

Interfaz Natural para el Control de Drone Mediante Kinect

Natural Interface Control Using Drone Kinect

Eduardo Berra^{a *}, José Rodrigo Cuautle^{a **}

^aUniversidad del Valle de Puebla, Calle 3 Sur No. 5759, Col. El Cerrito C.P. 72440, Puebla, México

Recibido: 28/06/2013; revisado: 5/07/2013; aceptado: 30/07/2013.

Berra, E., Cuautle, J. R.: Interfaz Natural para el Control de Drone Mediante Kinect. *Jou.Cie.Ing.* **5** (1): 53-65, 2013. ISSN 2145-2628.

Resumen

Se presenta el diseño e implementación de un sistema de control de radio frecuencia manipulado por pc para un drone por medio de una interfaz natural que comprende gestos corporales detectados con Kinect, sin la adición de herramienta alguna de control automático de lazo cerrado

Palabras Claves: Interfaz natural, Kinect, gesto corporal, control, drone.

Abstract

Se presenta el diseño e implementación de un sistema de control de radio frecuencia manipulado por pc para un drone por medio de una interfaz natural que comprende gestos corporales detectados con Kinect, sin la adición de herramienta alguna de control automático de lazo cerrado

Keywords: Natural interface, Kinect, body gesture, control, drone.

1. Introducción

Una interfaz natural de usuario es aquella que permite interactuar con un sistema sin utilizar dispositivos de entrada como el ratón o el teclado, entre otros. En cambio, se recurre a la interacción a través de movimientos gestuales realizados con las manos o con el cuerpo, así como a través de comandos de voz.

Generalmente, la parte del sistema que recibe los comandos emitidos por el usuario en un sistema con interfaz natural, es una pantalla táctil, la cual es un elemento adecuado para algunas aplicaciones de cómputo, pero no lo será cuando el objetivo de un usuario en el futuro cercano sea interactuar con dispositivos mecánicos situados en su entorno.

Pensemos por ejemplo en el caso del control de un sistema robótico que se desempeña en un ambiente pe-

ligroso o dañino para el humano, en el caso de una persona con discapacidad motriz que desee realizar alguna tarea doméstica a través de un dispositivo mecatrónico, o bien en el usuario que controla el sistema domótico de su casa a través de gestos.

Para el desarrollo de este tipo de interfaces han aparecido en años recientes dispositivos y entornos de desarrollo, como Kinect de la empresa Microsoft [12], WAVI Xtion de la empresa Asus [14], o Grasp de Omek [13], recientemente adquirida por Intel.

En el presente trabajo, se plantea responder a la pregunta sobre qué elementos intervienen y cómo se relacionan en la implementación de una interfaz natural soportada por Kinect para el control de dispositivos mecatrónicos.

En este contexto existen trabajos como el de Chaotic Moons Labs [1], donde han desarrollado una patineta

* 661@uwp.edu.mx

**rodrigoparodi@uwp.edu.mx

motorizada con llantas todo terreno que utiliza Kinect para moverse. Esta patineta interactúa con el usuario sensando movimientos a través de Kinect conectado a una Tablet, la cual toma los valores de los movimientos de Kinect para emitir comandos a un controlador de los motores, y así se muevan hacia donde el usuario ordena.

Por otra parte, Taylor Veltrop vincula el uso de Kinect con un enlace inalámbrico para controlar a un robot [2]. Básicamente el código desarrollado toma los datos adquiridos por Kinect y los envía a un software instalado en el robot para que este imite los movimientos.

Por su cuenta, Ar Drinct [3], un proyecto desarrollado por estudiantes franceses, provee una biblioteca para el control de drones ARDrone de la marca Parrot mediante el uso de java y Kinect. El sistema consiste en una interfaz llamada AR.Drinct que sigue los movimientos del usuario controlando al avión no tripulado de acuerdo a una barra de indicación que se muestra según los movimientos de la mano. Sin embargo su precisión no es muy buena. La línea ARdrone de Parrot ha impulsado el desarrollo de aplicaciones mediante el lanzamiento de su propia API que facilita a los programadores la interacción con este Drone.

Flying Machine Arena [4] es otro proyecto basado en drones que interactúan en un lazo cerrado ya sea controlado por un computador o mediante un operador captado por Kinect. Trata sobre el aspecto de automatismo y control de vuelo mediante la captura de imágenes para la planeación de vuelos en espacios reducidos. El tipo de drones que aquí se manejan son desarrollados por los mismos investigadores aplicando la no comercialización de los mismos.

De esta manera, el objetivo de este trabajo es diseñar e implementar el sistema de control para un drone mediante gestos corporales detectados con Kinect, con el

fin de acercar este tipo de sistemas a los estudiantes de nuestras universidades.

2. Interfaz Natural de Usuario con Kinect

2.1. Interfaz Natural de Usuario

Un principio de una interfaz natural es ser intuitiva, de tal modo que el usuario no requiera entrenamiento o éste sea mínimo [7, 11]. Esto implica una reducción de tiempos y costos de utilización de software, así como una mayor aceptación del usuario.

Un enfoque para la creación de interfaces es diseñarlas de tal forma que los usuarios realicen sus tareas de forma rápida; otro enfoque es crear interfaces que los usuarios prefieran emplear por su facilidad.

Las interfaces naturales son fluidas, es decir, permiten al usuario realizar tareas, percatándose apenas de que están utilizando una interfaz. Esto permite el uso sin interrupciones y de una manera cómoda y eficaz [8].

Dentro del diseño de interfaces el tiempo de completado es importante. Los usuarios son impacientes y no les gusta esperar a que las mismas reaccionen en tiempos lentos y con desfases a las actividades llevadas en los procesos que las mismas realizan.

2.2. Controlador de Videojuegos Kinect

Kinect comprende un arreglo de sensores y actuadores que permiten la interacción entre el programa y el usuario de manera natural: cámara de color, emisor infrarrojo, sensor infrarrojo de profundidad, arreglo de micrófonos, y led de estado [12], ubicados como muestra la figura 1.

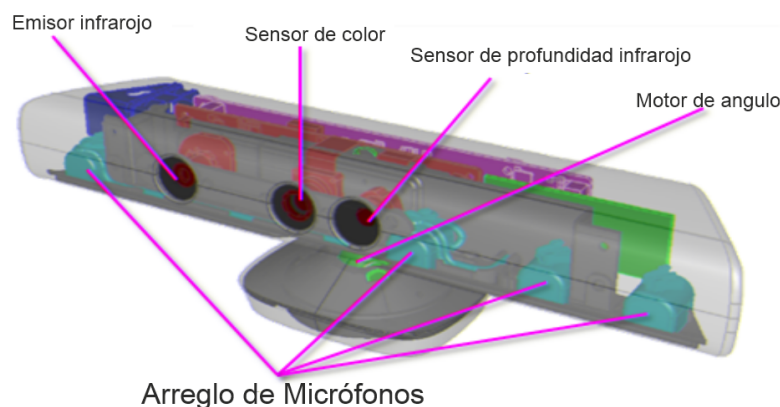


Figura 1. Ubicación de los sensores y actuadores contenidos en el controlador de videojuegos Kinect para Xbox 360.

La cámara de color soporta hasta 30 cuadros por segundos a una resolución de 640x480 píxeles, el máximo de resolución es de 1280x960. El rango del ángulo de la cámara es 43 grados en vertical y 57 en horizontal.

La pareja de emisor y sensor de profundidad infrarroja permite obtener la distancia entre un punto determinado de un objeto y la cámara, empleando el método de luz estructurada. La resolución soportada es de

640x480, 320x240 y 80x60 píxeles, la información obtenida se muestra utilizando gradientes para distinguir la profundidad entre objetos tridimensionales como se muestra en la figura 2.

El arreglo de micrófonos cuenta con cuatro elementos que captan la dirección de la onda de sonido de acuerdo a su ubicación.

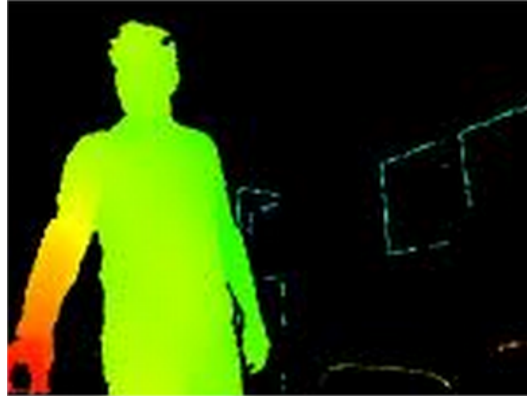


Figura 2. Imagen obtenida del sensor de profundidad.

2.3. Kit de Desarrollo de Software para Kinect

La arquitectura del kit de desarrollo de software (SDK) para Kinect, se conforma por tres partes principales (figura 3). La parte mostrada en el nivel supe-

rior de la figura basa su funcionamiento en el control del hardware y de los sensores o cámaras con las que cuenta Kinect, el control de arreglo de micrófonos, la conexión USB y los motores que permiten el enfoque del dispositivo.

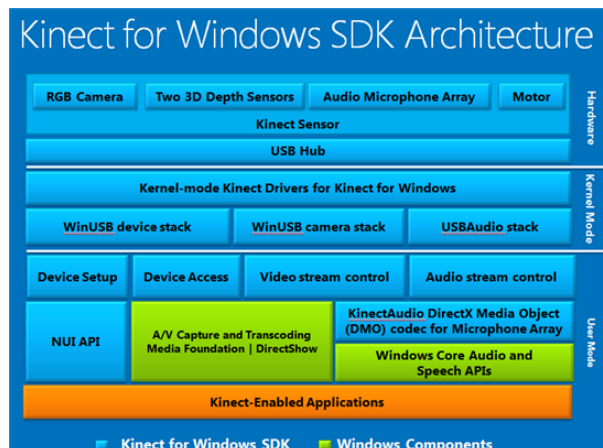


Figura 3. Arquitectura del kit de desarrollo de software de Kinect.

El nivel central está encargado de la comunicación, permite la carga de los datos en crudo que los sensores detectan.

El nivel mostrado en la parte inferior proporciona la interacción entre la aplicación C# y el usuario final, se encarga de enmarcar y ocultar los procesos para el análisis de los datos en crudo, presentado al programador una interfaz adecuada para el tratamiento de datos. El sitio para descarga y seccion de tools es posible encon-

trarlos en [10].

2.4. *Skeletal de Kinect*

Kinect representa las articulaciones y el esqueleto del ser humano en un grafo con 20 nodos sensados, la figura 4 muestra los nombres y la ubicación de los mismos, la figura 5 muestra la posición relativa del usuario con respecto al controlador Kinect[6].

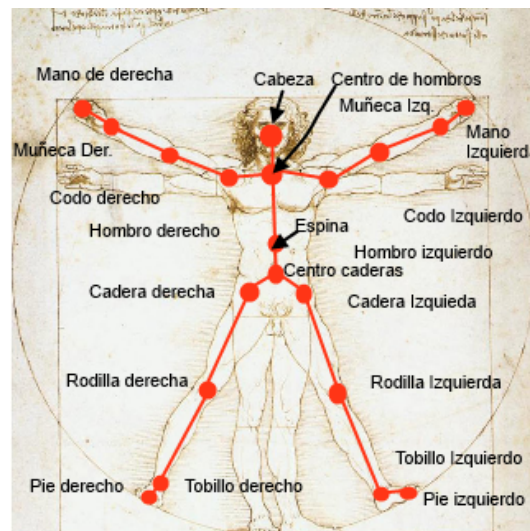


Figura 4. Puntos del cuerpo humano detectados por Kinect.

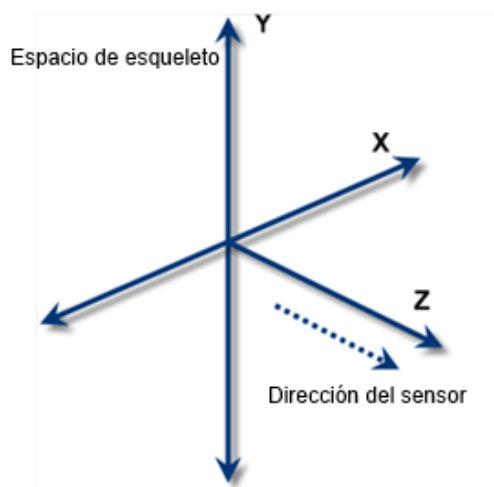


Figura 5. Ubicación del espacio del esqueleto con respecto al controlador Kinect.

2.5. Sistema de Profundidad

En tiempo de ejecución Kinect tiene la capacidad de procesar los datos de profundidad para identificar hasta seis figuras humanas en un mapa de segmentación, este mapa contiene los datos de proximidad de los objetos en el ángulo de visión de la cámara de Kinect. Para acceder a los datos de la profundidad de un objeto en el caso de la figura humana tiene que estar activo el seguimiento del esqueleto. El termino segmentación se entiende como el conjunto de seguimiento del esqueleto y la toma de los pixeles de profundidad del sensor de Kinect como un solo flujo de datos. Cuando el valor es un 0 en el flujo devuelto por la profundidad, nos

indica que no existe ninguna persona delante del sensor, si existe una variación del valor del 1 al 6 se tiene por consiguiente un flujo que identifica hasta 6 posibles personas. Los valores de profundidad se utilizan como una máscara para aislar las porciones del usuario o una cierta región de interés para su análisis.

El sistema de profundidad tiene un rango en la distancia de detección de 800 a 4000 mm. Los rangos para captar objetos a distancia tienen algunas limitantes, una de ellas es que la cámara de profundidad utiliza infrarrojos por lo que no es confiable para detectar obstáculos transparentes como los cristales, de la misma manera no se puede usar bajo la acción de la luz solar. La interpretación de los puntos a seguir se hace mediante la profundidad y la posición de cada pixel de la imagen.

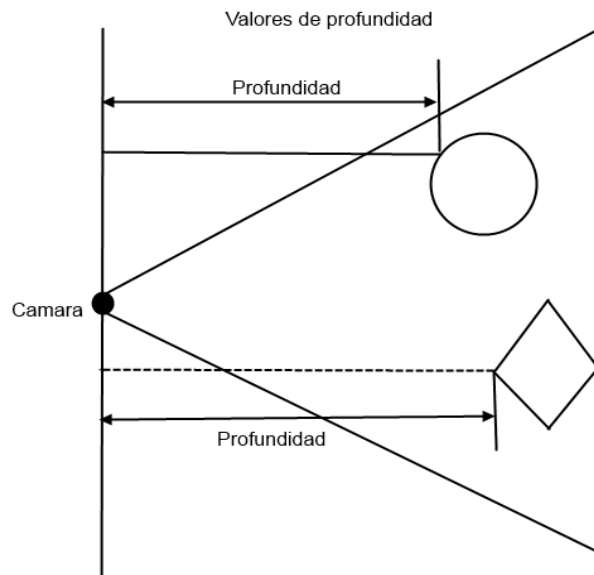


Figura 6. Acotación para la obtención de coordenadas bajo el sistema de profundidad

Algunas consideraciones a tomar en el sensor son las sombras que se pueden presentar durante la emisión de infrarrojo, por defecto las sombras provocan un patrón en el sensor que impide el cálculo de profundidad por lo deben evitarse para disminuir la distorsión en la medición.

3. Resultados

3.1. Arquitectura del sistema

La figura 7 muestra la organización del sistema: los gestos asumidos por el usuario en sus extremidades superiores son adquiridos a través de Kinect, interpretados en un computador anfitrión, y mapeados a tres escalas del 0 al 100 % correspondientes a los ejes espaciales x, y, z. Los valores calculados son codificados y enviados a un micro controlador encargado de manipular tres potenciómetros digitales, conectados al control de infrarrojo del drone.

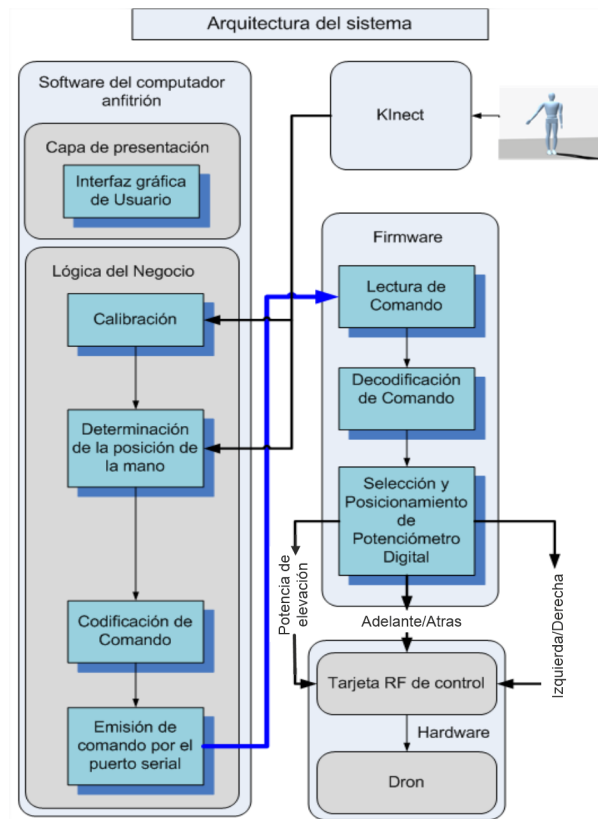


Figura 7. Arquitectura del sistema

3.2. Interfaz de Usuario y Gestos de Control

La figura 8 ilustra la forma propuesta para la interacción del usuario con el sistema: éste debe colocarse

frente al controlador Kinect para que a través de gestos corporales ocasione que el dron levante el vuelo y describa la trayectoria deseada.

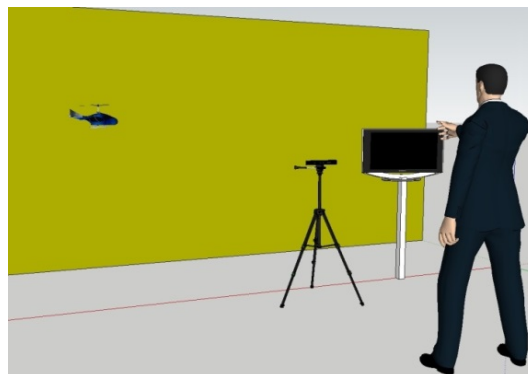


Figura 8. Ejemplificación del enfoque adoptado para el control del dron a través de gestos corporales

Para la generación de gestos de control, el usuario debe colocarse de pie frente al Kinect a una distancia aproximada de 2 m. Su mano derecha está encargada de los desplazamientos arriba-abajo y derecha-izquierda

(figuras 9 y 10), su mano izquierda controla el desplazamiento adelante-atrás. Estos ejes se encuentran relacionados con la API de profundidad y Skeletal de Kinect

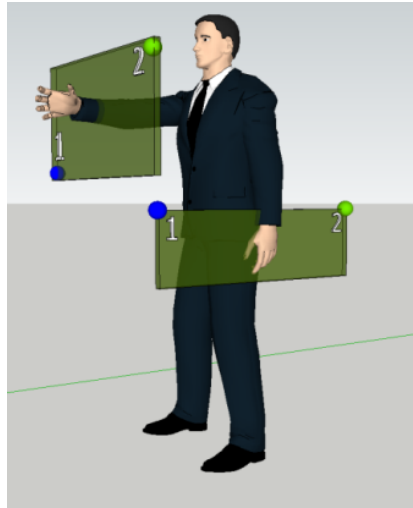


Figura 9. Rango de movimiento para los gestos del usuario.

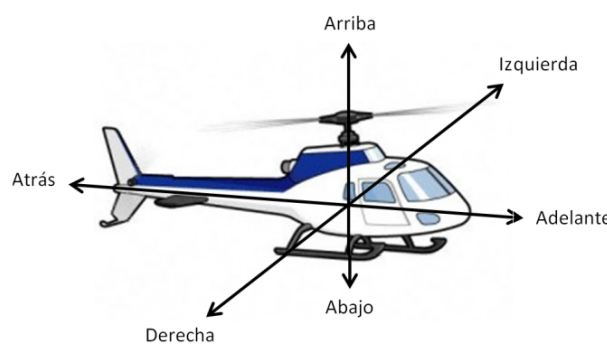


Figura 10. Dirección y sentido de movimientos comunicados al drone.

3.3. Calibración

Para que el vehículo pueda ser controlado bajo diferentes características anatómicas, resulta necesario realizar una calibración para el usuario en turno. Esto

se logra por medio de dos gestos corporales, como se muestra en la figura 11. Estos gestos, basados en las manos del usuario, definen los extremos del rango reconocido por el sistema para cada uno de los 3 ejes espaciales (figura 9).

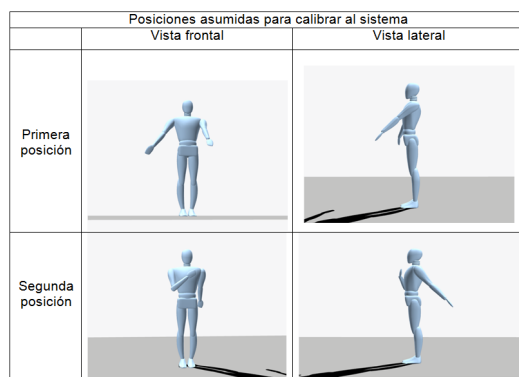


Figura 11. Gestos para calibrar el sistema.

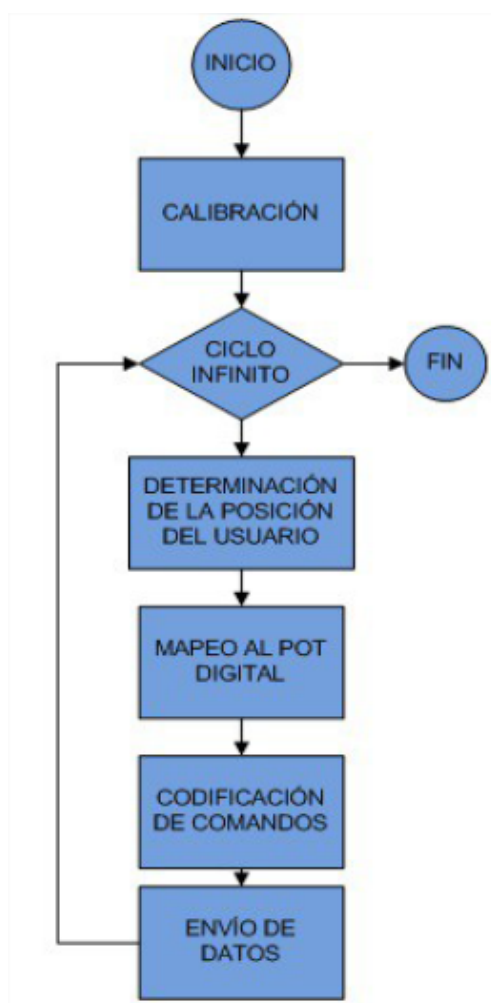


Figura 12. Gestos para calibrar el sistema.

3.4. Software del computador anfitrión

La figura 12 muestra un diagrama de flujo de alto nivel para el software del computador anfitrión. Ya que originalmente el drone fue diseñado para controlarse por un dispositivo de radiofrecuencia con potenciómetros para determinar el desplazamiento en cada uno de los tres ejes espaciales, se realiza un mapeo entre el rango definido para los gestos corporales y el rango de tres potenciómetros digitales sustitutos. La codificación de comandos comprende un byte que contiene la posición de estos potenciómetros vinculados cada uno a un eje espacial. El envío de los mismos a un micro con-

trolador se realiza de manera continua en intervalos del orden de milisegundos. La figura 12 presenta el diagrama de flujo de alto nivel del código que se ejecuta en el computador.

3.5. Hardware y Firmware

Los comandos enviados por el computador anfitrión son captados por una tarjeta Arduino [9] (figura 14), donde se decodifican y se transforman en pulsos adecuados para la configuración de potenciómetros digitales ds1669 (Figura 13).

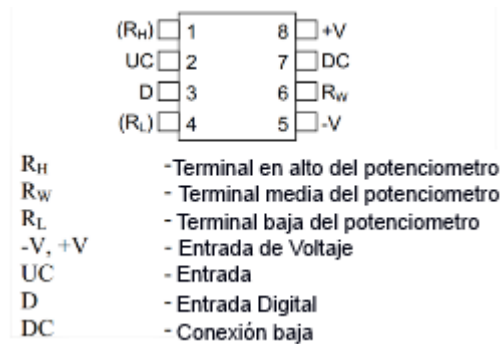


Figura 13. Potenciómetro Digital.

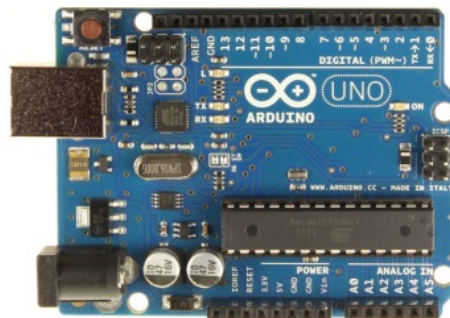


Figura 14. Placa Arduino R3.

3.6. Pruebas Realizadas

El sistema obtenido permite manipular al vehículo, a través de gestos corporales y Kinect, las figuras 15 a la

17 muestran algunos aspectos de la forma de interacción y del prototipo obtenido.



Figura 15. Interacción del usuario con el sistema.

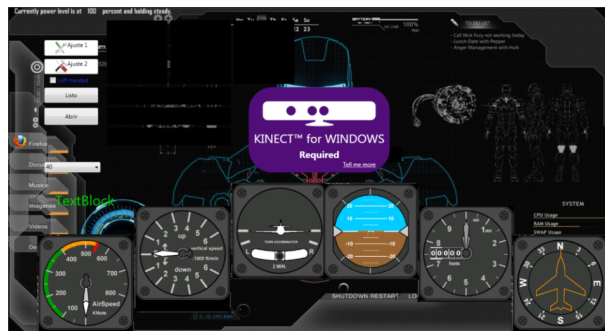


Figura 16. Interfaz gráfica de usuario.



Figura 17. Interfaz con potenciómetros digitales para comunicar el computador anfitrión con el control de radio frecuencia.

La implementación del prototipo y su función se puede observar en las figura 18 en la cual la interacción de Kinect mediante el código de la computadora anfitrión

maneja las posiciones a través de pulsos para después ser transmitidas al dron.

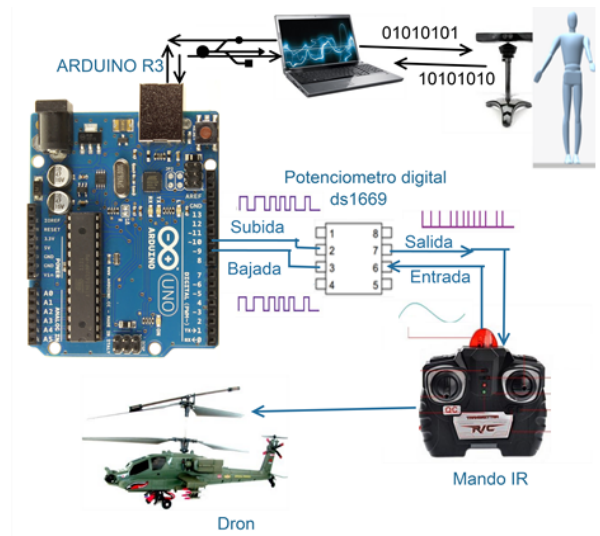


Figura 18. Interaccion del sistema.

Al llevar a cabo una prueba para aprender a usar la interface natural la cual fue realizada con 4 estudiantes, ajenos al proyecto se pudo observar el siguiente comportamiento:

1. En la primera prueba de 5 a 3 minutos de vuelo en un primer intento quedaba claro que la posición de altura con respecto al dron se ordena elevando la mano derecha, sin embargo los movimientos de adelante hacia a tras se eliminaban por completo debido a que la mano que controla dicho movimiento es la izquierda.
2. En la segunda de 5 a 3 minutos de vuelo los movimientos y control del dron mejoraron debido a

que en la primera etapa de pruebas se tenía contemplado elevar al dron pero por la misma inexperticia del uso de la interface natural es difícil coordinar los movimientos que controlen las posiciones del dron.

La siguiente tabla muestra el uso de cada uno de los ejes de los usuarios en sus intentos dicha tabla fue adquirida de la interface en los momentos que los usuarios tenían una interacción con el dron.

De los resultados obtenidos en la muestra se puede observar que en la primera interacción con la interfaz se tiene un grado de complejidad de uso que va disminuyendo conforme el usuario se adapta a la interfaz.

Alumno	Eje X % uso	Eje Y % uso	Eje Z % uso	Prueba
Alumno 1	30	60	60	1era. Prueba
Alumno 2	10	10	10	
Alumno 3	20	70	10	
Alumno 4	10	60	30	
	Eje X % uso	Eje Y % uso	Eje Z % uso	
Alumno 1	30	60	70	2da. Prueba
Alumno 2	40	80	70	
Alumno 3	50	80	40	
Alumno 4	10	60	30	
	Eje X % uso	Eje Y % uso	Eje Z % uso	
Alumno 1	30	65	80	3era. Prueba
Alumno 2	50	80	75	
Alumno 3	80	90	60	
Alumno 4	25	60	40	

Figura 19. Tabla 1. Prueba para la interfaz.

4. Conclusiones

Se han presentado detalles de la implementación de un sistema para controlar un drone por medio de una interfaz natural con Kinect. El sistema no contempla ningún sistema de control automático debido al interés que se tiene en medir la rapidez de respuesta del sistema.

La repuesta del sistema de profundidad manejado en la implementación tiene los siguientes problemas:

1. Congelamiento de la interfaz de profundidad
2. Calibración del sistema con desfases debido al buffer de video de profundidad.
3. Cualquier movimiento es censado en la interacción con el dron.

El congelamiento sobre el sistema de profundidad puede ser remediado aumentando el buffer para la captura de video, afortunadamente esta congelamiento en la imagen no interrumpe el flujo de información de movimiento posicional para el dron, sin embargo si afecta la fase de calibración de movimientos. El censado de los movimientos durante la interacción tiende a evitar que usuario pueda adoptar posturas diferentes a las de control por lo que lo ata en directo con la acción del dron, debido a esto se a pensado en la posibilidad de tener un gesto para detener el censado y control directo del dron.

Con los resultados obtenidos en el desempeño de la interface concluimos que es viable la implementación de interfaces naturales para el control de drones con un rango confiable de precisión para su manejo.

5. Trabajo a Futuro

La medición propuesta a futuro se describe en las tablas 2 a 3, que mostrarán los valores obtenidos de los parámetros seleccionados para evaluar la rapidez del sistema implementado. Los datos de la tabla 2 corresponderán al promedio obtenido luego de 10 sesiones de 20 minutos cada una. Para obtener los datos de la tabla 3 se seleccionarán 3 usuarios sin experiencia en el uso de dispositivos de control por radiofrecuencia, se les permitirá practicar a cada uno durante 5 minutos, tanto con el control RF como con la interfaz natural con Kinect. Acto seguido, cada usuario realizará 10 intentos para ejecutar un recorrido con el punto de despegue y de aterrizaje al nivel del suelo, situados a 10 metros entre sí y dentro de un recinto de 5 m de altura. Además se colocará una barrera a 3 m de altura entre estos puntos sobre la que deberán navegar para llegar al punto de aterrizaje. Cualquier colisión durante el vuelo cancelará el

intento en turno, por lo que los dos últimos parámetros de la tabla 3 se obtendrán sobre los vuelos exitosos.

Lo anterior se encamina a ayudar a decidir qué aplicaciones de control son pertinentes para dispositivos como Kinect y proponer soluciones a los problemas que se detecten.

Tabla 1. Velocidad Máxima Identificada.

Eje	Mano izquierda
X	
Y	
Z	

Tabla 2. Fluidez en la ejecución del software Identificada.

Trabamientos por minuto	Tiempo promedio por trabamiento

Tabla 3. Capacidad de control del usuario según el método empleado.

Criterio	Control directo por dispositivo de Rado Frecuencia
Vuelos exitosos	
Precisión del aterrizaje	
Tiempo requerido para completar el recorrido	

Referencias

- [1] Chaotic Moon Studios. (2013, Septiembre) CM.LABS. [Online].<http://www.chaoticmoon.com/labs>
- [2] Taylor Veltrop. (2012, Sep.) ROS.org. [Online]. [http://wiki.ros.org/openni/Contents/ROS %203D/Humanoid %20Teleoperation#Public_Exposure](http://wiki.ros.org/openni/Contents/ROS%203D/Humanoid%20Teleoperation#Public_Exposure)
- [3] Nathan Gaberel. (2012, Sep.) ArDrinet. [Online].<https://code.google.com/p/ardrinet/>
- [4] Institute for Dynamic Systems and Control. (2012, Nov.) Flying machine Arena. [Online]. <http://www.flyingmachinearena.org>
- [5] Parrot. (2012, Sep.) Parrot. [Online]. <http://ardrone2.parrot.com/apps>
- [6] Abhishek Kar, Skeletal Tracking using Microsoft Kinect, Department of Computer Science and Engineering, IIT Kanpur.

- [7] Gideon Steinberg, Natural User Interfaces, in CHI, Austin, Texas, USA, 2012.
- [8] Sergio Herman Peralta Benhumea, Interfaz de lenguaje natural usando Kinect. Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, Distrito Federal, México, 2012.
- [9] Arduino. (2012, Oct.) Arduino. [Online]. <http://www.arduino.cc/>
- [10] Microsoft. (2012, Aug.) Kinect for Windows-SDK. [Online]. <http://www.microsoft.com/en-us/kinectforwindowsdev/Start.aspx>
- [11] Maged N Kamel Boulos et al., Web GIS in practice X: a Microsoft Kinect natural user interface for Google Earth navigation, International Journal of health Geographics, 2011 10:45.
- [12] Microsoft. XBOX. [Online]. <http://www.xbox.com/es-MX/Kinect>
- [13] Omek. (2013, Feb.) Omek, Gesture Recognition and Tracking Technology.[Online]. <http://www.omekinteractive.com/products/grasp/>
- [14] Asus. (2013, Feb.) Wavi Xtion. [Online]. http://event.asus.com/wavi/product/WAVI_Xtion.aspx