

## **Guía para uso del Microsoft Kinect for Windows V1**

### **Sobre el producto**

El Microsoft Kinect for Windows es un producto que salió al mercado en el año 2012. El apoyo en cuanto a *drivers* y *software* (Kinect for Windows SDK) que sacó la compañía fue contaba con soporte para el sistema operativo Windows 7. En el 2014, Microsoft sacó un segundo modelo, para el cuál había bastantes cambios en cuanto al soporte. Con el tiempo, se fue perdiendo el soporte oficial para el aparato y no todas las librerías son compatibles con los nuevos sistemas operativos. Por lo mismo, surgió un esfuerzo colaborativo para generar el soporte necesario a través de unas librerías *open source* conocidas como Open Kinect, que buscan adaptar el aparato a diferentes sistemas operativos, como Linux y versiones más recientes de Windows. A través de diferentes pruebas que se estuvieron realizando con el aparato, surge esta guía para la instalación de las librerías con las que se pudo llamar el aparato.

### **Uso en Windows 11**

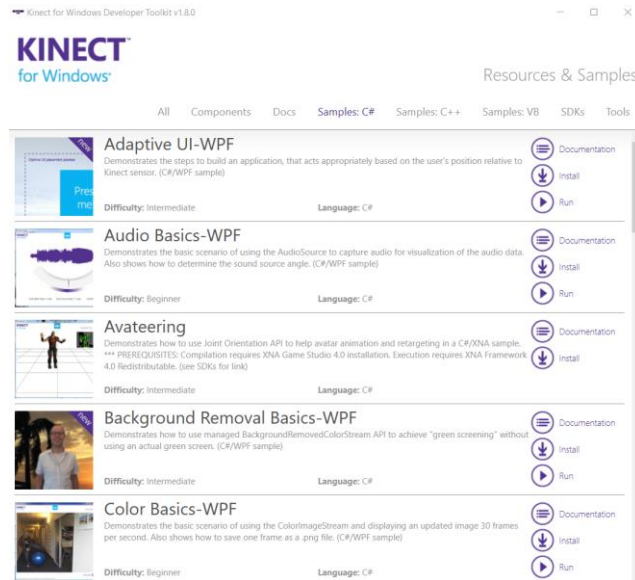
La versión original de Kinect for Windows SDK 1.0 no es compatible con sistemas operativos de Windows a partir de Windows 8. Además, a partir de versiones de las librerías 2.0, el aparato no es compatible ya que fueron diseñadas para el modelo 2.

Después de varias instalaciones fallidas, con las que el sistema detectaba la cámara, pero sacaba un mensaje de error en los *drivers* al mostrar el aparato en el Administrador de Dispositivos, se encontró el sistema compatible con Windows 11: la versión de Kinect for Windows SDK 1.8.

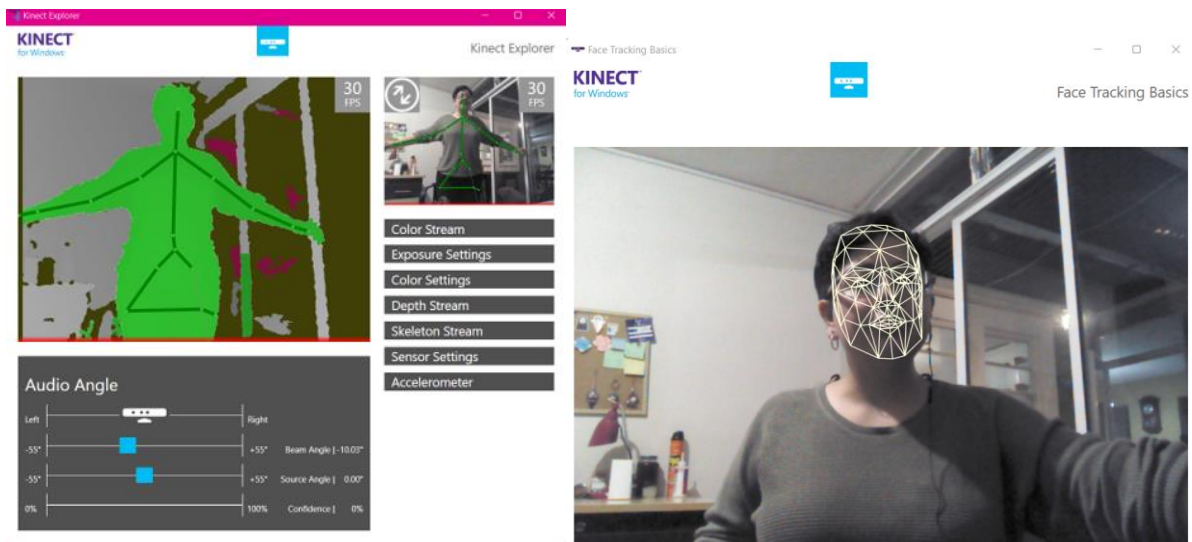
Este se puede encontrar en el siguiente enlace: <https://www.microsoft.com/en-us/download/details.aspx?id=40278>

Para su uso se recomienda también instalar la interfaz gráfica de Kinect for Windows Developer Toolkit v1.8, ya que cuenta con varios módulos instalados y permite una visualización rápida de ellos utilizando la cámara. Este se puede descargar desde el enlace: <https://www.microsoft.com/en-us/download/details.aspx?id=40276>

La combinación de ambos permite realizar pruebas de una gran cantidad de módulos en C#, C++ o Visual Basic, al igual de tener un acceso fácil a los parámetros de la cámara.



**Figura 1:** Pantalla de Kinect for Windows Desktop Toolkit v1.8 con ejemplos en C# que se pueden utilizar



**Figura 2:** Algunos de los módulos que se pueden probar: Kinect Explorer (izquierda), que permite ver los controles e imágenes del aparato y Face Tracking Basics (derecha), que identifica el rostro de una persona.

Esta aplicación es bastante útil, ya que sí permite tener acceso a la información de la cámara y probar diferentes módulos. La aplicación tiene un botón para leer la documentación de cada uno de los módulos, pero este no lleva a la página específica.

La información de esta versión del Toolkit, desde *troubleshooting* hasta la documentación de los módulos, se puede encontrar en el siguiente link: [https://learn.microsoft.com/en-us/previous-versions/windows/kinect-1.8/hh855347\(v=ie8.10\)](https://learn.microsoft.com/en-us/previous-versions/windows/kinect-1.8/hh855347(v=ie8.10))

Esta información es bastante general, sin embargo, y no incluye detalles de la programación de los módulos.

Sin embargo, para la programación e interacción de la cámara directamente se recomienda utilizar el programa Visual Studio. Se recomienda utilizar una versión de la aplicación pasada, empezando con Visual Studio 2010, ya que, para versiones nuevas, la versión que es compatible es el modelo más nuevo del Kinect, el Microsoft Kinect Azure. Para este tipo de aplicaciones existen diferentes referencias, incluyendo el libro: *Beginning Kinect Programming with the Microsoft Kinect SDK* por James Ashley y Jarrett Webb.

### Uso en Linux Ubuntu Desktop

Para el uso del Kinect en Linux se puede utilizar la librería de Open Kinect. Sin embargo, esta librería también ha ido perdiendo soporte oficial por lo que no es recomendable realizar la instalación directa con la instrucción de:

```
sudo apt-get install libfreenect
```

Al realizar la instalación de esta manera, no se obtienen los *drivers* más actualizados. Es necesario entonces realizar la instalación desde la fuente. Para esto se corren las siguientes instrucciones:

```
sudo apt-get update
```

```
sudo apt-get install cmake build-essential libusb-1.0-0-dev
```

```
git clone https://github.com/OpenKinect/libfreenect.git
```

```
cd libfreenect
```

```
mkdir build && cd build
```

```
cmake -L ..
```

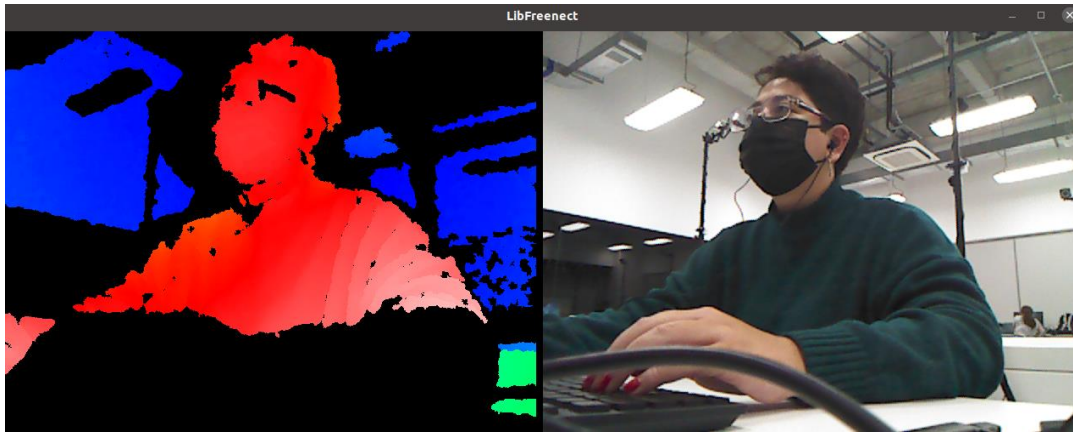
```
make
```

```
sudo make install
```

Una vez se haya terminado la instalación, se puede probar con un ejemplo al ingresar la instrucción:

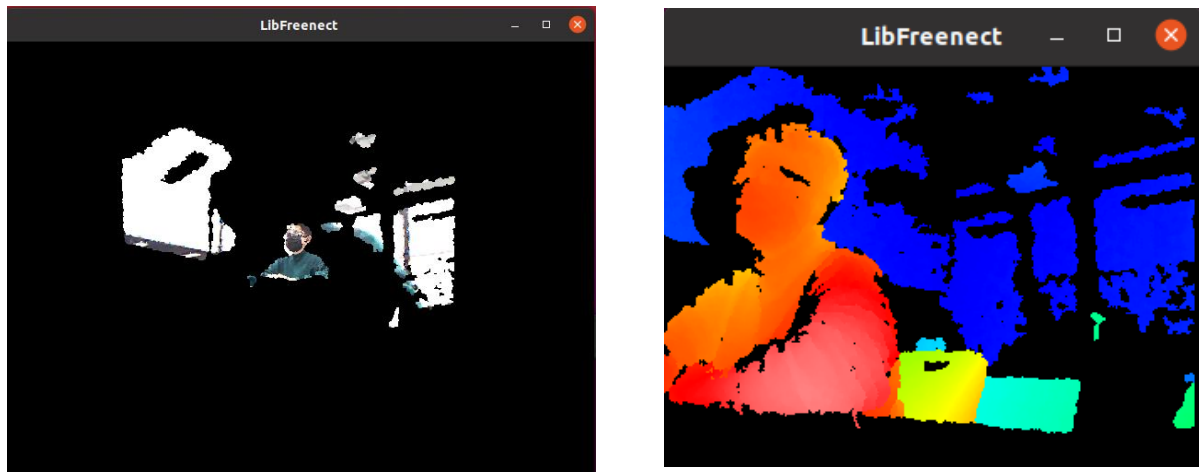
```
freenect-glview
```

La instrucción debe de abrir una pantalla con el video normal y otra con el video obtenido del sensor de profundidad, como se ve en la Figura 3. Si indica que es necesario realizar la instalación de una librería extra, realizar la instalación y luego volver a probar el ejemplo. En el folder que se creó con el nombre de “libfreenect” se encuentra un folder con el nombre de “examples”. Aquí se encuentran programas en C que se pueden correr para realizar las pruebas, incluyendo glview. Estos ejemplos se pueden correr con el mismo formato, colocando freenect- antes del nombre del ejemplo que se quiera probar.

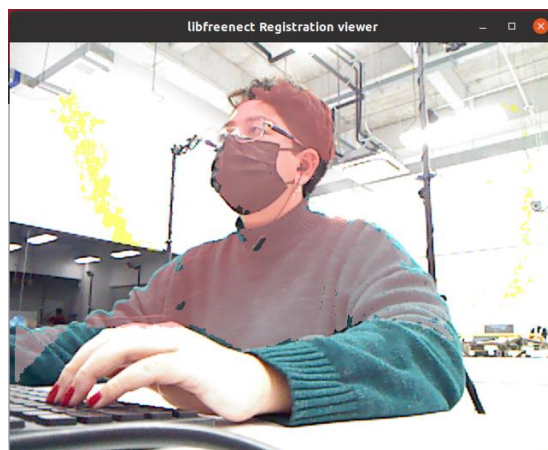


**Figura 3:** Resultado de correr el ejemplo freenect-glvie.

Otros ejemplos se pueden correr con el mismo formato, como glpclview (Figura 4 izquierda), chunkview (Figura 4 derecha) y regview (Figura 5).



**Figura 4:** Resultado de correr el ejemplo freenect-glpclview (Izquierda) y freenect-chunkview (Derecha).



**Figura 5:** Resultado de correr el ejemplo freenect-regview.

Por errores de compatibilidad con librerías de audio y del manejo de motores, ciertos ejemplos como `freenect-tiltdemo` o `freenect-wavrecord` no logran obtener resultados. En caso ninguno de los ejemplos esté funcionando se puede tratar de agregar unos permisos. Para esto, es necesario ingresar la siguiente instrucción en la terminal:

```
sudo nano /etc/udev/rules.d/51-kinect.rules
```

Luego, se copia y se pega el siguiente texto:

```
# ATTR{product}=="Xbox NUI Motor"
SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02b0", MODE="0666"
# ATTR{product}=="Xbox NUI Audio"
SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02ad", MODE="0666"
# ATTR{product}=="Xbox NUI Camera"
SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02ae", MODE="0666"
# ATTR{product}=="Xbox NUI Motor"
SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02c2", MODE="0666"
# ATTR{product}=="Xbox NUI Motor"
SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02be", MODE="0666"
# ATTR{product}=="Xbox NUI Motor"
SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02bf", MODE="0666"
```

Se guarda con `Ctrl+s` y se sale del programa con `Ctrl+x`.

Además, se puede tratar de descargar el *driver* para el audio directamente. Para esto es necesario correr la instrucción

```
sudo apt install kinect-audio-setup
```

Esto enseñará un mensaje de error al finalizar por lo que es necesario modificar un valor en un archivo. Se ingresa la instrucción:

```
sudo nano /usr/sbin/kinect_fetch_fw
```

El valor de `SDK_MD5` que se muestra al principio se reemplaza con el valor `945806927702b2c47c32125ab9a80344`.

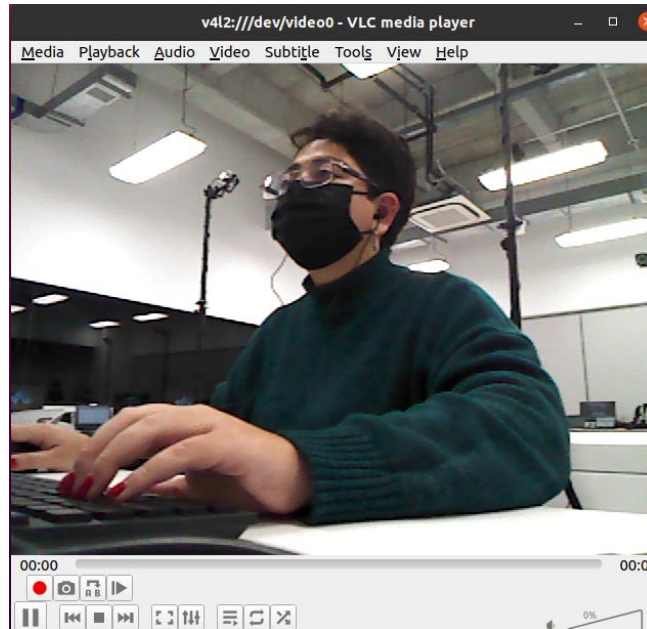
Al correr de nuevo la instrucción:

```
sudo apt install kinect-audio-setup
```

Se debería de poder realizar la instalación correctamente. De igual manera se puede probar corriendo los ejemplos de la librería, aunque no se pudo tener acceso completo a todos los ejemplos, como `tiltdemo` o `micview`.

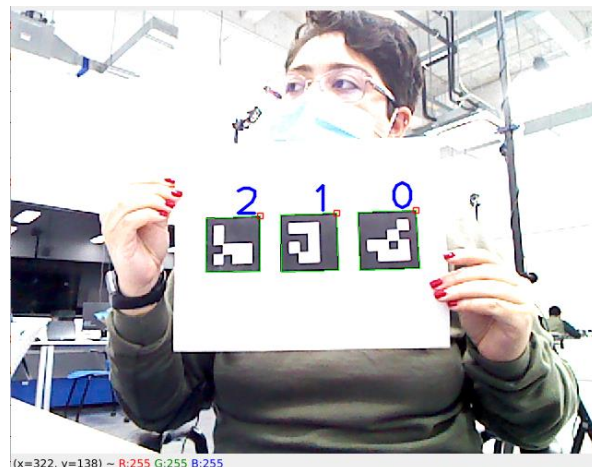
Ya con los *drivers* instalados, es posible utilizar el Kinect como cámara. Para esto es necesario correr la instrucción a continuación y llamarla como `/dev/video0` en aplicaciones como VLC, como se puede ver en la Figura 6.

```
sudo modprobe gspca_kinect
```



**Figura 6:** Ejemplo de Kinect como WebCam en VLC.

Además, con esto ya se puede llamar a la cámara por medio de OpenCV y realizar todo tipo de visión por computadora.



**Figura 7:** Kinect corriendo programa de detección de marcadores ArUco utilizando OpenCV.