

PROGRAMACIÓN DE GUANTES DE DATOS Y AMBIENTES VIRTUALES

M. en C. Morales Alvarado Rodríguez, M. en C. Mauricio Olguín Carbajal

Departamento de Realidad Virtual

Centro de Innovación y Desarrollo Tecnológico en Cómputo (CIDETEC)

Unidad Profesional López Mateos, Instituto Politécnico Nacional

Resumen

Este artículo trata sobre la realidad virtual como una herramienta para facilitar la comprensión y la realización de tareas complicadas y aburridas, a través de entornos de programación y el uso de guantes de datos. Se explican las características de un guante comercial, los lenguajes de programación disponibles y se proporciona una guía para desarrollar una aplicación que permite establecer la comunicación con el dispositivo, así como la presentación de los datos obtenidos en pantalla, para su posterior análisis e interpretación en una interfaz específica.

Palabras Clave: Realidad virtual, Guantes de datos, P5, Controlador Dual.

1. Introducción

La realidad virtual es una interfaz de computadora que comprende simulación e interacciones a través de diversos canales sensoriales en tiempo real, los cuales pueden ser visuales, auditivos, táctiles, olfativos y gustativos [?].

En la actualidad, la realidad virtual ha adquirido un gran auge debido a que, gracias al uso de simulaciones realistas por medio de dispositivos especializados como los guantes de datos, los cascos (*Head Mounted Display*), sistemas de rastreo y sistemas de retroalimentación, se han facilitado muchas tareas, tales como: el análisis de la información, la simulación de ambientes reales o ficticios, la simulación de prototipos en las diferentes áreas de conocimiento, la teleoperación de dispositivos remotos, y todo esto de una manera amigable y atractiva.

Los altos precios que caracterizan a los dispositivos de realidad virtual han provocado la búsqueda de alternativas menos sofisticadas como la simulación por medio de dispositivos de cómputo convencional, como el teclado, el ratón y el monitor. Esto se conoce como realidad virtual de escritorio y dentro de los principales programas de cómputo se puede mencionar: VRML (*Virtual Reality Modeling Language*) [?], Java 3D [?], Direct X [?], 3Ds Max [?], Maya [?], entre otros. La limitante de dichos programas es que la interacción es complicada para el usuario, ya que navegar en un ambiente de tres dimensiones se vuelve una tarea difícil, pues se deben combinar las funciones del ratón y el teclado para efectuar movimientos más complejos como caminar hacia delante y girar, o girar y levantarse. La implementación de un guante de datos como dispositivo de entrada a los ambientes virtuales de escritorio eleva la sensación de inmersión en el usuario y le permite comprenderla mejor desde dentro para el desarrollo de mejores ambientes virtuales [?].

2. Guantes de datos

Los guantes de datos son uno de los dispositivos de rastreo y manipulación de ambientes virtuales más populares hoy en día. El usuario se coloca un guante el cual posee sensores para detectar la flexión de los dedos, así como la posición, rotación y traslación de la mano. Dicha información es enviada a la computadora para ser convertida en instrucciones que afecten un ambiente virtual por medio de una interfaz de software.

Seleccionar un guante de datos para una aplicación no es una tarea sencilla. Algunos factores que deben ser considerados son: costo, tipo de conexión, modo de alimentación y herramientas de desarrollo disponibles [?].

La Tabla 1 presenta una relación de algunos guantes de datos, una breve descripción y su precio.

Cuadro 1 Guantes de datos comerciales

Imagen	Nombre	Descripción	Precio
	5DT Data Glove 5 Ultra	Ideal para animación realística en tiempo real.	995 dólares
	5DT Data Glove Ultra Wireless Kit	Kit <i>Plug-and-play</i> para el guante 5DT Ultra. Con compatibilidad Bluetooth.	1,495 dólares
	DG5-VHand	Guante de datos profesional multipropósito.	495 dólares
	P5 Glove	Guante de datos muy económico y versátil.	59 dólares
	Pinch Glove	Soporta una gran variedad de gestos con las yemas de los dedos.	1,899 dólares

Para la aplicación de este artículo se utilizará el guante P5 debido a su bajo costo, su conexión USB (*Universal Serial Bus*) que supera al puerto serie RS-232 de los guantes Pinch Glove y DG5-VHand. En cuanto al módulo de alimentación, el P5 obtiene la energía del mismo puerto; está provisto de 3 botones programables, uno de encendido/apagado, y a diferencia de los demás puede ser programado con diferentes lenguajes como Visual Basic, Java, Delphi y C++.

3. Guante P5

El guante P5 [?] es un dispositivo de realidad virtual para computadoras personales, económico e ideal para manipular ambientes en tres dimensiones y en general en cualquier ambiente que pueda ser controlado con la naturalidad con la que se usa la mano en el mundo real para manipular cualquier objeto o herramienta [?]. En la Figura 1 se puede observar una fotografía del guante y la torre de recepción en acción.



Figura 1 Guante P5 y torre de recepción.

Una de las características relevantes del guante es que está provisto de dos interfaces físicas que funcionan al mismo tiempo, las cuales son: comportarse como un ratón USB y funcionar como un guante virtual. No obstante, la mayoría de los programadores desactivan la función de ratón USB para que no interfiera en sus aplicaciones.

El modo de operación del guante es muy sencillo. El usuario mueve su mano frente a la torre receptora, la cual constantemente está sensando una serie de emisores infrarrojos, visibles o no, que se encuentran en la palma del guante y convierte esas señales en coordenadas para los ejes X, Y y Z, así como la orientación en grados para dichos ejes.

4. Software para programar el guante P5

Para programar el guante P5 existen diversos kits de desarrollo, de los cuales destacan los siguientes: el SDK (*Software Development Kit*) oficial de Essential Reality para Visual C++ 6.0, la versión beta del controlador Dual de Kenner [?] basado en el código liberado de Essential Reality, y la opción de desarrollar un controlador especial para implementar el guante en algún ambiente de desarrollo en 3D como VRML. De las opciones mencionadas, el controlador Dual es la más atractiva ya que posee las características del SDK oficial y añade mecanismos de filtrado que mejoran los datos que reporta el guante; asimismo, permite programar en diferentes plataformas como Java, Delphi y C/C++ [?].

5. Programación del guante P5

La programación del guante P5 se llevó a cabo en el lenguaje Java [?] sobre la plataforma Windows. Para ello se obtuvo de Internet la versión 3 del controlador dual de Kenner [?], el cual está compuesto de los siguientes archivos: P5DLL.dll y CP5DLL.java. La biblioteca P5DLL.dll es el controlador de Windows para el

guante, mientras que CP5DLL provee las rutinas necesarias para comunicar Java con dicha biblioteca.

Aunque es recomendable copiar la biblioteca P5DLL.dll dentro del directorio donde se ejecutará la aplicación, para este artículo se copió en <Directorio de Windows>\system32, de modo que sea cargada cada vez que se inicia el sistema operativo. Se sugiere hacer una copia de seguridad de la biblioteca original que viene con el guante y que se almacena en dicho directorio.

En cuanto al archivo CP5DLL.java, se debe compilar y empaquetar antes de poder llamar a las diferentes rutinas del mismo. Es importante mencionar que antes de programar el guante usando el controlador dual, se requiere tener instalado el Java SE Development Kit.

Para compilar se escribe:

```
javac -d . CP5DLL.java
```

Una vez compilado, se procede a empaquetarlo:

```
jar cvf CP5DLL.jar com
```

De ahí, se procede a copiar el archivo resultante en: <Directorio de java>\jre\lib\ext y <directorio de la máquina virtual de java>\lib\ext.

6. Contenido del paquete CP5DLL

La clase CP5DLL [?] consta de varias constantes, variables, métodos y de tres clases internas: P5Data, P5Info y P5State. La clase P5Data solo se conserva con fines de compatibilidad con versiones anteriores del controlador. En la Figura 2 se presenta un diagrama jerárquico del paquete CP5DLL.

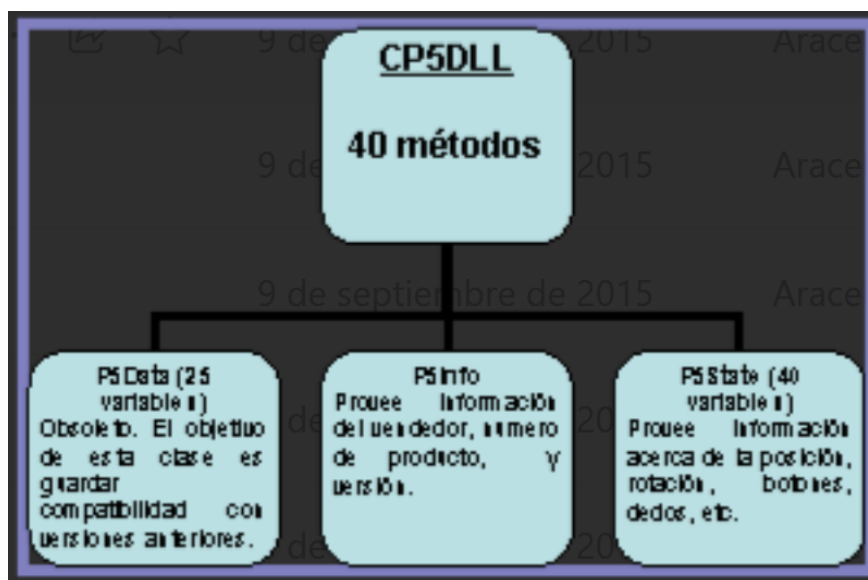


Figura 2 Estructura jerárquica de la clase CP5DLL.

Los 40 métodos públicos de CP5DLL son usados para inicializar y configurar los guantes conectados a la computadora (se recuerda que se puede conectar más de un guante a la vez). Por ejemplo, es posible ajustar el nivel de filtro de los datos que se reciben del guante. Por su parte, las 40 variables en P5State proveen información acerca de la posición del guante, rotación, estado de los botones, de los LED, etc. Existe un método llamado update() el cual se debe llamar para actualizar dichos valores.

Finalmente, las 25 variables públicas de la clase P5Info poseen información acerca de los detalles del guante, como el nombre del fabricante, el número del vendedor, número de serie, etc.

7. Desarrollo de una clase demostrativa del guante

A continuación se desarrollará una clase llamada `PruebaDatos` para desplegar la información que reporta el guante en modo consola de texto. De este modo es más fácil concentrarse en la programación del guante que en los detalles de la interfaz presentada al usuario. Esta clase está compuesta por un menú de opciones que permite seleccionar, por medio del teclado, los datos que se presentan al usuario acerca de la posición y orientación del guante y de las tiras que miden la flexión de los dedos. Se hará uso del teclado en el menú de la aplicación, y de los botones del guante para actualizar los datos. La Tabla 2 presenta el pseudocódigo de la aplicación, y en la Tabla 3 se puede observar el código.

Cuadro 2 Pseudocódigo de la clase `PruebaDatos`.

INICIO
Instanciar las clases <code>CP5DLL</code> y <code>CP5DLL.P5State</code> Establecer el intervalo de muestreo del guante (n) Comprobar la comunicación con el guante Si la comunicación falló: notificar al usuario y terminar Configurar el eje Z+ hacia el usuario Desactivar el modo ratón USB Llamar a <code>P5_GetCount()</code> para determinar el número de guantes Si hay más de 1 guante: avisar y terminar Referenciar el estado del guante 0 Mientras el usuario no desee salir: Desplegar menú y esperar selección Opción 1: Posición y rotación en modo absoluto Actualizar estado; mostrar X, Y, Z con botón B Mostrar rotación X, Y, Z con botón C Esperar n ms; salir con botón A Opción 2: Flexión de los dedos Actualizar estado; mostrar flexión con botón B Esperar n ms; salir con botón A Opción 3: Terminar la aplicación Restaurar el modo ratón USB Cerrar la conexión del guante FIN

Cuadro 3 Código de la clase PruebaDatos.

```
import com.essentialreality.*; // importar el controlador dual de Kenner
import java.io.*;

public class PruebaDatos {
    private CP5DLL gloves;
    private CP5DLL.P5State gloveState;
    private static final int GLOVE_DELAY = 100; // ms de sensado

    public PruebaDatos() {
        gloves = new CP5DLL();
        if (!gloves.P5_Init()) {
            System.out.println("Fallo la inicializacion del guante");
            System.exit(1);
        }
        gloves.P5_SetForwardZ(-1); // Eje Z positivo hacia el usuario
        gloves.P5_SetMouseState(-1, false); // Desactivar modo mouse

        int numGuantes = gloves.P5_GetCount();
        if (numGuantes > 1) {
            System.out.println("Muchos guantes detectados: " + numGuantes);
            System.exit(1);
        }
        gloveState = gloves.state[0];
        int res = 1;

        while (res == 1) {
            System.out.println("Menu de opciones");
            System.out.println("1: Posicion y rotacion del guante modo absoluto");
            System.out.println("2: Flexiones de los dedos");
            System.out.println("3: Salir");
            System.out.print(" ? : ");

            try {
                char opcion = (char) System.in.read();
                System.in.skip(System.in.available());

                switch (opcion) {
                    case '1':
                        System.out.println("Oprima B para posicion, C para rotacion, A
para salir");
                        gloveState.update();
                        while (!gloveState.button[0]) {
                            gloveState.update();
                            if (gloveState.button[1]) {
                                System.out.print("Posicion filtrada x:" + gloveState.
filterPos[0]);

                                System.out.print(" y:" + gloveState.filterPos[1]);
                                System.out.println(" z:" + gloveState.filterPos[2]);
                            }
                            if (gloveState.button[2]) {
                                System.out.print("Rotacion filtrada x:" + gloveState.
filterPitch);

                                System.out.print(" y:" + gloveState.filterYaw);
                                System.out.println(" z:" + gloveState.filterRoll);
                            }
                        }
                        try { Thread.sleep(GLOVE_DELAY); }
                        catch (InterruptedException e) {}
                    }
                    break;

                    case '2':
                        System.out.println("Oprima B para actualizar, A para salir");
                        gloveState.update();
                        while (!gloveState.button[0]) {
                            gloveState.update();
                        }
                    }
                }
            }
        }
    }
}
```

8. Resultados

En las Figuras 3, 4 y 5 se puede observar pantallas de la aplicación en tiempo de ejecución.

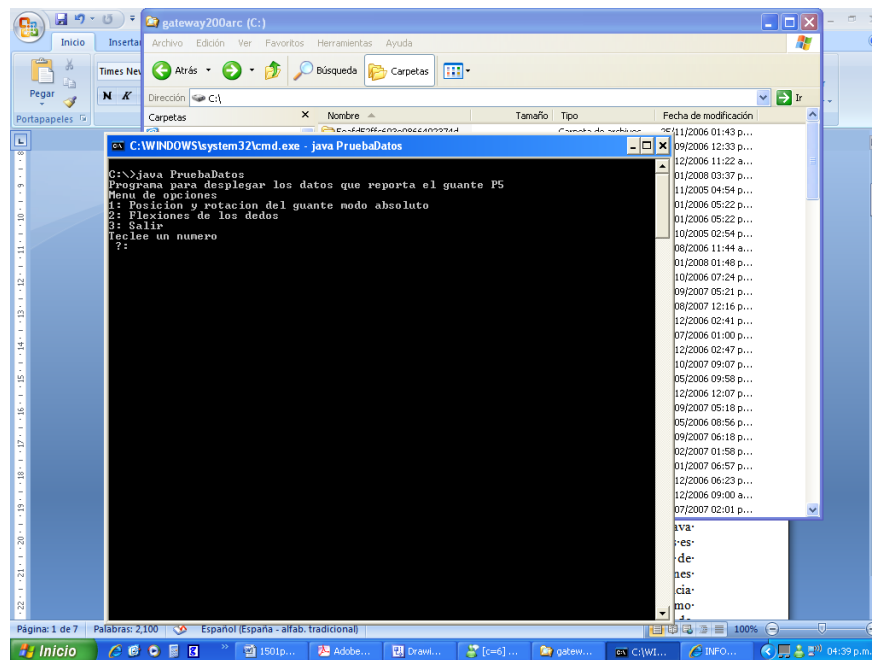


Figura 3 Menú principal de la clase PruebaDatos.

Los valores desplegados por la aplicación son reportados por el guante al momento de oprimir los botones B y C.

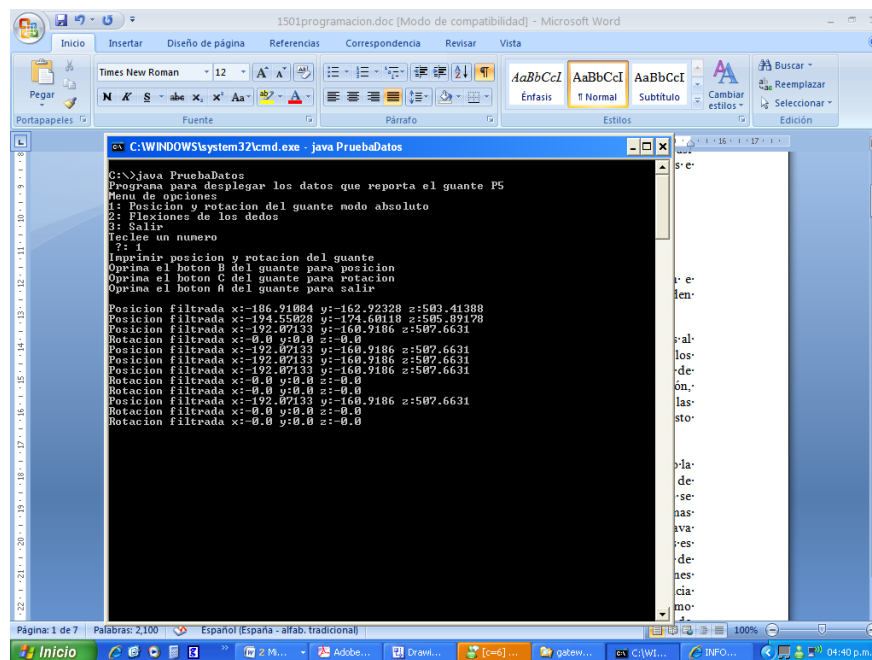


Figura 4 Posición y rotación del guante.

Dentro de la posición, rotación del guante y la flexión de los dedos, se observó que los valores más estables son los últimos y que la rotación presenta varias fluctuaciones aunque se mantenga la mano inmóvil por

periodos prolongados.

```

Imprimir las flexiones de los dedos
Oprima el boton B del guante para actualizar
Oprima el boton A del guante para salir

Dedo pulgar:696 indice:687 medio:774 anular: 915 me:ique: 988
Dedo pulgar:696 indice:687 medio:768 anular: 915 me:ique: 975
Dedo pulgar:696 indice:553 medio:648 anular: 563 me:ique: 743
Dedo pulgar:696 indice:553 medio:648 anular: 563 me:ique: 743
Dedo pulgar:696 indice:553 medio:648 anular: 563 me:ique: 743
Dedo pulgar:696 indice:553 medio:648 anular: 563 me:ique: 743
Dedo pulgar:696 indice:787 medio:883 anular: 851 me:ique: 1849
Dedo pulgar:696 indice:787 medio:981 anular: 851 me:ique: 1872
Dedo pulgar:696 indice:728 medio:987 anular: 872 me:ique: 1812
Dedo pulgar:696 indice:722 medio:987 anular: 862 me:ique: 988
Dedo pulgar:696 indice:553 medio:648 anular: 563 me:ique: 743
  
```

Figura 5 Flexión de los dedos.

Es importante mencionar que los datos reportados por el guante P5 sobre la posición están expresados en *P5 Units* y que equivalen a 0.0488 centímetros. Las unidades de la rotación y la flexión de los dedos también son internas del guante y se desconoce su valor con exactitud.

9. Resultados obtenidos

En el presente trabajo se analizaron las características más relevantes del guante P5, y se demostró por medio de una clase programada en Java la forma de obtener y desplegar los datos que reporta el guante a través del controlador dual. De este modo, el usuario puede valerse de esta aplicación para comprender la naturaleza de los datos obtenidos y darles la interpretación necesaria para futuros desarrollos.

10. Conclusiones

La realidad virtual es una técnica que facilita la realización de tareas complicadas, difíciles de comprender y aburridas, ya que por medio de la simulación y la interacción con los sentidos humanos es posible trabajar de una manera más sencilla, amigable y con un mayor nivel de retroalimentación. Los guantes de datos son dispositivos muy socorridos en esta área del conocimiento, debido a que hacen uso de la mano evitando la memorización de combinaciones de teclas y aprovechando una parte esencial del cuerpo.

Existen hoy en día diversos guantes comerciales que ofrecen distintas prestaciones de rendimiento, precisión de los datos, así como diferentes técnicas de rastreo; la decisión del usuario depende directamente del uso que se le dará y la plataforma en la que desea trabajar.

El guante P5 es un dispositivo económico y sencillo de utilizar, el cual puede ser implementado en ambientes virtuales de escritorio con el fin de mejorar la interacción con la computadora. Por esta razón es importante conocer las características de este dispositivo, las herramientas de programación disponibles y las plataformas en que éstas trabajan. Una vez que se establece la comunicación con el guante y se conoce la naturaleza de los datos que se reciben, es posible implementar interfaces dirigidas a un problema particular, con todos los beneficios que se adquieren en la interacción con el usuario.

La aplicación *PruebaDatos* desarrollada en este artículo permite a los desarrolladores conocer los datos devueltos por el guante en tiempo real y proporciona una base para familiarizarse con la plataforma Java, de modo que se puedan aprovechar las diversas tecnologías existentes como redes, programación web, entre otras.

Finalmente, se hace hincapié en la necesidad de desarrollar ambientes virtuales accesibles al usuario común, con herramientas gratuitas y dispositivos económicos.

Referencias

- [1] G. Burdea y P. Coiffet. *Virtual Reality Technology, 2nd edition*. Wiley-Interscience, E.U.A., 2003.
- [2] VRML 97 and related specifications, Web3D Consortium. <http://www.web3d.org/x3d/specifications/vrml/>
- [3] Java 3D, Sun Microsystems. <http://java.sun.com/products/java-media/3D/index.html>

- [4] DirectX, Microsoft. <http://www.microsoft.com/windows/directx>
- [5] 3Ds Max, Autodesk. <http://mexico.autodesk.com/adsk/servlet/index?siteID=1002155&id=9978871>
- [6] Maya, Autodesk. <http://mexico.autodesk.com/adsk/servlet/home?siteID=1002155&id=7659874>
- [7] R. Morales Alvarado. *Interfaz para el aprendizaje de la realidad virtual haciendo uso de un guante de datos*. IPN CIDETEC, México, 2007.
- [8] P5 Glove Wikia. http://scratchpad.wikia.com/wiki/P5_Glove
- [9] A. Dengel, S. Agne, B. Klein, A. Ebert et al. "Human-centered interaction with documents". *Knowledge Management Lab, Intelligent Visualization Lab, DFKI GmbH*, HCM, Alemania, 2006.
- [10] Java SE Development Kit, Sun Microsystems. <http://java.sun.com/javase/downloads/?intcmp=1281>
- [11] A. Davison. *Pro Java 6 3D Game Development*. Apress, E.U.A., 2007.
- [12] C. Kenner. Dual Mode Driver. http://www.geocities.com/car1_a_kenner/p5glove.html