

Control del Pan & Tilt desde C

Juan Andrade-Cetto
IRI-UPC-CSIC

Agosto 10, 2000

Introducción

El robot móvil "Marco" posee una cabeza de visión estereoscópica de Directed Perception, Inc. La cabeza tiene dos grados de libertad:

1. PAN θ_1 : giro en un eje vertical con respecto a la base del robot, (como si la cabeza dijera "no"), y
2. TILT θ_2 : giro en un eje paralelo al *baseline* de las cámaras, (como si la cabeza dijera "si").

El controlador de el Pan & Tilt está instalado dentro del cuerpo del robot, y está conectado físicamente al puerto serie 2 (COM2) del ordenador interno del robot.

La resolución de giro de ambos grados de libertad es de 185.1428 segundos de arco por paso de motor. Así por ejemplo, para girar la cabeza 10 grados debiera moverse alguno de los ejes

$$\frac{10\text{grados} * 3600 \frac{\text{segundos de arco}}{\text{grado}}}{185.1428 \frac{\text{segundos de arco}}{\text{paso}}} = 194\text{pasos}$$

Las transformaciones homogéneas que relacionan la base del Pan & Tilt con cada uno de sus grados de libertad son:

Del TILT al PAN:

$$H_1^2 = \begin{bmatrix} \cos(\theta_2) & 0 & \sin(\theta_2) & 0 \\ \sin(\theta_2) & 0 & -\cos(\theta_2) & 0 \\ 0 & 1 & 0 & -76.2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Del PAN a la BASE:

$$H_0^1 = \begin{bmatrix} \cos(\theta_1) & 0 & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & 0 & \cos(\theta_1) & 0 \\ 0 & -1 & 0 & 107.315 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Control local

Para controlarlo localmente (desde el ordenador instalado en Marco) basta con establecer comunicación con el puerto serie del robot y enviar comandos de control como los descritos mas adelante.

En el *desktop* de Windows del ordenador del robot (si te conectas usando el VNC lo podrás ver), hay un ícono de hyperterminal llamado *Pan & Tilt*. Esta sesión de hyperterminal ya está configurada con las características de velocidad de conexión , paridad, bits de stop, puerto, etc. propias del Pan & Tilt.

Una vez iniciada esa sesión de hyperterminal podrás teclear alguno de los siguientes comandos (y más, descritos en el manual de Directed Perception [1]). Nota: los comandos se ejecutan al oprimir la barra de espacio. No es necesario oprimir ENTER.

r Reset la unidad. Este comando moverá ambos ejes hasta sus límites físicos para inicializar los encoders.

pp Preguntar posición actual del PAN

tp Preguntar posición actual del TILT

ps Preguntar velocidad actual del PAN

ts Preguntar velocidad actual del TILT

pp100 Mover el PAN a la posición absoluta 100

tp100 Mover el TILT a la posición absoluta 100

ps200 Cambiar la velocidad del PAN a 200 pasos por segundo

ts200 Cambiar la velocidad del TILT a 200 pasos por segundo

h Cancelar movimiento de la unidad en sus dos ejes

hp Cancelar movimiento del PAN

ht Cancelar movimiento del TILT

Control desde C (local y remoto)

Para enviar comandos de movimiento de manera remota, Delfin Pereiro ha escrito un *servidor* usando sockets que permite enviar instrucciones a través de ethernet como si se enviaran a un puerto local, pero que en realidad se ejecutan en un ordenador remoto. De esta manera, programar el Pan & Tilt es exactamente igual para una conexión local como para una conexión remota. Solo hace falta definir la dirección IP a donde se encuentra el robot antes de establecer la conexión.

A continuación se muestran dos ejemplos de como ejecutar una serie de movimientos desde C en el Pan & Tilt

Conexión local

```
#include "stdio.h"
#include "ptu.h"

short dato;

...

// inicio de conexión
COMStream = open_host_port("COM2");
if(COMStream==(void *)0xc0000000)
    exit(0);
else
    set_mode(SPEED_CONTROL_MODE,PTU_INDEPENDENT_SPEED_CONTROL_MODE);

// resetea el PAN&TILT
// No es necesario si los encoders no han perdido información
reset_ptu():

// mueve el TILT a la posición 200 y el PAN a la posición 300
dato=200;
set_desired(TILT,POSITION,&dato,ABSOLUTE);
dato=300;
set_desired(PAN,POSITION,&dato,ABSOLUTE);

// cierra la conexión con el PAN&TILT
// el servidor se quedará esperando una nueva conexión
close_host_port(COMStream);
```

Conexión remota

```
#include "winsock2.h"
#include "stdio.h"
#include "ptu.h"
```

```

char IPAddress[13];
short dato;

...

// envía al socket inicio de conexión
sprintf(IPAddress, "147.84.76.240");
SetIP(IPAddress);
COMStream = open_host_port("COM2");
if(COMStream==(void *)0xcccccccc)
    exit(0);
else
    set_mode(SPEED_CONTROL_MODE,PTU_INDEPENDENT_SPEED_CONTROL_MODE);

// resetea el PAN&TILT
// No es necesario si los encoders no han perdido información
reset_ptu():

// mueve el TILT a la posición 200 y el PAN a la posición 300
dato=200;
set_desired(TILT,POSITION,&dato,ABSOLUTE);
dato=300;
set_desired(PAN,POSITION,&dato,ABSOLUTE);

// cierra la conexión con el PAN&TILT
// el servidor se quedará esperando una nueva conexión
close_host_port(COMStream);

```

La diferencia principal que permite esta portabilidad consiste en incluir para un proyecto de comunicación local los ficheros *ptu.cpp*, *ptu.h*, *w32serial.cpp*, *w32serial.h*, y *opcodes.h*.

En el caso de un proyecto de comunicación remota, requiere los ficheros *ptu.cpp*, *ptu.h*, *w32socket.cpp*, *w32socket.h*, y *opcodes.h*, además de que se deba encadenar el proyecto con la biblioteca *ws2_32.lib*.

Todos los ficheros necesarios están en *ptulocal.zip* y *pturemote.zip* que pueden bajar de <http://www-iri.upc.es/people/cetto/marco.html>.

References

- [1] Computer controlled pan-tilt unit model ptu user's manual, 1999. Directed Perception, Inc. Version 1.11.