

Body Position and Orientation Recognition Library KiNAO

Javier Acosta
Daniel Méndez
Willy Villalobos

Universidad de Costa Rica

2 de junio de 2014



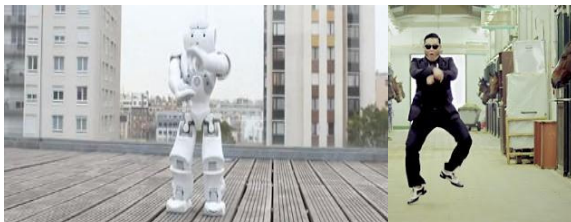
¿Qué es KiNAO?

Es una librería que detecta el cuerpo humano, brindando datos sobre posición y orientación. La idea se concibe como un paso para lograr controlar el movimiento de un robot NAO de forma que este replique las posiciones de una persona, utilizando un kinect como dispositivo de detección del movimiento.



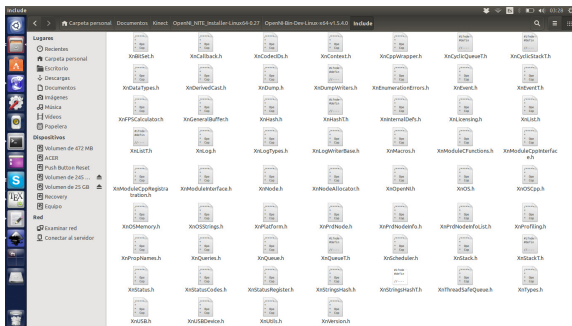
¿Qué es KiNAO?

A partir de la librería OpenNI se construyen los métodos adicionales necesarios para detectar el cuerpo y los movimientos, y posteriormente trasladar esa información al NAO para que la imite, o a cualquier autómatas compatible, realizando la debida adaptación si fuese necesario.



Funcionamiento de la librería

Al momento de instalarse, OpenNI contiene una serie de archivos y librerías necesarios para la compilación y funcionamiento de los archivos, particularmente de los ejemplos de aplicación (samples) que incluye por defecto.



Funcionamiento de la librería

Para la librería desarrollada se tienen los siguientes archivos:

- 1 XnTypes.h
- 2 definition.h
- 3 XnCppWrapper.h
- 4 cpp_global.h
- 5 kinao_global.h
- 6 joint.h
- 7 joint.cpp
- 8 larm.cpp (left arm), head.cpp...

Este archivo define los tipos de datos necesarios para manipular toda la información que el kinect percibe, y que luego se procesa y despliega.

Se definen nuevos tipos de datos necesarios para que los metodos de la nueva librería funcionen adecuadamente.

Esta librería contiene los métodos para calibración y reconocimiento del cuerpo, los samples de OpenNi se valen de esta librería, junto con otros archivos, para detectar a los usuarios y generar el "skeleton".

Este archivo se incluye únicamente en un eventual main para no tener que incluir por separado cada uno de los archivos de las partes del cuerpo que se van a reconocer con el kinect.

Este archivo se incluye en el joint.h ya que es la declaración de todas las clases y funciones de la librería. Al agregar nuevas librerías, se deben incluir aquí por cuestiones de orden a la hora de programar.

Se realizan las definiciones de métodos y se heredan las clases necesarias para el funcionamiento de la librería.

Doxygen Manual: Do... Kinao: Joint Class Refer... Kinao: Head Class Refer... Kinao: Joint Class Refer...

File:///home/daniel/UCR/EyAteam/ProyectoCpp/Desarrollo/Kinao/Doxy/Wizard/html/class_joint.html

Public Member Functions

joint()	Esta es la declaración de la funcionalidad de la clase base "joint", de aquí se derivan las demás.
virtual ~joint(void)	Destructor del Objeto.
virtual float length()	length (float[3], float[3]) Destructor del Objeto. More...
virtual float angle()	angle (float, float, float)
virtual float getAngle()	getAngle (XnVector3D, XnVector3D, XnVector3D) L1,L2,L3,L3 es el opuesto del ángulo a determinar) More...
virtual XnVector3D getNormalVector()	getNormalVector (XnVector3D, XnVector3D, XnVector3D) Genera el vector resultante de realizar el producto cruz entre los 2 vectores coplanarios. More...
virtual XnVector3D getProjectionVector()	getProjectionVector (XnVector3D, XnVector3D) Obtiene el vector resultante de proyectar un vector sobre otro. More...
virtual XnFloat getProjection()	getProjection (XnVector3D Vector1, XnVector3D Vector2) Obtiene la constante de la proyección. More...
virtual XnReferenceAxis generateReference()	generateReference (XnVector3D, XnVector3D, XnVector3D) Construye un nuevo eje de coordenadas a partir de 3 vectores ortogonales. More...

Detailed Description

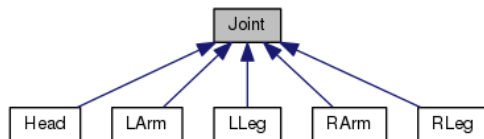
Aqui se describe la clase base "joint".

Member Function Documentation

Se declara la funcionalidad de la clase base joint.h y de los distintos métodos de la librería. Cada archivo de reconocimiento de extremidades, torso y cabeza emplea los métodos definidos en este archivo.



Estos archivos aprovechan los métodos definidos en joint.h y joint.cpp para empezar a generar un esqueleto virtual de la persona detectada, e incluye métodos para obtener y procesar datos de posición y orientación.



Tipos de datos importantes

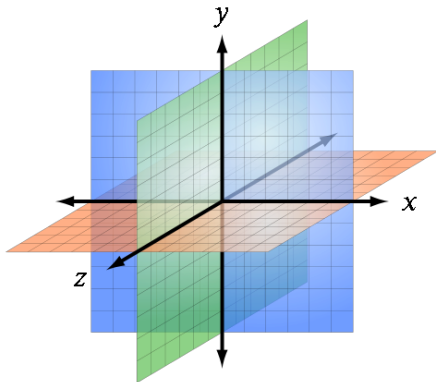
Para la implementación de la librería se recurre a la definición de nuevos tipos de datos y de la reutilización de otros tipos de datos que por defecto define las librerías base del OpenNI. En este caso tenemos:

- 1 XnFloat
- 2 XnVector3D
- 3 XnReferenceAxis

Es un tipo de dato que almacena un número con punto flotante, similar a float, para trabajar de forma aislada los datos e incluirlos más cómodamente en otras estructuras más complejas.

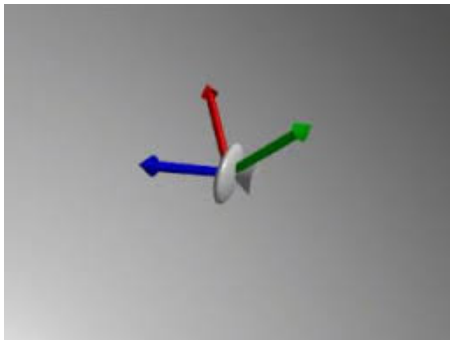
XnVector3D

Es un tipo de dato especial que genera un vector cuyas componentes son de tipo XnFloat definidas en el sistema de coordenadas rectangulares (X, Y, Z)



XnReferenceAxis

Este tipo de datos fue definido para efectos del proyecto, lo que hace es almacenar 3 vectores `XnVector3D` ortogonales entre si, formando un nuevo marco de referencia.



Tiempo para una demostración



Gracias por su atención