

Uso del Telepresencia con Irobot

Inteligencia Artificial

Carlos Cárdenas

Brando Dávila

Eleazar Aranciaga

Juan Ramos

Junior Lara

1 Requisitos:

1.1 Hardware

- OLPC o notebook o laptop con webcam incluida
- Bluetooth
- Irobot Create

1.2 Software

- Distribución Linux Ubuntu (para otro hardware diferente a la OLPC).
- Distribución de Linux por defecto de la OLPC.
- Escritorio Sugar.
- Librería PyCreate.
- Actividad MotionCapture-3.xo.

2 Programa:

2.1 Instalación del software base

Se Instalarán los siguientes paquetes en python:

Paquete	Versión	Detalle
python	2.5.2-0ubuntu1	An interactive high-level object-oriented la
python-at-spi	0.6.1-1ubuntu2	Assistive Technology Service Provider Interf
python-dbg	2.5.2-0ubuntu1	Debug Build of the Python Interpreter (versi
python-dbus	0.82.4-1ubuntu1	simple interprocess messaging system (Python
python-glade2	2.12.1-0ubuntu1	GTK+ bindings: Glade support
python-gst0.10	0.10.11-1	generic media-playing framework (Python bind
python-gtk2	2.12.1-0ubuntu1	Python bindings for the GTK+ widget set
python-hippocanvas	0.2.23-0ubuntu2	Python bindings to hippo-canvas
python-imaging	1.1.6-1ubuntu5	Python Imaging Library
python-opencv	1.0.0-4ubuntu1	Python bindings for the computer vision libr
python-opengl	3.0.0~b1-0ubuntu4	Python bindings to OpenGL
python-pyatspi	1.22.1-0ubuntu1	Assistive Technology Service Provider Interf
python-pyaudio	0.2.3	Python bindings for PortAudio v19
python-pygame	1.7.1release-4.1ubuntu1	SDL bindings for games development in Python
python-pysqlite2	2.4.0-2build1	python interface to SQLite 3
python-speechd	0.6.6-0ubuntu8	Python interface to Speech Dispatcher

python-telepathy	0.14.0-1ubuntu1	python language bindings for telepathy
python-xml	0.8.4-10ubuntu2	XML tools for Python
python2.5	2.5.2-2ubuntu4.1	An interactive high-level object-oriented la

Por medio del comando de:

```
sudo apt-get install python python-at-spi python-dbus python-gdata python-glade2 python-gst0.10
python-gtk2 python-hippocanvas python-libxml2 python-opencv python-opengl python-pyatspi
python-pyaudio python-pygame python-pysqlite2 python-speechd python-telepathy python-xml
python2.5
```

Se Instalarán los siguientes paquetes en speech para pasar texto a voz y tener mensajes de alerta del desplazamiento del robot:

Detalle	Versión	Paquete
espeak	1.36-0ubuntu1	A multi-lingual software speech synthesizer
espeak-data	1.36-0ubuntu1	A multi-lingual software speech synthesizer
libespeak-dev	1.36-0ubuntu1	A multi-lingual software speech synthesizer
libespeak1	1.36-0ubuntu1	A multi-lingual software speech synthesizer
libgnome-speech7	1:0.4.18-0ubuntu2	GNOME text-to-speech library
libgsm1	1.0.12-1	Shared libraries for GSM speech compressor
libgsm1-dev	1.0.12-1	Development libraries for a GSM speech compr
libspeechd2	0.6.6-0ubuntu8	Speech Dispatcher: Shared libraries
python-speechd	0.6.6-0ubuntu8	Python interface to Speech Dispatcher
speech-dispatcher	0.6.6-0ubuntu8	Common interface to speech synthesizers
speech-tools	1:1.2.96~beta-2	Edinburgh Speech Tools - user binaries

Por medio del comando de:

```
sudo apt-get install espeak espeak-data libespeak-dev libespeak1 libgnome-speech7 libgsm1
libgsm1-dev libspeechd2 python-speechd speech-dispatcher speech-tools
```

Instalación de Sugar en Ubuntu:

Detalle	Versión	Paquete
sugar	0.79.0-0ubuntu3	the Sugar framework used on the XO laptop
sugar-activities	0.79.0-0ubuntu3	metapackage for most Sugar activities
sugar-artwork	0.40+git20080131-0ubuntu1	the artwork for the OLPC project's XO laptop
sugar-base	0.3-1	Sugar graphical interface for the OLPC lapto
sugar-calculate-activity	18-0ubuntu1	calculator activity for the XO laptop
sugar-chat-activity	35-0ubuntu1	chat activity on the XO laptop
sugar-connect-activity	21-0ubuntu1	connect the dots game on the XO laptop
sugar-datastore	0.8.0-3ubuntu1	datastore for Sugar
sugar-journal-activity	86-0ubuntu1	journal activity on the XO laptop
sugar-logviewer-activity	6-0ubuntu1	log viewing activity on the XO laptop
sugar-memorize-activity	26-0ubuntu1	memory game activity on the XO laptop

sugar-pippy-activity	20-0ubuntu2	Python learning activity on the XO laptop
sugar-presence-service	0.79.0-0ubuntu2	presence service for Sugar
sugar-terminal-activity	8-0ubuntu1	terminal activity on the XO laptop
sugar-toolkit	0.79.0-0ubuntu1	the Sugar toolkit library used on the XO lap
sugar-turtleart-activity	7-0ubuntu1	Graphical programming activity on the XO lap
sugar-web-activity	86-0ubuntu1	web activity on the XO laptop

```
sudo apt-get install sugar sugar-activities sugar-artwork sugar-base sugar-calculate-activity sugar-
chat-activity sugar-connect-activity sugar-datastore sugar-journal-activity sugar-logviewer-activity
sugar-memorize-activity sugar-pippy-activity sugar-presence-service sugar-terminal-activity sugar-
toolkit sugar-turtleart-activity sugar-web-activity
```

2.2 Configuración del Bluetooth en Linux

1. Descargamos algunos modulos comprimido. (logre luego de 3 veces hacer una compilación del kernel en el ubuntu pero esto me demoró 3 hr de prueba y error, no lo recomiendo si no quieren aburrirse).

wget http://cvs.cs.brynmawr.edu/cgi-bin/viewcvs.cgi/*checkout*/Myro/packages/olpc-bluetooth.tgz?rev=HEAD

2. Descomprimos:

```
tar -xvzf olpc-bluetooth.tgz
```

3. Instalamos los módulos:

```
cd olpc-bluetooth
/sbin/insmod bluetooth.ko
/sbin/insmod hci_usb.ko
/sbin/insmod l2cap.ko
/sbin/insmod rfcomm.ko
```

4. Instalamos estos paquetes:

```
yum install bluez-utils bluez-libs pyserial
Para sistemas ubuntu:
sudo apt-get install bluez-utils bluez-libs pyserial
```

5. Con esto se tendrá los bluetooth instalados, es hora de activar el servidor:

```
/etc/init.d/bluetooth start
```

6. Luego scaneamos:

```
# hcitool scan
Scanning ...
00:0A:3A:22:6F:D0 Serial
```

```
# hcitool dev
Scanning ...
11:11:11:11:11:11 usb
```

7. Como dije en un post anterior. No es posible conectar directamente el irobot como con los bluetooth del telefono o pda. Así que se hará de forma manual. El código de conexion es el 000.

```
vi /etc/bluetooth/rfcomm.conf
```

8. Colocar en /etc/bluetooth/rfcomm.conf

```
rfcomm1{  
  bind yes;  
  device 00:0A:3A:22:6F:D0 ;  
  channel 1;  
}
```

9. Editamos el /etc/bluetooth/hcid.conf y colocamos:

```
passkey-agent "0000"
```

10. Otro inicio del servicio para tener el /dev/rfcomm1:

```
/etc/init.d/bluetooth start
```

2.3 Instalación del MotionCapture

1. Se ingresará a la interface se Sugar.
2. Se abrirá la terminal.
3. Se descargará la actividad de MotionCapture

```
wget http://www.dunmire.org/olpc/MotionCapture-3.xo
```

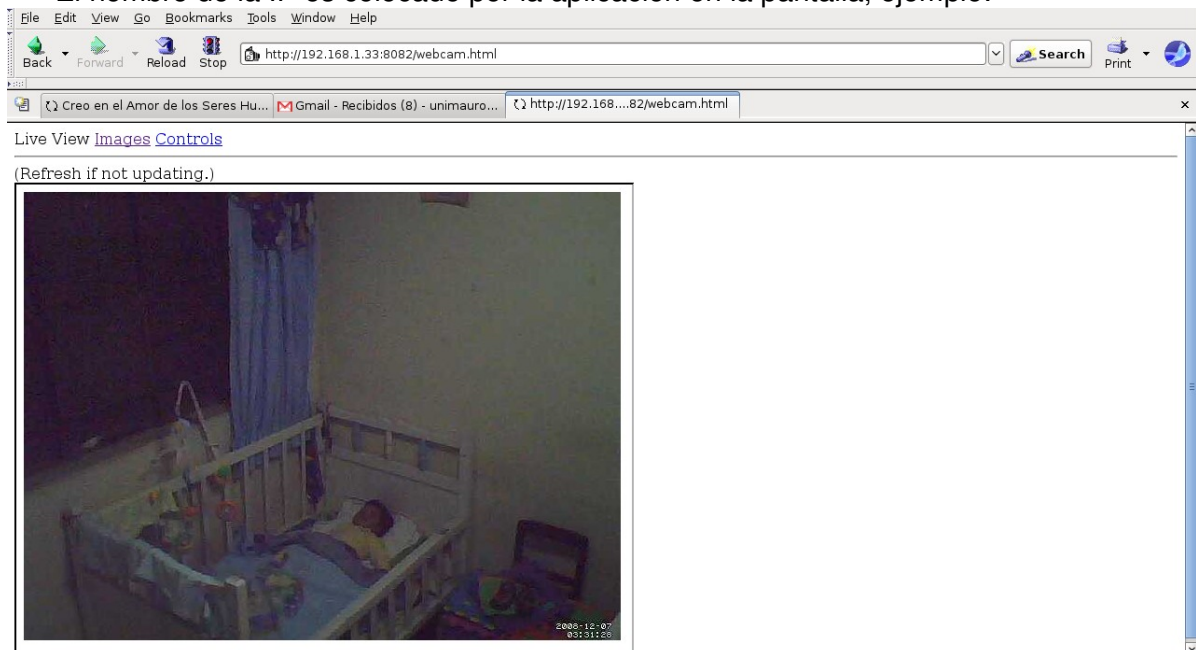
4. Se instalará la aplicación MotionCapture:

```
sugar-install-bundle MotionCapture-3.xo
```

5. Probar la instalación del MotionCapture. Para ver el resultado cada vez que se activa la actividad se podrá ingresar por medio de la web cam con la siguiente URL:

<http://localhost:8082> u <http://ipdelalaptop:8082>

El nombre de la IP es colocado por la aplicación en la pantalla, ejemplo:



2.4 Uso del PyCreate

El PyCreate es una librería usada en la universidad e HarveyMudd para conectar robot por

medio de terminales serial.

Se puede usar por defecto para dispositivos tipo el legorobotYN

Se descarga desde aquí:

<http://www.cs.hmc.edu/~dodds/erdos/pyCreate.zip>

Para el caso del Irobot se tiene los siguientes sensores, donde cero es el parametro de inicio:

```
self.casterDrop = 0
self.leftWheelDrop = 0
self.rightWheelDrop = 0
self.leftBump = golpe izquierda
self.rightBump = golpe derecha
self.wallSensor = buleano
self.leftCliff = ciclo de recorrido
self.frontLeftCliff = ciclo hacia la izquierda
self.frontRightCliff = ciclo hacia la derecha
self.rightCliff = ciclo hacia adelante
self.virtualWall = Recorrido virtual
self.driveLeft = direccion izquierda
self.driveRight = direccion derecha
self.mainBrush = 0
self.vacuum = 0
self.sideBrush = 0
self.leftDirt = 0
self.rightDirt = 0
self.remoteControlCommand = control remoto
self.powerButton = control de apagado
self.spotButton = control de alto
self.cleanButton = control de limpieza de informacion de entrada
self.maxButton = 0
self.distance = distancia
self.rawAngle = angulo de giro
self.angleInRadians = angulo en radianes
self.chargingState = cambio de estado
self.voltage = voltaje
self.current = actividad reciente
self.temperature = temperatura
self.charge = carga
self.capacity = capacidad.
```

Para usar el robot desde python se puede hacer lo siguiente teniendo la libreria Create.py en la ubicación de la terminal:

```
>>>import create ==> Con esot importamos la libreria Create.py
>>>r = create.Create('/dev/rfcomm1') ==> Conectamos al robot
>>> r.start() ==> Iniciamos los comandos
>>>r.demo(4) ==> Ejecutamos un demo el número 4
>>>r.demo() ==> Paramos el demo.
>>> r.stop() ==> Cerramos las conecciones
```

2.5 Uso del Programa

1. El programa a ejecutar se llama main.py.
2. Se ejecuta así:

python main.py

3. Se puede ejecutar conectando al robot usando bluetooth o wireless por medio de ssh o utilizando un cron desde otra olpc, laptop o pc de escritorio:

En nuestro caso usamos la olpc para enviar la señal de la webcam por wireles y una laptop para enviar la señal al robot por medio de bluetooth.

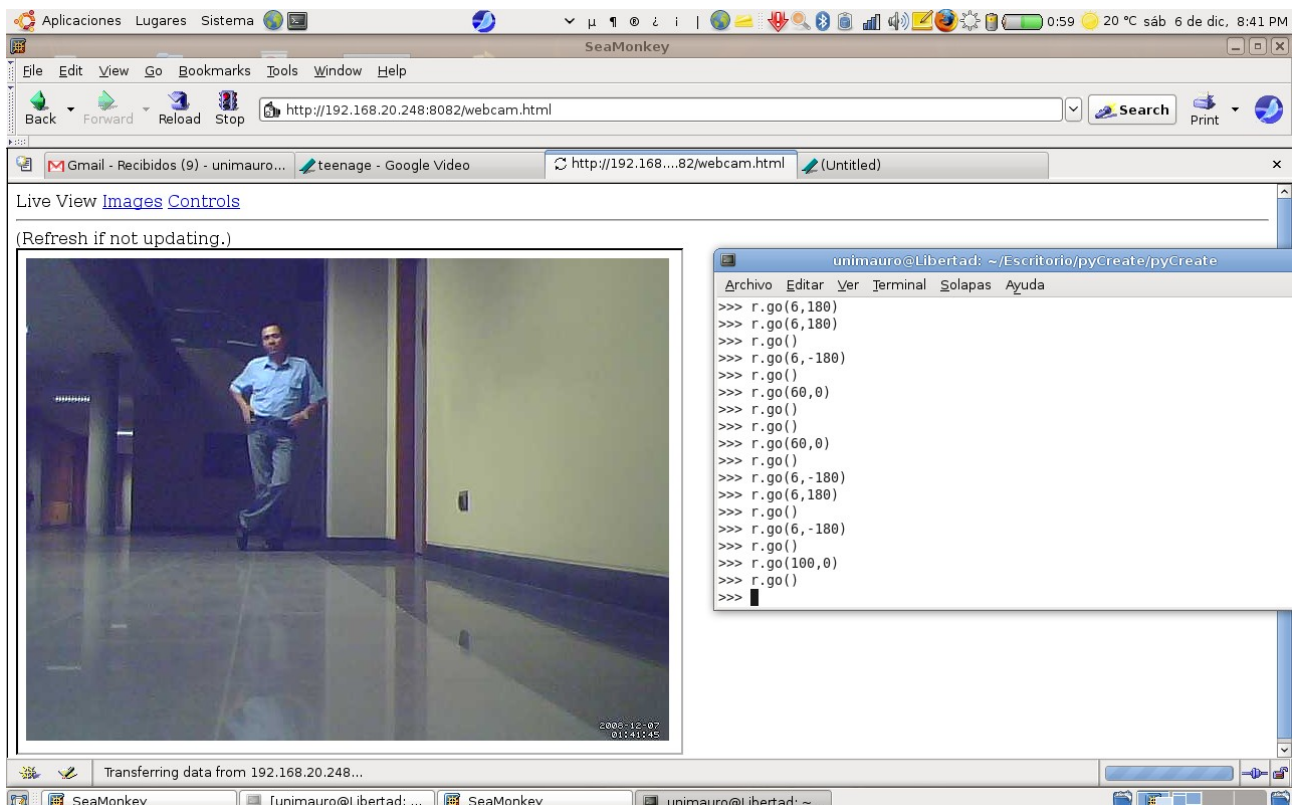
4. El código fuente es:

```
import os
import create
import time
import sys
import subprocess

def synth(txt):
    subprocess.call(["espeak", "-w", "test.wav", "-p", "60", "-s", "140", "-v", "es", txt], stdout=subprocess.PIPE)
    subprocess.call(["aplay", "test.wav"], stdout=subprocess.PIPE)

if __name__ == "__main__":
    # r=create.Create('/dev/rfcomm1')
    synth('Robot Activo')
    # r.demo(4)
    synth('Demo de Sonido')
    time.sleep(10)
    # r.demo()
    synth('Cerrando Demo de Sonido')
    # r.close()
    synth('Cerrando Conexion al Robot')
```

5. Resultados



Un robot que explora con la seguridad de los que lo conducen.