

UNIVERSIDAD: Universidad Nacional de La Plata / Facultad de Informática / Instituto de Investigación en Informática LIDI.

TÍTULO DEL TRABAJO: SISTEMA DISTRIBUIDO DE SEGUIMIENTO EN TIEMPO REAL APLICADO A ENTRENAMIENTO DEPORTIVO.

AUTOR(ES): Federico Cristina y Sebastián H. Dapoto.

E-MAIL DE LOS AUTORES: {fcristina, sdapoto}@lidi.info.unlp.edu.ar

PALABRAS CLAVES: Tiempo real, entrenamiento deportivo

ÁREA TEMÁTICA: Redes Académicas

INTRODUCCIÓN

El objetivo del ejercicio planteado es completar un recorrido con pelota dominada en el menor tiempo posible. El mismo se encuentra delineado por marcas, las cuales deben ser transitadas en forma de zig-zag sorteando obstáculos, hasta llegar a la meta final.

El jugador deberá realizar el recorrido accionando cada una de las marcas con la mano correspondiente. A su vez, existen semáforos a lo largo del mismo que indicarán al jugador si debe realizar una pausa antes de continuar con el ejercicio.

El recorrido tiene una longitud de 1495 metros y podrá ser realizado por hasta 30 jugadores simultáneamente. El mismo consta de tres grandes bloques los cuales, a su vez, contienen una serie de sectores parciales, cada uno identificado con un color que denota un nivel de dificultad. Los sectores parciales del mismo color son comparables entre sí y son utilizados para cálculos estadísticos de la performance de los jugadores a lo largo del recorrido.

Con el fin de poder desarrollar una solución acorde a los requerimientos planteados, se debieron contemplar los siguientes aspectos:

- Hardware a utilizar: determinar los equipos que mejor se adapten a las necesidades de relevamiento en tiempo real.
- Cómo realizar el relevamiento de la información: qué metodología de adquisición de datos utilizar y cómo informar los resultados obtenidos.
- Problemas de accesibilidad concurrente a los recursos (hardware y software): contemplar la posibilidad de que dos o más jugadores activen marcas o semáforos de manera simultánea.

DESARROLLO

Hardware utilizado

Una vez contempladas todas las posibilidades, se determinó utilizar una combinación de dispositivos, tal como se detalla a continuación:

- Tarjetas magnéticas: los jugadores usan un par de guantes que contienen una tarjeta magnética cada uno. Estas tarjetas se usan para identificar inequívocamente cada mano de cada jugador.

- Lectoras de tarjetas magnéticas: se dispone de una serie de lectoras a lo largo del recorrido por las cuales los jugadores deberán pasar sus guantes. Cada una de éstas representa un tipo de marca del recorrido.

- Sensores infrarrojos: se dispone de una serie de sensores infrarrojos a lo largo del recorrido por los cuales los jugadores deberán pasar. Cada una de éstos representa otro tipo de marca del recorrido. Los mismos son también utilizados para activar semáforos.

- WebPort Tunnel (WT): Son utilizados junto con las lectoras de tarjetas magnéticas para relevar la ubicación de los jugadores. [1]

- WebPort Remote (WR): es un dispositivo de captura que cuenta con 16 entradas digitales. Son utilizados para recibir las señales de los sensores infrarrojos. [2]

- Semáforos: los mismos serán encendidos/apagados mediante información enviada a un circuito lógico a través del puerto paralelo de la computadora central. [3][4]

La figura 1 presenta la dimensión del proyecto junto con la distribución del hardware de captura. La figura 2 detalla la configuración desarrollada sobre un tramo del total del recorrido.

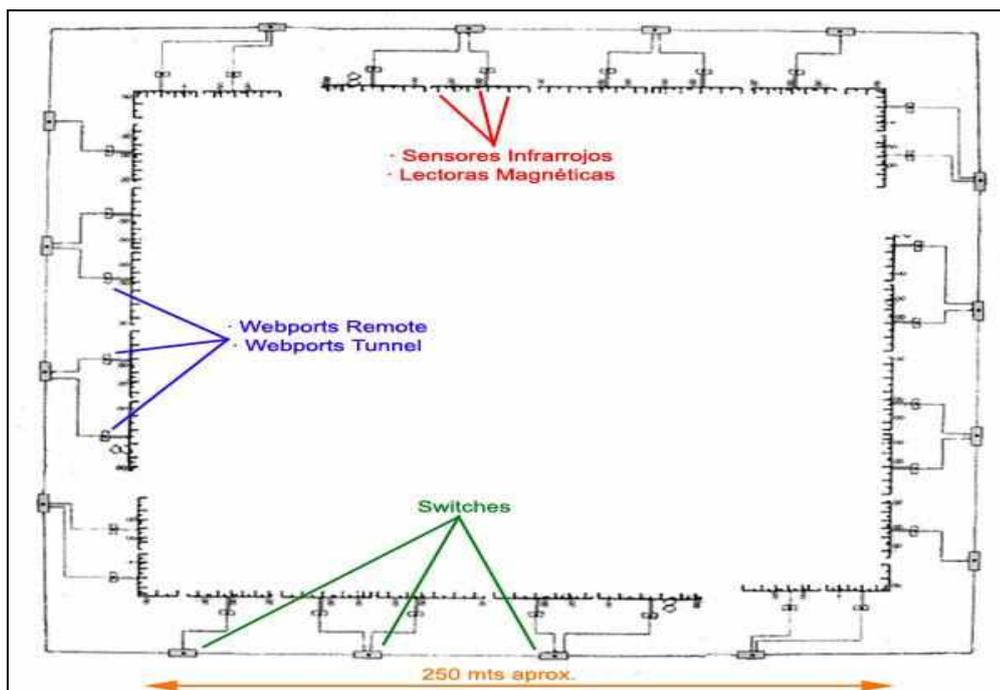


Figura 1 – Distribución del hardware de captura

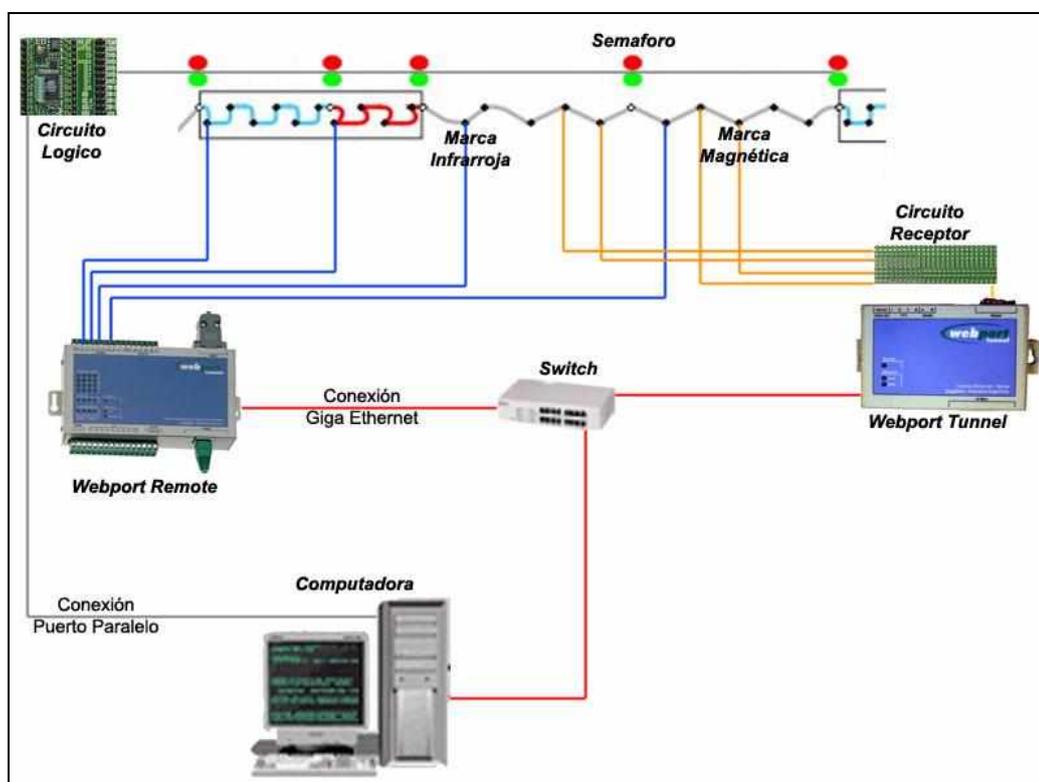


Figura 2 – Configuración de hardware (un tramo del recorrido)

Metodología de adquisición de datos e interacción

Como se expuso anteriormente, el relevamiento de datos se realiza tanto con dispositivos WR como con dispositivos WT. Se determinó utilizar una solución híbrida dada la necesidad de garantizar que ciertas marcas (infrarrojas) sean activadas indefectiblemente, las cuales indican el final de un parcial o activan un semáforo.

En la adquisición mediante WR se realiza un polling cada cierto intervalo de tiempo sobre dichos dispositivos, de forma de detectar cualquier señal proveniente de los sensores infrarrojos.

En la adquisición mediante WT, cada vez que un jugador pasa alguno de sus guantes (cada uno de éstos posee una tarjeta magnética con un identificador único) sobre una lectora, se genera un evento que indica el número de lectora y tarjeta leídos, de forma tal de llevar el registro de la posición de cada jugador en el recorrido.

Para realizar el encendido/apagado de un semáforo, se envía la información correspondiente a través del puerto paralelo a un circuito lógico encargado de realizar la acción correspondiente.

Estas actividades, que en principio no suponen un problema de difícil solución, comienzan a incrementar su complejidad al momento de contemplar la captura concurrente de 30 jugadores [5]. Se detallarán a continuación las principales complejidades a resolver.

Consultar el estado de todos los wr sin perder marcaciones: tal como se comentó anteriormente, los WR son consultados continuamente en intervalos de tiempo regulares con el fin de conocer el estado de los mismos. El intervalo de dicha consulta debe cumplir dos requisitos fundamentales. Primeramente, el intervalo deberá ser lo suficientemente pequeño como para supervisar todos los WR garantizando que entre una consulta y otra no se hayan perdido marcaciones por parte de los jugadores. Por otro lado, el intervalo deberá ser lo suficientemente grande como para permitir tiempos de CPU al resto de la aplicación (manejo de eventos de WT, chequeos de estados, información numérica en pantalla, renderización gráfica, etc). [6][7]

Conocer en todo momento la ubicación actual del jugador: en los WT existe una asociación entre los números de tarjeta y los jugadores. Es decir, para un recorrido cada jugador tiene asignado un par de tarjetas con identificador único (una por mano). Esta asociación deja de existir al momento de utilizar los WR, ya que los mismos solo reflejan el paso de un jugador a través del sensor infrarrojo, sin contar con la capacidad de conocer a qué

jugador corresponde. Para determinar esta información, el sistema desarrollado se basa en el registro que mantiene actualizada la posición de los jugadores en el recorrido [8]. El algoritmo recorre el mismo siguiendo una lógica que permite identificar cuál es el jugador que ha atravesado el sensor infrarrojo, contemplando incluso el caso en el que éste haya saltado una o más marcas anteriores al sensor.

Contemplar el caso en el que un jugador abandone el ejercicio: puede ocurrir el caso en que un jugador abandone el ejercicio, o bien el mismo es sobrepasado por otro jugador (lo cual también se interpreta como un abandono). Para el primer caso, se resolvió que si un jugador no activa una marca en un período de tiempo mayor a uno especificado, el mismo es eliminado del ejercicio. Para esto, se mantiene un registro que almacena el instante de tiempo en el que cada jugador accionó la última marca. La verificación correspondiente se realiza una vez por segundo. El segundo caso es mas trivial: se inhabilitan todos los jugadores cuya marca actual es alcanzada por otro jugador. El algoritmo se basa en el registro que mantiene actualizada la posición de los jugadores en el recorrido.

Contemplar el caso en el que varios jugadores activan semáforos simultáneamente: la interacción hardware-software para el encendido/apagado de los semáforos llevó a la necesidad de realizar un circuito lógico que interprete la información proveniente del puerto paralelo y realice la acción correspondiente sobre alguno de los semáforos dispuestos a lo largo del recorrido. Dado que existen 37 semáforos; éstos se dividieron en 5 grupos de 8 semáforos cada uno, tal como se muestra a continuación:

Semáforos	Datos LPT	Control LPT
1 al 8	00000001 al 10000000	1000
9 al 16	00000001 al 10000000	0100
17 al 24	00000001 al 10000000	0010
25 al 32	00000001 al 10000000	0001
33 al 40	00000001 al 10000000	1100

El canal de datos del puerto paralelo indica qué semáforo dentro del grupo debe ser encendido/apagado, mientras que el canal de control indica el grupo y es además el disparador de la lectura del puerto paralelo.

La metodología es la siguiente: primeramente se direcciona la información correspondiente al canal de datos y luego de los milisegundos necesarios para estabilización (*hold time*) se direcciona la información al canal de control.

La complejidad inmediata que surge es el caso en que dos jugadores activen sendos semáforos de manera simultánea; lo que podría llevar a que se sobrescriba información en el puerto paralelo, perdiéndose así el encendido o apagado de alguno de los semáforos. Existe entonces la necesidad de secuencializar el uso del puerto paralelo, garantizando la no pérdida de peticiones al mismo.

La solución desarrollada consiste en implementar una cola (vector circular), en la que se almacenan las peticiones de uso del puerto paralelo (qué información direccionar en canal de datos y de control). La estructura de datos a almacenar en la cola es la siguiente:

```
struct lptStruct {
    int data;        // informacion al bus de datos
    int ctrl;       // informacion al bus de control
    bool accion;    // TRUE=AND(apagar sem.), FALSE=OR(encender sem.)
    int estado;    // Instante actual enviar 1 (Datos) o 0 (Control)
};
```

Al momento de encolar una petición, se indican los valores de *data* y *ctrl*, junto con la *accion* a realizar (encender o apagar el semáforo). El *estado* inicialmente se setea en 1, lo que indica que se deberá direccionar la información de *data*.

Cada 10ms se lleva a cabo el procesamiento de la cola (si la misma posee un tamaño mayor a cero), la cual se realiza en dos etapas, según se indica en *estado*:

- La primera etapa consiste en acceder a una petición de la cola, enviando la información especificada en *data* al bus de datos y modificando el *estado*, seteando en valor 0, lo que indica que se deberá direccionar la información de *ctrl*.

- La segunda etapa consiste en acceder a la misma petición de la cola, enviando la información especificada en *ctrl* al bus de control y eliminando de la cola dicha petición.

Podría llegar a darse el caso en el que todos los jugadores activen simultáneamente sendos semáforos, en el cual la cola tendría un tamaño de 30. Dado que procesar cada petición demora 20ms en total, el tiempo máximo de error sería de 0,6 segundos; lo cual no presenta un problema para los requerimientos del ejercicio. De ser necesario, se podría almacenar el timestamp de cada petición y calcular así el tiempo de compensación a favor del jugador.

Otros chequeos a realizar: otros chequeos debieron contemplarse para garantizar la correcta funcionalidad del sistema, como por ejemplo: ignorar marcaciones repetidas, las cuales pueden darse si la marcación de un jugador sobre una marca conlleva un lapso de tiempo mayor a dos encuestas a los WR; contemplar el caso en el que un jugador active una marca para la cual aún no se encuentre habilitado; por ejemplo, si el mismo se encuentra demorado por un semáforo en rojo.

RESULTADOS

Funcionalidades que provee el sistema:

- Brindar un total control de configuración de manera visual sobre todos los aspectos del ejercicio (posición de los semáforos, marcas, sensores infrarrojos, zonas de bloques, etc.).

- Registrar en tiempo real el momento en el que cada marca es accionada, tiempos parciales y el tiempo total requerido para completar el recorrido; así como registrar si el jugador falló en el accionar de alguna de las marcas.

- Activar/desactivar de manera automática los semáforos de acuerdo a las características del jugador y los tiempo logrados en los parciales.

- Visualizar en tiempo real la información tanto gráfica como numérica que cada jugador va generando a lo largo del recorrido.

Para cada área del sistema se desarrollaron módulos independientes de configuración de hardware, configuración de marcas y de captura.

Módulo de configuración de hardware

Tal como lo muestra la figura 3, este módulo permite:

- Configurar las direcciones IP de cada Webport, su tipo (Remote o Tunnel).
- Asociar cada marca del recorrido con un contacto (WR) o una lectora (WT)

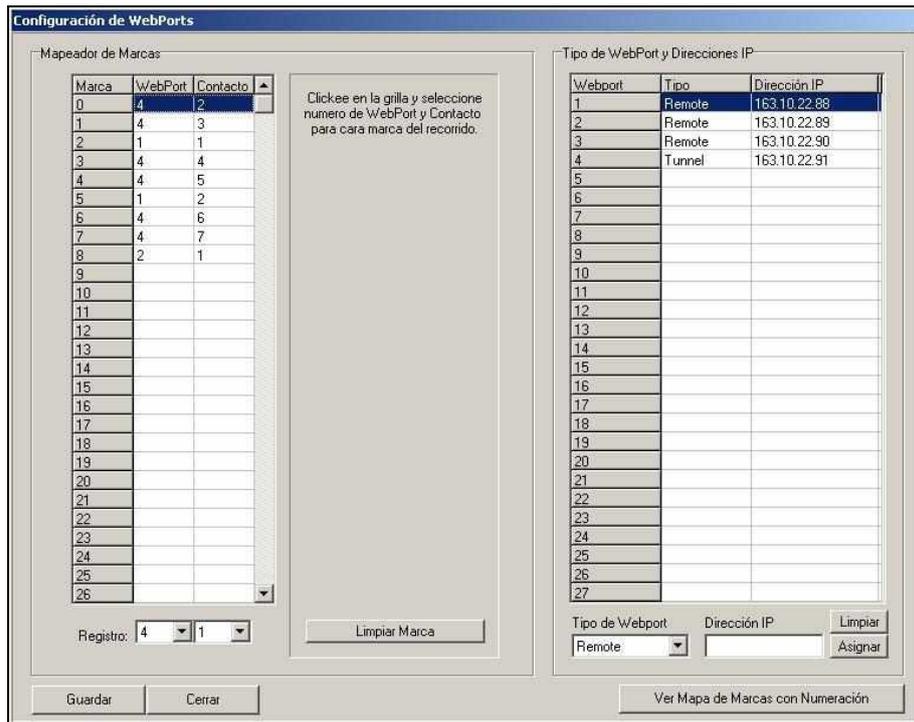


Figura 3 – Módulo de configuración de hardware

Módulo de configuración de marcas

La figura 4 muestra el módulo de configuración marcas, el cual permite especificar el recorrido de entrenamiento, indicando posición de los semáforos, distancias entre las marcas, indicación de finalización de parciales y bloques, etc.

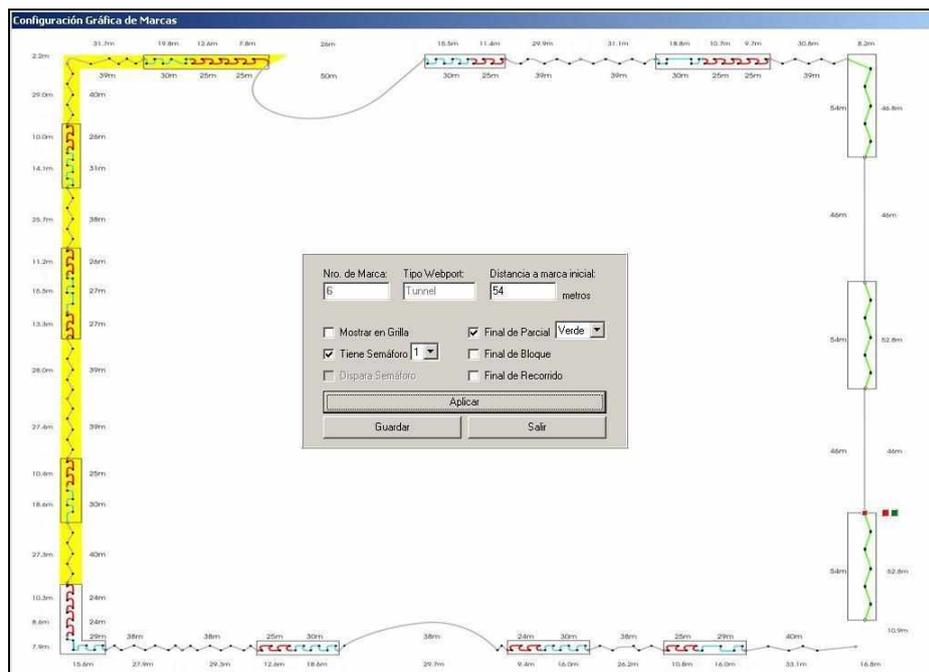


Figura 4 – Módulo de configuración de marcas

Módulo de captura y visualización de recorridos

Las figuras 5 y 6 muestran el entorno de captura y visualización en tiempo real de recorridos. El mismo brinda la información gráfica y numerica de la posición y tiempos de los jugadores. Se muestra además el estado de los semáforos, las marcas perdidas, y otros datos de interés. [9][10][11]

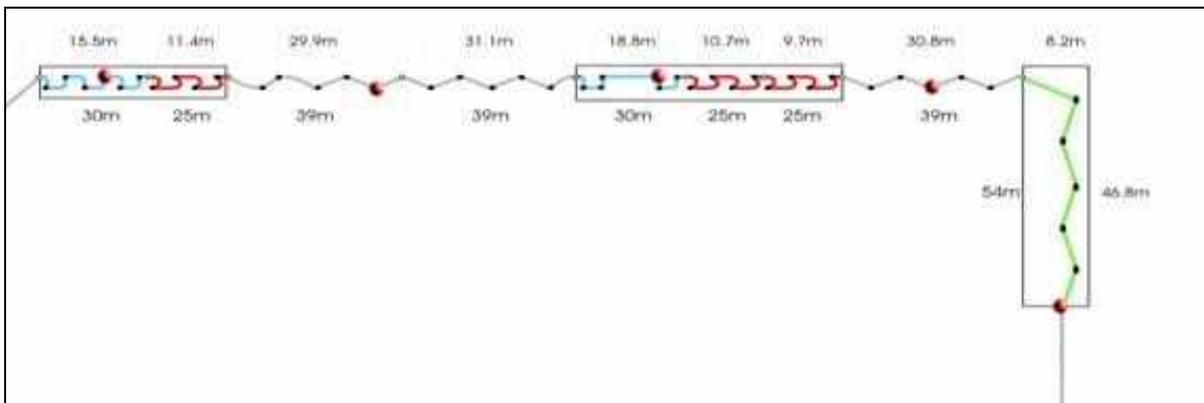


Figura 5 – Visualización gráfica de la posición de los jugadores sobre un tramo del recorrido

Registrar Jugadores		Iniciar Captura									Tiempo		Marcas		
Cerrar		Detener Captura									Total	Act.	Per.	Verd.	X
	p.1	p.2	p.3	p.4	p.5	p.6	p.7								
j.1														Verd.	X
j.2														Verd.	X
j.3														Verd.	X
j.4														Verd.	X
j.5														Verd.	X
j.6														Verd.	X
j.7														Verd.	X
j.8														Verd.	X
j.9														Verd.	X
j.10														Verd.	X
j.11														Verd.	X
j.12														Verd.	X
j.13														Verd.	X
j.14														Verd.	X
j.15														Verd.	X
j.16														Verd.	X
j.17														Verd.	X
j.18														Verd.	X
j.19														Verd.	X
j.20														Verd.	X
j.21														Verd.	X
j.22														Verd.	X
j.23														Verd.	X
j.24														Verd.	X
j.25														Verd.	X
j.26														Verd.	X
j.27														Verd.	X
j.28														Verd.	X
j.29														Verd.	X
j.30														Verd.	X

Figura 6 – Información numérica de posiciones y tiempos de los jugadores

CONCLUSIÓN

Se ha desarrollado un sistema de seguimiento en tiempo real que permite mejorar la técnica deportiva de los jugadores.

El desarrollo del mismo dejó en claro las complicaciones que acarrea la captura concurrente de información. Cada una de estas problemáticas han sido resueltas tal como se explicó en apartados anteriores.

Dadas las características versátiles de la implementación, la misma es totalmente adaptable a otras configuraciones y disciplinas deportivas.

REFERENCIAS

- [1] Especificación Webport Tunnel. Interface Ethernet/Serial.
<http://www.sage.com.ar/Links/tunnel.htm>
- [2] Especificación Webport Remote. Monitoreo, registro y conversión de datos analógicos y digitales. Interface ethernet / serial. <http://www.sage.com.ar/Links/remote.htm>
- [3] Parallel port Interfacing Tutorial. <http://www.logix4u.net/paralleport1.htm>
- [4] Control de potencia por PC.
<http://www.pablin.com.ar/electron/circuito/computer/lpt32/index.htm>
- [5] ANDREWS, Gregory R.; SALLY, Elliott. Concurrent Programming: Principles and Practice, Ed. Addison-Wesley ISBN: 0805300864 (1991).
- [6] STALLINGS, William. Organización y Arquitectura de Computadoras: Diseño para optimizar prestaciones. Ed. Prentice Hall (5ta edición) España, 2000. ISBN 84-205-2993-1.
- [7] TANENBAUM, Andrew. Organización de Computadoras. Ed. Prentice Hall (4ta edición), 2000. ISBN 9701703995.
- [8] WEISS, Mark. Data Structures and Algorithm Analysis in C. Ed. Addison Wesley. ISBN: 0321189957.
- [9] DE LA ESCALERA, Arturo. Visión por Computador, Fundamentos y Métodos. Prentice Hall - 2001. ISBN: 8420530980
- [10] JAIN, Ramesh; KASTURI, Rangachar; SCHUNCK, Brian G. Machine Vision. Mac Graw-Hill Internacional Edition - 1995. ISBN: 0071134077
- [11] OpenCV - Open Source Computer Vision Library Community.
<http://groups.yahoo.com/group/OpenCV>.