar recursos de hardware como son la memoria RAM y el tiempo de procesamiento del CPU, lo que resulta en un mejor desempeño.

Rendimiento: el sistema debe ser capaz de dar respuesta a los movimientos motrices del usuario minimizando el tiempo de respuesta de la aplicación, es decir, que se vea real el movimiento y no haya retrasos en los movimientos del usuario y su correspondiente movimiento de baquetas en la escena virtual.

3. Arquitectura General del Sistema

La figura 2 muestra la arquitectura del sistema, en la que se pueden observar los componentes de cada nivel que permiten cumplir con los requerimientos funcionales.

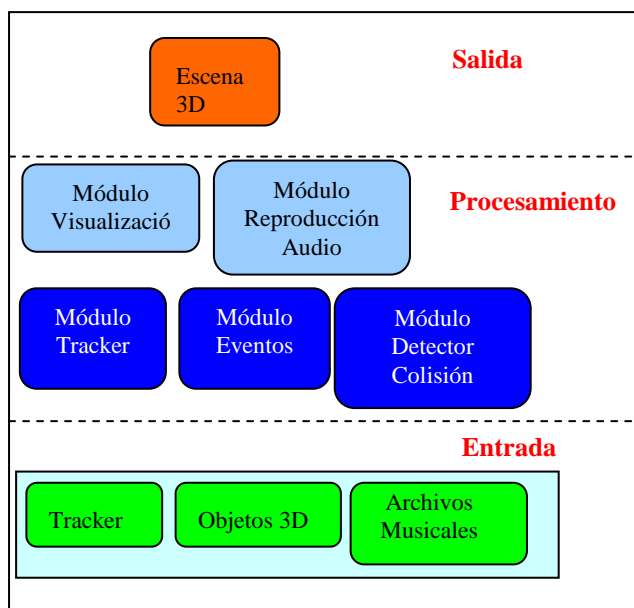


Figura 2. Arquitectura de la Batería

3.1. Módulos del Sistema

Modulo de Visualización

Carga de la Escena 3D en el ambiente virtual: Por medio de la librería Open Scene Graph (OSG) cargamos la escena virtual con las imágenes de los objetos musicales y las baquetas, además se hace la iluminación del ambiente. Se establecen los parámetros de configuración 3D para que el

dispositivo de despliegue visual pueda mostrar imágenes en tres dimensiones y se ubica la cámara en posición inicial.

Retroalimentación Visual: Al momento de producirse una colisión entre la baqueta y la batería, además de reproducir el sonido musical correspondiente, se cambia de color al objeto que ha sido golpeado, de esta forma el usuario recibe una retroalimentación visual del objeto con el que interactúa.

Módulo Tracker: Con la librería propia del dispositivo de posicionamiento "Polhemus Liberty"(PDI), se obtienen las coordenadas X, Y y Z que registran los cambios de movimiento de las baquetas; estas coordenadas son usadas para mover sus correspondientes imágenes en la escena virtual.

Módulo de Eventos: La función principal de este módulo es poder tocar el bombo por medio de un pedal, que actúa como evento click del ratón del computador. Además adiciona la funcionalidad necesaria para poder tocar la batería completa sin el dispositivo de rastreo "Tracker".

Módulo Detector de Colisiones: Este módulo es usado para detectar colisiones, las cuales ocurren cuando hay una intersección entre dos objetos. Cuando existe una colisión entra en funcionamiento el módulo de reproducción de audio.

Módulo de Reproducción de Audio: Se reproduce el sonido asociado al objeto musical que el usuario golpea por medio del uso de colisiones de imágenes. Esta reproducción va asociada con diferentes niveles de volumen, de tal forma que a mayor fuerza de golpe elevamos el volumen de reproducción.

Cada instrumento musical tiene un canal de reproducción individual, con lo que se logra independencia entre los sonidos, por lo que no hay que esperar que finalice uno para iniciar otro sonido.

3.2. Clases del Sistema

Las clases participantes en la interacción son:

CPDIDev: Es una clase que pertenece a las librerías PDI, Polhemus Device Interface, está fuera de nuestro diseño. Representa al dispositivo físico.

PDIconsole: Esta clase representa la entidad que establece la comunicación entre el dispositivo Polhemus Liberty y nuestro sistema de batería virtual.

Osg: Es la clase encargada de establecer la escena 3D y todos los métodos que se pueden ejecutar en la interfaz grafica, como por ejemplo las colisiones.

MyOsgNodoMusical: Entidad que representa a cada instrumento musical como tambores y platillos e inclusive a las baquetas. Tiene asociado un sonido y su objeto imagen correspondiente, si es un elemento baqueta no tiene sonido.

ElMouseEventHandler: Esta entidad hereda de GUIEventHandler la utilizamos para capturar el evento onClick(), que se dispara cuando el usuario presiona el pedal del bombo.

PlayAudio: Esta clase utiliza la librería SDL_Mixer la cual abre el dispositivo de audio con los parámetros deseados, además usamos un canal diferente para reproducir cada sonido, también se le envía un nivel de volumen que va de 0 a 128.

3.3. Diagramas de Interacción de Objetos

A continuación se presentan los diagramas de interacción de objetos que permiten alcanzar los requerimientos establecidos.

Escenario 1.1: Ejecución exitosa del esterbil.

Se ejecutan lecturas continuas del tracker, las cuales registran los sensores conectados y las posiciones sobre las coordenadas 'x', 'y', 'z', y sus giros respectivos 'roll', 'yaw', 'pitch'. Cuando un usuario hace un movimiento de la baqueta -sobre la cual está pegado un sensor del tracker-, se realiza el cambio de posición geométrica de la imagen asociada a la baqueta, luego se ejecuta un proceso para determinar si hubo una colisión de la imagen de la baqueta con la imagen de un instrumento, si se da la colisión se reproduce el sonido que corresponde al instrumento musical y se muestra una retroalimentación visual que representa el golpe dado. Ver figura 3.

Escenario 1.2: Visión panorámica de la escena 3D.

Se ejecutan lecturas continuas del tracker, las cuales registran los sensores conectados y las posiciones sobre las coordenadas 'x', 'y', 'z', y sus giros respectivos 'roll', 'yaw', 'pitch'. Ver figura 4.

Cuando un usuario hace un movimiento de la cabeza -sobre la cual está pegado un sensor del tracker-, se realiza el cambio de posición geométrica de la cámara de la escena. El campo de visión que tiene el usuario dentro de la escena virtual es de 140 grados en la horizontal y 140 grados en la vertical y la relación del movimiento real y el generado es de 1 a 1, es decir, si el usuario gira 1 grado, la escena se mueve 1 grado.

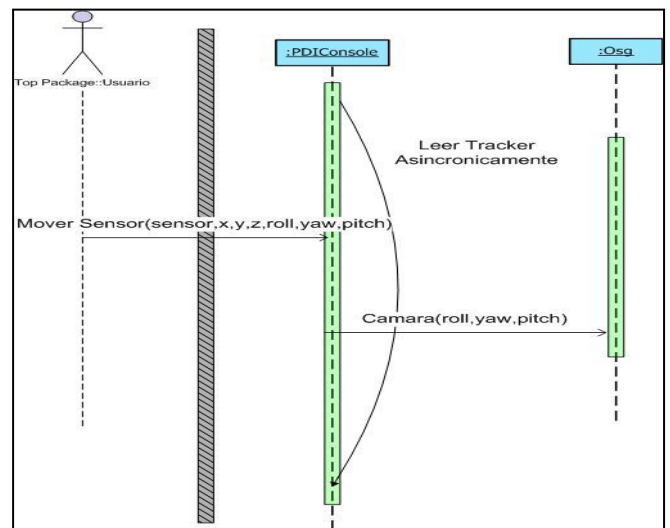


Figura 4. Vista panorámica.

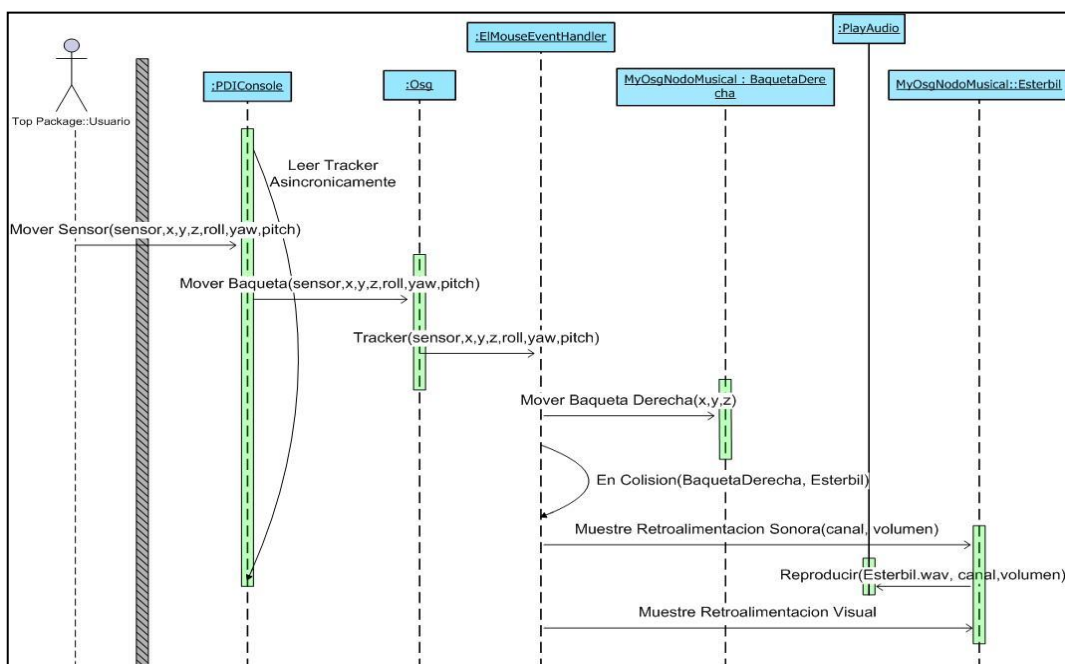


Figura 3. Ejecución del Esterbil

Escenario 1.3: Ejecución exitosa del bombo.

La ejecución del bombo difiere de los otros instrumentos debido a que se la realiza por medio de un pedal.

Al presionar el pedal, este emite una señal al computador, que actúa como un click, por lo que al producirse el evento se reproduce el sonido del bombo y se muestra la retroalimentación visual. Ver figura 5.

El canal de audio es manejado por la librería SDL y su función es reproducir más de un sonido al mismo tiempo.

4.3. Lectura de coordenadas de las baquetas

La clase PDIconsole tiene el método “main” el cual desencadena la ejecución, lo primero que se realiza es Inicializar, Conectar y Configurar el dispositivo de rastreo.

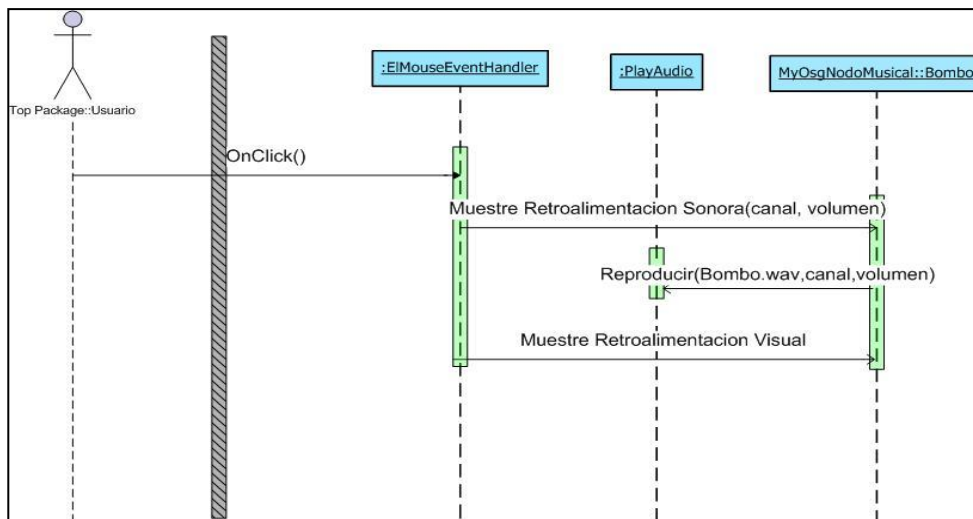


Figura 5. Ejecución del Bombo

4. Implementación del Sistema

4.1. Herramientas de desarrollo

A continuación se listan las herramientas de software usadas para desarrollar esta aplicación.

- Polhemus Developer Interface (PDI).
- Microsoft Visual C++ 2008 Express Edition (MSVC).
- OpenSceneGraph-2.8.1.
- SDL-1.2.13 y SDL_mixer-1.2.8.
- Blender.

4.2. Carga de escena 3D

Para la carga de la escena en 3D se utiliza una clase que representa a cada parte de la batería, se la denomina MyOsgNodoMusical. Al crear una instancia de esta clase se asocia la imagen del instrumento musical, también se configura la posición geométrica en la escena, el archivo de sonido que debe reproducir y el canal del audio.

Luego se configura en el dispositivo el modo continuo de lectura, esto es para obtener las lecturas de las coordenadas X, Y y Z del sensor, una vez que tenemos las lecturas, empezamos a mover los objetos baquetas, en el caso de que haya colisión, se reproduce el sonido asociado a la clase MyOsgNodoMusical. Cada clase tiene un método destructor, que se ejecuta antes de salir del programa, cerrando las conexiones de la clase PlayAudio, del PDIconsole, y otras.

4.4. Conexión física de los dispositivos de realidad virtual.

El sensor de la cabeza lo insertamos en el primer puerto (1), en cualquiera de los puertos restantes impares (3, 5, 7) conectamos la baqueta derecha y en cualquiera de los puertos pares (2, 4, 6, 8) conectamos la baqueta izquierda.

En la figura 6 mostramos un diagrama de conexiones de los dispositivos usados.

5. Pruebas y Resultados

Las pruebas miden los indicadores de latencia y de inmersión, a continuación se hace un detalle del proceso.

Latencia

La capacidad de acoplar los movimientos motrices del usuario con su representación gráfica es la base de muchos sistemas de realidad virtual, la sincronización

Para tomar estas mediciones se realizan solo golpes simples, se hizo uso de un temporizador “timer” a nivel de código fuente [5], el cual es inicializado con el valor

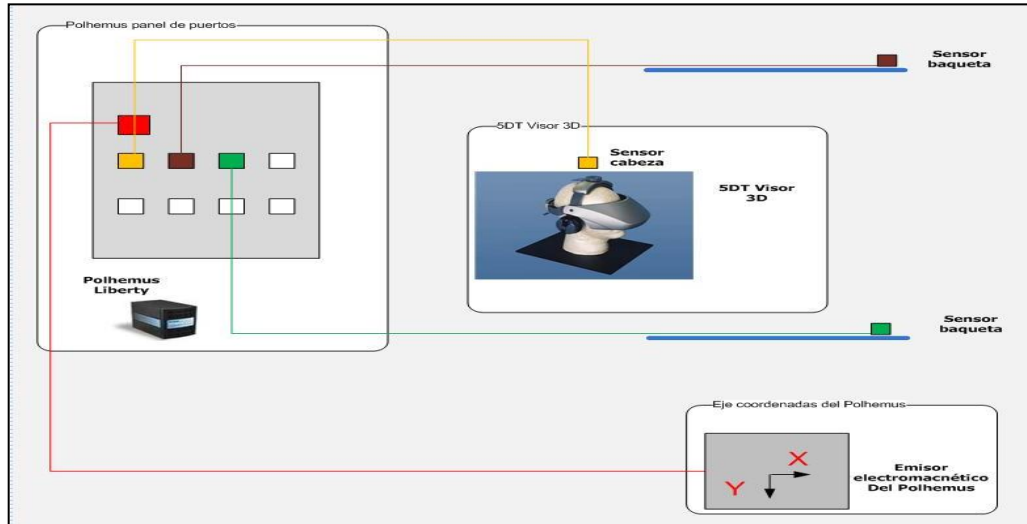


Figura 6. Diagrama de conexión de dispositivos

exacta de los movimientos motrices y la representación gráfica es crítica: retrasos mayores a 100 ms puede causar discrepancias entre lo real y virtual, llegando incluso a causar mareos [4]. Esta es una de las razones por las que se ha incluido a la latencia como un indicador a ser considerado.

Se estableció la latencia como el tiempo transcurrido entre el movimiento motriz de la baqueta por parte del usuario y el instante de colisión con la batería.

cero cada vez que se toma una lectura de posiciones del Polhemus, luego se mide el tiempo hasta que se produce un evento de colisión y se le suma el retardo del dispositivo de rastreo Polhemus.

Se toman 30 muestras (basados en el teorema del límite central [6]). La tabla 2 muestra los resultados obtenidos, el bombo es el elemento con mayor latencia 93 ms, este tiempo es muy cercano al límite del tiempo recomendado para sistemas de realidad virtual (100ms) [4].

Tabla 2. Cálculos de Media, varianza y desviación estándar de la latencia de la Batería

	tom izquierdo	tom derecho	tom de piso	ride
Sumatoria (s)	0.41739	0.398028	0.412817	0.380119
Media (s)	0.013913	0.0132676	0.013760567	0.012670633
Varianza	1.68E-07	1.04E-08	2.4E-08	2E-08
Desviación estándar	0.0004	0.00010	0.00015	0.00015
Retardo Polhemus (s)	0.0035	0.0035	0.0035	0.0035
Resultado (s)	0.017413	0.0167676	0.017260567	0.016170633
Resultado (ms)	17.413	16.7676	17.26056667	16.17063333

	crash	redoblante	esterbil	bombo
Sumatoria (s)	0.394316	0.410011	0.442566	2.707233
Media (s)	0.013143867	0.013667033	0.0147522	0.0902411
Varianza	1.41E-08	3.5E-08	5.29538E-08	1.4E-05
Desviación estándar	0.00012	0.0002	0.0002	0.0038
Retardo Polhemus (s)	0.0035	0.0035	0.0035	0.0035
Resultado (s)	0.016643867	0.017167033	0.0182522	0.0937411
Resultado (ms)	16.64386667	17.16703333	18.2522	93.7411

La latencia del crash es de 16 ms, del redoblante 17 ms, del esterbil es 18 ms, del bombo es 93 ms, del tom izquierdo es 17 ms, del tom derecho es 16 ms, del tom de piso es 17 ms, del ride es 16 ms. La varianza de cada instrumento es muy pequeña.

Inmersión

La inmersión es un factor que describe, desde el punto de vista físico, la experiencia en la realidad virtual [1].

Este indicador está relacionado a la configuración física de la interfaz de usuario. La cantidad de sentidos que se alimentan establece los niveles de inmersión: Completamente Inmersivo (todos los sentidos), Semi-Inmersivo (la mitad o menos), No Inmersivo (ninguno) [1].

En una batería real, por lo regular son tres sentidos los que son experimentados por el baterista, debido a que sólo se simulan dos (la visión y la audición), se puede afirmar que el ambiente virtual es semi-inmersivo.

6. Conclusiones y Recomendaciones

El tiempo de respuesta es una variable que hay que tener muy en cuenta en los sistemas de realidad virtual, ya que juega un papel crucial para dar la sensación de ser más real y no causar discrepancias entre lo real y virtual.

La latencia de los instrumentos que conforman la batería varía desde 16 a 18 ms, a excepción del bombo que tuvo un tiempo de 93.74 ms, como resultado tenemos que este instrumento musical reduce la sensación de presencia que experimenta el usuario en el ambiente virtual.

No se puede “sentir la batería” debido a que no se implementó una retroalimentación táctil, esto dificulta el manejo de las baquetas para generar golpes en los instrumentos.

El dispositivo de despliegue gráfico personal (Head Mounted Display) permite aislar al usuario del mundo real incrementándose la sensación de inmersión en el ambiente virtual, este efecto no se hubiese alcanzado con el proyector 3D o con el monitor.

La implementación es escalable, dejando abierta la posibilidad de que se añadan más instrumentos de percusión en la escena.

En una batería real el sonido generado depende de la forma en que el baterista hace colisionar la baqueta, en el caso de esta implementación existe esta

limitante, los sonidos son grabaciones de una batería real, por lo que el sonido generado va a ser siempre el mismo independientemente de la forma en que se produzca la colisión.

Debido a que el bombo tiene una alta latencia por el uso del pedal, recomendamos que se implemente su funcionamiento con otro sensor del equipo de rastreo (tracker).

Tomando ventaja del diseño escalable de este prototipo, se recomienda que en trabajos futuros se implemente la opción de escoger el tipo de batería que se va a tocar, por ejemplo una de rock, una con tambores africanos, una de estilo japonés.

Se recomienda que se implemente la retroalimentación táctil para hacer más real la interacción con la batería virtual y de esta forma incrementar la sensación de inmersión en el ambiente virtual.

7. Referencias

- [1] Mario A. Gutiérrez A, Frédéric Vexo, Daniel Thalmann, “Stepping into Virtual Reality”, 2008.
- [2] Federico Miyara, La música por computadora, 1995.
- [3] Michael Kriegel, Musical instruments in virtual environments using optical tracking systems, 2005.
- [4] M. Friedmann, T. Starner, A. Pentland, Synchronization in Virtual Realities, Enero 1991.
- [5] How do I evaluate the time difference between two events?, disponible en:
<http://www.codeguru.com/forum/showthread.php?t=379581>, último acceso junio 21, 2010.
- [6] MSc. Gaudencio Zurita, PROBABILIDAD Y ESTADÍSTICA: Fundamentos y Aplicaciones, 2008.
- [5] Grigore C Burdea, Philippe Coiffet, Virtual Reality Technology, 2003.
- [6] 5DT “Fifth Dimension Technologies” disponible en:
<http://www.5dt.com/hardware.html#hmd>, ultimo acceso junio 21, 2010.
- [7] Motion Tracking, Polhemus Liberty disponible en:

- http://www.polhemus.com/?page=Motion_Liberty
- [8] Installing OpenSceneGraph 2.8.0, disponible en:
<http://dwrightdesign.com/2009/05/installing-openscenegraph-280/>, ultimo acceso junio 21, 2010.
- [9] Virtual Drums, 2006, disponible en:
<http://www.virtual-drums.com>, último acceso junio 21, 2010.
- [10] Virtual Air Guitar, disponible en:
<http://airguitar.tml.hut.fi/>, último acceso junio 21, 2010.
- [11] Liberty Manual.pdf, user manual, Octubre 2005.
- [12] Grigore C. Burdea, Haptic Feedback for Virtual Reality, 1999.