

ESCUELA POLITÉCNICA DEL EJÉRCITO

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA ELECTRÓNICA,
AUTOMATIZACIÓN Y CONTROL

PROYECTO DE GRADO PARA LA OBTENCIÓN DEL
TÍTULO EN INGENIERÍA

“TELEOPERACIÓN DE UNA PLATAFORMA MÓVIL ASISTIDA
MEDIANTE EL USO DE UN SISTEMA DE VISIÓN ARTIFICIAL”

Autor:

Richard Alfonso Andrade Alfaro

SANGOLQUÍ - ECUADOR

2012

CERTIFICACIÓN

Certificamos que el presente proyecto de grado, “TELEOPERACIÓN DE UNA PLATAFORMA MÓVIL ASISTIDA MEDIANTE EL USO DE UN SISTEMA DE VISIÓN ARTIFICIAL”, fue desarrollado en su totalidad por el señor Richard Alfonso Andrade Alfaro bajo nuestra dirección.

Atentamente,

Ing. Alejandro Chacón

DIRECTOR

Ing. Jaime Andrango

CODIRECTOR

AGRADECIMIENTO

Agradezco a mis padres por el apoyo brindado a lo largo de mi vida, quienes me han acompañado en mis victorias y en mis derrotas, que siempre me han dado ánimos para continuar y cumplir con mis sueños y anhelos. Ellos que con su esfuerzo diario y sus enseñanzas me han permitido crecer como persona y como profesional.

Les doy un especial agradecimiento a mi abuelito Julio Alfonso y a mi tía Mariana porque gracias a ellos tuve la oportunidad de estudiar la carrera que me apasiona y tanto quiero, la ingeniería Electrónica Automatización y Control, y que sin su ayuda no podría estar donde me encuentro ahora.

Por ultimo pero no menos importante le doy un especial agradecimiento a Nathaly quien me ha llevado hacer cosas que jamás imagine que podría hacer y que me ayudo a descubrir de lo que era capaz de hacer por mí mismo. Gracias por estar a mi lado mientras desarrollaba este proyecto y por ayudarme en el mismo.

Richard Alfonso Andrade Alfaro

DEDICATORIA

Le dedico a mi padre por su esfuerzo, por enseñarme que con dedicación se puede llegar a conseguir lo que uno se propone, a mi madre por el amor incondicional que me brinda y los ánimos de seguir adelante pese a las dificultades que se presenten, a mis abuelitos por demostrarme que el trabajo duro siempre da sus frutos.

Richard Alfonso Andrade Alfaro

PROLOGO

Desde tiempos antiguos, el hombre ha venido utilizando distintas herramientas para poder acrecentar el alcance de su capacidad de manipulación. En un principio, no se trataba más que de palos utilizados para hacer caer la fruta madura de un árbol. Actualmente existen dispositivos más complejos, como por ejemplo pértigas las cuales accionando una pinza a distancia, permiten alcanzar objetos ubicados en estanterías o lugares de almacenamiento de difícil acceso.

Estos desarrollos desembocaron finalmente en lo que se conoce como sistemas de teleoperación maestro-esclavo, en donde un manipulador denominado esclavo reproduce los movimientos de un dispositivo o manipulador maestro, controlado a su vez manualmente por un operador humano.

Teleoperación se define como un conjunto de tecnologías para permitir la operación o gobierno a distancia de un dispositivo por un ser humano. En teleoperación un parámetro importante es la Telepresencia, la cual se consigue realimentando coherentemente al ser humano suficiente cantidad de información sobre el entorno remoto, para manejar este parámetro se hace uso de interfaces.

Dentro de las interfaces existen las de control supervisado, en las cuales se da toda la información necesaria para permitir al operador sentir el máximo de telepresencia en el sitio remoto, hace utilización de dispositivos hápticos con realimentación de fuerza y torque, utiliza dispositivos de realidad virtual, como guantes, lentes, trackers, con el objetivo de provocar el máximo grado de telepresencia.

El uso de dispositivos Head Mounted Display (HMD) o cascos de realidad virtual, permite la realimentación visual, El HMD usa dos pequeñas pantallas (LCD o CRT) que se colocan delante de los ojos a modo de gafas. Un sistema óptico entre el ojo y las pantallas permite crear una imagen virtual más grande y más alejada del ojo, lo que redundará en una visión mucho más cómoda.

El presente trabajo pretende combinar un sistema teleoperado basado en la plataforma móvil Pioneer 3-AT, un HMD equipado con un tracker y un joystick, una cámara de visión y analizar el grado de telepresencia e inmersión sensorial alcanzada.

ÍNDICE DE CONTENIDOS

1. SISTEMAS TELEOPERADOS	9
1.1. CONCEPTOS BÁSICOS	9
1.2. ORÍGENES DE LOS SISTEMAS TELEOPERADOS	10
1.3. DESCRIPCIÓN GENERAL DE UN SISTEMA TELEOPERADO.....	13
1.3.1. ELEMENTOS DE UN SISTEMA TELEOPERADO	13
1.4. TELEOPERACIÓN DIRECTA	14
1.5. MODOS DE CONTROL POR TELEOPERACIÓN DIRECTA O MANUAL	15
1.5.1. CONTROL EN POSICIÓN.....	15
1.5.2. CONTROL EN VELOCIDAD.....	15
1.5.3. CONTROL DE ESFUERZOS.....	15
1.6. TELEPRESENCIA	15
1.6.1. FACTORES QUE INTERVIENEN EN LA TELEPRESENCIA.....	16
1.7. GRADOS DE INMERSIÓN SENSORIAL.....	18
1.8. REALIDAD AUMENTADA	19
1.9. SISTEMA TELEOPERADO ROMOTEVI.....	20
1.9.1. DESCRIPCIÓN DEL SISTEMA	20
1.9.2. DISEÑO DEL SISTEMA TELEOPERADO ROMOTEVI.....	21
1.9.3. VERSIÓN 1.1 DEL SISTEMA ROMOTEVI	22
1.10. VERSIÓN 1.2 DEL SISTEMA ROMOTEVI.....	23
2. SISTEMAS DE CASCOS VIRTUALES (HMD).....	24
2.1. INTRODUCCIÓN	24
2.2. CASCO DE REALIDAD VIRTUAL	25
2.3. SENTIDO VISUAL	25
2.3.1. CAMPO VISUAL	26
2.4. PERCEPCIÓN VISUAL	28
2.4.1. PROFUNDIDAD	29
2.5. CREACIÓN DE IMÁGENES EN UN CASCO DE REALIDAD VIRTUAL	31
2.5.1. PANTALLAS TRANSMISIVAS	31
2.5.2. PANTALLAS REFLECTIVAS.....	31
2.5.3. PANTALLAS AUTO-EMISIVAS.....	32
2.5.4. TIPOS DE PANTALLAS USADAS EN CASCOS VIRTUALES.....	33
2.5.5. RESOLUCIÓN GRÁFICA DE IMAGEN	33
2.6. SISTEMA DE TRACKING	36

2.6.1.	GRADOS DE MOVIMIENTO DE UN CASCO DE REALIDAD VIRTUAL	37
2.6.2.	MONITORES	39
2.7.	SONIDO ESTÉREO	42
2.8.	ANÁLISIS COMPARATIVO.....	43
3.	IMPLEMENTACIÓN DEL HARDWARE DEL SISTEMA DE TELEOPERACIÓN	45
3.1.	HARDWARE DEL SISTEMA ROMOTEVI.....	45
3.2.	SERVIDOR DEL SISTEMA ROMOTEVI.....	47
3.2.1.	HARDWARE DEL ROBOT MÓVIL PIONER P3-AT	48
3.2.2.	CÁMARA DE VIDEO VC-C50I	49
3.2.3.	COMPUTADORA DE ABORDO	50
3.2.4.	DISPOSICIÓN Y ANÁLISIS DEL HARDWARE PARA LA VERSIÓN 1.1	51
3.2.5.	DISPOSICIÓN Y ANÁLISIS DEL HARDWARE PARA LA VERSIÓN 1.2.....	54
3.2.6.	CONSUMO ENERGÉTICOS DEL ROBOT MÓVIL	56
3.3.	CLIENTE DEL SISTEMA ROMOTEVI	57
3.3.1.	DESCRIPCIÓN DEL CASCO DE REALIDAD VIRTUAL.....	57
3.3.2.	COMPUTADOR CLIENTE DEL SISTEMA ROMOTEVI	58
3.3.3.	DESCRIPCIÓN DEL SISTEMA DE COMUNICACIÓN INALÁMBRICO.	59
3.4.	PRUEBAS DE FUNCIONAMIENTO DEL HMD	59
3.5.	PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA DE COMUNICACIÓN INALÁMBRICA. ...	61
4.	IMPLEMENTACIÓN DEL SOFTWARE DEL SISTEMA	64
4.1.	DESCRIPCIÓN DEL SOFTWARE DEL SISTEMA.....	64
4.1.1.	LIBRERÍAS UTILIZADAS EN EL SOFTWARE DEL SISTEMA	65
4.2.	DESCRIPCIÓN DEL SDK DEL CASCO DE REALIDAD VIRTUAL	65
4.2.1.	CREACIÓN DE LA DLL REA	67
4.3.	DESCRIPCIÓN DE LA LIBRERÍA ARIA.....	68
4.4.	LIBRERÍA ARNETWORK.....	69
4.5.	LIBRERÍA OPENCV.....	70
4.6.	LIBRERÍA QT	70
4.7.	SOFTWARE DE CONTROL DEL SERVIDOR ROMOTEVI.....	71
4.7.1.	ESTRUCTURA DEL SERVIDOR DE CONTROL.....	71
4.7.2.	FUNCIONAMIENTO DEL SERVIDOR DE CONTROL.....	75
4.8.	SOFTWARE CLIENTE DEL SISTEMA ROMOTEVI.....	78
4.8.1.	ESTRUCTURA DEL CLIENTE ROMOTEVI.....	78
4.8.2.	FUNCIONAMIENTO DEL CLIENTE ROMOTEVI.....	80
4.9.	SOFTWARE DEL SERVIDOR DE VIDEO ROMOTEVI.....	82
4.9.1.	ESTRUCTURA DEL SERVIDOR DE VIDEO.....	82

4.9.2.	FUNCIONAMIENTO DEL SERVIDOR DE VIDEO	83
4.10.	SOFTWARE DEL CLIENTE DE VIDEO	84
4.10.1.	ESTRUCTURA DEL CLIENTE DE VIDEO	85
4.10.2.	FUNCIONAMIENTO DEL CLIENTE DE VIDEO	86
4.11.	INTERFAZ GRAFICA DEL CLIENTE ROMOTEVI.....	87
5.	PRUEBAS Y RESULTADOS.....	89
5.1.	DESCRIPCIÓN DE LA PRUEBA EXPERIMENTAL.....	89
5.2.	METODOLOGÍA DE EVALUACIÓN PARA EL SISTEMA ROMOTEVI.....	91
5.2.1.	CUESTIONARIO DE TENDENCIA INMERSIVA.....	91
5.2.2.	CUESTIONARIO DE EVALUACIÓN DE PRESENCIA.....	92
5.3.	RESULTADOS OBTENIDOS EN EL CUESTIONARIO DE PRESENCIA PARA EL SISTEMA ROMOTEVI.....	95
5.3.1.	RESULTADOS DE LOS FACTORES PRINCIPALES.....	95
5.3.2.	RESULTADOS DE LOS FACTORES SECUNDARIOS.....	97
6.	CONCLUSIONES Y RECOMENDACIONES.....	99
6.1.	CONCLUSIONES	99
6.2.	RECOMENDACIONES	100
ANEXOS	¡ERROR! MARCADOR NO DEFINIDO.	
ANEXO 1	¡ERROR! MARCADOR NO DEFINIDO.	
ANEXO 2	¡ERROR! MARCADOR NO DEFINIDO.	
ANEXO 3	¡ERROR! MARCADOR NO DEFINIDO.	
ANEXO 4	¡ERROR! MARCADOR NO DEFINIDO.	
ANEXO 5	¡ERROR! MARCADOR NO DEFINIDO.	
ANEXO 6	¡ERROR! MARCADOR NO DEFINIDO.	
REFERENCIAS BIBLIOGRAFICAS.....	101	
ÍNDICE DE FIGURAS	102	
ÍNDICE DE TABLAS	105	
GLOSARIO	106	

CAPÍTULO 1

SISTEMAS TELEOPERADOS

1.1. CONCEPTOS BÁSICOS

Se entiende por teleoperación la extensión de las capacidades sensoriales y destrezas humanas a una localización remota^{[5][6]}. Para explicar mejor lo que es teleoperación nacen dos conceptos que la complementan que son teleactuación y telesonorización, entendiéndose como teleactuación los aspectos específicos de generación de órdenes a los actuadores y entendiéndose a telesonorización como la captación y visualización de la información sensorial^[6].

De la teleoperación nace un concepto más evolucionado en el cual se toma en cuenta la autonomía del sistema remoto, este concepto es telerobótica entendiéndose como la teleoperación convencional con un aumento en las características autónomas tanto en la capacidad de decisión como actuación del sistema remoto.

Se podría decir que en la telerobótica el nivel de comunicación entre el operador y el robot es mucho más elevado que en la teleoperación, y es por esto que según Lumina y otros autores en 1994 los métodos de control telerobótico se pueden clasificar en tres tipos:

- Control manual.
- Control supervisor.
- Control completamente automático

Generalmente el término teleoperación describe a todos los métodos de control telerobótico pero también se lo asocia al control manual, donde es el método más básico en el cual todos los movimientos del robot son controlados por un flujo continuo de información proveniente del operador humano.

El control supervisor es donde los movimientos del robot pueden ser provocados por un operador humano o por una computadora y para el control completamente automático todos los movimientos del robot son provocados por una computadora^[5].

Otro concepto importante en la teleoperación es la inmersión del operador en un lugar remoto en el que este se encuentre y para esto existe lo que se denomina como telepresencia que se refiere a la situación o circunstancia que se da cuando un ser humano tiene la sensación de encontrarse presente en un lugar remoto^[6].

Otro concepto que está estrechamente relacionado con la telepresencia pero orientado más hacia un mundo virtual que puede o no reflejar un lugar real es la realidad virtual, entendiéndose como la situación o circunstancia que se da cuando un ser humano tiene la sensación de encontrarse en un lugar distinto de donde físicamente se encuentra en ese momento gracias a una serie de información generada exclusivamente por una computadora.

Como se puede ver estos dos conceptos están estrechamente relacionados, ya que los dos están orientados a generar sensaciones que hace que un operador o usuario de un sistema crea que se encuentra en un lugar remoto o distinto, pero la diferencia radica en que las sensaciones en el caso de la realidad virtual serán generadas por algoritmos de simulación mientras que en la telepresencia será el mismo lugar el que genere dichas sensaciones.

Podríamos aseverar que por la relación tan estrecha que existe entre estos dos términos los conceptos básicos que representan a la realidad virtual se pueden aplicar a la telepresencia como puede ser los factores de simulación interactiva, interacción implícita y niveles de inmersión sensorial del sistema.

1.2. ORÍGENES DE LOS SISTEMAS TELEOPERADOS ^{[5][6]}

Los orígenes de los sistemas teleoperados están posiblemente en los mecanismos que se crearon para el manejo de sustancias radioactivas de las primeras plantas nucleares, donde el contacto directo con los elementos radioactivos representaba un peligro potencial para los trabajadores. Estos sistemas mecánicos hacían posible que un operador protegido y con visión directa diera órdenes a un manipulador que se encontraba a pocos metros.

Luego se desarrollaron primitivos sistemas mecánicos maestro-esclavo en donde el operador controlaba un manipulador equivalente en el cual ejecutaba las tareas y eran transmitidas mecánicamente al manipulador remoto. Como ejemplo de esto tenemos las

investigaciones que lideró Raymond Goertz en 1947, en el laboratorio Argonne National en los Estados Unidos, orientadas al desarrollo de este tipo de mecanismos maestro-esclavo.

Como fruto de estas investigaciones en 1948 se desarrolló el primer manipulador teleoperado mecánicamente llamado M1 que fue el antecesor de todos los sistemas maestro-esclavo existente actualmente



Figura 1-1. Fotografía correspondiente a Raymond Goertz manipulando químicos a través de un cristal de protección, hecha en 1948 en el laboratorio nacional de Argonne USA.[3]

Posteriormente, en los años 50, se desarrollaron manipuladores con actuadores eléctricos dotados de transmisión eléctrica para los movimientos y monitores para visualizar el medio a través de cámaras. Como ejemplo tenemos el primer manipulador maestro-esclavo con accionamiento eléctrico y servo-controlado llamado M2, también desarrollado por Raymond Goertz en 1954.

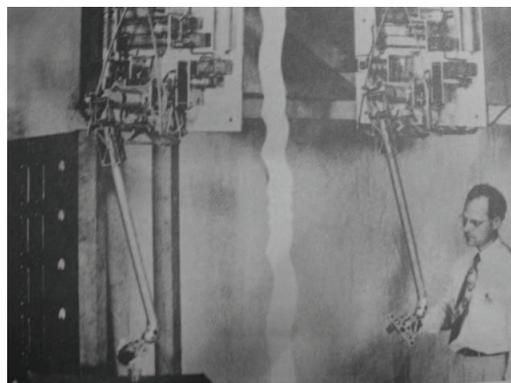


Figura 1-2. Fotografía correspondiente a Raymond Goertz haciendo tareas de aprehensión de objetos, 1954, laboratorio nacional de Argonne USA.[6]

Luego se comenzó a implementar sistemas con reflexión de esfuerzos, esto para mejorar la experiencia de teleoperación; se llegó a comprobar que estos sistemas eran muy útiles siempre que los retardos de transmisión eran despreciables, pero muy perjudiciales cuando dichos retardos eran significativos provocando oscilaciones e incluso la pérdida de la estabilidad en la teleoperación.

Ya por los años 60, el campo de aplicación de los sistemas teleoperados se amplió a sistemas teleoperados submarinos y se comenzaba a experimentar con sistemas de teleoperación espacial como por ejemplo en la misión espacial Surveyor 1, misión en la cual se teleoperaba un manipulador para recoger muestras en la luna.

Con el mejoramiento de las computadoras se crearon sistemas informáticos de teleoperación de donde surgieron conceptos tales como sistemas de control supervisado, útiles para el mejoramiento de la eficiencia de la teleoperación, sistemas para compensar retardos como el control con visualización predictiva.

Por otra parte, surgieron muchos dispositivos para mejorar la interactividad humano-maquina como son dispositivos hápticos y dispositivos visualizadores como HMDs, dichos dispositivos abrieron el camino para la reciente utilización de realidad virtual en la teleoperación.

Como se puede ver, la mayor parte de los desarrollos se orientaron a la teleoperación de Manipuladores pero paralelamente se desarrollaron sistemas teleguiados de vehículos entre los cuales se puede incluir a vehículos terrestres, submarinos y aéreos, como por ejemplo, las aplicaciones submarinas que se desarrollaron por los años 60 provocaron el desarrollo de vehículos pilotados remotamente a través de cable y mostrando en monitores lo que las cámaras montadas en los submarinos mostraban.

Posteriormente, se empezaron a utilizar vehículos terrestres y aéreos pilotados con radio control, pero el control se encontraba restringido a la visualización directa del vehículo. En los últimos 20 años, con el desarrollo de las telecomunicaciones, se pudo dotar a los vehículos de video transmitido por radio haciendo posible la teleoperación de estos vehículos sin cables.

Por ultimo cabe mencionar los últimos desarrollos de vehículos teleoperados a través de internet, redes LAN o rede Wifi las cuales se encuentran fuertemente limitadas por los

tiempos de respuesta muy altos pero que con la creación de dispositivos de red más rápidos y protocolos de comunicación en tiempo real dichos retardos son cada vez más pequeños.

1.3. DESCRIPCIÓN GENERAL DE UN SISTEMA TELEOPERADO

1.3.1. ELEMENTOS DE UN SISTEMA TELEOPERADO

En un sistema teleoperado existe un operador el cual envía las distintas órdenes de movimiento a través de una interfaz de entrada, estas órdenes son enviadas desde el sistema maestro hasta los mecanismos actuadores del sistema esclavo, el cual ejecuta dichas órdenes.

El sistema remoto está dotado de todo un sistema sensorial que mide todas las variables que son de interés, estas variables posteriormente serán transmitidas hasta el sistema maestro a la interfaz de salida llegando así hasta el operador creando de esta forma un bucle de control como se puede ver en la Figura 3, de esta forma se logra una interacción entre el operador y el entorno remoto a través de todo el sistema maestro-esclavo.

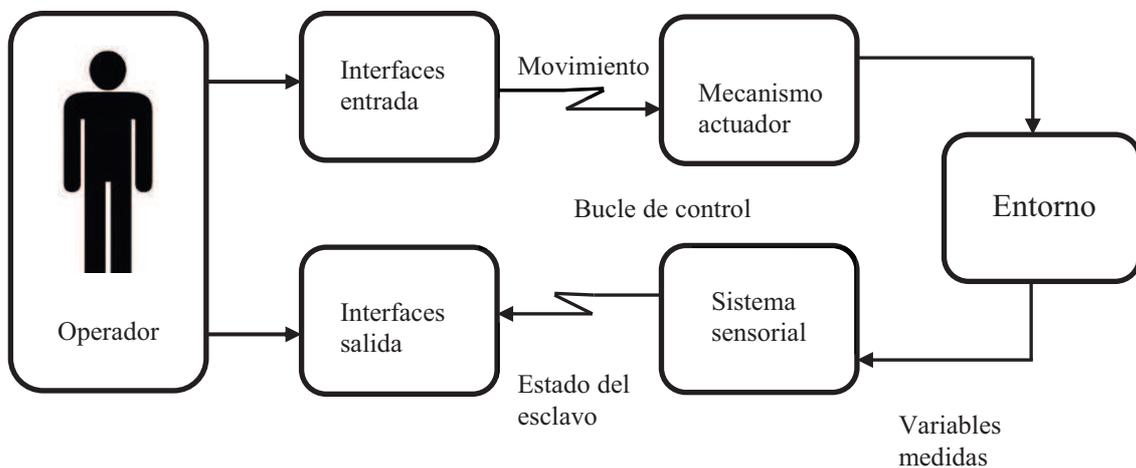


Figura 1-3. Esquema general de un sistema de teleoperación [6]

Por tanto un sistema de teleoperación está constituido por los siguientes elementos:

Operador o Teleoperador: es un ser humano que realiza el control a distancia del dispositivo. Su acción puede ir desde un control continuo hasta una intervención

intermitente, con la que únicamente se ocupa de monitorizar y de indicar objetivos y planes cada cierto tiempo.

Dispositivo teleoperado: podrá ser un manipulador, un robot, un vehículo o dispositivo similar. Es la máquina que trabaja en el lugar remota y que está siendo controlada por el operador.

Interfaz: Pueden ser de entrada o de salida y son un conjunto de dispositivos que permiten la interacción del operador con el sistema de teleoperación. Se considera a los monitores de vídeo, o cualquier otro dispositivo que permita al operador mandar información al sistema y recibir información del mismo.

Canales comunicación: conjunto de dispositivos que modulan, transmiten y adaptan el conjunto de señales que se transmiten entre la zona remota y la local.

Sensores: conjunto de dispositivos que recogen la información, tanto de la zona local como de la zona remota, para ser utilizada por el interfaz y el control.

1.4. TELEOPERACIÓN DIRECTA

Es un término que fue creado por Schneiderman en 1990, cuya idea principal es generar movimiento directamente con la utilización de vario objetos colocados en la interfaz de usuario. Dichos objetos pueden ser joysticks, botones, potenciómetros, etc.

Con esta misma idea, posteriormente se implementaron interfaces de usuario por computadora, ofreciendo ciertas ventajas sobre los dispositivos tradicionales ya que son muy flexibles y se las puede cambiar, reconfigurar, desarrollar a través de módulos de software muy fácilmente sin tener que cambiar físicamente las interfaces.

Actualmente estos dos tipos de interfaces se han conjugado para generar unas más complejas, al utilizar dispositivos visualizadores sobre la cabeza o HMDs, dispositivos apticos junto con interfaces echas en computadora realimentando video o datos importantes en la teleoperación mejorando mucho más la experiencia del usuario.

1.5. MODOS DE CONTROL POR TELEOPERACIÓN DIRECTA O MANUAL

1.5.1. CONTROL EN POSICIÓN

En este caso se envían órdenes que especifican posiciones o referencias para los sistemas de control realimentado de bajo nivel. Este modo de control es el más fácil en lo que a teleoperación directa se refiere, en el caso de robots móviles las órdenes de posición que son enviadas dependen del sistema de movimiento del robot; estas son típicamente de establecimiento de curvaturas y velocidad de avances.

1.5.2. CONTROL EN VELOCIDAD

Las órdenes están expresadas como velocidades cartesianas o articulares, dichas órdenes son ejecutadas directamente por el robot y según transcurre el tiempo la velocidad se traduce en una modificación de la posición del robot.

Como el tiempo que dura la orden de velocidad determina la trayectoria del robot o vehículo provoca que este modo sea más sensible a los retardos de comunicación, y mucho más si las ordenes son variables.

1.5.3. CONTROL DE ESFUERZOS

Este modo es más usado en el control de manipuladores robóticos. Se basa en realimentar al operador mediante reflexión de esfuerzos en un manipulador maestro activo, generando de esta manera una arquitectura de control bilateral.

Este modo de control se encuentra estrechamente relacionado con el concepto maestro-esclavo. En donde el esclavo trata de seguir al maestro de la forma más exacta posible al maestro que es controlado por el operador.

1.6. TELEPRESENCIA

Existen ciertas situaciones o trabajos en que un ser humano no debería estar, debido a la peligrosidad del entorno o del mismo trabajo en sí, para este tipo de trabajos se utilizan robots, pero no se pretende que el robot haga dicho trabajo de manera semiautónoma o

autónoma sino que se pretende que sea una extensión del operador proyectando toda la información sensorial a este y realizando todos los movimientos que este le mande, logrando una conjugación de las capacidades de un robot como fuerza, precisión y robustez con el razonamiento, inteligencia y la capacidad de tomar decisiones de un ser humano.

Las principales características son que el robot debe imitar los movimientos del cuerpo del operador como cabeza, brazos, piernas o manos y realimentar completamente al operador tratando de abarcar la mayor capacidad de sentidos como visual, auditiva y táctil de una forma y calidad comparable a la que normalmente producen los ojos, oídos y tacto.

Por lo expuesto anteriormente, se puede definir a la teleoperación como:

“Situación o circunstancia que se da cuando un ser humano tiene la sensación de encontrarse físicamente en el lugar remoto. La telepresencia se consigue realimentando coherentemente al ser humano suficiente cantidad de información sobre el entorno remoto.”

La telepresencia se centra en crear interfaces de un usuario lo más inmersivas posibles mediante la utilización interfaces visuales como HMDs, interfaces apticas como guantes e interfaces auditivas, interfaces cuyo propósito es crear lo que se conoce como inmersión sensorial.

1.6.1. FACTORES QUE INTERVIENEN EN LA TELEPRESENCIA

Por conceptualización se determinó que la realidad virtual es un concepto análogo a la telepresencia puesto que los dos crean una ilusión al operador de encontrarse en un lugar distinto al que se encuentran físicamente, la diferencia está en que la realidad virtual genera las sensaciones del lugar remoto a partir de simulaciones por computador y la telepresencia genera las sensaciones extendiéndolas desde el lugar remoto al operador por lo que los factores que intervienen en la realidad virtual son llevables a la telepresencia.

Existen tres factores que componen a la telepresencia:

- Interactividad con el medio.
- Interacción implícita.
- Inmersión sensorial.

Interactividad con el medio.

La simulación interactiva por concepto es la interactividad que tiene el usuario con el mundo virtual, sin embargo para la telepresencia el mundo en el que el robot se encontrara ya es totalmente interactivo puesto que se trata del mundo real, pero, la existencia de este factor está dada debido a que el robot debe tener las capacidades necesarias para que el operador pueda interactuar con el lugar remoto.

Para que este factor exista al robot se lo debería dotar con dispositivos como grippers o brazos robóticos que le permitan mover objetos de un lugar a otro o en general dispositivos que le permitan modificar de alguna manera el lugar remoto donde se encuentre.

Este factor no solo trata de dotar al robot con dispositivos que le permitan interactuar con el medio sino que exista una representación física o simulada de lo que está haciendo dicho dispositivo al operador como por ejemplo si se tratase de un gripper, realimentar visualmente el estado actual de este, si está abierto o cerrado, si se encuentra sosteniendo un objeto mostrar el peso que se encuentra cargando.

Interacción implícita

Un usuario utiliza interacción explícita cuando quiere transmitir una orden o deseo de una manera que no es natural para el como por ejemplo utilizando interfaces de tipo W.I.M.P (window-icon-menu-pointing device) que es como se los conoce a los sistemas basados en ventanas e iconos. Al tener este tipo de interfaces el usuario debe hacer un pequeño esfuerzo recordando el comando gráfico que necesita para hacer una determinada acción.

En cambio para que exista una interacción implícita el sistema de interfaces de entrada debería capturar la voluntad del operador implícitamente en sus movimientos naturales. El ejemplo más claro es el de una cámara de video con PAN y TILT la cual actualiza sus movimientos a través de los movimientos de la cabeza del usuario.

El objetivo primordial de la interacción implícita es que el usuario deje de percibir la interfaz de entrada y que solo interactúe con el robot directamente.

Inmersión sensorial

Dentro de lo que es telepresencia la inmersión sensorial es la desconexión de los sentidos del operador del lugar donde se encuentra físicamente y la conexión de dichos sentidos hacia el lugar remoto. De esta manera el operador deja de percibir el entorno que lo rodea y queda inmerso en el lugar remoto.

De los diferentes órganos sensoriales que posee el ser humano, la vista es la que más información nos presenta y es la que a su vez más sensación de presencia nos genera. Por eso todo sistema de telepresencia debe realimentar como mínimo al sentido de la vista, es decir debe generar una imagen del lugar remoto.

Una de las principales mejoras en lo que es visión es la visión estereoscópica, ya que permite un nivel de percepción de las distancias y ubicación de los objetos mucho más elevado y porque genera un estímulo mucho más grande debido a que el sistema de visión de ser humano también es estereoscópico.

1.7. GRADOS DE INMERSIÓN SENSORIAL.

El grado de inmersión sensorial está estrechamente relacionado con el número de sentidos para los cuales el sistema genera los estímulos y en segunda instancia el alcance, la calidad y la coherencia de dichos estímulos.

El ser humano, como ya se sabe, posee cinco sentidos: la vista, el oído, el tacto, el olfato y el gusto. A estos sentidos actualmente se añadió el sentido del equilibrio, que nos permite conocer la situación del cuerpo con respecto a la gravedad, y la cinestesia, que es la sensación de percibir movimiento, peso y posición de los miembros del cuerpo.

Como ya se mencionó uno de los factores más importantes para el grado de inmersión es el número de sentidos que intervienen en el sistema, cuando mayor sea el número de sentidos que intervienen mayor será el grado de inmersión.

Otro factor que interviene es la inmersión total o parcial, se entiende como inmersión total cuando al usuario se lo desconecta totalmente del lugar en el que se encuentra como por ejemplo la utilización de cascos HMDs que bloquean totalmente la visualización externa y muestra en pantallas las imágenes que deseamos.

Pero la inmersión total no es precisamente deseada, puesto que al no percibir el entorno real donde se encuentra el operador, puede provocar desorientación, como por ejemplo, los usuarios de sistemas HMDs por el mismo hecho de tener la vista aislada pierden la ubicación de sus extremidades provocando un cierto grado de desorientación.

Otro factor que determina el grado de inmersión es la calidad del estímulo, la velocidad de actuación y los tiempos de latencia del sistema. La calidad de la imagen y del sonido puede incrementar o disminuir el grado de inmersión, puesto que, como ya se mencionó antes la calidad debe ser comparable a la del sentido que la percibe.

La velocidad de actuación también es importante ya que si el operador realiza una acción determinada en el robot pero este tiene un tiempo muy grande de respuesta disminuye el grado de inmersión y mucho más si los tiempos de latencia son grandes puesto que el sentido no se realimenta debidamente y puede provocar desorientación.

1.8. REALIDAD AUMENTADA

En algunos casos, la finalidad del sistema no es extender la información sensorial que percibe del lugar remoto, sino que también se pretende aumentar dicha información. Esta técnica, cuyo objetivo es ampliar la percepción del mundo real con información crítica generada por un computador, se denomina *realidad aumentada*.

Y se puede definir a la realidad aumentada como una situación o circunstancia que percibe un operador cuando la información sensorial que le es realimentada de un entorno es modificada previamente por un computador con el objetivo de añadirle nueva información creada artificialmente, y que es no accesible directamente de la realidad por los sentidos del operador, aunque éste se encontrase en la zona remota^[3].

1.9. SISTEMA TELEOPERADO ROMOTEVI

1.9.1. DESCRIPCIÓN DEL SISTEMA

El sistema ROMOTEVI es un sistema teleoperado el cual posee un vehículo todo terreno que se encuentra equipado con un sistema de visión artificial el cual tiene como objetivo teletransportar a su operador al lugar remoto donde el vehículo se encuentra y así poder explorar dicho entorno, teniendo la capacidad de experimentar todas las sensaciones visuales que el entorno posea desde un lugar seguro.

El sistema teleoperado ROMOTEVI es un sistema que se encuentra clasificado dentro del tipo de teleoperación directa por control de velocidad, puesto que se envían órdenes a un vehículo robótico que tienen que ver con su velocidad de avance o retroceso. El sistema se encuentra desarrollado bajo la estructura maestro-esclavo ya que el operador es el encargado de enviar las distintas órdenes y el robot las realiza en el entorno remoto.

Para que el operador pueda realizar las tareas en el entorno remoto el sistema debe contar con dos características importantes un sistema de visión y uno de control, por este motivo, se ha dividido a todo el sistema en estos dos subsistemas importante tal como se muestra en la Figura 1-4, donde, el subsistema de visión es el encargado de presentar el entorno remoto al operador y el subsistema de control sirve para realizar tareas e interactuar con ese entorno remoto.

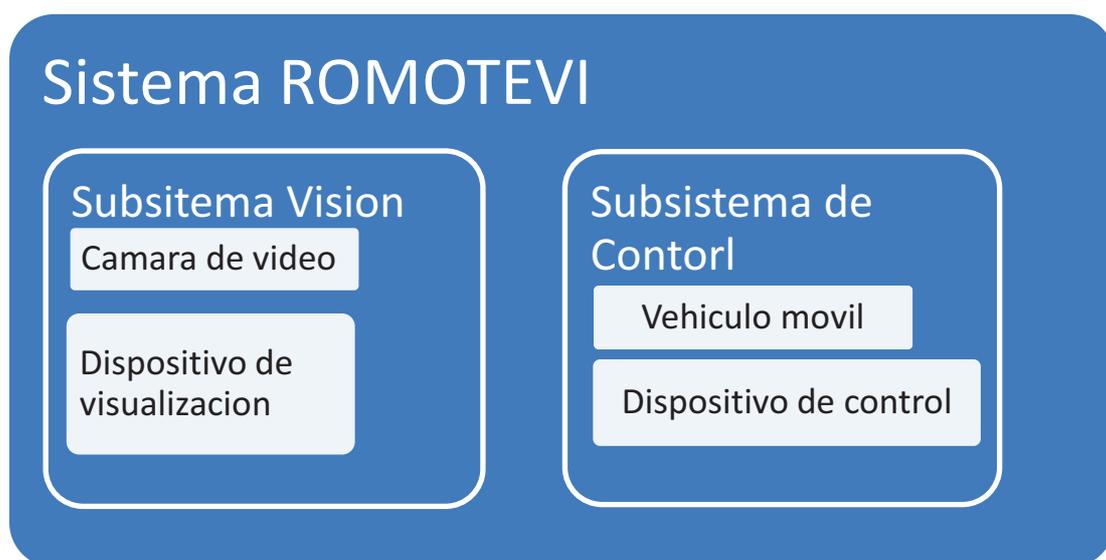


Figura 1-4. Esquema descriptivo del sistema ROMOTEVI

Además de lo mencionado anteriormente, el sistema ROMOTEVI posee interactividad con el medio porque lo puede explorar y porque lo puede modificarlo debido a que el vehículo robótico posee una herramienta para sujetar objetos. También posee interacción implícita debido a que todo el control del robot se lo ha hecho con dispositivos que recogen información de los movimientos del operador, tanto para la parte de visión como para la parte de control, y se los envía al vehículo robótico y debido a que el sistema ROMOTEVI usa un casco de realidad virtual y este dispositivo tiene la característica de ser puramente inmersivo, le brinda al sistema ROMOTEVI una de sus principales características la cual es poseer inmersión sensorial para el operador.

Debido a las tres características mencionadas anteriormente el sistema cuenta con todos los elementos para poseer telepresencia, además, posee un cierto grado de inmersión sensorial el cual fue medido mediante la realización de unas pruebas y de unos test, cuyos resultados fueron analizados en el capítulo 5.

1.9.2. DISEÑO DEL SISTEMA TELEOPERADO ROMOTEVI

En el diseño inicial del sistema se pretendía crear un robot móvil el cual se encuentre equipado con una cámara de video, el video y control se lo realizaría comunicando al robot a través de algún tipo de comunicación inalámbrica que en lo posterior fue definida como se verá más adelante. Del otro lado se pretendía tener un casco de realidad virtual junto con un joystick para controlar la cámara y el robot respectivamente.

Debido a que el robot debe comunicarse y enviar el video inalámbricamente, es mejor utilizar un protocolo de comunicación estándar para realizar dichas comunicaciones, por esto se determinó que lo más conveniente es utilizar comunicación WIFI con un protocolo TCP/IP de comunicación.

Para realizar este tipo de comunicación es necesaria la utilización de computadores, por ende se colocó un computador de abordo en el robot y un computador de escritorio como el maestro del robot.

1.9.3. VERSIÓN 1.1 DEL SISTEMA ROMOTEVI

El sistema teleoperado en su diseño inicial utiliza una plataforma móvil Pioneer 3-AT como vehículo robótico el cual posee un gripper rectangular en su parte frontal inferior como herramienta para la interacción con el medio y una cámara de video en su parte frontal superior para visualizar el entorno remoto, tal como se presenta en la Figura 1-5.

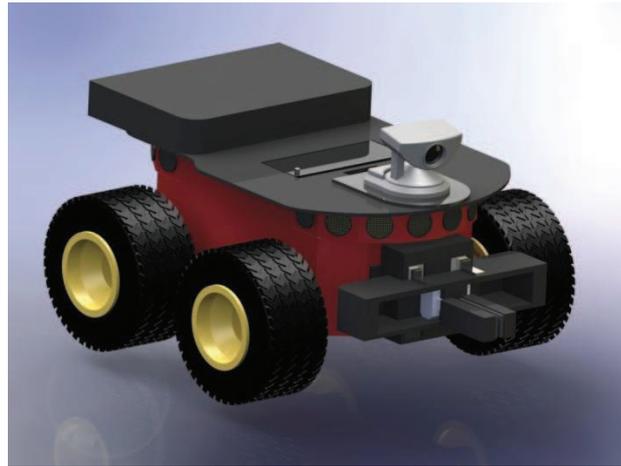


Figura 1-5.- Robot móvil Pioneer P3-AT

Como se mencionó anteriormente es necesario que el robot posea una computadora de abordo, por este motivo, se creó un contenedor para dicho computador que se colocó en la parte trasera de robot como se puede ver en la Figura 1-5.

Para el subsistema de visión, se utiliza un casco de realidad virtual junto con la cámara de video antes mencionada, utilizándolos en conjunto para obtener dos cosas importantes dentro del sistema, la primera es mostrar el entorno remoto al operador y la segunda es otorgar un cierto grado de inmersión al operador.

En el diseño inicial, como el casco de realidad virtual posee un resolución de 800X600 se utilizó esta misma resolución para presentar el video obtenido desde la cámara y como posee un grupo de sensores para obtener el movimiento de la cabeza del operador se utilizaron estos datos para controlar la cámara de video que posee dos grados de libertad generando de esta manera la característica de interacción implícita para el operador.

Para el subsistema de control se utiliza un joystick junto con la plataforma móvil para la exploración e interacción del operador con el entorno remoto, se determinó la utilización de un joystick porque es un dispositivo que genera la característica de interacción implícita en el ámbito de vehículos móviles.

1.10. VERSIÓN 1.2 DEL SISTEMA ROMOTEVI

Algunos aspecto del diseño inicial tanto en hardware como en software fueron modificados debido a circunstancias que se explicaran en los capítulos 3 y 4 respectivamente, dichas modificación generaron que en la cámara sea posicionada en la parte trasera superior de la plataforma, sobre el contenedor para la computadora de abordo y se coloque una flecha de color amarilla sobre la superficie de la plataforma en la parte frontal, todos estos cambios son comparables en la Figura 1.5.

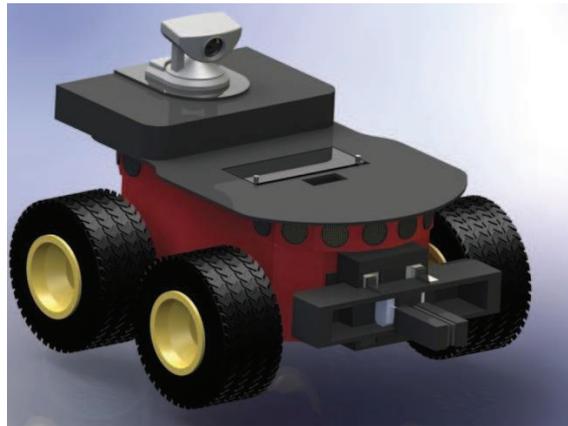


Figura 1-6.- Disposición actual de los elementos en la plataforma móvil

En cuanto al software se modificó el protocolo de comunicación y la resolución con la que se pretendía mostrar el video según el diseño, donde la resolución original seria de 800x600 y la resultante es de 160x120, la explicación del porqué de dichas modificaciones se explican en el capítulo 4 correspondiente al software que maneja todo el sistema ROMOTEVI.

CAPÍTULO 2

SISTEMAS DE CASCOS VIRTUALES (HMD)

2.1. INTRODUCCIÓN

La realidad virtual y la telepresencia tienen un gran énfasis en la creación de estímulos sensoriales para generar un engaño y hacer creer a un usuario que se encuentra en un lugar distinto a donde físicamente este se encuentre.

Dentro de los 5 sentidos que posee el ser humano, el más importante y el que más presencia genera es el sentido visual, por ende, en el presente capítulo se abordara su funcionamiento y los distintos tipos de dispositivos visuales que se utilizan para generar sensaciones en este sentido, poniendo particular énfasis en los sistemas de cascos virtuales o HMD.

Un cascos virtuales o HMD es un dispositivo que posee dos pantallas al frente de los ojos, con una compleja configuración óptica que pretende simular una pantalla mucho más grande de lo que realmente lo es y un sensor de orientación de la cabeza que permite presentar en las pantallas la imagen correspondiente a la dirección de la cabeza, una representación gráfica de un casco de realidad virtual se tiene en la Figura 2-1.



Figura 2-1.- Elementos que componen un casco de realidad virtual.

2.2. CASCO DE REALIDAD VIRTUAL

El casco de realidad virtual o HMD (head mounted display) es un casco que tiene dos pequeños monitores que se sitúan delante de cada ojo, un sensor de orientación de la cabeza, que permite presentar en pantalla la imagen correspondiente a la dirección a la que mira el observador y un sistema de audio que permite percibir todos los sonidos del mundo virtual, por tanto, es un sistema puramente inversivo.



Figura 2-2.- Imagen de un casco de realidad virtual.

Sus inconvenientes son la molestia de llevar un aparato acoplado a la cabeza, la posible fatiga visual que genera, su uso individual y el coste prohibitivo que tiene si se requiere alta resolución y/o gran campo de visión, el ejemplo de un casco de realidad virtual se tiene en la Figura 2-2.

2.3. SENTIDO VISUAL

El sentido visual es el sentido más desarrollado en el ser humano, este tiene la capacidad de capturar la luz que llega al campo visual, reflejada por los cuerpos, para obtener información del entorno. De esta información se puede distinguir tamaños y formas de los cuerpos, la distancia a la cual se encuentran e incluso el tipo de superficie.

El sistema visual está conformado por el ojo el cual es el encargado de recibir y transformar la luz en impulsos nerviosos y una parte del cerebro que se encarga de recoger

estos impulsos nerviosos y convertirlos en percepción visual. El sentido visual se encuentra conformado por tres características, como se lo enuncia en la Tabla 2-1.

Características	Descripción
Campo visual	Son los grados de visión que tiene el ser humano en cada ojo y en su visión estereoscópica
Sistemas oculomotores	Es todo el sistema muscular utilizado para producir movimiento de los ojos.
Fotoreceptores	Son los encargados de percibir la luz que ingresa al ojo e interpretarla con el cerebro humano.

Tabla 2-1.- Elementos que conforman el sentido visual humano.

2.3.1. CAMPO VISUAL ^[7]

Este factor es el más importante dentro del sistema visual, puesto que es lo más complicado de simular en el sistema computacional. El campo visual de cada ojo humano es de 150 grados en línea horizontal y de 120 a 135 grados en línea vertical, el campo visual combinado de los dos ojos es de 180 a 200 grados en línea horizontal con una zona de solapamiento de 180 grados dejando de 30 a 35 grados de visión monocular a cada lado, como se muestra en la Figura 2-3.

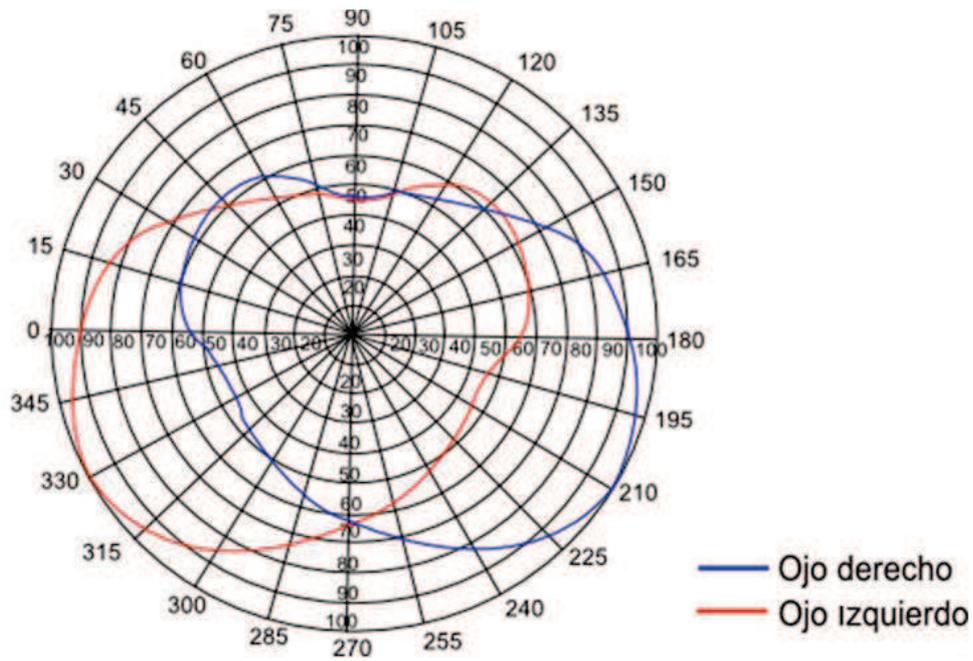


Figura 2-3.- Esquema de la disposición del campo visual del ser humano.

Este factor es difícil de simular cuando se pretende presentar el mundo virtual utilizando una cámara, puesto que, una cámara no tiene un rango tan amplio de visión, además, la imagen de una cámara no es estéreo siendo recomendable la utilización de un sistema estereoscópico si se pretende representar el mundo real a través de una cámara.

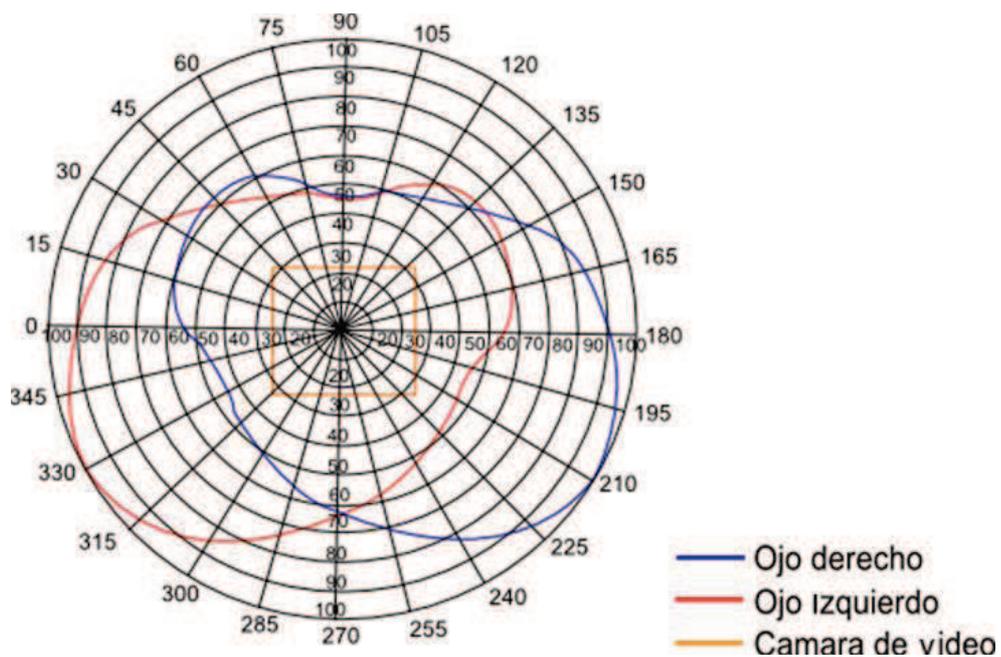


Figura 2-4. Campos de visión de los elementos que intervienen en el sistema.

Como se mencionó anteriormente, existe un problema que se detectó en la ejecución de las pruebas hechas en el Capítulo 5, donde se observa que el campo de visión que se presenta al operador es muy reducido cuando se mira cosas cercanas a la cámara. El problema se genera debido a que el ángulo de visión de la cámara es muy reducido, como se muestra en la Figura 2-4, es de apenas 47 grados horizontalmente y de 42 grados verticalmente. Dichos grados, comparando con el amplio campo de visión humano solo en su parte binocular, es demasiado pequeño generando problemas de adaptación en la visión.

Para este análisis no se ha tomado en cuenta el campo de visión del casco de realidad virtual, si bien es parecido al de la cámara de unos 40 grados en diagonal, este es un dispositivo que se usa para generar inmersión y es la cámara la que verdaderamente limita o promueve el campo de visión de todo el sistema.

2.4. PERCEPCIÓN VISUAL

La luz es una estrecha banda de radiación electromagnética de longitud de onda comprendida entre los 400 nm y 700 nm, caracterizada por tres parámetros los cuales son: color, brillo y contraste.

Estos parámetros juntos permiten una correcta percepción del entorno que nos rodea, donde el color o espectro se refiere a las diferentes longitudes de onda de las que está compuesta la luz, el brillo es la amplitud o intensidad de la luz y el contraste o saturación es la pureza del color.

Los tres parámetros se complementan con todos los parámetros que genera la percepción de profundidad, como se puede ver en la Tabla 2-2. La percepción de profundidad a su vez contiene tres parámetros los cuales son: parámetros monoculares, oculomotores y binoculares, para generar una correcta percepción visual del mundo que nos rodea.

Es importante debido a que dentro de los indicadores de profundidad oculomotores son los que presentan problemas al utilizar un casco de realidad virtual puesto que su uso causa fatiga en los ojos.

Elementos de la percepción visual	Características
Color	Una visión a color es importante para percibir correctamente los objetos en el mundo real o virtual.
Brillo	Es la intensidad luminosa reflejada por el cuerpo que se pretende percibir.
Percepción de profundidad	Se encuentra caracterizado por: <ul style="list-style-type: none"> ● Parámetros monoculares. ● Parámetros oculomotores. ● Parámetros binoculares.

Tabla 2-2.- Tabla de los elementos que conforman la percepción visual.

2.4.1. PROFUNDIDAD

Una característica del sistema visual humano es aprovechar cualquier oportunidad para distinguir profundidad, estos indicios se los puede clasificar como indicadores monoculares, oculomotores, binoculares y de movimiento. Cabe recalcar que para tener una percepción de profundidad no es estrictamente necesario poseer visión estereoscópica gracias a que por nuestra experiencia también podemos percibirla mediante el uso de otros métodos.

La característica principal de los indicadores monoculares es que usan un solo ojo y no utilizan la ayuda de los músculos oculares, y se basan principalmente en la experiencia personal. Existen varios indicadores monoculares los cuales son:

- *Perspectiva lineal.*
- *Interposición u oclusión.*
- *Gradientes de textura.*
- *Covarianza de luminancia.*
- *Sombras.*
- *Altura en el campo visual*

Los indicadores oculomotores tienen que ver con los músculos que producen movimientos en los ojos y con la deformación del cristalino, dentro de estos indicadores se pueden encontrar los siguientes:

- **Acomodación.-** Se basa en modificar la forma del cristalino para poder observar de 6 mts en adelante con el cristalino relajado o mirar a menos de 6 mts con el cristalino deformado.
- **Convergencia.-** Trata de la rotación de los ojos hacia la nariz para mirar objetos más cercanos.

Estos dos indicadores son los que vienen a producir ciertos problemas con la utiliza de los distintos tipos de visualizador ya son los causantes de fatiga en los ojos debidos a la cercanía de las pantallas del casco a los ojos produce demasiada convergencia en los ojos y también obliga a deformar constantemente el cristalino para mirar las pantallas.

Otro indicador de profundidad, y el más importante para distancias medias, es la denominada disparidad o paralaje binocular el cual se produce debido a la diferencia de imágenes, de un mismo objeto, producidas en las retinas de cada ojo.

Entendemos como disparidad binocular a la distancia entre puntos homólogos medida sobre la retina. Esto se produce debido a que los ojos están separados horizontalmente una distancia denomina distancia interocular. Se sabe que dos puntos en ambas retinas son homólogos si proceden del mismo punto del mundo real (o sintético). La disparidad binocular está provocada por el hecho de que cada ojo ve el mundo desde un punto de vista diferente.

Existe un concepto análogo a la disparidad binocular llamado Paralax el cual es aplicado a pantallas o pantallas de proyección, como el caso de la visión plano-estereoscópica generada a partir de dos imágenes 2D. Si este concepto es aplicado correctamente con un dispositivo adecuado puede producir disparidad binocular.

Y por último tenemos el indicador de paralaje de movimiento, este se basa en la velocidad a la que un objeto se mueve a través de la retina; este indicador es el más importante para percibir profundidad a grandes distancias. Los objetos más cercanos se moverán más rápido que los objetos lejanos aunque los dos estén a la misma velocidad.

2.5. CREACIÓN DE IMÁGENES EN UN CASCO DE REALIDAD VIRTUAL ¹⁹¹

Existen cuatro tipos de formas para crear imágenes en dispositivos externos, las cuales, se encuentran clasificadas debido a la fuente de luz que poseen para iluminar los píxeles, entre ellas se tienen:

- Transmisivas.
- Reflectivas.
- Auto-emisivas.

2.5.1. PANTALLAS TRANSMISIVAS

Son pantallas que poseen una fuente de luz que ilumina la pantalla desde la parte posterior de esta, la pantalla posee píxeles que son prendidos o apagados o parcialmente prendidos para escala de grises, como se puede ver en la Figura 2-5. Los transistores que controlan estas pantallas se encuentran a lo largo de los lados del píxel.

La ventaja de este tipo de pantallas es su diseño simple de iluminación, sus imágenes producidas son de alta calidad, pero tienen un factor de relleno menos eficiente y requiere una etapa más para integrar el color.

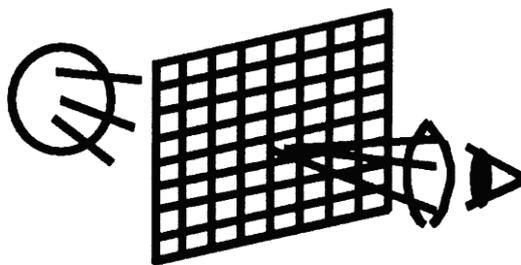


Figura 2-5.- Pantalla transmisiva.

2.5.2. PANTALLAS REFLECTIVAS.

Son pantallas que poseen una fuente de luz que ilumina la matriz de píxeles de la pantalla desde la parte frontal de esta, con una superficie reflectante bajo cada píxel. El píxel es prendido o apagado o parcialmente prendido para escalas de grises, como se puede

ver en la Figura 2-6. Los transistores que controlan cada pixel se encuentran debajo de los pixeles.

Poseen una alta eficiencia lumínica y su factor de relleno es de alta densidad debido a que tiene los transistores atrás de cada pixel. La desventaja de estas pantallas es controlar la dispersión de la luz que se genera y necesitan de una etapa de integración para el color.

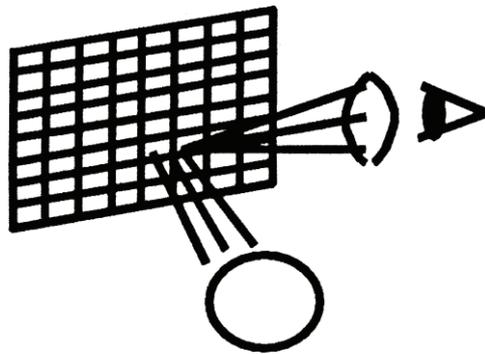


Figura 2-6.- Pantallas reflectivas.

2.5.3. PANTALLAS AUTO-EMISIVAS.

En estas pantallas los mismo pixeles emiten luz, cada pixel es prendido y apagado o parcialmente prendido para manejar escalas de grises, como se puede ver la Figura 2-7. Los transistores que controlan los pixeles se encuentran debajo de cada uno de ellos y los controladores electrónicos se encuentran alejados de las fuentes de imagen.

Como ventajas en este tipo de pantallas es que todo el conjunto es de menor peso, tiene un alto factor de relleno debido a que sus transistores se encuentran detrás de cada pixel y trabaja en un amplio rango de temperaturas. Estas pantallas poseen una luminancia limitada.

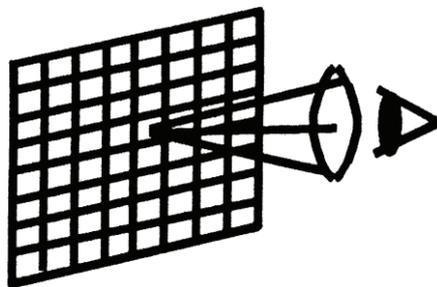


Figura 2-7.- Pantallas auto-emisivas.

2.5.4. TIPOS DE PANTALLAS USADAS EN CASCOS VIRTUALES

A partir de los años 2000, ya existían los cascos virtuales los cuales se encontraban equipados con pantallas CTR, ósea pantallas de tubos catódicos, debido a que en esta época dicha tecnología era la más madura y cumplía con todas las necesidades de alta luminosidad requerida.

Sin embargo, en las últimas décadas nuevas tecnologías como pantallas pequeñas y delgadas han sido desarrolladas para los cascos virtuales, debido a su bajo consumo, reducido peso y tamaño.

Actualmente existen dos categorías para las fuentes de imagen de los cascos virtuales, las cuales son de emisión y no emisión. Las fuentes no emisivas son aquellas que modulan la imagen pixel a pixel en una base para mostrar la imagen deseada, dichas fuentes son dos las cuales son enunciadas a continuación:

- Pantallas de cristal líquido transmisivo (LCD).- Son pantallas transmisivas, un campo eléctrico modula y controla la transmisión de la luz de fondo a través de celdas llenas de cristal líquido.
- Pantalla de cristal líquido en silicio (LCOS).- Estas son pantallas reflectivas. La luz es transmitida a través de una celda, si el pixel está transmitiendo se refleja en una superficie como un espejo pero si el pixel está apagado esta luz se dispersa.

Los dispositivos de emisión contienen una amplia categoría de fuentes de imagen en la que el plano de la imagen del dispositivo emite luz sin la necesidad de iluminación adicional. Las pantallas que se encuentran dentro de esta categoría son las siguientes:

- Diodos orgánicos emisores de luz (OLED).- Una unidad de disco de bajo voltaje, a través de una delgada capa de materia orgánica, hace que emita luz visible cuando se recombinan los portadores de carga en el material.

2.5.5. RESOLUCIÓN GRAFICA DE IMAGEN

La resolución de una imagen por concepto es el número de píxeles que componen su alto y largo, dependiendo de la resolución que posea la imagen esta tendrá mayor o menor definición.

Actualmente la resolución que presentan los diferentes tipos de pantalla que se mencionaron en el apartado anterior depende mucho de las capacidades que posea la tarjeta gráfica, los cascos de realidad virtual básicamente manejan 2 tipos de tecnología grafica las cuales son:

- VGA: Arreglo grafico de video.
- XGA: Arreglo grafico extendido.

Tecnología VGA

La tecnología VGA es una tecnología que maneja resoluciones pequeñas, fue la primera en aparecer y se la utilizo para monitores de computadoras. Actualmente esta tecnología se encuentra dividida en varias versiones que fueron aumentando o disminuyendo la resolución según sea el caso, tal como se muestra en la Tabla 2-3.

Tecnología	Resolución
QQVGA	160X120
HQVGA	240X160
QVGA	320X240
WQVGA	400X240
HVGA	480X320
VGA	640X480
WVGA	800X480
FWVG	854X480
SVGA	800X600
DVGA	960X640

WSVGA	1024X576
WSVGA	1024X600

Tabla 2-3. Tabla de resoluciones para la tecnología grafica VGA

De esta tabla las tecnologías que más se usan en cascos virtuales es la VGA y SVGA debido a que es más barata y no sube tanto los costos del casco como se puede ver en la Tabla 2-5.

Tecnología XGA

Este tipo de tecnología surgió después de la VGA debido a la necesidad de manejar resoluciones más grandes, esta tecnología es utilizada en monitores de resoluciones más amplias, actualmente se encuentra dividida en varias versiones que manejan distintas resoluciones como muestra la Tabla 2-4.

Tecnología	Resolución
XGA	1024X768
WXGA	1280X720-1280X800-1360X768
XGA+	1152X864
WXGA+	1440X900
SXGA	1280X1024
SXGA+	1400X1050
WSXGA+	1680X1050
UXGA	1600X1200

WUXGA	1920X1200
-------	-----------

Tabla 2-4. Tabla de las resoluciones existentes en la tecnología gráfica XGA.

De la Tabla 2-4, la tecnología que más se usa actualmente en cascos virtuales es la SXGA, la cual alcanza resoluciones de 1280x1024, esta resolución viene incorporada en cascos que poseen más característica como visión estereoscópica y en 3D debido a que su costo es más elevado en comparación a los que usan resoluciones VGA.

2.6. SISTEMA DE TRACKING

Para lograr una sensación integral de inmersión en el mundo virtual es importante tener una relación directa entre el movimiento del cuerpo y la acción en el ambiente computacional, lo que se denomina paralelismo cinético (motion parallax). Es especialmente importante para controlar la visión del entorno digital, al desplazar la cabeza o para la localización de las distintas partes del cuerpo.

Para poder controlar el movimiento es necesario colocar rastreadores (trackers) en las distintas partes del cuerpo. Estos rastreadores pueden ser mecánicos, ultrasónicos, ópticos o magnéticos y permiten conocer la posición tridimensional y la orientación (seis grados de libertad), definiendo exactamente la posición en el espacio.

Los más sencillos y utilizados son los denominados giroscopios. Se instalan en la parte frontal de los cascos y permiten reconocer giros de la cabeza para adecuar la imagen visualizada en el casco o en una pantalla.

Los HMDs actuales tienen rastreador de cabeza de diseño avanzado de alta velocidad utilizan sistemas micro-electro mecánicos (MEMS) giroscopios y acelerómetros, uno para cada eje X, Y y Z), para percibir el movimiento de la cabeza, y transmite la información a un programa en ángulos de giro, inclinación y rotación conocidos como pitch, yaw y roll tal y como lo demuestra la Figura 2-8.



Figura 2-8. Imagen de los movimientos sentidos por los trackers que posee el HMD.

2.6.1. GRADOS DE MOVIMIENTO DE UN CASCO DE REALIDAD VIRTUAL

Los grados de movimiento de un casco de realidad virtual están dados por los sensores que este posee, estos sensores generalmente son sensores inerciales MEMS, Figura 2-9, los cuales, consisten en dispositivos en miniatura combinados con componentes de detección inerciales eléctricos y mecánicos. Están diseñados para convertir una fuerza de inercia en una salida de tensión lineal con una sensibilidad determinada. Los sensores de esta tecnología más usados en cascos de realidad virtual son:

1. Acelerómetros.
2. Giroscopios.
3. Brújulas.

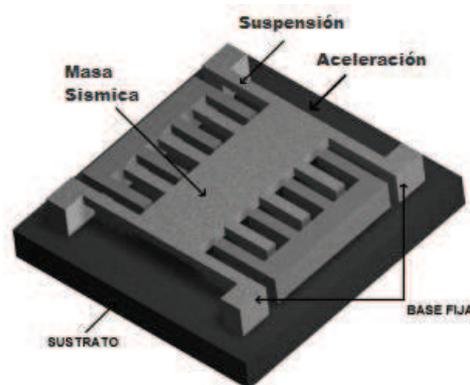


Figura 2-9. Acelerómetro capacitivo MEMS de movimiento horizontal

Estos dos son ampliamente utilizados en muchas aplicaciones en las industrias aeroespacial, militar, automotriz y marina. También existen en cascos de realidad virtual en los cuales miden los movimientos de la cabeza del operador.

El casco de realidad virtual usado en el sistema y descrito en el Capítulo 2, tiene la capacidad de medir movimientos en los rangos descritos en la Figura 2-10, los cuales son 120 grados en vertical y 360 grados en horizontal. Si se compara con los rangos de movimientos de la cabeza del ser humano, también descritos en la Figura 2-10, se puede ver que solventa totalmente la medición de dichos movimientos. Como estos movimientos deben ser enviados a una cámara, también es importante tomar en cuenta los rangos de movimiento que posee la misma.

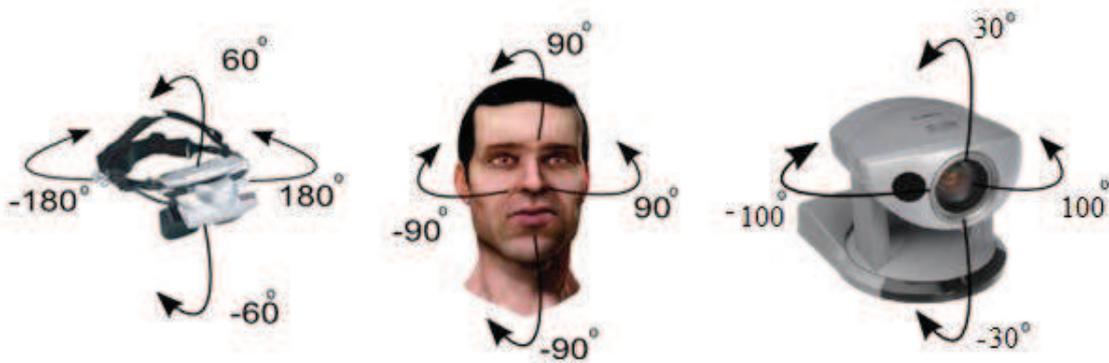


Figura 2-10. Rangos de movimientos de los elementos que intervienen en el sistema de visión.

En el sistema que se implementó, la cámara posee un rango de movimiento descrito en la Figura 2-10, donde se observa que puede sustentar parte de los movimientos de la cabeza, pero el ángulo de inclinación es muy reducido, pues para el ser humano es de ± 90 grados y la cámara solo puede cubrir ± 30 grados. Esto representa un problema dentro del sistema, pues aunque el ángulo de inclinación negativo de la cámara solo muestra la parte superior de la plataforma, el problema radica en el ángulo de inclinación superior, puesto que éste no alcanza a cubrir el movimiento total de la cabeza.

Cabe recalcar que para el resto de movimientos posibles, la limitante de dichos movimientos será la cabeza del operador; sin embargo, no se colocó seguridades basadas en los movimientos de la cabeza, debido a que si es necesario mirar más hacia el lateral se

lo puede hacer pues el casco posee la capacidad de medir dichos movimientos y el operador puede hacerlo moviendo todo el cuerpo.

2.6.2. MONITORES

Los monitores se comportan como pequeñas ventanas dentro de nuestro campo de visión, normalmente son sistemas monousuario. La tecnología empleada es muy variada: CRT, LCD y plasma. Para conseguir un efecto más inmersivo hay dos opciones: o bien usar una pantalla más grande, o acercar la pantalla a los ojos. Este último caso lo implementa uno de los dispositivos más típicos de realidad virtual que es el casco de realidad virtual o HMD.

El HMD usa dos pequeñas pantallas que se colocan delante de los ojos a modo de gafas. Un sistema óptico entre el ojo y las pantallas permite crear una imagen virtual más grande y más alejada del ojo, las imágenes tienen un tamaño equivalente a una pantalla de cine del sistema internacional de medida que es de 2.67 metros a una distancia de 3.66 metros, lo que genera una visión mucho más cómoda. En la Figura 2-11 se muestra el esquema de un HMD donde se quiere representar en el espacio un punto C. Cada pantalla muestra el punto tal y como lo vería cada ojo por separado (A1 ojo izquierdo y A2 ojo derecho). El usuario no ve estos puntos sino sus equivalentes virtuales (B1 y B2 respectivamente). El sistema visual hace converger estos dos puntos en uno sólo: el punto C. La convergencia depende de la distancia interpupilar.

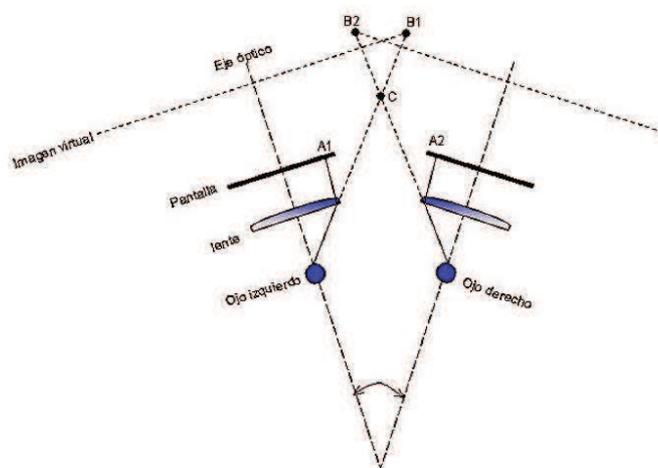


Figura 2-11. Esquema de la disposición de las pantallas en un HMD.

Configuración óptica de las pantallas de un HMD. ^[9]

En un casco de realidad virtual existen tres características que se deben cumplir con ciertas configuraciones ópticas, las cuales son:

- Colime la fuente de imagen, lo cual significa, crear una imagen virtual que aparenta estar más lejos de lo que realmente está.
- Magnificar la fuente de imagen, para crear una imagen que aparente ser mucho más grande de lo que realmente es.
- Repetir la fuente de imagen, es decir, crear una imagen virtual de la fuente de imagen fuera del frente de la cara.

Existen dos enfoques que se utilizan comúnmente en los cascos de realidad virtual que tratan de cumplir con estas tres características los cuales son: el diseño de formación no pupilar y el diseño de formación pupilar.

Diseño de formación no pupilar.

Es una formación de lentes de aumento simple, es el más fácil de diseñar, el menos costoso de fabricar, más ligero y más pequeño. Esta configuración sufre de una distancia de proyección corta entre el origen de la imagen y la imagen virtual, poniendo a todo el sistema óptico en la parte frontal de la cabeza, cerca de los ojos. En la Figura 2-12 se tiene la representación gráfica de un diseño no pupilar. Este enfoque se utiliza normalmente para aplicaciones de visualización simple, tales como las HMD médicas.

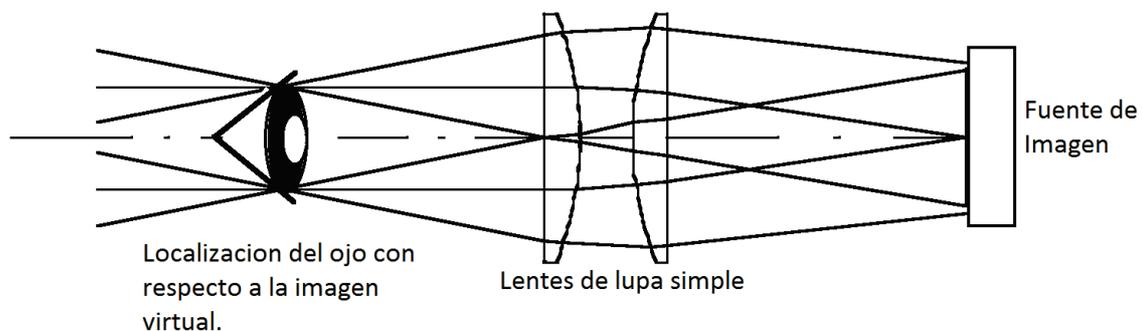


Figura 2-12.- Diseño de una formación de lentes no pupilar.

Diseño de formación pupilar

Este diseño es un poco más complejo, es más parecido al arreglo óptico que posee un microscopio compuesto, o un periscopio submarino en el que un primer grupo de lentes crea una imagen intermedia de la fuente de imagen. Esta imagen intermedia se retransmite por otra serie de lentes, donde se crea la imagen impresa de la imagen intermedia, tal como se puede ver en la Figura 2-13.

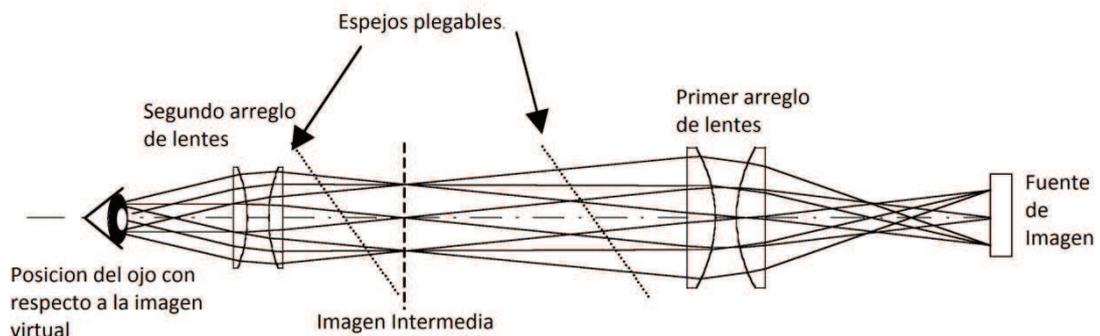


Figura 2-13.- Diseño de una formación de lentes pupilar.

La ventaja de este diseño es que proporciona más longitud de camino desde el plano de la imagen al ojo. Esto le da al diseñador una mayor libertad para insertar cuantos espejos plegables necesite en el tren óptico, mejorando el centro de gravedad. Las desventajas son: el aumento de peso por los lentes adicionales y el coste del casco de realidad virtual.

Estos sistemas son muy importantes ya que son los encargados de generar la ilusión de estar mirando una pantalla mucho más grande lo cual genera un nivel de inmersión; pero esta pantalla y el nivel de inmersión van a estar relacionados con el campo de visión que el casco abarca con relación al del ser humano. Mientras más campo de visión pueda abarcar el casco, mayor será el nivel de inmersión del operador.

El casco de realidad virtual usado en el sistema posee un campo de visión de 40 grados en diagonal, esto es aproximadamente 47 grados horizontalmente y 42 grados verticalmente, que comparado con el campo visual que el ser humano posee en su parte binocular es pequeño. Debido a como se lo está usando, éste no influye en el campo de

visión que el operador ve, ya que el mismo está determinado por la cámara de video; lo que sí determinará es el nivel de inmersión que experimenta el operador.

2.7. SONIDO ESTÉREO

Un sistema de sonido estéreo está formado por dos canales de sonido y sus respectivos parlantes. Jugando con la intensidad y retardo de la señales de cada parlante se puede lograr ubicar fuentes sonoras, llamadas fuentes fantasmas, en cualquier punto situado entre los dos altavoces (Figura 2-14). Esto se debe a que la primera onda de un sonido es la que toma el cerebro como dirección de éste, suprimiendo la información direccional del otro sonido similar, que llega inmediatamente después de percibir el primer sonido. Estos sonidos tienen que llegar dentro de una ventana de tiempo de unos 50ms.

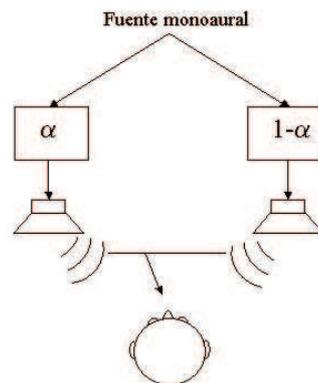


Figura 2-14.- Sistema de sonido estereofónico.

Si las señales son iguales en intensidad y retardo en ambos canales, la señal parece proceder de una fuente central del sistema. Por el contrario, si se introduce un retraso en el canal derecho, la fuente de sonido parecerá proceder del lado izquierdo, a pesar de que las dos intensidades sean iguales. Si a continuación, se atenúa el canal izquierdo o se amplifica el derecho, la fuente de sonido volverá a desplazarse al centro del sistema. De esta manera existe una relación entre retardo e intensidad para ubicar el sonido en el espacio que hay delante del oyente.

Algunos televisores estéreo cuando reproducen un programa grabado en sonido monofónico, simulan el estéreo retrasando ligeramente la señal de uno de los canales, esto da un efecto pseudo-estéreo que resulta más agradable que el sonido monofónico.

El sonido estéreo también se puede conseguir con auriculares, aunque hay limitaciones: la respuesta a frecuencias bajas es peor; y, a la fuente fantasma puede ubicársele dentro de la cabeza, en lugar de enfrente del oyente.

2.8. ANÁLISIS COMPARATIVO

Como resumen del capítulo se determinó los aspectos importantes a tomar en cuenta para una correcta configuración del sistema; por ejemplo, se determinó que el campo de visión de todo el sistema está directamente relacionado con el campo de visión de la cámara, por ende, se debería mantener una estrecha relación entre el campo de visión de la cámara y del casco, pues si el casco de realidad virtual abarca una gran parte del campo de visión si la imagen y la imagen que se va a presentar en muy estrecha, esto provocaría que toda la inmersión del casco se pierda.

También, se determinó como una parte importante del sistema a la resolución que se va a manejar en el casco de realidad virtual, pero esta resolución va a estar estrechamente relacionada con la resolución del video que se está manejando; por ejemplo, en el sistema el video esta con una resolución de 160x120 y el casco posee una de 800x600, por tanto, específicamente para este diseño, esa resolución es adecuada pero si se pretende mejorar la calidad de la imagen mejorando la resolución es conveniente utilizar un casco con una resolución más grande.

Otro aspecto a tomar en cuenta es el rango de medición de los movimientos que posee el casco de realidad virtual, pues siempre tienen que ser iguales o mayores a los rangos de movimiento normales de una cabeza humana. Por otro lado están los rangos de movimientos que posee la cámara, éstos tienen que ser preferentemente análogos a los rangos de medida del casco de realidad virtual, e iguales o mayores a los rangos de movimiento que posee la cabeza del operador.

En cuanto a la parte de costos, un casco de realidad virtual se ve afectado específicamente por la resolución y el tipo de pantalla que éste posee. Los cascos virtuales

relativamente más baratos, cuyos valores van desde los 400 a 600 dólares, son los que poseen resoluciones VGA, tal como se puede ver en la Tabla 2-5, y luego les siguen los cascos virtuales de resolución SVGA que oscilan entre 800 a 3000 dólares. Dentro de los cascos virtuales con resolución VGA podemos encontrar aquellos que usan pantallas LCD u OLED, lo cual también se refleja en el costo. Como se puede ver en la Tabla 2-5, un casco de realidad virtual con la misma resolución pero con pantalla LCD es más barato que uno que posee pantallas OLED.

Los cascos virtuales que usan la tecnología gráfica XGA, actualmente trabajan solo en el rango de resoluciones SXGA (1280x1024); este tipo de casco es el que posee el costo más elevado en el mercado los cuales oscilan entre 3000 a 12000 dólares.

A su vez, estos cascos utilizan tres tipos de pantalla: LCD, OLED y LCOS, donde los cascos que utilizan pantallas LCD reportan ser los más baratos dentro de su clase y los que usan pantallas LCOS los más caros, tal como se puede ver en la Tabla 2-5.

Nombre del casco	Resolución del casco	Tipo de pantalla	Imagen virtual	Campo de visión	Entrada de señal	Precio \$
i-Trek 3D	VGA 640x800	LCD	2,28mx3m	35°	NTSC/PAL	599
z800 pro dual	FULL SVGA 800x600	OLED	2.66mx3.65m	40°	NTSC/PAL	2295
Virtual viewer 3D	SVGA 800x600	LCD	2.28mx3m	35°	NTSC/PAL	799
VR HMD Pro 3D	SVGA 800x600 SXGA 1280x1024	OLED	-	42,60°	NTSC/PAL	2999
VR HMD Pro 3D	SVGA 800x600	OLED	2.16mx1.52m	35°	NTSC/PAL	10500
Virtual Visor	SVGA 800x600	LCD	2.16mx3m	35°	NTSC/PAL	499
5DT HMD 800	SVGA 800x600	LCOS	-	40°	NTSC/PAL	3895
VR Pro SXGA	SXGA 1280x1024	LCD	-	60°	NTSC/PAL	10500
VR1280	SXGA 1280x1024	LCOS	-	60°	NTSC/PAL	12000

Tabla 2-5. Tabla comparativa de los precios de cascos virtuales

CAPÍTULO 3

IMPLEMENTACIÓN DEL HARDWARE DEL SISTEMA DE TELEOPERACIÓN

3.1. HARDWARE DEL SISTEMA ROMOTEVI

El sistema ROMOTEVI consta de un robot móvil Pioneer 3-AT, equipado con una cámara de video y un gripper rectangular en su parte frontal. También posee una computadora de a bordo Acer ONE (servidor) y un computador cliente, que a su vez posee un casco de realidad virtual o HMD y un joystick. El sistema se encuentra conectado como se ilustra en la Figura 3-1.



Figura 3-1.- Gráfico general del sistema

En la Tabla 3-1 se detalla cada uno de los dispositivos que conforman el sistema, su función y su medio de comunicación.

Nombre del dispositivo	Función del dispositivo	Medio de comunicación
HMD	El eMagin Z800 permite visualizar el entorno remoto y controlar a la cámara de video.	USB
Joystick	El joystick Genius F-23 controla remotamente a la plataforma y al gripper	USB
Computador cliente	Realiza la conexión hacia el servidor del robot a través de un <i>access point</i> .	ETHERNET
Wireless access point	Realiza el enlace entre el computador cliente y el computador servidor del robot.	<ul style="list-style-type: none"> • ETHERNET • WIFI
Computador servidor	Recibe las peticiones del cliente para el robot, controla los movimientos del mismo y transmite el video de la cámara.	<ul style="list-style-type: none"> • WIFI • RS-232 • S-VIDEO
Plataforma robótica Pioneer 3-AT.	Plataforma móvil encargada de realizar la exploración del entorno remoto.	<ul style="list-style-type: none"> • RS-232
Cámara de video	Canon VC-C50i capta y transmite video del entorno remoto.	<ul style="list-style-type: none"> • RS-2323 • S-VIDEO

Tabla 3-1.- Tabla con los dispositivos y sus funciones en el sistema.

El hardware implementado en todo el sistema ROMOTEVI se encuentra dividido en dos partes importantes, las cuales se encuentran enlazadas por una conexión Wifi:

- Cliente
- Servidor

Adicionalmente, posee 2 tipos de configuraciones posibles, las cuales han sido analizadas para encontrar la que mejor se adapte a un mejor manejo y control del robot. Dichas configuraciones han sido nombradas como versiones 1.1 y 1.2, que fueron levemente descritas en el Capítulo 1 y serán abordadas a mayor profundidad en este capítulo.

3.2. SERVIDOR DEL SISTEMA ROMOTEVI

Tal como se muestra en la Figura 3-2, el hardware del servidor consta de tres partes fundamentales:

- Un robot móvil Pioneer 3-AT
- Una cámara de video VC-C50i
- Una computadora de abordo Acer Spire One.



Figura 3-2.- Hardware que conforma el servidor del sistema ROMOTEVI

Estos tres elementos son los básicos para realizar la teleoperación de un robot móvil: la cámara muestra el entorno por el cual se está desplazando este; la computadora de

abordo es la encargada de realizar las acciones que pide el cliente sobre el robot; y, el robot móvil es el elemento físico a teleoperar. Además de estos tres elementos el robot móvil consta de un gripper.

3.2.1. HARDWARE DEL ROBOT MÓVIL PIONER P3-AT

El P3-AT es una plataforma robótica todo terreno, tiene 4 motores y llantas todo terreno, circula por superficies de tierra, piedra o pavimento y puede subir pendientes de hasta 45 grados. Es capaz de alcanzar velocidades de hasta 0.7 metros por segundo y una capacidad de carga de máximo 15 Kg.



Figura 3-3.- Robot móvil Pioneer P3-AT

En la Figura 3-3 se puede apreciar la forma física de la plataforma Pioneer P3-AT, a continuación se describirán todos los componentes, tanto internos como externos, que posee esta plataforma:

- Microprocesador Renesas SH2.
- Tarjeta de potencia para la alimentación de los dispositivos.
- Un gripper.

Microprocesador Renesas SH2

El microprocesador Renesas SH2 tiene una velocidad de procesamiento de 44.2368Mhz y 32 bit RISC, posee una memoria RAM de 32K y 128K de memoria flash en el que se encuentra embebido el sistema ARCOS, el cual es el encargado de administrar todos los recursos que se encuentran implementados en el robot. El microprocesador se

encuentra colocado en una placa donde tiene todos los puertos de entrada y salida para controlar el robot, como se muestra en el Anexo 1.

Gripper

El gripper Pioneer posee 2 grados de libertad y está colocado en la parte frontal de la plataforma robótica Pioneer P3-AT. Los dos movimientos que posee radican en la apertura y cierre de las pinzas, y en el ascenso y descenso de todo el mecanismo de cierre y apertura de las mismas, lo que le permite recoger objetos que se encuentren en el suelo, alzarlos y llevarlos a una nueva ubicación. Las especificaciones del gripper se detallan en el Anexo 1.

3.2.2. CÁMARA DE VIDEO VC-C50I

Es una cámara que se asemeja a los movimientos de la cabeza, posee 2 grados de libertad tal como se muestra en la Figura 3-4. Es una cámara de video que posee la capacidad de control de sus movimientos a través de una conexión RS-232 y una salida video tipo S-Video, las cuales fueron utilizadas para este proyecto. Dichas conexiones se encuentran descritas en el Anexo 2.

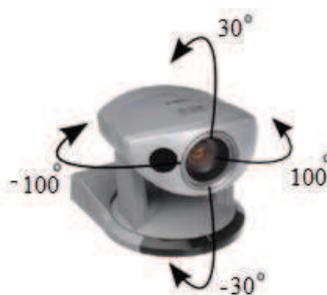


Figura 3-4.- Cámara de video Canon VC-C50i

La cámara está conectada a la computadora de abordo a través de una tarjeta de adquisición de video, y a la plataforma móvil Pioneer P3-AT a través de un cable RS-232 full-dúplex al puerto auxiliar 2.

3.2.3. COMPUTADORA DE ABORDO

La Acer Spire One es una ultra portátil, la cual ejecutará el programa servidor que controlará la plataforma móvil y realizará la transmisión del video. Es ideal para integrarse al robot debido a su pequeño tamaño y a su bajo consumo (1.1 Ah); además cuenta con la especificaciones técnicas necesarias para realizar su función dentro del sistema, las cuales se enuncian en la Tabla 3-2.

Características	Descripción
Procesador	Intel Atom 1,6 GHz.
Sistema operativo	Windows XP
Memoria RAM	Memoria RAM de 1GB.
Conexiones inalámbricas	Conexiones: 10/100 Mbps ethernet. Conectividad 3G
Dimensiones	248mm x 170mm x 29mm.
Peso	995 g. en su versión más básica.
Consumo	19 VDC 1.58A 30W

Tabla 3-2.- Especificaciones técnicas de la computadora de abordó.

Como se puede ver en la Tabla 3-2, el consumo de la ultra portátil es de 30 W a 19 Volts dc y la plataforma opera únicamente con 12 y 5 volts, por lo que fue necesaria la implementación de un convertor de 12 a 19 Volts, que fue colocado para que la ultra portátil pueda operar, las especificaciones del convertor se ha colocado en la Tabla 3-3.

Características	Descripción
-----------------	-------------

Entrada de voltaje	12 VDC
Entrada de corriente	8 A max
Salida de voltaje	15-24 VDC
Potencia de salida	80 W

Tabla 3-3.- Especificaciones técnicas del convertor de voltaje.

3.2.4. DISPOSICIÓN Y ANÁLISIS DEL HARDWARE PARA LA VERSIÓN 1.1



Figura 3-5.- Versión 1.1 del robot móvil del sistema ROMOTEVI

La principal diferencia entre las dos versiones es la posición de la cámara, pues, en la versión 1.1 ésta se encuentra en la parte frontal de la plataforma, Figura 3-5. Esta ubicación fue escogida debido a que es la posición recomendada por el fabricante.

Como se mencionó en el Capítulo 2, la cámara que se está usando en el sistema ROMOTEVI es una Canon VC-C50i, cuyas características técnicas principales son:

- Conectividad con RS-232C o S-Video.
- Inclinación de 100 grados y un rango de 200 grados de vista panorámica (100 grados a la izquierda y 100 grados a la derecha).
- Campo de visión de 45 grados horizontal y 47 vertical.

Con la colocación de la cámara en la parte frontal y tras la ejecución de pruebas de manejo en la plataforma, se determinaron las siguientes características del diseño:

- Es imposible saber la ubicación de la cámara con respecto al robot, por este motivo, al momento de realizar algún movimiento no se podía determinar la dirección que tomaría la plataforma al moverse. Como se observa en la Figura 3-6, la cámara está apuntando hacia un ángulo a la izquierda pero si se procede a mover la plataforma, ésta se moverá hacia donde indica la flecha roja en la figura.

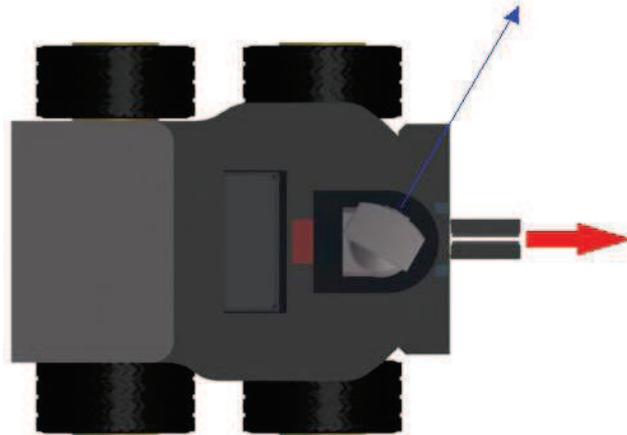


Figura 3-6.- Análisis del posicionamiento de la cámara con respecto al robot móvil

- No se puede determinar el espacio físico que ocupa la plataforma en un espacio determinado pues, como se muestra en la Figura 3-7, todo lo que se encuentra debajo de la línea celeste no se puede ver ya que es obstruido por la placa superior de color negro, impidiendo de esta forma observar la rueda y por ende no se puede apreciar el ancho total del robot.

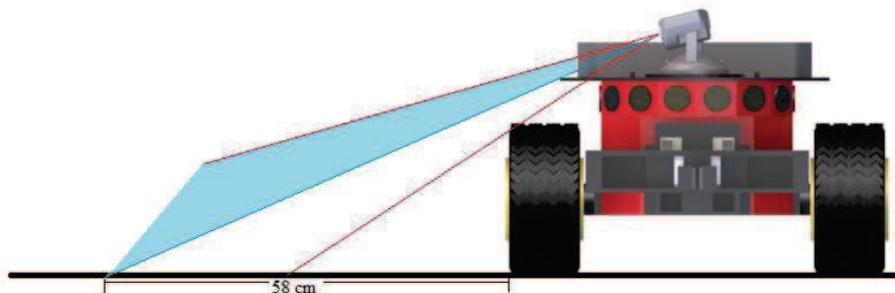


Figura 3-7.- Campo de visión de la cámara con respecto al robot móvil.

Con la consideración mencionada, es importante analizar lo que sucede con la cámara mirando hacia adelante. Como se puede ver en la Figura 3-8, el campo de visión de la cámara no alcanza a ver el gripper que posee el robot, por esta razón trabajar con el gripper para recoger un objeto es muy difícil; además, la cámara capta una imagen plana haciendo difícil apreciar las distancias.

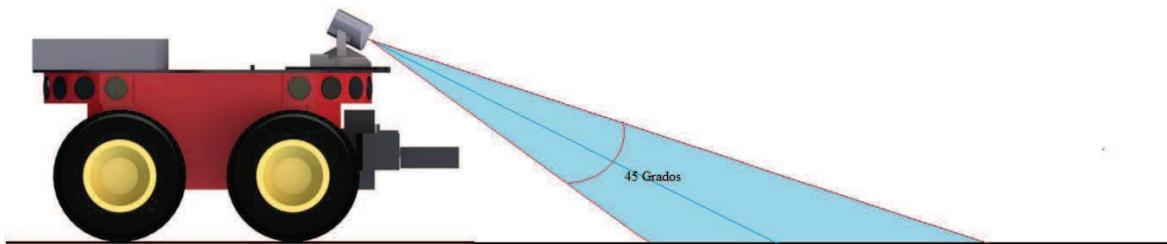


Figura 3-8.- Campo de visión de la cámara apuntando a la parte delantera del robot.

- La cámara tiene un rango de movimiento horizontal de ± 100 grados, los cuales abarcan toda la parte frontal de la plataforma. Como se aprecia en la Figura 3-9, no se puede ver todo lo que está atrás de la cámara, y tomando las explicaciones anteriores se puede afirmar que con la utilización de esta configuración no es posible apreciar ninguna parte del robot.

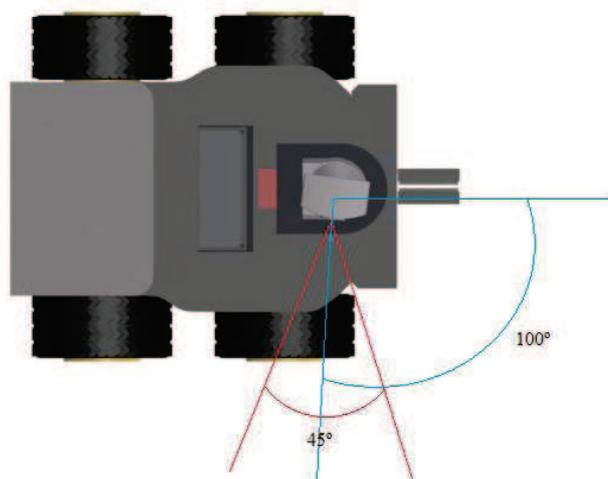


Figura 3-9.- Rango de movimiento de la cámara con respecto al robot.

3.2.5. DISPOSICIÓN Y ANÁLISIS DEL HARDWARE PARA LA VERSIÓN 1.2



Figura 3-10.- Versión 1.2 del robot móvil del sistema ROMOTEVI

Para esta versión se utiliza la cámara colocada en la parte posterior del robot, intentando lograr el mismo efecto que se tiene cuando se maneja un automóvil, como se muestra en la Figura 3-10, tratando de tener una clara visibilidad del frente y los laterales del robot móvil. Adicionalmente, se coloca una marca amarilla en la parte frontal de la plataforma que indica al usuario que se encuentra mirando hacia adelante de la plataforma.

Con la colocación de una marca visual y una previa explicación al usuario de lo que significa dicha marca se pretende eliminar de cierta forma la sensación de desorientación que tenía el usuario con la versión 1.1 del robot móvil, como se muestra en la Figura 3-11, esto debido a que el usuario busca la marca para conocer que está viendo hacia adelante y que esa será la dirección a la que el robot va a desplazarse.

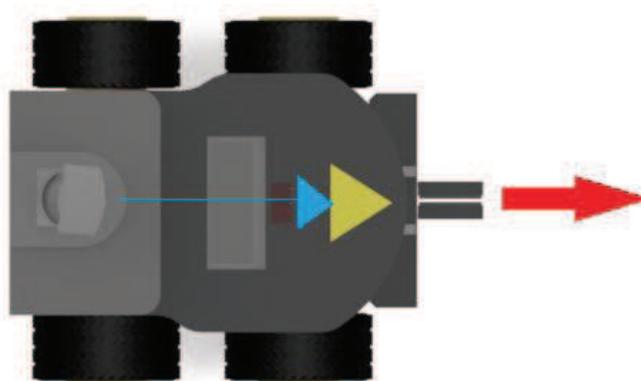


Figura 3-11.- Análisis de la ubicación de la cámara con respecto al robot móvil para la versión 1.2

Con la colocación de la cámara en la parte posterior del robot móvil se logra tener un rango más amplio de visibilidad de la plataforma. Como se puede ver en la Figura 3-12, el rango de movimiento de la cámara es de 200 grados en total, los cuales abarcan casi toda la plataforma, por lo que es posible calcular el espacio que ésta va necesitar para pasar por un sitio cualquiera.

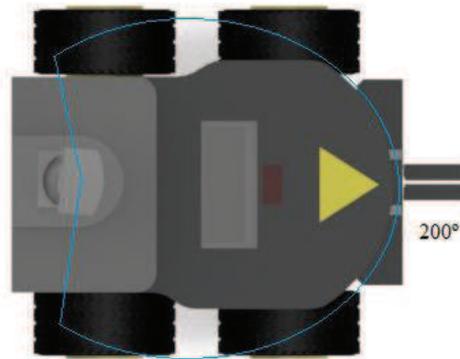


Figura 3-12.- Rangos de movimiento de la cámara para la versión 1.2

Si se observa la Figura 3-13, el espacio resaltado de color celeste indica lo que la cámara puede ver y se aprecia la persistencia del problema que se tenía en la versión 1.1, en la cual no se puede mirar el gripper y tampoco se puede mirar nada por debajo de la línea azul. Contrastando lo que se puede ver con los grados de movimiento que tiene la cámara, con esta configuración se puede mirar los contornos de la plataforma lo que permite calcular los espacios por los que deba podría pasar.

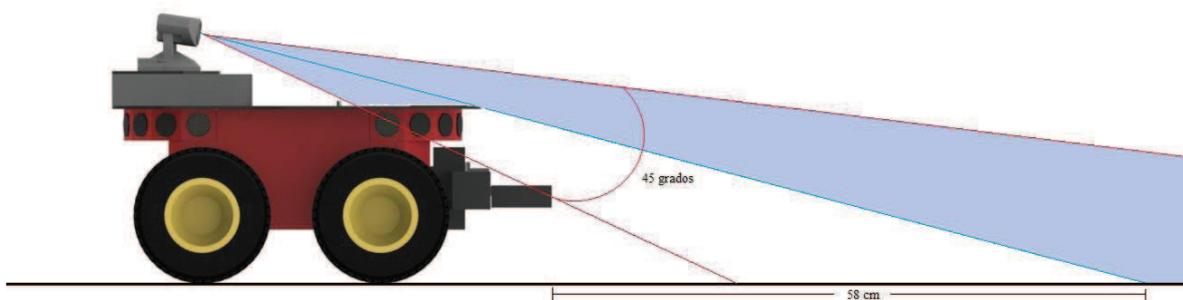


Figura 3-13.- Campo de visión de la cámara apuntando a la parte delantera del robot para la versión 1.2

Debido a que en esta versión se corrigieron muchos problemas de la versión anterior, las pruebas y mediciones del Capítulo 5 se realizaron bajo la versión 1.2 para obtener el mayor grado de presencia e inmersión sensorial posible con el sistema ROMOTEVI.

3.2.6. CONSUMO ENERGÉTICOS DEL ROBOT MÓVIL

La plataforma móvil 3-AT posee un sistema de alimentación de 3 baterías de 12 voltios a 7.5 Ah; estas tres baterías están dispuestas en paralelo lo que le brindan un total de 22 Ah a todo el sistema.

Elemento	Consumo(A)
Cámara de video	0.600
Plataforma con computadora de abordo	5
Motores(2 motores-max. consumo)	10
Gripper	0.300

Figura 3-14.- Tabla del consumo energético del robot móvil.

En la Figura 3-14, se puede apreciar el consumo de cada elemento que conforma el servidor del sistema ROMOTEVI, el cual es de 16 amperios; con este valor se calcula el tiempo máximo de autonomía de la plataforma con la ecuación 1:

$$t. aut = \frac{\text{capacidad bateria}}{\text{consumo maximo}} = \frac{22 Ah}{16 A} = 1.37 h \quad (1)$$

Idealmente, la plataforma debería tener un tiempo de autonomía de 1 hora y 22 minutos, pero al realizar pruebas en el robot móvil se determinó que el tiempo útil es de 45 minutos hasta un máximo de 1 hora. La diferencia de tiempos radica en que la ecuación 1

sirve para una batería ideal, en la cual, el tiempo de descarga es proporcional a la carga ejercida sobre la batería, pero en la realidad el tiempo de descarga de la batería depende de una curva de descarga proporcionada por cada fabricante y del porcentaje de carga que se utilice. El porcentaje actual de carga sobre la batería en la plataforma es de un 62%, además la carga ejercida sobre la batería no es constante, ésta varía el momento de desplazarse, provocando que el tiempo de autonomía se reduzca.

3.3. CLIENTE DEL SISTEMA ROMOTEVI

El cliente del sistema ROMOTEVI se encuentra constituido por tres elementos, los cuales están descritos en la Figura 3-15 y son:

- Casco de realidad virtual eMagin Z800
- Joystick Genius F-23
- Computador cliente.



Figura 3-15.- Elementos que conforman el cliente del sistema ROMOTEVI.

3.3.1. DESCRIPCIÓN DEL CASCO DE REALIDAD VIRTUAL

El eMagin Z800 es un casco de realidad virtual cuyas características brindan a los usuarios inmersión sensorial. Este casco aporta al sistema ROMOTEVI la capacidad de captar los movimientos de la cabeza del usuario y generar un ambiente virtual con el video captado por la cámara del robot.



Figura 3-16.- Casco de realidad virtual eMagin Z800

Para captar los movimientos, el casco consta de varios giroscopios en la parte frontal dispuestos vertical y horizontalmente, éstos son los encargados de captar los movimientos que luego son traducidos a ángulos de desplazamiento. El casco posee un rango de movimiento horizontal de 0 a 360 y verticalmente posee un rango de -60 a 60 grados, para más información refiérase al Anexo 3.

3.3.2. COMPUTADOR CLIENTE DEL SISTEMA ROMOTEVI

Este computador es el encargado de recopilar todas las acciones que se quieren ejecutar en el robot y enviarlas al servidor, para realizar esto se lo ha dotado de dos dispositivos importantes: un casco de realidad virtual, el cual recopilará la información de los movimientos de la cabeza del operador; y, un joystick, el cual recopilará los movimientos que se desean ejecutar en la plataforma. Posteriormente, estos datos, son enviados al servidor a través de comandos que se encuentran declarados en él.

Paralelamente, recibe los datos de video desde el servidor y las muestras en el HMD; las especificaciones técnicas que posee la computadora están enunciadas en la Tabla 3-4.

Característica	Descripción
Procesador	Intel Core 2 Quad Q6600 2.40 GHz
Disco duro	MAXTOR STM 250 Gb

Memoria RAM	3.23 Gb
Sistema operativo	Windows XP Professional SP2

Tabla 3-4.- Datos técnicos del computador cliente.

3.3.3. DESCRIPCIÓN DEL SISTEMA DE COMUNICACIÓN INALÁMBRICO.

El tipo de comunicación que se ha utilizado para teleoperar la plataforma es Wifi con un alcance máximo de 30 a 100 metros en interior y de 100 a 300 metros en aire libre, con línea de vista. La comunicación se la ha realizado a manera de cliente-servidor, como se muestra en la Figura 3-17, donde el computador cliente es el encargado de enviar todas las peticiones de control al computador servidor que se encuentra en la plataforma y que transmite las peticiones del cliente al microprocesador del robot

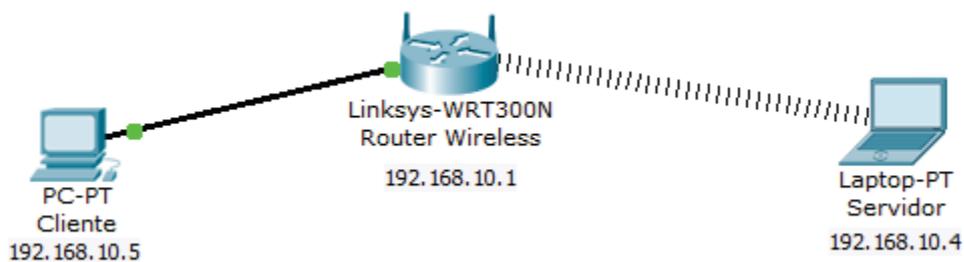


Figura 3-17.- Esquema de la red de comunicación del sistema.

3.4. PRUEBAS DE FUNCIONAMIENTO DEL HMD

Para realizar una prueba adecuada del funcionamiento del casco de realidad virtual eMagin Z800, se ha dividido en dos partes importantes las cuales son:

- La configuración y visualización del escritorio del computador en el casco.
- La ejecución de un programa que muestra los ángulos medidos por el tracking del casco.

Visualización del escritorio en el visor del casco

En esta prueba se verifica el correcto funcionamiento del casco de realidad virtual y la pantalla externa.

Es necesario colocar la resolución de la pantalla en 800 x 600 debido a que el casco trabaja con tecnología VGA. Esto se realiza en el computador cliente: Menú Inicio → Panel de Control → Pantalla → Configuración, aquí se modifica la resolución de la pantalla a 800x600; si todo está correcto el escritorio debería mostrarse tanto en la pantalla externa como en el visor del casco.

Ejecución del programa de prueba

Para esta parte de la prueba se requiere instalar el controlador del casco, el cual se encuentra en la carpeta “Driver” del cd adjunto. Posterior a la instalación, se ejecutará el programa Test_casco.exe. Este programa se encargará de verificar si el casco está correctamente conectado y si el computador cliente lo ha reconocido. Si todo esto es correcto, debajo del “Estado del Casco” se mostrará “Casco Iniciado” de lo contrario se mostrara “Casco no Iniciado”, tal como se muestra en la Figura 3-18.

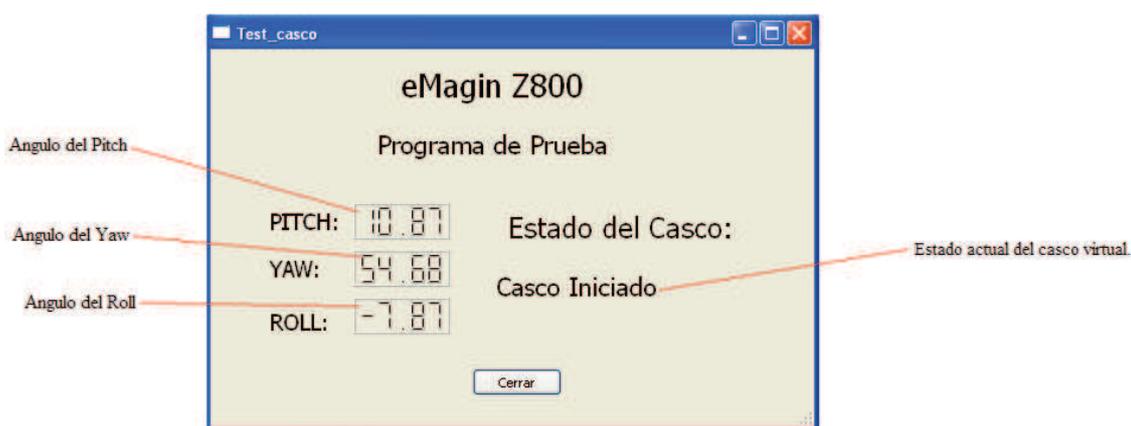


Figura 3-18.- Interfaz gráfica del programa de prueba para el casco de realidad virtual.

Una vez establecida la conexión del casco al computador, se empezará a mostrar los ángulos medidos por el rastreador del casco, que medirá los ángulos en el que se encuentra la cabeza.

3.5. PRUEBAS DE FUNCIONAMIENTO DEL SISTEMA DE COMUNICACIÓN INALÁMBRICA.

Esta prueba de funcionamiento pretende verificar la correcta conexión y funcionamiento de todo el sistema de comunicación inalámbrico de la plataforma móvil, para esto se ha elaborado un programa que verificará que el robot se encuentre conectado al computador servidor y a su vez verificará que el computador cliente esté enlazado a él.

Previo a la ejecución del programa de prueba se debe configurar las direcciones de red tal como se muestra en la Figura 3-5.

	Dirección de red	Máscara de subred	Puerta de salida
Servidor	192.168.10.5	255.255.255.0	192.168.10.1
Cliente	192.168.10.4	255.255.255.0	192.168.10.1

Tabla 3-5.- Configuración de red del sistema de comunicación inalámbrico.

El programa de prueba posee 4 etapas, las cuales verifican:

- Si el servidor se encuentra en la red
- Si el cliente puede conectarse al servidor
- Si se puede escribir y leer datos desde el servidor; y,
- Si en el servidor se encuentra conectada la plataforma móvil.

```

c:\Documents and Settings\ST-2000\Mis documentos\Visual St
Conectando al Servidor
Cliente conectado al servidor.
Servidor dice: "Hola Cliente"
Cliente dice: Hola Servidor.
Servidor dice: "Robot Conectado"
Socket del servidor cerrado.
  
```

Figura 3-19.- Programa de prueba para la comunicación inalámbrica de la plataforma.

En la Figura 3-19 se encuentra toda la conversación que hace el cliente con el servidor y notifica si la prueba se realizó con éxito. Por ejemplo, si notifica que el cliente está conectado al servidor, quiere decir que el servidor está en red y que la conexión se la pudo realizar. Si el servidor saluda al cliente, quiere decir que el servidor puede escribir datos al cliente, de igual manera, para la parte del cliente. Por último, indica si el robot está o no conectado al servidor.

En el Tabla 3-6 se encuentra un resumen de las pruebas que se realizaron para la comprobación del correcto funcionamiento del sistema, donde se explica cuando falla y las posibles soluciones para corregir dichas fallas.

Pruebas	Respuestas	Estado	Observaciones
Casco de realidad virtual			
Interfaz del programa de prueba	Casco iniciado	Correcto	
	Casco no iniciado	Fallo	Revisar la conexión y reinstalar driver
Comunicación con la plataforma			
Comunicación de red (Consola del programa de prueba)	Cliente conectado	Correcto	
	Error conectando al servidor	Fallo	Revisar la conexión de red y firewall del servidor
	Servidor dice:.....	Correcto	
	Error leyendo al servidor	Fallo	Firewall del cliente
	Cliente dice: "Hola Servidor"	Correcto	

	Error escribiendo al servidor	Fallo	Firewall del servidor
Comunicación serial (Consola del programa de prueba)	Robot conectado	Correcto	
	Robot desconectado	Fallo	Revisar la conexión serial del robot

Tabla 3-6. Tabla de registro de las pruebas del sistema.

CAPÍTULO 4

IMPLEMENTACIÓN DEL SOFTWARE DEL SISTEMA

4.1. DESCRIPCIÓN DEL SOFTWARE DEL SISTEMA

El sistema de teleoperación ha sido concebido en una arquitectura cliente-servidor, como se puede apreciar en la Figura 4-1, y es así como se estructuró el software; además se separó la parte de control del robot y el sistema de video para que éstas sean paralelas. La interacción de las librerías del software de video se encuentra representada en la Figura 4-2.

En este apartado se describirán las librerías utilizadas para desarrollar los programas, además se detallará su funcionamiento en base sus diagramas UML de clase.

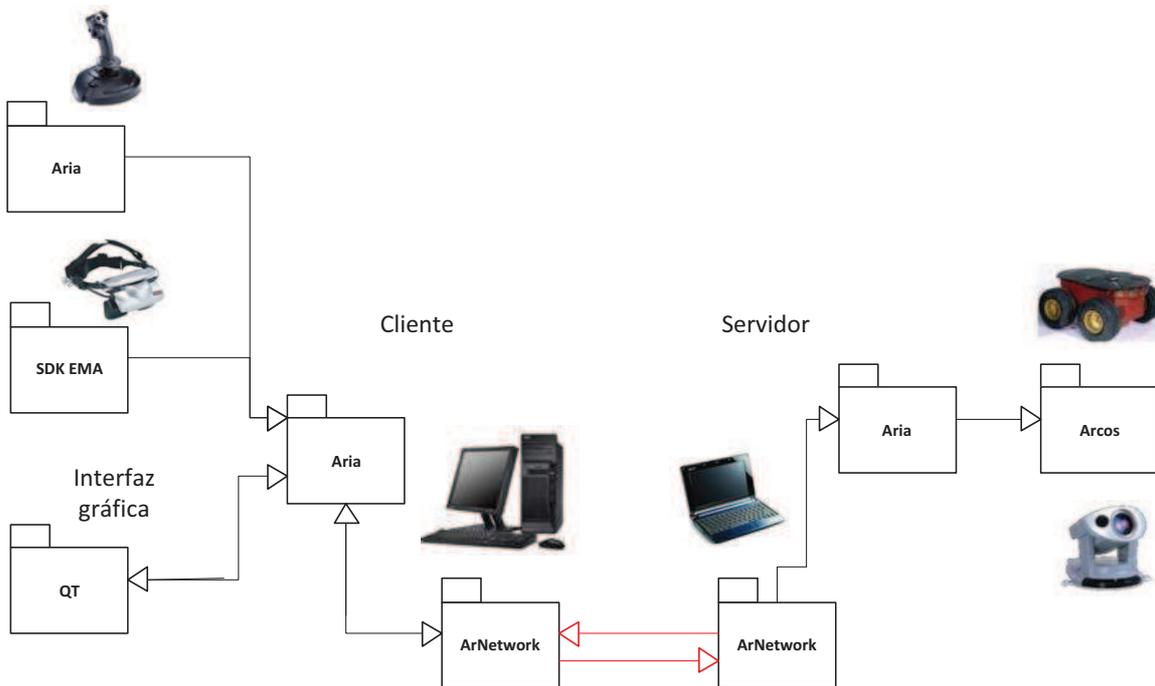


Figura 4-1.- Interacción de librerías para el sistema de control.

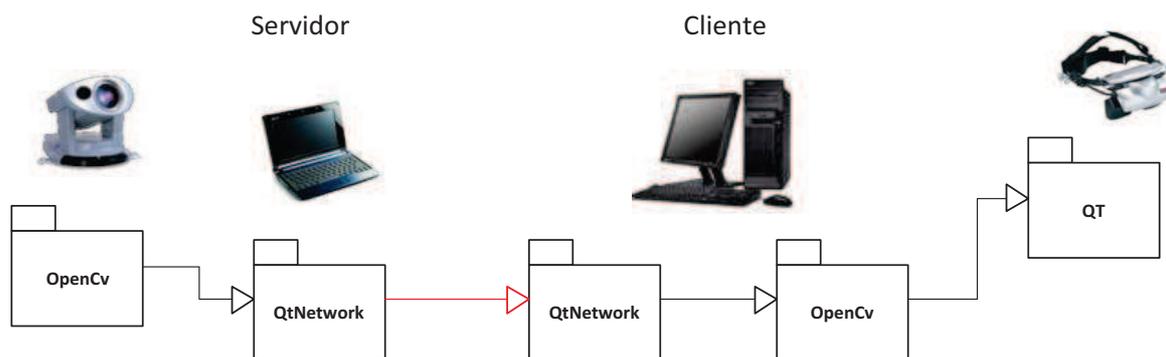


Figura 4-2.-Interacción de librerías para el sistema de video

4.1.1. LIBRERÍAS UTILIZADAS EN EL SOFTWARE DEL SISTEMA

Para el desarrollo del presente sistema se ha utilizado básicamente 4 librerías, las cuales son:

- ARIA, desarrollada por MobileRobots, la cual es la encargada de realizar la conexión del servidor al microprocesador de la plataforma móvil, además de administrar y manejar todos los recursos del robot.
- ArNetwork, igualmente desarrollado por MobileRobots. Es el encargado de gestionar la comunicación TCP/IP de la plataforma y es capaz de implementar clientes y servidores para comunicar al robot con otro computador en la red.
- OpenCV, una librería desarrollada por Intel para procesamiento de imágenes. En este sistema se encarga de captar el video de la cámara integrada en el robot y presentarlo en el cliente.
- Qt, que es un grupo de librerías utilizadas para desarrollar interfaces gráficas mediante el uso de gadgets. Cuenta también con librerías para desarrollar aplicaciones de red, manejo hilos y otras funcionalidades mencionadas más adelante. Dentro del sistema es utilizado para crear la interfaz gráfica y realizar el streaming de video.

4.2. DESCRIPCIÓN DEL SDK DEL CASCO DE REALIDAD VIRTUAL

El SDK del casco de realidad virtual eMagin Z800 está dividido en dos interfaces: una de alto nivel y otra de bajo nivel, las cuales se utilizan dependiendo del tipo de aplicación que se desea hacer.

La interfaz de bajo nivel es llamada EMADevice, la cual se encarga de comunicarse directamente con el driver y posee funciones básicas para controlar el casco de realidad virtual como son:

- Conectar o desconectar el casco
- Configurar los parámetros de visualización del casco, como brillo y color.
- Controlar el activado del modo 3D o retomar el modo 2D.
- Controlar el tiempo del protector de pantalla.
- Recopilar datos de velocidad angular, aceleración y posición.
- Funcionalidad para controlar el cursor con el casco.

Esta librería también posee métodos que permiten la inicialización y la finalización del rastreador del casco cuando un programa arranca o termina, lo que permite tener un gran control sobre el mismo.

La estructura total de la librería se encuentra descrita en la Figura 4-3, donde se aprecian las capas de alto y bajo nivel comunicadas entre sí; la capa de bajo nivel a su vez controla el casco a través de la capa de comunicación del casco de realidad virtual.

También, en la Figura 4-3, se puede observar parte de la programación que se utilizó para este proyecto, puesto que la aplicación no necesariamente tiene que pasar por la capa de alto nivel para llegar a la de bajo nivel, sino que puede prescindirse de ella y trabajar directamente con la capa de bajo nivel, lo cual fue aplicado en este caso. Para mayor información, consúltese la referencia de programación de la librería EMADevice o el programa de ejemplo adjuntos a la tesis.

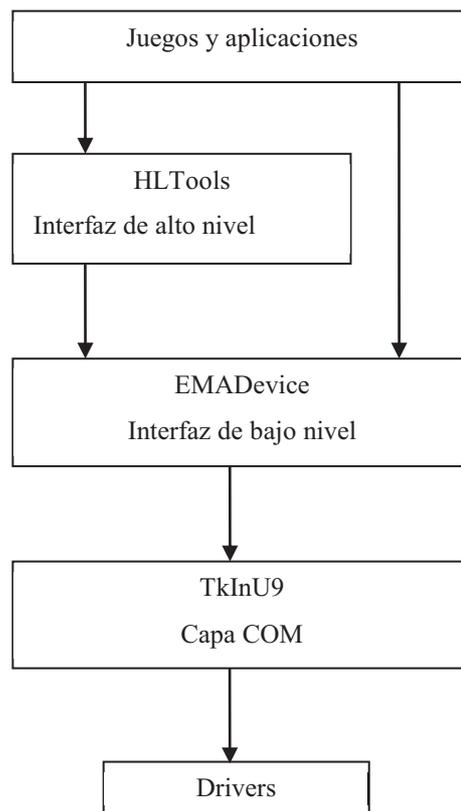


Figura 4-3.- Estructura del SDK de EMagin

4.2.1. CREACIÓN DE LA DLL REA

La librería dinámica Rea fue creada con el fin de realizar una conexión del SDK de EMADevice con la librería Aria de la plataforma móvil Pioneer P3-AT. Dentro de esta librería se usaron cuatro funciones básicas de EMADevice:

InitializeHeadset.- Inicializa el casco de realidad virtual y devuelve un entero con valor 0 si no se inicializó, o un valor de 1 si se inicializó el casco.

DeInitializeHeadset.- Desactiva el casco de realidad virtual.

MantenerVivo.- Es una función que se utiliza para que el casco no entre en suspensión, además le indica al casco que está siendo usado.

GetHeadsetData.- Es una función que se utiliza para obtener los valores pitch, yaw y roll del casco de realidad virtual. Esta función requiere la instalación de DirectX 9.0.

La librería dinámica Rea es llamada por el programa cliente cuando éste desea inicializar el casco y obtener los datos que el casco está captando para posteriormente indicar los ángulos que debe moverse la cámara de video. Para más información, referirse a la librería Rea.

4.3. DESCRIPCIÓN DE LA LIBRERÍA ARIA

ARIA (Interfaz robótica avanzada para aplicaciones) es una biblioteca desarrollada en C++, la cual es utilizada en todas las plataformas MobileRobots. ARIA se encarga de manejar todos los detalles de bajo nivel de la plataforma, como por ejemplo: controlar la velocidad del robot, la partida, la partida relativa, y otros parámetros de movimiento. También recibe estimaciones de la posición, las lecturas de sonar, y todos los datos actuales de operación enviados por la plataforma móvil, evitando de esta manera que el programador se preocupe de estos detalles y se concentre en las tareas que la plataforma realizará.

ARIA proporciona herramientas para la integración de E/S con el hardware digital, analógicos y de comunicación serial RS 232C, e incluye soporte completo para todos los accesorios de los robots móviles, entre ellos los láser telemetro, cámaras pan-tilt-zoom, pinzas y GPS.

Aria es un API que se encuentra estructurada como se muestra en la Figura 4-4, sin embargo contiene diferentes librerías que sirven de base para incrementar capacidades adicionales, como rutinas de navegación avanzadas, las cuales son desarrolladas con librerías de ARNL, síntesis y reconocimiento de voz, reproducción de efectos de sonido, funciones matemáticas, implementaciones de hilos y manejo de red bajo protocolos UDP y TCP en la librería ArNetwork, que están disponibles como bibliotecas independientes.



Figura 4-4.- Estructura de ArRobot de Aria

Si bien, como se mencionó anteriormente, ARIA está escrita en lenguaje C++, la mayor parte de la API también está disponible en java y python. Es compatible con GNU Linux usando G++ (3.4 o posterior), pese a que se compila previamente con el compilador por defecto de la plataforma, puede re-compilarse con cualquier otra versión de G++ (superiores a 3.4) y en Windows con MS Visual C++ NET. 2003 (7.1), Visual C++ 2008 (9,0) y Visual C++ 2010 (10.0).

4.4. LIBRERÍA ARNETWORK

La biblioteca ArNetworking es la capa más alta de comunicación que posee MobileRobots, trabaja bajo la modalidad de cliente-servidor en donde el servidor se encuentra implementado en un computador de abordo dentro de la plataforma. Por otro lado, el cliente se encuentra implementado en un computador externo. Ambos están enlazados entre sí por cualquier tipo de red, lo cual crea un marco completo para recibir actualizaciones de datos regulares del servidor al cliente y enviar comandos desde el cliente al servidor.

El servidor posee varios servicios ya implementados que van a ser reflejo de un comando para el robot, como por ejemplo la conducción, y a su vez existen comandos para conseguir diferentes tipos de información de regreso, como por ejemplo la odometría de la plataforma.

ArNetwork utiliza un formato de paquete similar al formato de los paquetes del robot, transmitido a través de sockets TCP o UDP.

4.5. LIBRERÍA OPENCV

OpenCV (Open source computer vision library) es una librería gratuita desarrollada por Intel. Esta librería proporciona un alto nivel de funciones para el procesamiento de imágenes y permiten a los programadores crear aplicaciones poderosas en el dominio de la visión digital.

OpenCV es una librería opensource y multiplataforma que implementa una gran variedad de herramientas para la interpretación de la imagen. Es compatible con Intel Image Processing Library (IPL) que realiza operaciones en imágenes digitales. OpenCV es principalmente una librería que contiene algoritmos para las técnicas de la calibración (calibración de la cámara), detección de rasgos, para rastrear (flujo óptico), análisis de la forma (geometría, contorno que procesa), análisis del movimiento (plantillas del movimiento, estimadores), reconstrucción 3D (transformación de vistas), segmentación de objetos y reconocimiento (histograma, etc.).

4.6. LIBRERÍA QT

Qt es una biblioteca multiplataforma ampliamente usada para desarrollar aplicaciones con una interfaz gráfica de usuario, así como también para el desarrollo de programas sin interfaz gráfica, como herramientas para la línea de comandos y consolas para servidores.

Qt utiliza el lenguaje de programación C++ de forma nativa, adicionalmente puede ser utilizado en varios otros lenguajes de programación a través de *bindings*. También es usada en sistemas informáticos empotrados para automoción, aeronavegación y aparatos domésticos como frigoríficos.

Distribuida bajo los términos de General Public License (y otras), Qt es software libre y de código abierto. Es desarrollada por la división de software Qt de Nokia y es completamente gratuita.

4.7. SOFTWARE DE CONTROL DEL SERVIDOR ROMOTEVI

Como se mencionó en la sección 4.1, el software del robot está dividido en dos partes: control y visión, y éstas a su vez se encuentran bajo una estructura cliente-servidor. En este apartado se tratará el servidor de control del robot, su estructura y funcionamiento.

4.7.1. ESTRUCTURA DEL SERVIDOR DE CONTROL

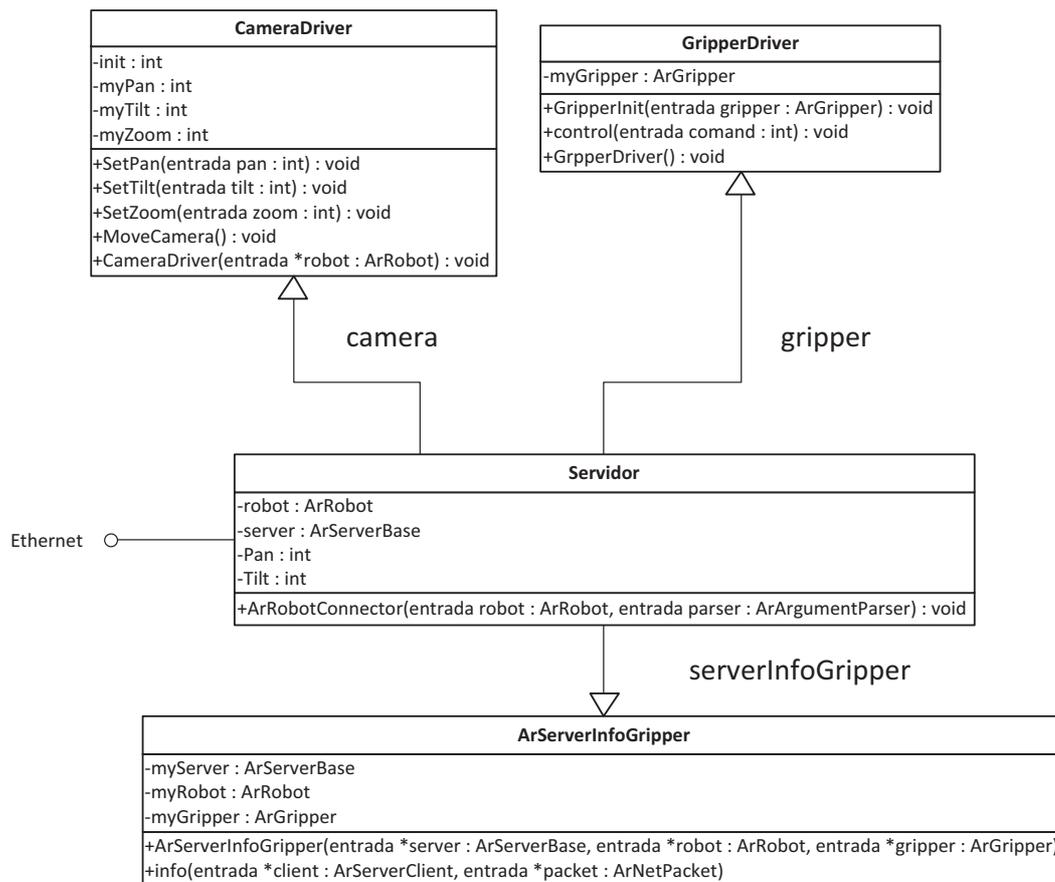


Figura 4-5.- Diagrama UML de clases para el servidor ROMOTEVI

El servidor de control se encuentra conformado por un núcleo en el cual están las librerías de Aria para controlar el robot y ArNetworking para administrar la conexión de red. En la Figura 4-5 se muestra un diagrama de clases con la interacción de las mismas con el núcleo llamado servidor. A su vez el núcleo es complementado con dos clases adicionales:

- La clase *CameraDriver*, la cual controla el movimiento de pan y tilt de la cámara del robot.
- La clase *GripperDriver*, encargada de manejar el gripper.

Adicionalmente, existe la clase *ArServerInfoGripper* que se encarga de realimentar datos sobre el estado del gripper.

CameraDriver, tiene tres funciones:

- *SetPan*, que ingresa el valor del Pan que se quiere mover en la cámara
- *SetTilt*, de igual manera ingresa el valor del tilt que se quiere mover la cámara
- *MoveCamera*, la cual ejecuta los movimientos antes ingresados por las otras dos funciones.

GripperDriver, tiene como funciones:

- *Initgripper*, la cual inicializa el objeto perteneciente al gripper del robot
- *Control*, que es una función a la cual se le envía un valor entero entre 2 y 5 que representan las distintas acciones que puede ejecutar el gripper como son:
 - 2 para subir
 - 3 para bajar
 - 4 para cerrar
 - 5 para abrir

Para manejar las clases antes mencionadas se declaran comandos en el servidor que representan servicios que se brindan al cliente. Los comandos utilizados en el servidor se encuentran en la Tabla 4-1; estos comandos se clasifican en dos tipos: comandos implícitos en el núcleo, como es *ModeRatioDriver*; y, comandos personalizados, los cuales son creados como funciones en el servidor y relacionados con un nombre de comando que ejecuta esa función.

Dentro del servidor se crearon cuatro comandos personalizados los cuales son:

- *ComandoGripper*, el cual ejecuta la función control de la clase *GripperDriver*.
- *PanCamara*, ejecuta la función *SetPan* de la clase *CameraDriver*.
- *TiltCamara*, ejecuta la función *SetTilt* de la clase *CameraDriver*.
- *MoveCamara*, ejecuta la función *MoveCamera* de la clase *CameraDriver*.

También se creó el comando *ServerInfoGripper*, pero se lo hizo de tal forma que sea un comando implícito en el núcleo (véase la clase *ArServerInfoGripper* para más información), debido a que los comandos personalizados solo reciben datos del cliente y no permiten enviárselos. Por este motivo se creó este comando para enviar datos al cliente sobre el estado del gripper en el robot. Adicionalmente, se usan dos comandos ya establecidos en el robot los cuales son:

- *ServerInfoDriver*, el cual envía al cliente datos que tienen que ver con el robot como son: valores de batería, movimiento en X y Y con respecto al punto en el que fue encendido el robot, velocidades de traslación, rotación, y dirección actual del robot, también tomada con referencia al punto donde fue encendido el mismo.
- *ModeRatioDriver*, el cual recepta valores enteros con signo de la velocidad de traslación y rotación que reflejan los movimientos del robot; por ejemplo si la velocidad de traslación es positiva el robot va hacia delante, si es negativa se dirige hacia atrás, de igual forma con la velocidad de rotación: si es positiva gira a la derecha, si es negativa gira a la izquierda.

Control	Comando	Resumen
Robot	ModeRatioDriver	Movimiento del robot
Cámara PTZ	PanCamara,	Envió del valor de pan
	TiltCamara	Envió del valor del tilt
	MoveCamara	Ejecución del movimiento
Griper	ComandoGriper	Envió del botón presionado en el joystick

Tabla 4-1.- Comandos utilizados en el servidor.

Creación de un comando personalizado

Para crear un comando personalizado en el servidor, en primer lugar es necesario crear una función que será el código que se ejecutará cuando dicho comando sea llamado por el cliente. A continuación, se describe el ejemplo de una función llamada *Funcion_personalizada*, la cual al ser llamada ejecuta una función del objeto *camera*, enviando el parámetro recibido desde el cliente a esta función.

```
void Funcion_personalizada (ArArgumentBuilder *args)
{
    //Se agrega el código deseado por ejemplo el envío del dato a una
    función de un objeto camera.;
    if (args && args->getArg(0)) {
        int tilt=(int)args->getArgDouble(0);
        camera->SetTilt (tilt);
    }else{
        ArLog::log (ArLog::Normal, "Comando recibido sin dato");
    }
}
```

Después de crear la función que ejecutará el comando, se debe crear el evento que ejecutará a esa función. Para ello se llama a las funciones *Functor*, las cuales son de varios tipos y constituyen un manejador de eventos. En este ejemplo se va a manejar al *Functor ArGlobalFunctor1*, el cual puede manejar un parámetro de recepción y tiene el nombre de *FuncionPersonalizadaFunctor*, la cual llama a *Funcion_personalizada*.

```
ArGlobalFunctor1<ArArgumentBuilder*>
FuncionPersonalizadaFunctor (&Funcion_personalizada);
```

Por último, se crea un administrador de comandos personalizados llamado *commands*, el cual tiene como parámetro el servidor; luego se otorga un nombre de comando al *Functor* previamente creado el cual será llamado por el cliente. Para mayor información refiérase a la documentación del programa *Servidor_Control* en *Servidor.cpp*

```
ArServerHandlerCommands commands (&server);

commands.addStringCommand ("ComandoPersonalizado", "Ejemplo de comando
Personalizado", & FuncionPersonalizadaFunctor);
```

Creación de un comando implícito

Para la creación de comandos implícitos que devuelvan datos al cliente es necesario crear un clase nueva, dicha clase estará formada por funciones AREXPORT. Este tipo de funciones tienen la capacidad de ser exportadas por ARIA. Para inicializar dichas clases se lo hace como el ejemplo a continuación:

```
ArServerInfoGripper serverInfoGripper(&server, &robot, &gripper);
```

Esta inicialización de ejemplo lleva parámetros del server, como la conexión actual del cliente, los comandos y otros aspectos del servidor, también un parámetro del robot como la conexión, y un parámetro del gripper para controlarlo en el robot. Adicionalmente dentro del constructor de la clase se agrega el nombre del comando y la función que va a ejecutar éste. En el ejemplo colocado a continuación se demuestra cómo agregar el comando al servidor, donde *myServer* es una copia del servidor, en la clase el primer parámetro es el nombre del comando, el segundo parámetro es una descripción de lo que hace; y, el tercer parámetro es un objeto que contiene al función a ejecutar.

```
myServer->addData("InfoGripper", "Obtiene el estatus del Gripper",  
                &myinfoCB, "none", "byte2: infGrip", "RobotInfo",  
                "RETURN_SINGLE");
```

Para más información sobre cómo construir esta clase refiérase a la documentación del programa *Servidor_Control* en la clase *ArServerInfoGripper*, en ella se encuentra implementado un comando para obtener datos del estado del griper.

4.7.2. FUNCIONAMIENTO DEL SERVIDOR DE CONTROL

En este apartado se explicará el algoritmo de funcionamiento del servidor en base a diagramas de flujo, de esta forma se podrá generar un idea más clara de cómo está

trabajando el programa sobre el robot y el cliente. En la Figura 4-6 se tiene la primera etapa de funcionamiento del servidor.



Figura 4-6.- Diagrama de flujo para el servidor ROMOTEVI (parte 1)

Cuando el servidor inicia, su primera acción es establecer comunicación con el robot e inicializar todos los comandos que va a ejecutar; después espera por una conexión desde el cliente y permanece allí hasta tener una.

Una vez establecida la conexión entra a una etapa de recepción de los paquetes desde el cliente, tal como se muestra en la Figura 4-7.

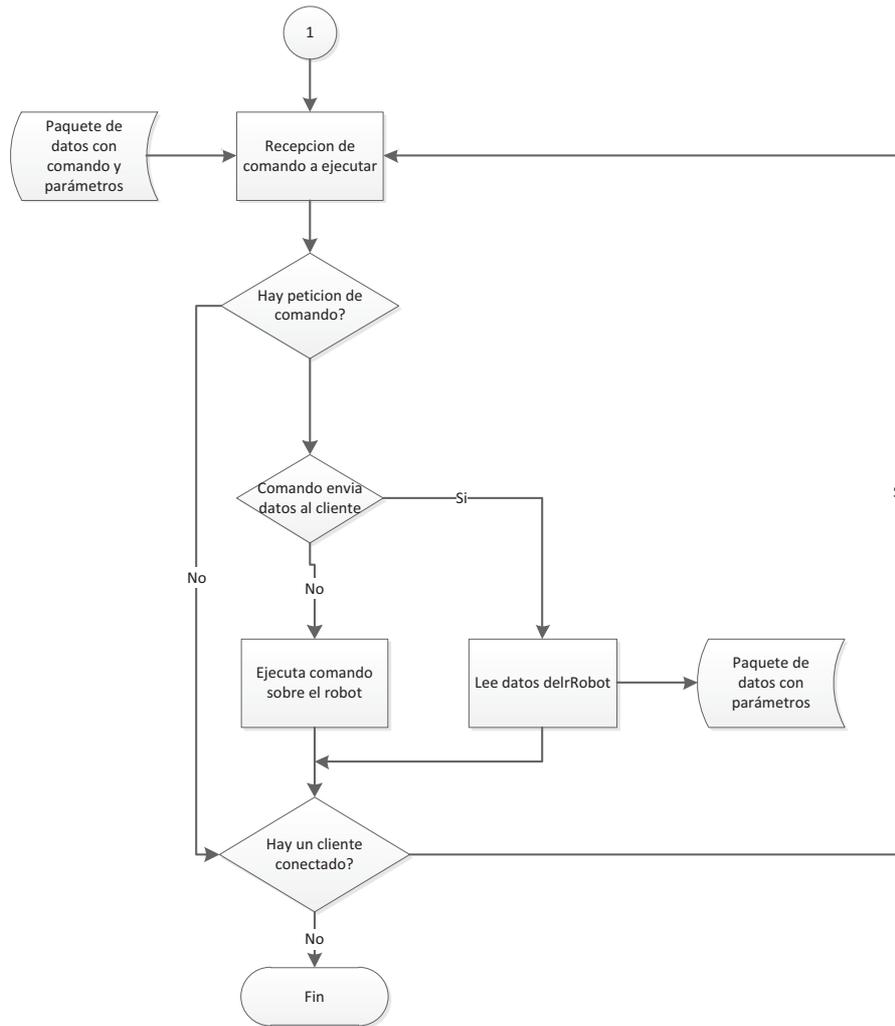


Figura 4-7.- Diagrama de flujo para el servidor ROMOTEVI (parte 2)

Cuando un paquete llegó al servidor se evalúa si éste contiene una petición de comando o no; si contiene una petición de comando dicho comando es analizado para ver si devuelve o no datos al cliente. En el caso de que devuelva datos al cliente, el servidor lee los datos necesarios desde el robot y los envía al cliente.

En caso de no regresar datos al cliente, el servidor toma los parámetros recibidos en el paquete y los envía al comando que se quiere ejecutar, al finalizar cualquiera de las dos opciones posibles verifica si el cliente aún está conectado, si aún lo está regresa a leer el paquete de datos, en caso contrario finaliza y cierra el servidor.

4.8. SOFTWARE CLIENTE DEL SISTEMA ROMOTEVI

El cliente se ejecuta en una localidad remota distinta a donde el robot se encuentra. Su función es recopilar datos de los movimientos de la cabeza del operador y del joystick para enviarlos a la cámara PTZ y al robot respectivamente.

4.8.1. ESTRUCTURA DEL CLIENTE ROMOTEVI

Al igual que el servidor, el cliente posee un núcleo conformado por las librerías ARIA y ArNetwork, en este caso ARIA es la encargada de manejar el joystick y ArNetwork es el encargado de administrar la conexión de red hacia el servidor. Todas las clases que se relacionan con el núcleo del cliente se encuentran representadas en la Figura 4-8 y son:

- *hmdDrive* es la clase encargada de la administración del casco de realidad virtual; esta usa a su vez la librería dinámica Rea (descrita en el apartado 4.2.1), creada para realizar la comunicación entre el cliente y el SDK del casco de realidad virtual. De la librería Rea se obtienen tres datos del casco: pitch, yaw y roll; estos datos son enviados al núcleo para posteriormente ser colocados en la clase *InputHandler*.
- La clase *JoyDriver*, tiene la función de obtener los valores horizontales y verticales del eje de movimiento del joystick. Los datos son obtenidos solo si el botón de disparo del joystick se encuentra presionado, de lo contrario devuelve 0 en estos dos parámetros.

Esta clase también entrega un entero llamado *comando*, que no es más que el número del botón presionado en el joystick aparte del botón de disparo; dicho comando se utiliza para realizar los distintos movimientos del gripper. Los valores que retorna esta clase son almacenados en el núcleo en las variables *myrobotrot*, *myrobotrans* y *comando*.

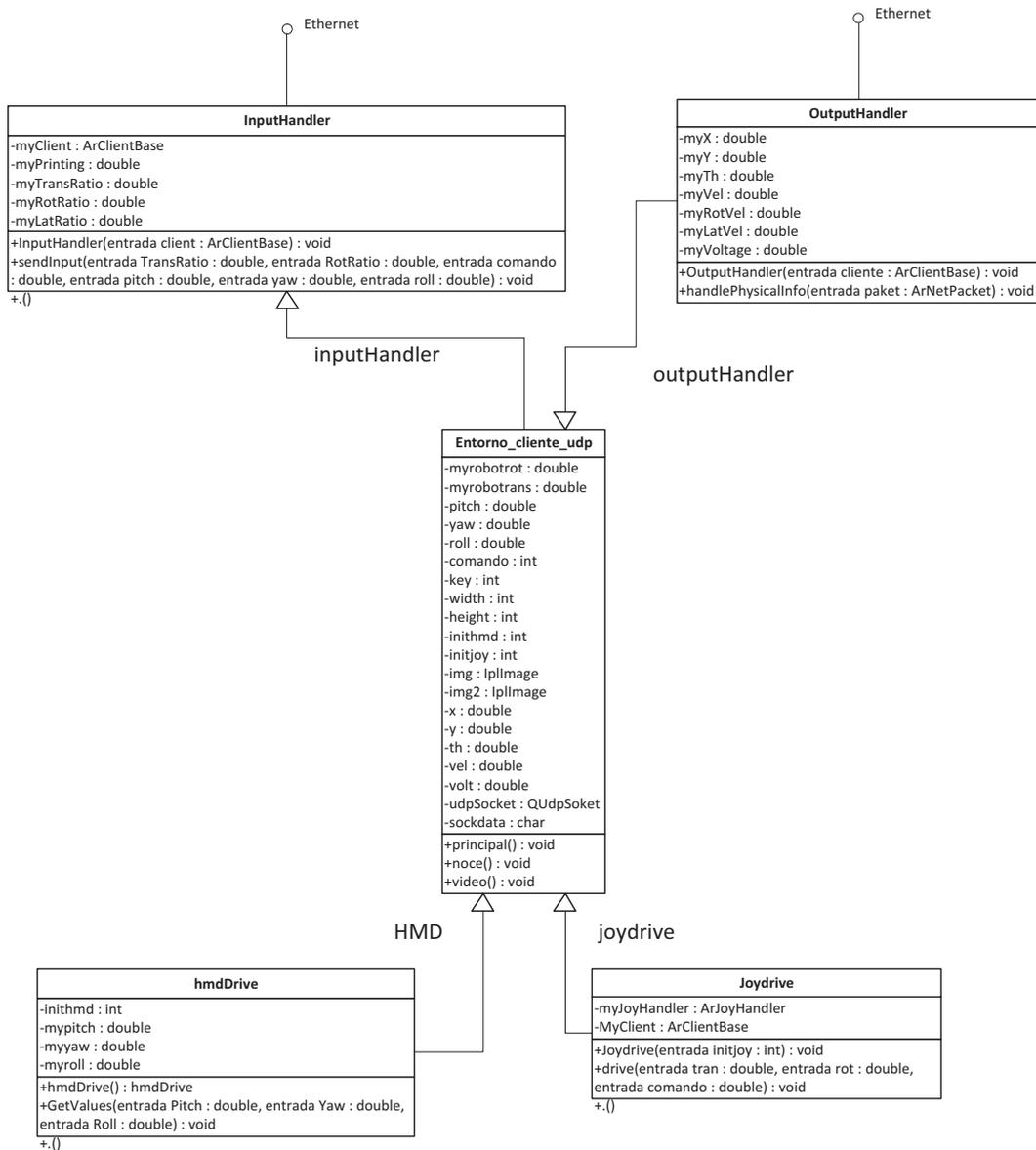


Figura 4-8.- Diagrama UML de clases del cliente ROMOTEVI.

En la Figura 4.8 se puede observar que existen dos clases que tienen conexión directa a la interfaz ethernet, *InputHandler* y *OutputHandler*. Estas dos clases usan la librería *ArNetworking* y son usadas para el envío y recepción de datos, respectivamente, desde el cliente.

- La clase *OutputHandler* recibe los parámetros de odometría del robot como son: velocidades de traslación, giro, estado de la batería y posición en X,Y con relación al punto de partida. Adicionalmente, recibe datos sobre el estado del gripper, por ejemplo si está abierto o cerrado y si está arriba o abajo. Cabe mencionar que esta clase se encuentra

en un hilo aparte del hilo principal que maneja el cliente, y es debido a esto que todos los datos se actualizan cada 100ms.

- La clase *InputHandler* es la encargada de llamar los comandos de movimiento de la cámara, el robot y el gripper en el servidor. Esta llamada se hace mediante un temporizador en el núcleo, el cual ejecuta la función de envío en la clase *InputHandler*. Los datos son enviados cada 10 ms.

4.8.2. FUNCIONAMIENTO DEL CLIENTE ROMOTEVI

El funcionamiento general del cliente se encuentra descrito en el diagrama de flujo en la Figura 4-9. El programa, en primer lugar, inicializa el casco de realidad virtual y el joystick; posteriormente inicia la conexión hacia el servidor y establece los tiempos en los que se harán la actualización de los datos y el envío de datos de los distintos comandos al mismo.

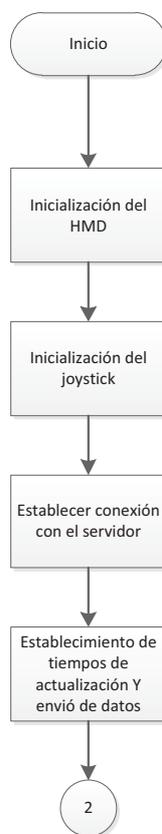


Figura 4-9.- Diagrama de flujo del cliente ROMOTEVI (Parte 1).

Después de culminar la inicialización de los dispositivos y establecer conexión con el servidor, el cliente ROMOTEVI valida si está enlazado, de no ser así finaliza el programa. Como se mencionó en el apartado 4.5.1, la actualización de datos se hace en un hilo diferente al del cliente, por ello en la Figura 4-9 se puede ver que al estar conectado el cliente al servidor surgen dos líneas diferentes de ejecución:

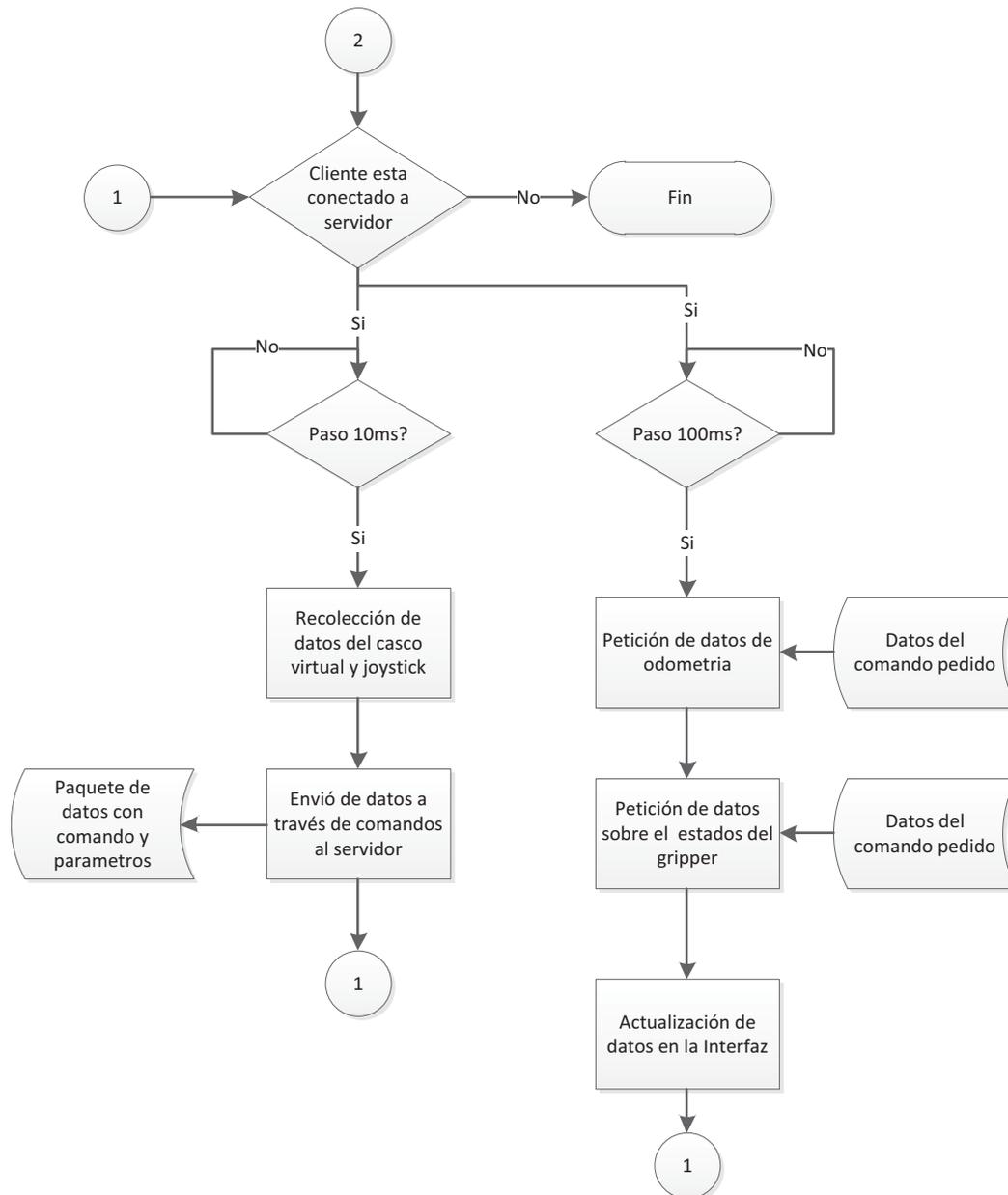


Figura 4-10.- Diagrama de flujo del cliente ROMOTEVI (Parte 2).

- El hilo que actualiza los datos cada 100 milisegundos primero pide los datos de odometría, luego hace la petición de los datos del gripper y por último actualiza todos los datos obtenidos en la interfaz gráfica.
- El hilo que envía los datos del casco y joystick primero verifica que hayan transcurrido 10 milisegundos, luego recolecta los datos de los dos dispositivos antes mencionados y finalmente envía los datos a través de los comandos en el servidor.

Al finalizar los dos hilos vuelven a verificar si el cliente esta enlazado al servidor y continua el ciclo.

4.9. SOFTWARE DEL SERVIDOR DE VIDEO ROMOTEVI

El servidor de video toma un cuadro de la cámara colocada en el robot y lo envía a través de la red al cliente de video ROMOTEVI. Cabe recalcar que el servidor de video ROMOTEVI envía un cuadro cada 30 ms, esto hace que la transmisión sea de 33 fps, la transmisión de video no se lo hace directamente al cliente sino como broadcast.

4.9.1. ESTRUCTURA DEL SERVIDOR DE VIDEO.

El servidor de video es un programa bastante simple por lo que solo posee una clase en la cual se toma el cuadro de la cámara se lo ingresa en el datagrama, ya que la transmisión se la hace mediante el uso del protocolo UDP/IP, y se lo envía a la red por broadcast.

Esta clase usa la librería OpenCv para capturar la imagen desde la cámara y posteriormente usa la librería QtNetwork para enviar el cuadro. La estructura de la clase está representada en la Figura 4-11.

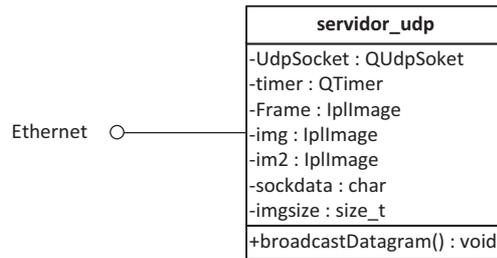


Figura 4-11.- Diagrama UML del servidor de video.

Conversión de un IplImage a datos.

Un IplImage es un dato que contiene toda la información para construir la imagen tomada desde la cámara. Este dato después de ser tomado debe ser reducido a una resolución de 160X120 pixeles, debido a que el datagrama no maneja mucha memoria. Posteriormente, se copian los datos de la imagen al datagrama y se los envía por la red. Para la copia de los datos se usa una función llamada *memcpy* de la librería OpenCv, como se muestra en el siguiente ejemplo:

```

frame = cvQueryFrame( capture );//Toma la imagen.
cvResize(frame,img,CV_INTER_LINEAR);// Redimensiona la Imagen
memcpy(datagram.data(),img->imageData,160*120*3);//Copia los datos
udpSocket->writeDatagram(datagram.data(), datagram.size(),
QHostAddress::Broadcast,45454);//Envia los datos.
  
```

4.9.2. FUNCIONAMIENTO DEL SERVIDOR DE VIDEO

Como se puede ver en la Figura 4-12, el servidor de video inicia abriendo el puerto de comunicación por el cual se van a transmitir los datos, que es el 45454. Posteriormente, inicializa la conexión a la cámara de video y configura los tiempos de envío a 30 ms y verifica si ya transcurrieron, de no ser así espera hasta que la condición anterior se cumpla. Al termino de este tiempo captura la imagen de la cámara, la almacena en el datagrama y la envía por el broadcast; luego confirma si se dio una orden para terminar el video, de no ser así vuelve a contabilizar el tiempo y sigue el ciclo, caso contrario cierra el programa.

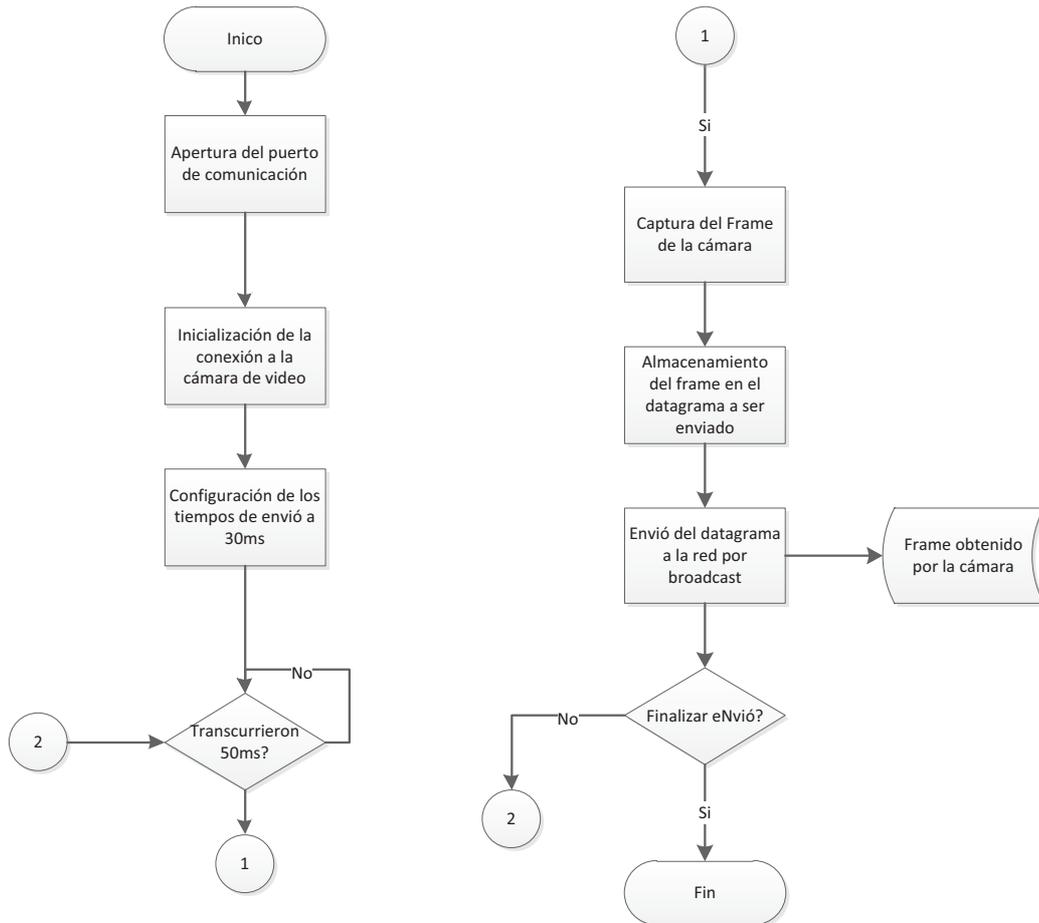


Figura 4-12.- Diagrama de flujo del servidor de video.

4.10. SOFTWARE DEL CLIENTE DE VIDEO

El cliente de video ROMOTEVI es el encargado de detectar cuando una imagen fue enviada por el broadcast desde el servidor de video, tomar esa imagen y colocarla en la interfaz gráfica. Las imágenes son capturadas el momento en que son detectadas en la red, por tanto el video tiene la misma velocidad de reproducción que la velocidad de captura del servidor.

Como se mencionó en el Capítulo 3, existen 2 versiones del sistema ROMOTEVI en hardware pero también existen en software. El sistema ROMOTEVI, para la versión 1.1 utilizó el protocolo TCP/IP en la transmisión de datos del video a una resolución de 640x480, esto provocaba que la transmisión de video sea muy lenta con retardos entre 2 a 3 segundos.

Para la versión 1.2, se modificó el protocolo de comunicación a UDP/IP, pero esta modificación provocó que la resolución del video sea disminuida a 160x120 pixeles el momento de transmitir y redimensionada a 512x384 pixeles al presentar la imagen. Esto causa que el video se pixele y la calidad de la imagen decaiga.

4.10.1. ESTRUCTURA DEL CLIENTE DE VIDEO

El cliente de video no tiene una estructura definida puesto que solo se encuentra en una función alojada en el núcleo del cliente de control ROMOTEVI. Éste no posee una clase dedicada con la cual se lo pueda estructurar, pese a que está integrado en el núcleo del cliente funciona separado de él, pues trabaja en un puerto distinto, utiliza una conexión mediante el uso de socket distinta a la del cliente y es llamada por un evento al detectar paquetes en el puerto 45454, mas no por alguna función del cliente de control. La función mencionada se muestra en el código a continuación:

```
1 QByteArray datagram;
2 datagram.resize(imgsize);
3 udpSocket->readDatagram(datagram.data(), datagram.size());
4 memcpy(img->imageData, datagram.data(), 160*120*3);
5 cvResize(img, img2, CV_INTER_LINEAR);
6 imagen= QImage((unsigned char *)img2->imageData, img2->width,
7               img2->height, img2->widthStep, QImage::Format_RGB888).rgbSwapped();
8 ui.label->setPixmap(QPixmap::fromImage(imagen, 0));
9 ui.video->setText("Correcto");
```

Del código mostrado, la parte más importante es la conversión de los datos guardados en el datagrama a un `IplImage`. Para convertir los datos se hace una copia de memoria de los datos en el datagrama, en la parte correspondiente a los datos en el `IplImage`, especificando la resolución de la imagen que se quiere copiar. Lo antes mencionado se realiza con la línea siguiente:

```
memcpy(img->imageData, datagram.data(), 160*120*3);
```

La conversión de `IplImage` a `QImage` para ser colocada en la interfaz gráfica se logra con la siguiente línea de código, donde `img2` es el `IplImage` a convertir y `Format_RGB888`

es el formato correspondiente a una imagen de color de 3 canales basada en tecnología RGB.

```
imagen= QImage((unsigned char *)img2->imageData,img2->width,
img2->height,img2->widthStep, QImage::Format_RGB888).rgbSwapped();
```

4.10.2. FUNCIONAMIENTO DEL CLIENTE DE VIDEO

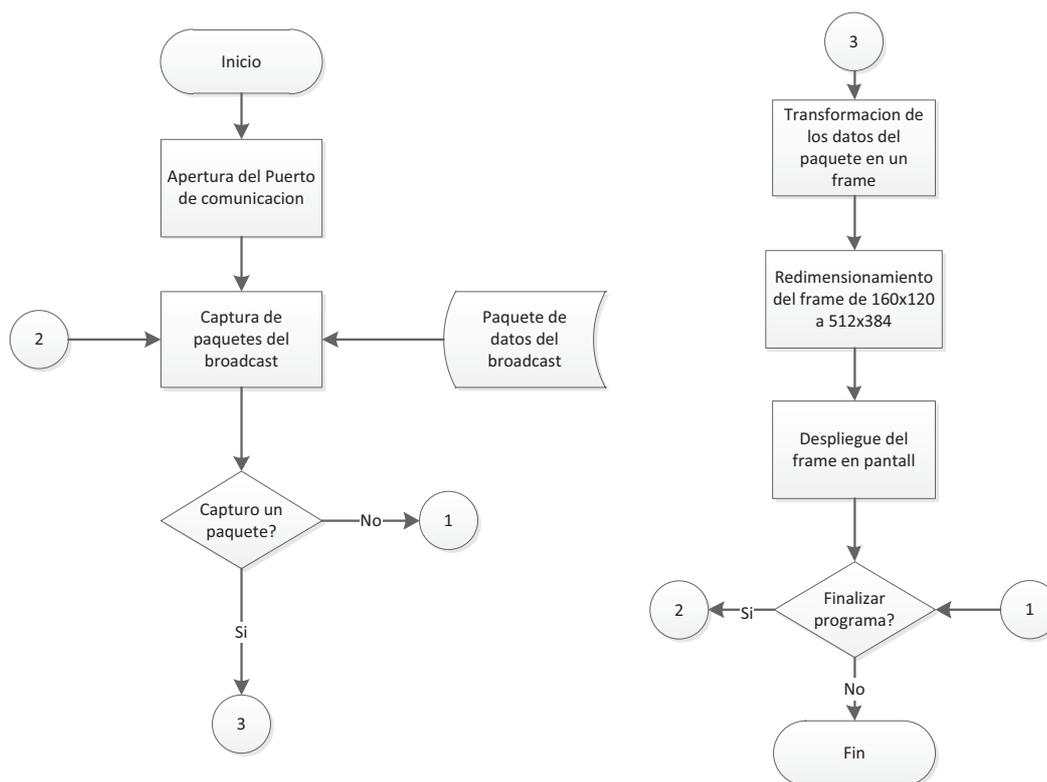


Figura 4-13.- Diagrama de flujo del cliente de video ROMOTEVI.

Como se puede ver en la Figura 4-13, el cliente de video empieza creando una conexión al puerto de comunicación; posteriormente inicia la captura de paquetes por broadcast. Cuando se captura un paquete enviado por el servidor de video se realiza la transformación de esos datos en una imagen, dicha transformación se encuentra explicada en el apartado 4.7.1; después redimensiona la imagen de 160x120 píxeles a una resolución de 512x384 píxeles para finalmente presentar la imagen en la interfaz gráfica.

Al terminar los pasos antes descritos verifica si aún debe correr el programa, si es así repite el ciclo, caso contrario finaliza su ejecución.

4.11. INTERFAZ GRAFICA DEL CLIENTE ROMOTEVI.

La interfaz gráfica del cliente ROMOTEVI se encuentra desarrollada bajo la librería Qt, como se mencionó en el apartado 4.1.1. En el cliente fue utilizada para crear la interfaz gráfica mostrada en la Figura 4.14.



Figura 4-14.- Interfaz gráfica del cliente ROMOTEVI

Como se puede ver, la interfaz gráfica está dividida en tres partes importantes, cada parte está orientada a dar una información específica, dicha división se encuentra resumida en la Tabla 4-2.

Número de parte	Descripción
Parte 1	Estado general del robot
Parte 2	Retroalimentación visual de la cámara
Parte 3	Estado general de los dispositivos conectados al cliente.

Tabla 4-2.- Descripción de las partes de la interfaz gráfica del cliente ROMOTEVI

La parte 1 es la encargada de retroalimentar la posición, velocidad y la capacidad de la batería, también informa sobre el estado del gripper, por ejemplo si está abierto o cerrado y si está arriba o abajo.

La parte 2 de la interfaz gráfica muestra el video captado por la cámara colocada en el robot.

La parte 3 está dividida en tres partes: la parte izquierda indica si el casco fue reconocido, la parte media si está conectado el joystick y la parte derecha si se está recibiendo el video correctamente. Para más información respecto a la interfaz gráfica refiérase al manual de usuario.

5.2. METODOLOGÍA DE EVALUACIÓN PARA EL SISTEMA ROMOTEVI

La metodología escogida para determinar el grado de telepresencia e inmersión sensorial del sistema ROMOTEVI, es la ejecución de dos cuestionarios a los usuarios que van a operar el sistema. El propósito de los cuestionarios es evaluar las experiencias del usuario, generadas al utilizar el sistema. Los cuestionarios usados son dos: uno es para determinar la capacidad de inmersión que posee una persona en un juego, libro o una tarea específica; y, el segundo cuestionario evalúa el nivel de presencia que posee cualquier sistema virtual. Dichos cuestionarios fueron tomados de [10].

5.2.1. CUESTIONARIO DE TENDENCIA INMERSIVA

Este cuestionario pretende determinar la capacidad que posee una persona a involucrarse en una tarea y mide algunos parámetros que son necesarios para evaluar un ambiente de realidad virtual, dichos parámetros son tres y se encuentran descritos en la Tabla 5-2.

Factores	Significado de la medición
INVOL	Tendencia de una persona a involucrarse en una actividad.
FOCUS	Tendencia de una persona a concentrarse en una actividad.
GAMES	Tendencia de una persona a jugar videojuegos.

Tabla 5-2.- Factores de medición del cuestionario CTI

El cuestionario consta de 29 preguntas; la escala de calificación del cuestionario es de 1 a 5, donde 1 es la peor situación y 5 la mejor. A cada pregunta se le encuentra asignado un peso dentro de la calificación total; los pesos y factores de medición del cuestionario se encuentran en el Anexo 4.

Para determinar la calificación máxima y mínima del cuestionario se multiplicaron los pesos por la mínima nota, se sumaron dichos pesos y se determinó el valor mínimo, de igual manera para el valor máximo. Los valores obtenidos se encuentran expresados en la Tabla 5-3.

Valor máximo del test	49,1
Valor mínimo del test	9,82

Tabla 5-3.- Valores máximos y mínimos del test CTI

El cuestionario se lo aplicó antes de realizar la prueba de experimentación con el sistema ROMOTEVI, para determinar a las personas con más tendencias inmersivas y posteriormente convocar a las personas seleccionadas para experimentar con la plataforma.

El número de personas escogidas para realizar este test fue de 31, provenientes de la Escuela Politécnica del Ejército, Departamento de Eléctrica y Electrónica, de octavo y décimo nivel, y se seleccionaron aquellas personas que tuvieron un porcentaje mayor a 64% en la evaluación del cuestionario. Se escogieron personas con este porcentaje debido a que representaría aquellos con un nivel de tendencia inmersiva mayor a lo normal, que gustan de los videojuegos, leer y se concentran con facilidad en una tarea determinada, para de este modo conseguir una evaluación mucho más acertada del sistema ROMOTEVI.

5.2.2. CUESTIONARIO DE EVALUACIÓN DE PRESENCIA.

Este cuestionario determina el nivel de presencia que genera un ambiente virtual hacia una persona que lo esté utilizando. El objetivo de este test es determinar el grado de presencia remota generada por el sistema ROMOTEVI en su operador. Para determinar este nivel el test evalúa 4 factores principales, descritos en la Tabla 5-4.

Factores	Medición del factor	Medición en el sistema ROMOTEVI
CF	Mide el control del sistema virtual.	Control del robot móvil
SF	Mide el nivel de inmersión de los sentidos de la vista, tacto y audio.	Medición del nivel de captura del sentido de la vista.
DF	Mide la distracción del sistema.	Distracción generada por el HMD y joystick.
RF	Mide el realismo del sistema virtual.	Mide el grado de coherencia del mundo real con el mundo mostrado por la interfaz.

Tabla 5-4.-Factores de medición principales.

El factor CF mide el grado de control que se tiene sobre el sistema, en este caso el control que se tiene sobre el robot móvil. El factor SF mide cuánto se han abordado los sentidos de la vista, tacto y audio en el sistema virtual, en el caso del sistema ROMOTEVI el único sentido tomado en cuenta es la vista. El factor RF mide el realismo que posee el sistema virtual, en el caso del sistema reflejará el nivel de coherencia entre el entorno remoto y la información mostrada en la interfaz.

El factor DF mide la cantidad de distracción que tienen las distintas interfaces del sistema virtual, en este caso mediría la distracción que genera el uso del casco de realidad virtual y del joystick sobre el sistema. A diferencia de los factores CF, SF, RF el factor DF debe tener la menor medición posible para mejorar el sistema.

El cuestionario también posee 6 subclases que permiten una mejor apreciación de la presencia que está generando el sistema. En la Tabla 5-5 se enuncian estos 6 factores y lo que miden dentro del sistema ROMOTEVI.

Subclases	Medición del factor	Medición en el sistema
NATRL	Naturalidad.	Mide la interacción implícita.
INV/C	Inmersión/control.	Mide el nivel de inmersión del sistema
AUD	Sentido de audición.	Mide el nivel de estímulo que provoca el sistema sobre el sentido de audición.
HAPTC	Sentido del tacto.	Mide el nivel de estímulo que provoca el sistema sobre el sentido del tacto.
RESOL	Sentido de la vista.	Mide el nivel de estímulo que provoca el sistema sobre el sentido de la vista.
IFQUAL	Calidad de la interfaz.	Mide la calidad de la interfaz del Sistema virtual.

Tabla 5-5.- Subclases del cuestionario de presencia.

Para medir el sistema fueron descartadas las preguntas orientadas a medir los factores AUD y HAPTC, puesto que se relacionan con los sentidos del tacto y del oído, los cuales no se los está usando el sistema, y; se consideró a la subclase INV/C la más importante, puesto que determinará el nivel de inmersión que posee el sistema ROMOTEVI. Sin embargo, todos los factores son analizados para llegar a determinar los problemas generales que posee el sistema.

El cuestionario posee 28 preguntas, las cuales se encuentran en el Anexo 5; su escala de calificación y cálculos de los valores máximos y mínimos son iguales a la del cuestionario de tendencias inmersivas, apartado 5.2.2. Los valores máximos y mínimos se encuentran descritos en la Tabla 5-6.

Valores máximos	56,1
Valores mínimos	10,3

Tabla 5-6.- Valores máximos y mínimos del test de presencia.

Como se mencionó en el apartado 5.2.2, el cuestionario de presencia se lo realizó a un grupo seleccionado anteriormente, el cual fue de 7 personas y la metodología con la cual se trabajó fue la siguiente:

- Se explicó en qué consistía la tarea que realizarían con el sistema ROMOTEVI. Véase apartado 5.1.
- Se realizó un pequeño tutorial sobre el control del sistema ROMOTEVI.
- Se procedió a realizar la prueba de experimentación con las 7 personas, las cuales desconocían totalmente el entorno remoto.
- Una vez finalizada la prueba experimental, se procedió a realizar el cuestionario.

5.3. RESULTADOS OBTENIDOS EN EL CUESTIONARIO DE PRESENCIA PARA EL SISTEMA ROMOTEVI.

El resultado global del cuestionario es de un 62,95%, este porcentaje representa cuán presente está el operador en el entorno remoto. Es necesario realizar un análisis de cada uno de los factores primarios y secundarios de medición para detectar cuáles serían las posibles mejoras para el sistema.

5.3.1. RESULTADOS DE LOS FACTORES PRINCIPALES.

Los resultados obtenidos del cuestionario para los factores principales de medición se encuentran descritos en la Figura 5-3, donde se puede apreciar en porcentaje los valores obtenidos en cada uno.

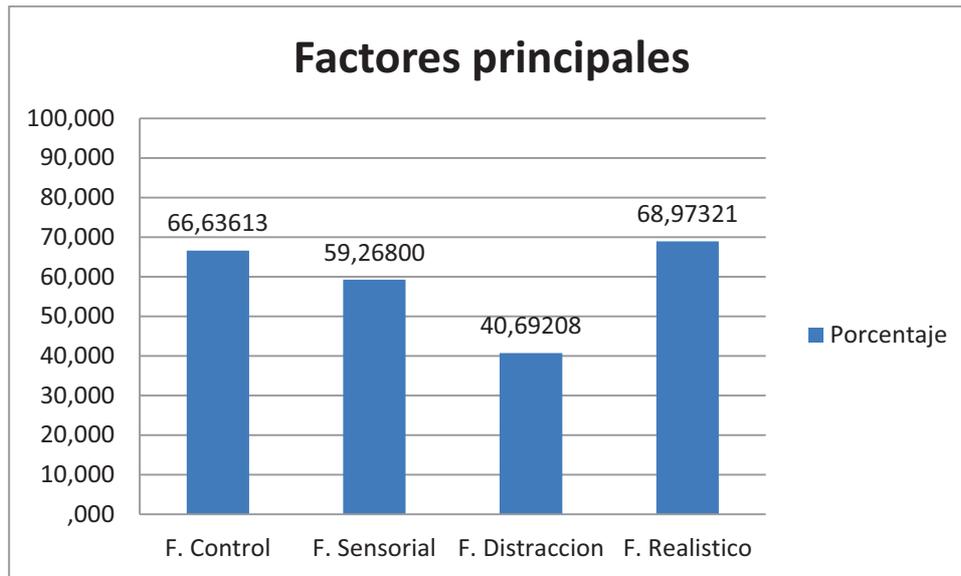


Figura 5-3.- Resultados obtenidos en los factores principales de medición de presencia en el sistema ROMOTEVI

Como se puede ver en la Figura 5-3, se tiene dos valores problemáticos:

- El factor sensorial refleja el problema descrito en el Capítulo 2, cuando se mencionó la diferencia existente entre el campo de visión humano y de la cámara. Como se esperaba, esto produjo una reducción en la generación de presencia por el sentido visual debido a que la diferencia del campo visual produce incomodidad al momento de ver a través de la cámara del robot.
- El factor de distracción también es un problema debido a que es muy alto, esto nos indica que las interfaces usadas para manejar el sistema ROMOTEVI producen cierto grado de distracción hacia el usuario. Lo que se detectó en las pruebas experimentales es que al usar el casco de realidad virtual el usuario ya no puede ver dónde se encuentra el joystick y el momento de controlar el *gripper* el usuario perdía la ubicación de los botones, provocando que éste regresara a ver el joystick para encontrarlos y generando una distracción muy grande de la tarea que se estaba realizando.

Para el caso del factor de control, se detectó en la prueba experimental que la respuesta de la cámara ante un movimiento de la cabeza del operador es muy lenta, obligando al operador a realizar movimientos más suaves para controlar la cámara. Esto no es un gran problema pero reduce el factor de control del sistema.

El factor realístico es el más alto de los anteriores, y esto es coherente dado que la imagen que se está presentando al operador es tomada del mundo real. Cabe recalcar que aquí existe un problema (ver Figura 5-4 en el factor de resolución): como se explicó en el apartado 4.7, la resolución de la imagen es baja y provoca que el realismo de la interfaz gráfica baje creando un menor grado de presencia.

5.3.2. RESULTADOS DE LOS FACTORES SECUNDARIOS.

Los resultados de los factores secundarios obtenidos en el cuestionario de presencia se encuentran expresados en la Figura 5-4, de éstos el más importante es el de inmersión y control ya que este factor nos da el nivel de inmersión sensorial que se obtuvo en el sistema ROMOTEVI, el cual es de 58,78%. Hay que tomar en cuenta que el único sentido involucrado en esta medición es el visual y también refleja el problema mencionado en el apartado 5.3.1 sobre la diferencia de campos visuales entre del ser humano y la cámara del robot.

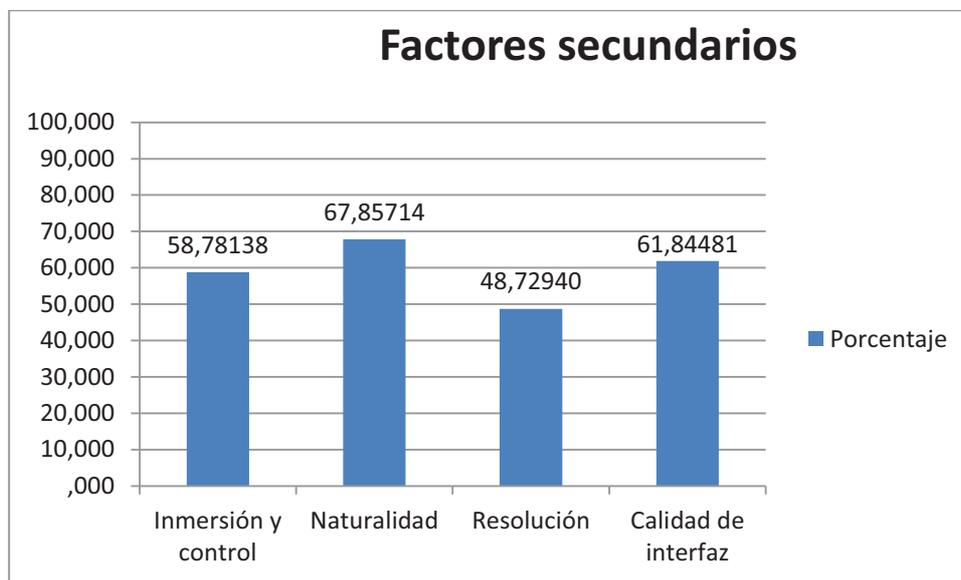


Figura 5-4.- Resultados obtenidos en los factores secundarios de medición de presencia en el sistema ROMOTEVI

El factor de naturalidad es normal que sea el más alto debido a la interacción implícita que por concepto genera el casco de realidad virtual, tal como se mencionó en el

Capítulo 1 en el apartado 1.5.1; además, el uso del joystick también genera una gran naturalidad debido a que su uso es muy común en los video juegos.

El factor de resolución, como se ve en la Figura 5-4, es notoriamente bajo y esto es debido a que la resolución del video es muy baja, aunque es suficiente para reconocer los objetos claramente en el entorno remoto, es un factor que reduce tanto el realismo como la inmersión sensorial del sistema.

El factor de calidad de la interfaz indica que la interfaz elaborada para realimentar al operador con los datos del robot es la adecuada, en la Figura 5-4 se muestra que posee un 61,84%. Cabe mencionar que este factor también se ve reducido debido a al factor de resolución antes mencionado.

Adicionalmente, como realimentación de las personas que realizaron la prueba experimental con el sistema ROMOTEVI, se obtuvo que es necesario la implementación de sonido en el sistema, para generar mayor nivel de presencia e inmersión sensorial ya que al carecer de éste el usuario experimenta una sensación poco natural de observar y no escuchar el entorno remoto.

CAPÍTULO 6

CONCLUSIONES Y RECOMENDACIONES

6.1. CONCLUSIONES

Al término de este proyecto se implementó un sistema teleoperado para un robot móvil utilizando visión artificial, el cual posee un 62.95% de telepresencia; este porcentaje representa cuánto se siente inmerso el operador dentro del entorno remoto. El sistema también permite una inmersión sensorial del 58.78% del sentido visual.

De los datos obtenidos en el capítulo 5, se determinó que el sistema captura el sentido visual en un 59.3%; esto se generó abarcando el 22.2% del campo visual binocular del ser humano. Sin embargo, el grado de presencia e inmersión sensorial se ve disminuido porque el usuario experimenta desorientación debido a la reducción de su campo visual.

Se consiguió que la cámara de video realice los mismos movimientos de la cabeza del operador, mediante el uso del casco de realidad virtual; sin embargo, dicho seguimiento presenta un retardo, debido al tiempo de respuesta de la cámara entre comandos. Además, su resolución en cuanto al movimiento es limitada, provocando desplazamientos de la cámara sin fluidez.

De las observaciones, se determinó que la utilización del conjunto HMD y joystick no es la adecuada. El grado de distracción presentado es del 40% y se genera porque el usuario no posee una vista directa al joystick, presentando una tendencia a extraviar la posición de los botones de éste.

Se observó que la resolución del video en el sistema ROMOTEVI provoca una imagen pixelada (carente de nitidez), esto provoca una reducción del nivel de inmersión sensorial. Sin embargo, es suficiente para reconocer los objetos en el entorno remoto y su velocidad resulta adecuada (33fps), con la cual se ha brindado naturalidad al video presentado al usuario.

6.2. RECOMENDACIONES

Es recomendable usar una cámara PTZ con un menor tiempo de respuesta para cada movimiento y una mayor resolución en los movimientos, esto con el propósito de mejorar la fluidez y naturalidad a la interacción cabeza-cámara del sistema. Además, debería tener un rango más amplio de movimiento, sobre todo en su eje vertical positivo, el cual debería ser cercano a los 90 grados, para tener un movimiento proporcional al de la cabeza del ser humano.

Debido a que el joystick provoca distracción por la posición de sus botones, se recomienda la utilización de un *gamepad*, ya que sus mandos poseen posiciones más adecuadas, disminuyendo la tendencia del usuario a desviar la mirada para ubicar la posición de los botones.

Se debería implementar un algoritmo de compresión de imágenes en el servidor del robot para aumentar la resolución de ésta a 800 x 600 píxeles, resolución que se tenía en la versión 1.1 del sistema ROMOTEVI, manteniendo una velocidad de transmisión de 24fps. Dicha velocidad es la mínima posible para mantener la naturalidad del video.

Se recomienda el uso de un accesorio en la cámara ubicada en el robot para incrementar el campo de visión de la misma. Una opción sería la utilización de un objetivo gran angular, el cual proporciona un incremento de 60 a 180 grados sin distorsionar la imagen. Cabe considerar que éste tiende a exagerar las líneas de perspectiva de los objetos muy cercanos y se debería investigar cuánto influye ello en la percepción visual del usuario.

Al término de las pruebas experimentales, algunos usuarios reportaron la necesidad de poseer audio en el sistema ROMOTEVI, debido a que retroalimentar video desde la cámara sin audio provoca una sensación poco natural al ver y no poder escuchar el lugar remoto que se está explorando. Debido a esto, es recomendable investigar cuánto incrementaría, el grado de presencia e inmersión sensorial, la suma de este factor

REFERENCIAS BIBLIOGRÁFICAS

- [1] CARABAYO, Jesús Ledesma, *Introducción a la Realidad Virtual*, Universidad Politécnica de Madrid, Departamento de Electrónica, <http://insn.die.upm.es/docs/VR0304.pdf>
- [2] BOO, Javi, *Introducción a la Realidad Virtual*, Universidad Politécnica de Cataluña, España, <http://www.lsi.upc.edu/~virtual/SGI/guions/ArquitecturaRV.pdf>
- [3] NUÑO y BASAÑEZ, Emmanuel y Luis, *Teleoperación: técnicas, aplicaciones, entorno sensorial y teleoperación inteligente*, Tesis Doctoral, 2004, <http://upcommons.upc.edu/e-prints/bitstream/2117/570/1/IOC-DT-P-2004-05.pdf>
- [4] PADILLA, Osca, *Manipulador Teleoperado inalámbicamente*, Ingeniería Mecatrónica, 2008-05-15, http://catarina.udlap.mx/u_dl_a/tales/documentos/lmt/padilla_m_o/capitulo2.pdf
- [5] GÓMEZ DE GABRIEL, J.M y Ollero Bauturone, Garcia Cerezo, *Teleoperación y Telerobótica*, A. Pearson Education S.A, Madrid,2006
- [6] OLLER, Anibal B., *Robótica Manipuladores y robots móviles*, MARCOBO S.A, Barcelona, 2001.
- [7] REYES, Arcadio L, *Realidad Virtual y Presencia*, Universidad de Málaga, Grupo de Investigación DIANA, 2004.
- [8] CAPELL, Ingrid M., *Sistema estereoscópico para tele operación asistida y supervisión de tareas robotizadas*, Escola Superior d'Enginyeria Industrial de Barcelona, 2006.
- [9] E. MELZER, James, *"Head Mounted Display" en Digital Avionics HandBook*, Segunda edición , CRC press, USA , 2006.
- [10] WITMER,Bob, G.Michael J. Singer, *Measuring Presence in Virtual Environments: A Presence Questionnaire*, U.S. Army Research Institute for the Behavioral and Social Sciences, Orlando

ÍNDICE DE FIGURAS

Figura 1-1. Fotografía correspondiente a Raymond Goertz manipulando químicos, hecha en 1948 en el laboratorio nacional de Argonne USA.[3]	11
Figura 1-2. Fotografía correspondiente a Raymond Goertz haciendo tareas de aprehensión de objetos, 1954, laboratorio nacional de Argonne USA.[6]	11
Figura 1-3. Esquema general de un sistema de teleoperación [6]	13
Figura 1-4. Esquema descriptivo del sistema ROMOTEVI	20
Figura 1-5.- Robot móvil Pioneer P3-AT	22
Figura 1-6.- Disposición actual de los elementos en la plataforma móvil	23
Figura 2-1.- Elementos que componen un casco de realidad virtual	24
Figura 2-2.- Imagen de un casco de realidad virtual.	25
Figura 2-3.- Esquema de la disposición del campo visual del ser humano.	27
Figura 2-4. Campos de visión de los elementos que intervienen en el sistema.....	27
Figura 2-5.- Pantalla transmisiva.....	31
Figura 2-6.- Pantallas reflectivas.	32
Figura 2-7 .- Pantallas auto-emisivas.	32
Figura 2-8. Imagen de los movimientos sensados por los trackers que posee el HMD.	37
Figura 2-9. Acelerómetro capacitivo MEMS de movimiento horizontal.....	37
Figura 2-10. Rangos de movimientos de los elementos que intervienen en el sistema de visión.	38
Figura 2-11. Esquema de la disposición de las pantallas en un HMD.	39
Figura 2-12.- Diseño de una formación de lentes no pupilar.	40

Figura 2-13.- Diseño de una formación de lentes pupilar.	41
Figura 2-14.- Sistema de sonido estereofónico.	42
Figura 3-1.- Gráfico general del sistema	45
Figura 3-2.- Hardware que conforma el servidor del sistema ROMOTEVI	47
Figura 3-3.- Robot móvil Pioneer P3-AT.....	48
Figura 3-4.- Cámara de video Canon VC-C50i.....	49
Figura 3-5.- Versión 1.1 del robot móvil del sistema ROMOTEVI.....	51
Figura 3-6.- Análisis del posicionamiento de la cámara con respecto al robot móvil.....	52
Figura 3-7.- Campo de visión de la cámara con respecto al robot móvil.....	52
Figura 3-8.- Campo de visión de la cámara apuntando a la parte delantera del robot.	53
Figura 3-9.- Rango de movimiento de la cámara con respecto al robot.....	53
Figura 3-10.- Versión 1.2 del robot móvil del sistema ROMOTEVI.....	54
Figura 3-11.- Análisis de la ubicación de la cámara con respecto al robot móvil para la versión 1.2	54
Figura 3-12.- Rangos de movimiento de la cámara para la versión 1.2	55
Figura 3-13.- Campo de visión de la cámara apuntando a la parte delantera del robot para la versión 1.2	55
Figura 3-14.- Tabla del consumo energético del robot móvil.	56
Figura 3-15.- Elementos que conforman el cliente del sistema ROMOTEVI.....	57
Figura 3-16.- Casco de realidad virtual eMagin Z800.....	58
Figura 3-17.- Esquema de la red de comunicación del sistema.....	59
Figura 3-18.- Interfaz gráfica del programa de prueba para el casco de realidad virtual. ...	60

Figura 3-19.- Programa de prueba para la comunicación inalámbrica de la plataforma.....	61
Figura 4-1.- Interacción de librerías para el sistema de control.	64
Figura 4-2.-Interacción de librerías para el sistema de video.....	65
Figura 4-3.- Estructura del SDK de EMagin.....	67
Figura 4-4.- Estructura de ArRobot de Aria.....	69
Figura 4-5.- Diagrama UML de clases para el servidor ROMOTEVI.....	71
Figura 4-6.- Diagrama de flujo para el servidor ROMOTEVI (parte 1).....	76
Figura 4-7.- Diagrama de flujo para el servidor ROMOTEVI (parte 2).....	77
Figura 4-8.- Diagrama UML de clases del cliente ROMOTEVI.	79
Figura 4-9.- Diagrama de flujo del cliente ROMOTEVI (Parte 1).	80
Figura 4-10.- Diagrama de flujo del cliente ROMOTEVI (Parte 2).	81
Figura 4-11.- Diagrama UML del servidor de video.....	83
Figura 4-12.- Diagrama de flujo del servidor de video.	84
Figura 4-13.- Diagrama de flujo del cliente de video ROMOTEVI.....	86
Figura 4-14.- Interfaz gráfica del cliente ROMOTEVI.....	87
Figura 5-1.- Descripción de la prueba experimental (Primera parte).....	89
Figura 5-2.- Descripción de la prueba experimental (Segunda parte).....	90
Figura 5-3.- Resultados obtenidos en los factores principales de medición de presencia en el sistema ROMOTEVI.....	96
Figura 5-4.- Resultados obtenidos en los factores secundarios de medición de presencia en el sistema ROMOTEVI.....	97

ÍNDICE DE TABLAS

Tabla 2-1.- Elementos que conforman el sentido visual humano.....	26
Tabla 2-2.- Tabla de los elementos que conforman la percepción visual.....	29
Tabla 2-3. Tabla de resoluciones para la tecnología grafica VGA.....	35
Tabla 2-4. Tabla de las resoluciones existentes en la tecnología gráfica XGA.....	36
Tabla 2-5. Tabla comparativa de los precios de cascos virtuales.....	44
Tabla 3-1.- Tabla con los dispositivos y sus funciones en el sistema.	46
Tabla 3-2.- Especificaciones técnicas de la computadora de abordó.	50
Tabla 3-3.- Especificaciones técnicas del conversor de voltaje.	51
Tabla 3-4.- Datos técnicos del computador cliente.	59
Tabla 3-5.- Configuración de red del sistema de comunicación inalámbrico.	61
Tabla 3-6. Tabla de registro de las pruebas del sistema.	63
Tabla 4-1.- Comandos utilizados en el servidor.	73
Tabla 4-2.- Descripción de las partes de la interfaz gráfica del cliente ROMOTEVI.....	87
Tabla 5-1.- Tabla de los factores experimentados.....	90
Tabla 5-2.- Factores de medición del cuestionario CTI	91
Tabla 5-3.- Valores máximos y mínimos del test CTI	92
Tabla 5-4.-Factores de medición principales.....	93
Tabla 5-5.- Subclases del cuestionario de presencia.	94
Tabla 5-6.- Valores máximos y mínimos del test de presencia.	94

GLOSARIO

Campo visual: 1. m. Espacio que abarca la vista estando el ojo inmóvil. 2. m. *Ópt.* Extensión de la superficie o del espacio reproducida en la imagen de un instrumento óptico.

HMD: En el ámbito de la realidad virtual, un casco virtual es un tipo de periférico entrada/salida que cuenta con un sistema de visión que permite a sus usuarios involucrarse en un mundo virtual generalmente en 3D. También puede ser llamado HMD, Casco 3D, anteojos virtuales, anteojos 3D, gafas 3D, etc.

Inmersión sensorial: Es la desconexión de los sentidos del operador del lugar donde se encuentra físicamente y la conexión de dichos sentidos hacia el lugar remoto.

Interfaz: Interfaz es un término que procede del vocablo inglés *interface* (superficie de contacto). En informática, esta noción se utiliza para nombrar a la conexión física y funcional entre dos sistemas o dispositivos.

Odometría: Se conoce como "odometría" a las técnicas de posicionamiento que emplean información de sensores propioceptivos (aquellos que adquieren datos del propio sistema), para obtener una aproximación de la posición real a la que se encuentra un sistema móvil, en un determinado instante, respecto a un sistema de referencia inicial.

Percepción: 1. f. Acción y efecto de percibir. 2. f. Sensación interior que resulta de una impresión material hecha en nuestros sentidos. 3. f. Conocimiento, idea.

Robot móvil: Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas solo a las personas. Se desplazan mediante una plataforma rodante (ruedas); estos robots aseguran el transporte de piezas de un punto a otro.

Teleoperación: Teleoperación: conjunto de tecnologías que comprenden la operación o gobierno a distancia de un dispositivo por un ser humano.

Telepresencia: Telepresencia: situación o circunstancia que se da cuando un ser humano tiene la sensación de encontrarse físicamente en el lugar remoto.

Pixelar: Imagen en la cual son apreciables los pixeles que la componen debido a un incremento excesivo en su tamaño

FECHA DE ENTREGA DEL PROYECTO

El presente proyecto de grado fue entregado al Departamento de Eléctrica y Electrónica, reposando en la Escuela Politécnica del Ejército desde:

Sangolquí, 20 de Noviembre del 2012

Ing. Víctor Proaño.

**COORDINADOR DE LA CARRERA DE INGENIERÍA ELECTRÓNICA
AUTOMATIZACIÓN Y CONTROL**

Richard Alfonso Andrade Alfaro

AUTOR

AUTORIZACIÓN

Yo, Richard Alfonso Andrade Alfaro

Autorizo a la Escuela Politécnica del Ejército la publicación, en la biblioteca virtual de la Institución del trabajo “Teleoperación de una plataforma móvil asistida mediante el uso de un sistema de visión artificial”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría

Sangolquí, 20 de Septiembre del 2012

Richard Alfonso Andrade Alfaro