



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES
INDUSTRIALES

“IMPLEMENTACIÓN DE UN PROTOTIPO PARA EL CONTROL DE
AMBIENTE DE UNA HABITACIÓN PARA PERSONAS CUADRIPLÉJICAS
CON UN 80% DE DISCAPACIDAD”

Trabajo de titulación

Tipo: Dispositivo Tecnológico

Presentado para optar al grado académico de:

INGENIERO EN ELECTRÓNICA, CONTROL Y REDES
INDUSTRIALES

AUTOR: YUGCHA ALOMALIZA CHRISTIAN ALEXIS

DIRECTOR: ING. EDWIN VINICIO ALTAMIRANO SANTILLÁN

Riobamba – Ecuador

2020

©2019, Christian Alexis Yugcha Alomaliza

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, Christian Alexis Yugcha Alomaliza, declaro que el presente trabajo de titulación es de mi autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 21 de agosto del 2020.

Christian Alexis Yugcha Alomaliza

C.I: 1803747912

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA ELECTRÓNICA EN CONTROL Y REDES
INDUSTRIALES

El Tribunal del Trabajo de Titulación certifica que: El trabajo de titulación: Tipo: Dispositivo Tecnológico, “**IMPLEMENTACIÓN DE UN PROTOTIPO PARA EL CONTROL DE AMBIENTE DE UNA HABITACIÓN PARA PERSONAS CUADRIPLÉJICAS CON UN 80% DE DISCAPACIDAD**”, realizado por el señor: **CHRISTIAN ALEXIS YUGCHA ALOMALIZA**, ha sido minuciosamente revisado por los Miembros del Trabajo de Titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

FIRMA

FECHA

Ing. Pablo Eduardo Lozada Yanez
PRESIDENTE DEL TRIBUNAL

Ing. Edwin Vinicio Altamirano Santillán
DIRECTOR DEL TRABAJO DE
TITULACIÓN

Ing. Jonny Israel Guaiña Yungán
MIEMBRO DEL TRIBUNAL

DEDICATORIA

El presente trabajo de titulación es dedicado principalmente a Dios, quien supo guiarme por el buen camino, darme fuerzas para seguir adelante, y permitirme haber llegado hasta este momento tan importante de mi formación profesional, a mis padres por su apoyo incondicional, a mis hermanos quienes supieron darme ánimos, a mi hijo quien es el motor principal para nunca rendirme y ser mi mayor motivación para seguir adelante, a mi novia por apoyarme y ser una fuente de motivación, finalmente a mis amigos que de una u otra manera aportaron a la culminación de este trabajo de titulación.

Christian

AGRADECIMIENTO

Principalmente quiero agradecer a Dios por haberme acompañado y guiado a lo largo de este arduo caminar estudiantil, por ser mi fortaleza en los momentos de debilidad y por brindarme una vida llena de aprendizajes y experiencias, agradezco a mi madre María Laura Alomaliza Z. por su amor y apoyo incondicional que aportaron a la culminación de este trabajo de titulación, a mis hermanos John Y. y Vladimir Y. por su apoyo en momentos difíciles de mi vida, a mi hijo Dylan Alexis Yugcha Lescano que con su sonrisa y amor me permitió seguir adelante hasta culminar la meta planteada, a mi novia Adriana Salinas quien con su sonrisa supo brindarme momentos de alegría y su presión fue fundamental para iniciar el desarrollo de este trabajo de titulación, a mis amigos con quienes compartir experiencias y momentos alegres durante este proceso de mi vida. A la ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO, por haberme educado y formado como un profesional. A mi Tutor de tesis Ing. Edwin Altamirano, por la motivación, dedicación y soporte en todo el proceso de realización de la misma.

Christian

TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	xi
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE GRÁFICOS.....	xii
ÍNDICE DE ANEXOS.....	xiv
RESUMEN.....	xvii
SUMMARY.....	xviii
INTRODUCCIÓN.....	1

CAPÍTULO I

1. MARCO TEÓRICO REFERENCIAL.....	2
1.1 Problematización.....	2
1.2 Formulación del problema.....	3
1.3 Justificación.....	3
1.4 Alcance.....	4
1.5 Objetivos.....	5
1.5.1 <i>Objetivo general</i>	5
1.5.2 <i>Objetivos específicos</i>	5
1.6 Marco Teórico Conceptual.....	5
1.6.1 <i>Análisis Biomecánico</i>	5
1.6.2 <i>Biomecánica clínica</i>	6
1.6.3 <i>Cinemática del movimiento</i> :.....	6
1.6.4 <i>Clasificación de la cinemática</i>	7
1.6.4.1 <i>Cinemática lineal</i>	7
1.6.4.2 <i>Cinemática angular</i>	7
1.6.5 <i>Médula espinal</i>	8

1.6.6	<i>Lesiones de la Médula espinal</i>	8
1.6.7	<i>Paraplejía y Cuadriplejía</i>	9
1.6.8	<i>Visión Artificial</i>	10
1.6.9	<i>Componentes de un sistema de Visión Artificial</i>	11
1.6.10	<i>Sistema de reconocimiento facial</i>	11
1.6.11	<i>Métodos de segmentación- Respuesta Espectral</i>	12
1.6.12	<i>Reconocimiento Facial</i>	12
1.6.13	<i>Librerías Java de desarrollo de aplicaciones de Visión por Computador</i>	13
1.6.14	<i>Entorno de LabVIEW</i>	13
1.6.15	<i>OpenCV</i>	14
1.6.16	<i>Conectividad EJS - Matlab</i>	14
1.6.17	<i>Herramientas en LabVIEW</i>	15
1.6.18	<i>Adquisición Digital de la imagen</i>	16
1.6.18.1	<i>Vision Express</i>	16
1.6.19	<i>Sistema de captura de imagen (cámara)</i>	17
1.6.20	<i>Plataforma Arduino</i>	17
1.6.20.1	<i>Características Generales</i>	19

CAPÍTULO II

2.	PROPUESTA Y DISEÑO DEL PROTOTIPO	¡Error! Marcador no definido.
2.1	Determinación del caso de estudio	20
2.2	Definición de requerimientos	22
2.2.1	<i>Propósito del Sistema</i>	23
2.2.2	<i>Definición de variables</i>	23
2.3	Selección del equipamiento	24
2.3.1	<i>Equipamiento del sistema de control</i>	24
2.3.1.1	<i>Cámara adquisición de información para el sistema de control</i>	25
2.3.1.2	<i>Selección de la pasarela del sistema de control con el entorno físico</i>	26

2.3.1.3	<i>Definición de Actuadores</i>	28
2.4	Algoritmo de Visión Artificial	29
2.4.1	<i>Selección del Software para el desarrollo del Algoritmo de V.A.</i>	29
2.4.2	<i>Desarrollo del sistema de Visión Artificial</i>	30
2.5	Vinculación del algoritmo de Visión Artificial con el de Control	39
2.6	Programación del microcontrolador	42

CAPITULO III

3.	VALIDACIÓN DEL PROTOTIPO ;Error! Marcador no definido.	
3.1	Implementación del sistema prototipo	45
3.1.1	<i>Conexión del hardware</i>	46
3.1.2	<i>Instalación del Hardware</i>	47
3.2	Interfaz Gráfica	49
3.3	Pruebas del prototipo	50
3.3.1	<i>Calibración del sistema</i>	51
3.3.2	<i>Prueba de distribución de elementos del ambiente de la habitación</i>	53
3.4	Validación de resultados obtenidos en las pruebas	55
3.5	Consumo eléctrico	62

CAPITULO IV

4.	EVALUACIÓN ECONÓMICA	64
4.1	Costos	64
4.1.1	<i>Costos directos</i>	64
4.1.1.1	<i>Costos eléctricos</i>	64
4.1.1.2	<i>Costos mecánicos</i>	64
4.1.1.3	<i>Costos de mano de obra</i>	65
4.1.2	<i>Costos directos totales</i>	65

4.1.3	<i>Costos indirectos</i>	65
4.1.4	<i>Costo total</i>	65

CONCLUSIONES	66
---------------------------	----

RECOMENDACIONES	67
------------------------------	----

GLOSARIO

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

Tabla 1-1:	Escala de clasificación de la lesión medular	9
Tabla 2-2:	Variables del sistema.....	24
Tabla 3-2:	Características de la cámara.	25
Tabla 4-2:	Valoración de características placas Arduino.....	26
Tabla 5-2:	Determinación de acciones según los movimientos establecidos.....	40
Tabla 6-3:	Evaluación del sistema “posición semi - sentado	55
Tabla 7-3:	Evaluación del sistema posición acostado.....	56
Tabla 8-3:	Evaluación del sistema pruebas experimentales.....	57
Tabla 9-3:	Posición y movimientos variación de eficiencia	59
Tabla 10-3:	Aciertos vs Desaciertos pruebas movimiento 1B	60
Tabla 11-3:	Resultado prueba T-Student movimiento 1B	60
Tabla 12-3:	Aciertos vs Desaciertos pruebas movimiento 3A.....	61
Tabla 13-3:	Resultado prueba T-Student movimiento 3A	61
Tabla 14-3:	Consumo base.....	62
Tabla 15-3:	Consumo a máxima carga	62
Tabla 16-4:	Costos eléctricos	64
Tabla 17-4:	Costos mecánicos	64
Tabla 18-4:	Costos de mano de obra.....	65
Tabla 18-4:	Costos directos totales	65
Tabla 20-4:	Costo indirectos totales.....	65
Tabla 21-4:	Costos directos totales	65

ÍNDICE DE FIGURAS

Figura 1-1:	Biomecánica del movimiento.....	6
Figura 2-1:	Relación entre la Cinemática Lineal y Angular	7
Figura 3-1:	Movimiento Humano	8
Figura 4-1:	Paraplejia y Cuadriplejia.....	10
Figura 5-1:	Esquema de la Visión artificial para personas con cuadriplejia.....	10
Figura 6-1:	Diagrama de bloques de un Sistema de Visión Artificial	11
Figura 7-1:	Fases de un Sistema de Visión Artificial	12
Figura 8-1:	OpenCV aplicaciones.....	14
Figura 9-1:	Conectividad EJS con Matlab	15
Figura 10-1:	Interfaz IMAQ	17
Figura 11-1:	Modelos de Arduino	18
Figura 12-1:	Partes fundamentales del Arduino	19
Figura 13-2:	Metodología	20
Figura 14-2:	Niveles de Lesión – Caso de estudio Lesión C4.....	21
Figura 15-2:	Movimientos del cuello.....	21
Figura 17-2:	Etapas físicas del sistema.....	23
Figura 18-2:	Características ordenador empleado para desarrollo del sistema.....	24
Figura 19-2:	Caracterización de la Cámara disponible.....	25
Figura 20-2:	Recursos - Arduino (UNO).....	27
Figura 21-2:	Módulo de Relés Arduino.....	28
Figura 22-2:	Librería IMAQ Vision	29
Figura 23-2:	Pantalla instalación NI Modulo Development Vision	30
Figura 24-2:	Caracterización de la Cámara disponible.....	31
Figura 25-2:	Determinación de la Cámara a emplearse	31
Figura 26-2:	Formato de adquisición de imagen	32
Figura 27-2:	Determinación parámetros calidad de imagen	33
Figura 28-2:	Parámetros de control e indicadores.	33
Figura 29-2:	Filtro de imagen	34
Figura 30-2:	Filtro un solo color de un plano	34
Figura 31-2:	Asistente de Visión	35
Figura 32-2:	Creación New Template	36
Figura 33-2:	Tolerancia de rotación de la plantilla para ser identificada.....	36
Figura 34-2:	Definición área de interés	37
Figura 35-2:	Área segunda plantilla.....	37

Figura 36-2:	Parámetros obtenidos de las plantillas definidas.....	38
Figura 37-2:	Desglose de parámetros del Clúster de las plantillas	38
Figura 38-2:	Recursos monitoreo del proceso	39
Figura 39-2:	Posición de los marcadores	39
Figura 40-2:	Enlace Arduino – Plataforma Informática	41
Figura 41-2:	Bloques para comparación de posiciones y comunicación Serial.....	42
Figura 42-2:	Desarrollo de la Interfaz gráfica del sistema.....	43
Figura 43-2:	IDE de Arduino.....	43
Figura 44-2:	Declaración de recursos iniciales empleados en el microcontrolador.....	44
Figura 45-3:	Dimensiones del ambiente para experimentación.....	45
Figura 46-3:	Diagrama de conexiones del sistema.	46
Figura 47-3:	Diagrama de esquemático con cargas físicas	46
Figura 48-3:	Tablero eléctrico – circuito de control.	47
Figura 49-3:	Tablero eléctrico – circuito con cargas.	48
Figura 50-3:	Cargas del ambiente controlado	48
Figura 51-3:	Panel frontal Interfaz gráfica.....	49
Figura 52-3:	Pruebas del sistema	50
Figura 53-3:	Definición posición inicial	51
Figura 54-3:	Definición de rangos	52
Figura 55-3:	Desplazamiento lateral de la cabeza hacia la izquierda	53
Figura 56-3:	Desplazamiento lateral de la cabeza hacia la derecha.....	53
Figura 57-3:	Distribución ambiente (1)	54
Figura 58-3:	Distribución ambiente (2)	54
Figura 59-3:	Prueba detección de marcadores – error por condiciones no ideales de luz	55

ÍNDICE DE GRÁFICOS

Gráfico 1-3:	Porcentajes de eficiencia de movimientos en los dos casos de posición.....	59
Gráfico 2-3:	Consumo base vs Consumo máximo del sistema.....	633

ÍNDICE DE ANEXOS

ANEXO A: DATASHETT ARDUINO UNO

ANEXO B: DATASHETT MEGA

ANEXO C: HOJAS DE VALORES DE PRUEBAS

ANEXO D: HOJA DE CÁLCULO DE T- STUDENT

RESUMEN

El presente trabajo describe la implementación de un prototipo para el control del ambiente de una habitación para personas cuadripléjicas con el fin de proporcionar una alternativa para que este tipo de pacientes puedan alcanzar un nivel de independencia. Se partió por definir el caso de estudio, determinando casos con un 80% de discapacidad, con una lesión medular tipo C4, que implica inmovilidad desde los hombros hacia abajo y movimiento de la cabeza por medio del cuello. Se planteó la manipulación de: una cortina, lámpara, ventilador y una sirena de alerta. Se desarrolló una aplicación informática basada en un algoritmo de visión artificial usando como plataforma de desarrollo a LabVIEW, en el que haciendo uso del asistente de visión se planteó un algoritmo que parte de la adquisición de imágenes continuas, su procesamiento y tratamiento para la identificación de patrones pre establecidos, que al ser detectados entrega información sobre su ubicación en modo de coordenadas cartesianas, recursos empleados para estimar rangos de tolerancia y discriminación de movimientos del paciente usados como fundamento para la emisión de señales de control gestionadas a través de un Arduino hacia una interfaz de potencia para conexión de las cargas. En conclusión, mediante pruebas realizadas a diez personas en dos situaciones específicamente en estado de semi-sentado y acostado se determinó una eficiencia del 87,75% y 77,87% respectivamente, y validado mediante una prueba de T-Student se determinó que el sistema discrimina los movimientos establecidos con buen eficiencia. Se recomienda tomar en cuenta condiciones de luminosidad que fue un factor determinante en este trabajo.

Palabras clave: <TECNOLOGÍA Y CIENCIAS DE LA INGENIERÍA>, <VISION ARTIFICIAL>, <CONTROL REMOTO>, <ALGORITMO DE VISION ARTIFICIAL>, <SISTEMA ASISTENCIAL>, CUADRIPLÉJICA, <AUTOMATIZACIÓN>, <SISTEMA INTELIGENTE DE ASISTENCIA>

ABSTRACT

This work describes the implementation of a prototype for controlling the environment of a quadriplegic room in order to provide an alternative for this type of patients can reach a level of independence. It was started by defining the case study, determining cases with 80% disability, with a C4 type spinal injury, which implies immobility from the shoulders down and movement of the head through the neck. It was proposed to manipulate: a curtain, lamp, fan and a warning siren. A computer application based on an algorithm of artificial vision using as a development platform LabVIEW was established in which making use of the view assistant an algorithm was proposed that starts from the acquisition of continuous images, its processing and treatment for the identification of pre-established patterns, that when detected it provides information on its location in Cartesian coordinate mode, resources used to estimate tolerance and discrimination ranges of patient movements applied as the basis for the emission of control signals managed through an Arduino to a power interface for connection of loads. In conclusion, through tests performed on ten people in two situations specifically in a semi-sitting and lying down, an efficiency of 87.75% and 77.87% respectively was determined, and validated by a T-Student test it was deduced that the system discriminates established movements with good efficiency. It is recommended to take into account light conditions that was a determining factor in this work.

Key words: <TECHNOLOGY AND ENGINEERING SCIENCES >, <ARTIFICIAL VISION>
< REMOTE CONTROL >, < ARTIFICIAL VISION ALGORITHM>, <CARE SYSTEM>
< QUADRIPLÉGIA>, <AUTOMATION>, <INTELLIGENT ASSISTANCE SYSTEM>

INTRODUCCIÓN

Una Lesión Medular es un proceso patológico que origina alteraciones de la función motora, sensitiva o autónoma, con diversas secuelas psicosociales para la persona y para su familia, la misma que es generadora de importantes procesos de discapacidad, acorde las últimas estadísticas presentadas por el CONADIS (Consejo Nacional de Discapacidades), indican que en el Ecuador existe un 2.63 % de la población total (368.245 personas carnetizadas) que presenta algún tipo de discapacidad sea esta física o intelectual, de este porcentaje, las discapacidades físicas y motrices son las más comunes, con un porcentaje del 37%, seguida por las intelectuales con un 24,5% y las discapacidades múltiples con un 13%, bajo estos datos se conoce que aproximadamente el 12% de la urbe total del país, muestra algún tipo de discapacidad. Acorde estudios de la OMS (Organización Mundial de la Salud), en Latinoamérica directamente un porcentaje del 2% de personas con discapacidad puede acceder a algún tipo de beneficio o proceso de rehabilitación. (Organización Mundial de la Salud, 2017)

En el Ecuador se estima que 33 de cada 100 personas no acceden a programas de rehabilitación, lo que ha motivado al estado ecuatoriano a la creación de fundaciones dedicadas a la inclusión de personas con capacidades diferentes. Fundaciones como “Manuela Espejo” de la Vicepresidencia son un claro ejemplo de cómo el estado busca la inclusión social y laboral, pero gracias al desarrollo tecnológico se está implementando sistemas con un grado de autonomía o inteligencia los mismos que requieren el uso de hardware y software, de tipo open source o de acceso libre con el fin de crear sistemas de bajo costo haciendo que la tecnología esté disponible para todo tipo de usuario y un sin número de aplicaciones en diversos ámbitos de desarrollo, como es el campo médico, a través del uso del software libre, el cual proporciona a desarrolladores la libertad para leer, redistribuir y modificar el código fuente, en el caso de hardware libre se hace referencia al hardware que está a disposición del público su diseño para su libre estudio, abierto a ser transformado y comercializado, es entonces que surge la necesidad en base a la problemática social generada a nivel global y haciendo uso de este tipo de tecnología, presentar soluciones puntuales a problemas sociales mejorando la calidad de vida de pacientes con un tipo de discapacidad la misma que limita la independencia autónoma del individuo, a través de un control autónomo de actividades cotidianas de las personas con cuadriplejía, por medio de un sistema de visión artificial, dotando al individuo de autonomía optimizando la eficacia de vida del mismo. (Organización Mundial de la Salud, 2017)

CAPÍTULO I

1. MARCO TEÓRICO REFERENCIAL

El desarrollo de la automatización y la revolución industrial plasmada en la industria 4.0, en la actualidad han generado un progreso en la calidad de vida de las personas y de los entornos laborales indistintamente del ámbito en el cual se desempeñe la fuerza laboral, ya que estos permiten realizar desde actividades cotidianas hasta complejas para personas con movilidad reducida o con una patología específica debido a los múltiples desarrollos tecnológicos generados en esta última década, permitiendo anticipar sucesos que pueden afectar el desempeño en el trabajo, dentro del sector productivo, acorde a los reglamentos que estipula el estado ecuatoriano de inclusión social, para un sector de personas con patologías congénitas o adquiridas que pertenecen a un grupo vulnerable de la sociedad, como son las personas con discapacidad las cuales se encuentran inmersas en los sistemas de inclusión social de diverso índole, por tal motivo es conveniente desarrollar nuevas tecnologías que permitan mejorar notablemente la calidad de vida de este sector vulnerable de la sociedad porque se estima que en el país 33 de cada 100 personas con problemas de discapacidad no pueden acceder a los programas de rehabilitación por no contar con los recursos necesarios, razón por la cual el dispositivo electrónico que se desarrollara está enfocado a mejorar las condiciones de vida de un paciente con cuadriplejía, en el control de ambiente de una habitación, a través de la ejecución de actuadores eléctricos que se tornarán de gran importancia al momento de realizar una determinada actividad cotidiana dentro de una habitación, pudiendo ser replicados en el futuro para nuevos desarrollos tecnológicos y la integración de este grupo vulnerable de la sociedad en la industria.

1.1 Problematicación

Acorde diversos estudios realizados la incidencia de las lesiones medulares es variable entre países y entre regiones. (2) (Wyndaele y Wyndaele, 2006, pp. 523-529), por tratarse de una complicada patología, la misma que afecta a diversas personas en el planeta y en una revisión sistemática de publicaciones científicas entre 1977 y 2006 sobre incidencia, prevalencia y epidemiología de lesión medular a nivel mundial concluyen que la incidencia global oscila entre 10,4 y 83 por millón de habitantes por año, y la prevalencia entre 223 a 755 por millón de habitantes. Según el estudio realizado por (3) Mazaira et al., (1998); estiman que en países desarrollados la

incidencia de la lesión medular varía en un rango entre 9 y 53 por millón de habitantes (3). Cada año suceden 12.000 nuevos casos en Norteamérica (4) (DeVivo et al., 2002, pp. 335-338), estimaciones actuales sugieren que la incidencia anual de lesiones medulares en los Estados Unidos es alrededor de 40 casos por millón (6) (Krause y Broderick, 2004, pp. 355-362).

Una lesión de la médula espinal (LME) generalmente se define como un daño o trauma en la médula espinal que a su vez resulta en una pérdida o función deteriorada que resulta en una movilidad o sensación reducida. Los fabricantes de equipos dicen que diseñar aplicaciones para usuarios con discapacidades no es rentable. La mayoría de estos sistemas están diseñados para usuarios que no están deshabilitados; por lo tanto, los sistemas que se dirigen a usuarios discapacitados necesitan interfaces especiales para ser accesibles. Utilizando una técnica de infrarrojos, tecnología de visión por computadora y diseño mecánico, los usuarios pueden realizar algunas actividades para mejorar su calidad de vida y dar cierto grado de independencia.(5) (Plaza et al., 2013)

De esta manera los sistemas de visión actuales operan sobre estructuras en lugar de píxeles. Dichos sistemas necesitan procesadores de alta velocidad para poder procesar la gran cantidad de datos recibidos a la entrada del sistema y la calidad digital de las imágenes.(6) (Domínguez, 2003). Por ende, las ventajas que se presenta al hacer uso de esta tecnología es la infinidad de aplicabilidad para crear nuevos usos enfocados a mejorar la calidad de vida de las personas que sufren una cuadriplejía, porque ofrece alta precisión, siendo de fácil manejo e integración, pudiendo ser instalado en el interior o exterior de un domicilio entre otros, haciendo necesario el uso de esta tecnología para mejorar la calidad de vida de los pacientes.

1.2 Formulación del problema

¿Cómo implementar un prototipo para el control de ambiente de una habitación para persona cuadripléjicas con un 80% de discapacidad?

1.3 Justificación

El desarrollo del trabajo de titulación consiste en implementar un prototipo para el control de ambiente de una habitación para personas cuadripléjicas con un 80% de discapacidad por medio de la implementación de actuadores eléctricos que se tornan de gran importancia al momento de realizar una actividad común en una habitación; el control de estos se basará en movimientos realizados por la persona cuadripléjica por medio de la cabeza. La cámara por cable se

encargará de capturar los movimientos que realiza, y enviar a un sistema electrónico seleccionado que cuenta con una interfaz que permite el almacenamiento de información para el procesamiento del movimiento captado y con el algoritmo desarrollado cumpla los requerimientos establecidos.

Los eventos físicos a desarrollar mediante los movimientos de la persona cuadripléjica serán encendido y apagado de una lámpara, encendido y apagado de un ventilador, encendido y apagado de un televisor y solicitud de ayuda. Todo este proceso se realizará mediante una tarjeta de desarrollo, la cual permite emitir un tipo de alerta a la persona de cuidado, contribuyendo de forma significativa a un aspecto muy importante cuya finalidad está enfocado a mejorar la calidad de vida de las personas cuadripléjicas, de esta manera abordando el objetivo 3 del Plan Nacional del Buen Vivir el cual hace mención a: “Mejorar la calidad de vida de la población”, porque el incremento de la expectativa de vida de esta población con esta patología hace que gracias a los avances tecnológicos y científicos en prevención, cuidado y manejo de complicaciones ha llevado a la necesidad de proporcionar productos de rehabilitación funcionales y que permitan la inclusión social.

1.4 Alcance

El presente trabajo de titulación está orientado a desarrollar las condiciones y herramientas necesarias para la captura de los movimientos que realiza, una persona cuadripléjica para enviar a un sistema electrónico seleccionado, el cual cuenta con una interfaz que permitirá el almacenamiento de información para el procesamiento del movimiento captado y con el algoritmo que se desarrollará pueda cumplir los requerimientos de control dentro de una habitación, brindándole de esta manera autonomía para ciertos requisitos básicos de asistencia a las personas que padecen determinada condición, el estudio será realizado en base a las distintas consideraciones y parámetros de diseño, como factor importante una valoración previa del especialista de la salud en referencia al tipo de patología, posteriormente se procederá a su desarrollo e implementación tomando en consideración las herramientas necesarias, además al final del desarrollo del presente dispositivo electrónico se pretende desarrollar un prototipo en base a un costo asequible a la sociedad, permitiendo reproducirlo posteriormente para el beneficio de toda la sociedad mejorando la calidad de vida de los pacientes con esta condición.

1.5 Objetivos

1.5.1 *Objetivo general*

Implementar un prototipo para el control de ambiente de una habitación para persona cuadripléjicas con un 80% de discapacidad

1.5.2 *Objetivos específicos*

Determinar los movimientos corporales correspondientes para la activación de los eventos físicos dentro de una habitación.

Establecer los requisitos de diseño que debe cumplir el prototipo.

Seleccionar componentes de hardware y herramientas de software adecuados que permita cumplir con los requisitos establecidos en el prototipo a construir.

Evaluar si el prototipo cumple con los requisitos planteados, mediante pruebas y resultados.

1.6 Marco Teórico Conceptual

1.6.1 *Análisis Biomecánico*

Para entender de una forma concisa y sistemática el funcionamiento y anatomía del cuerpo humano y cuán importante es el aparato locomotor en el organismo, el mismo que está comprendido por: el sistema óseo y sistema muscular, es preciso conocer la anatomía básica , porque en base a la literatura tradicional y moderna existen conceptos los cuales son importantes cuando nos referimos a la función locomotora, la cual involucra a: las extremidades, huesos, por tal motivo, es necesario tener un conocimiento previo desde el punto de vista anatómico.

1.6.2 *Biomecánica clínica*

La biomecánica clínica analiza las patologías que aquejan al cuerpo humano para generar soluciones capaces de evaluarlas, repararlas o paliarlas. La mayor parte de las afecciones del aparato locomotor pueden enfocarse desde el punto de vista mecánico, bien porque la causa del problema es mecánica o bien porque tiene consecuencias mecánicas. Siempre que se hace un estudio biomecánico hay que valorar no solo a la persona, sino también al medio en que se desenvuelve y a las herramientas que utiliza (prótesis, órtesis, bastones...). Se debe conocer muy bien la biomecánica de los diferentes tejidos y la de la postura y los tipos de movimientos básicos. Entre los movimientos básicos para el ser humano se encuentra la marcha, ya que es una de las actividades que le aporta mayor independencia, por lo que su alteración puede conducir a una situación de incapacidad y/o minusvalía importante. (Miguel Ángel Martínez Barca, 2009, p. 99)



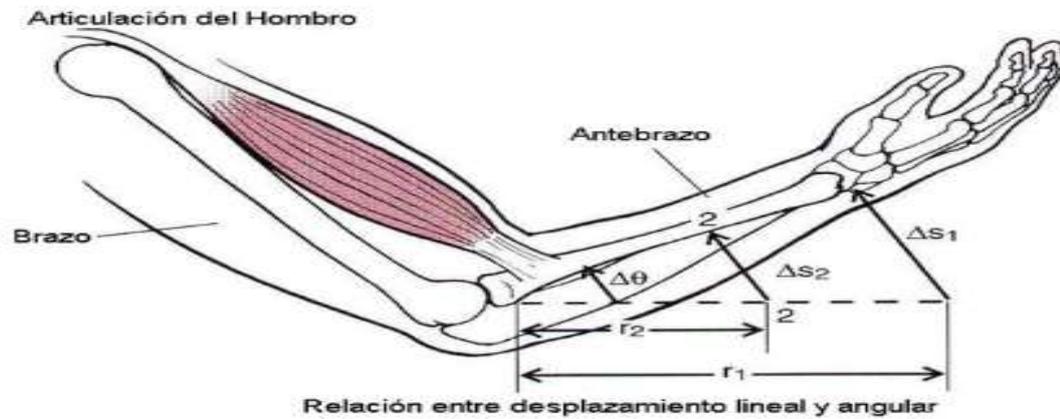
Figura 1-1. Biomecánica del movimiento

Fuente: <https://cutt.ly/HrehG3x>

1.6.3 *Cinemática del movimiento:*

La interpretación clínica de los trastornos motores empieza con la identificación de las desviaciones cualitativas y cuantitativas de los patrones patológicos respecto a los patrones normales. El análisis cinemático de la marcha comprende el registro y la evolución de parámetros tales como: desplazamiento articular, velocidad de marcha, aceleración y otros que están expresados en función del tiempo, y que se pueden estudiar debido a que están dados en términos del por ciento del ciclo de marcha. Los movimientos humanos poseen un porcentaje de variabilidad, entre sujetos de una misma población (inter-sujetos), pero también para cada sujeto

(intra-sujeto). El hecho de que los registros de los desplazamientos angulares de una misma articulación difieren temporalmente para un mismo sujeto, entre una observación y otra, se hace necesario el acondicionamiento de los mismos. (Díaz Novo, 2012, pp. 14-16)



$$v = \omega \times r$$

Figura 2-1. Relación entre la Cinemática Lineal y Angular

Fuente: <https://cutt.ly/Brejqqj>

1.6.4 Clasificación de la cinemática

La cinemática del movimiento se puede clasificar en:

1.6.4.1 Cinemática lineal

Descripción del movimiento que se realiza en una línea recta, i.e., cinemática de la traslación o movimiento lineal.

1.6.4.2 Cinemática angular

Descripción del movimiento que se lleva a cabo alrededor de un punto fijo (el eje o centro de giro/rotación que mantiene su posición en el interior o exterior del cuerpo), i.e., cinemática de la rotación o movimiento angular. (Corsino, 1998, p. 1).

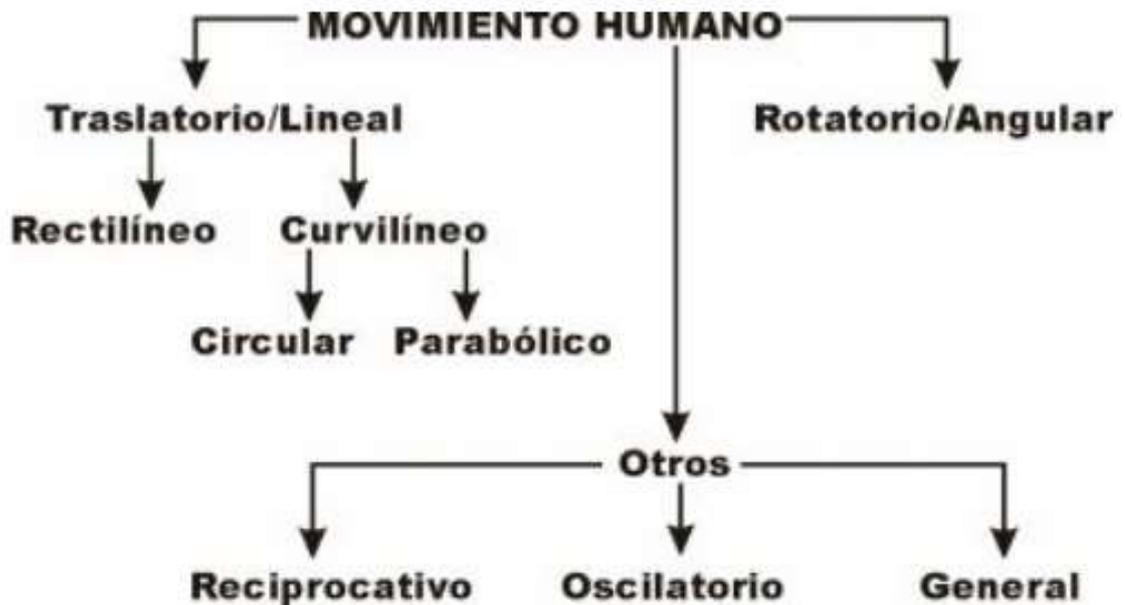


Figura 3-1. Movimiento Humano

Fuente: <https://bit.ly/2PMuUNy>

1.6.5 *Médula espinal*

La médula espinal es la parte del sistema nervioso central que controla los músculos voluntarios de las extremidades y el tronco, y que recibe información sensorial de estas regiones. También controla la mayoría de las vísceras y los vasos sanguíneos del tórax, el abdomen y la pelvis. El cordón se describe como compuesto por una serie de componentes segmentarios, pero en el adulto el cordón es en realidad un cilindro continuo de tejido nervioso central. La médula espinal humana tiene 12 segmentos torácicos, 5 segmentos lumbares, 5 segmentos sacros y 1 segmento coccígeo, lo que hace un total de 31 segmentos. Los pares de nervios espinales surgen de la médula espinal y salen de la columna vertebral a través de los agujeros intervertebrales. (Watson y Kayalioglu, 2009, pp. 1-7)

1.6.6 *Lesiones de la Médula espinal*

Las lesiones medulares se denominan de acuerdo con el último nivel intacto, es decir, aquella metámera que preserve todas las funciones neurológicas: motoras, sensitivas y autonómicas. En este sentido, pueden clasificarse de acuerdo con la funcionalidad clínica del paciente como cervicales (C1 a C8), torácicas altas (T1 a T6), torácicas bajas (T7 a T12), lumbosacras (L1 a S1) y del cono medular (sacro coccígeas). Entre más alta la lesión mayor será el compromiso

funcional del paciente. De acuerdo con la extensión de la lesión, éstas pueden considerarse como transversales y longitudinales; en la mayoría de los pacientes existe una combinación de ambas. (Henao-Lema y Pérez-Parra, 2010, pp. 161-162)

Tabla 1-1: Escala de clasificación de la lesión medular

Lesión completa A	Ausencia de función motora y sensitiva que se extiende hasta los segmentos sacros S4-S5
Lesión incompleta B	Preservación de la función sensitiva por debajo de nivel neurológico de la lesión, que se extiende hasta los segmentos sacros S4-S5 y con ausencia de función motora.
Lesión incompleta c	Preservación de la función motora por debajo de nivel neurológico y mas de la mitad de los músculos llave por debajo del nivel neurológico tienen un balance muscular menor de 3
Lesión incompleta D	Preservación de la función motora por debajo del nivel neurológico y más de la mitad de los músculos llave por debajo del nivel neurológico tienen un balance muscular de 3 o más.
Lesión Normal E	Las funciones sensitiva y motora son normales

Fuente: (HENA O -LEMA Y PEREZZ -PARRA, 2010)

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

1.6.7 Paraplejía y Cuadriplejía

El término paraplejía se utiliza para referirse a una debilidad parcial o completa de ambas piernas y el término cuadriplejía indica una debilidad parcial o completa de los cuatro miembros, eliminando así la necesidad de los términos paraparesia y cuadriparesia. Una misma alteración medular puede producir paraplejía o cuadriplejía, dependiendo de la localización de la lesión. (Fenichel, 2006, p. 255)

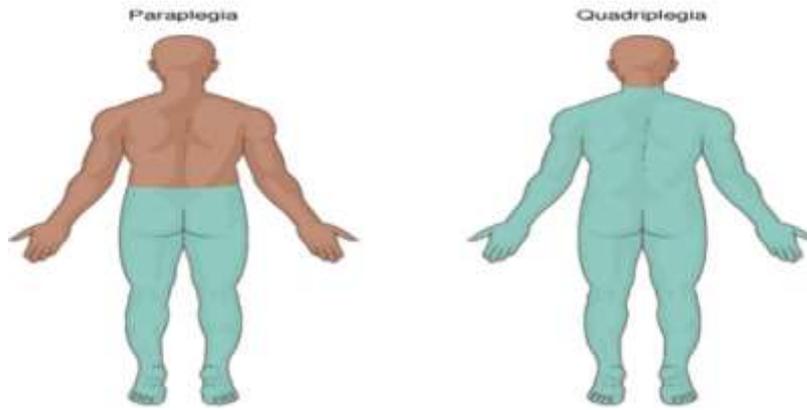


Figura 4-1. Paraplejia y Cuadriplejia

Fuente: <https://mayocl.in/35QwyTJ>

1.6.8 *Visión Artificial*

La Visión Artificial o Visión por Computador es un área multidisciplinar que pretende, en cierta medida, reproducir artificialmente el sentido de la vista mediante el procesamiento e interpretación de imágenes, captadas con distintos tipos de sensores (fundamentalmente, cámaras), y utilizando para ello las prestaciones de los ordenadores. El desarrollo de sistemas de Visión Artificial requiere una combinación de etapas de más bajo nivel (p. ej. operaciones de eliminación de ruido o de aumento del contraste) para mejorar la calidad de las imágenes capturadas junto con otras etapas de más alto nivel (de reconocimiento de patrones y de interpretación de imágenes) para reconocer los elementos presentes en una escena. (Sánchez Calle, 2003, p. 4)

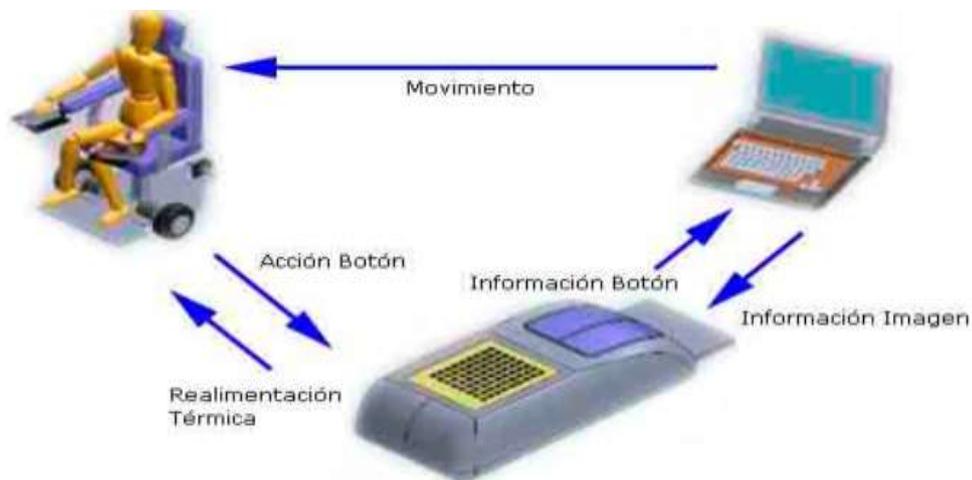


Figura 5-1. Esquema de la Visión artificial para personas con cuadriplejia

Fuente: <https://bit.ly/2Q7P0ke>

1.6.9 Componentes de un sistema de Visión Artificial

Los elementos de hardware mínimos necesarios para un sistema de visión artificial (SVA) son los siguientes:

Sensor Óptico: El sensor óptico puede ser una cámara color o monocromo que produce una imagen completa del dominio del problema cada 1/30 segundos. Los sensores podrían ser también una cámara scanner que produce una línea en cada instante. En este caso el movimiento del objeto por la línea de escáner (o al revés) produce la imagen bidimensional. La naturaleza del sensor y la imagen que produce vienen determinadas por la aplicación.

Tarjeta de adquisición de imagen: permite digitalizar la señal de video entregada por el subsistema anterior.

Computador: Una vez digitalizada la imagen, debe ser almacenada en la memoria de un computador para su posterior procesamiento y manipulación por programa.

Monitor de video: Permite visualizar tanto las imágenes o escenas captadas como los resultados del procesamiento y manipulación por programa. (González Marcos et al., 2006, p. 16)

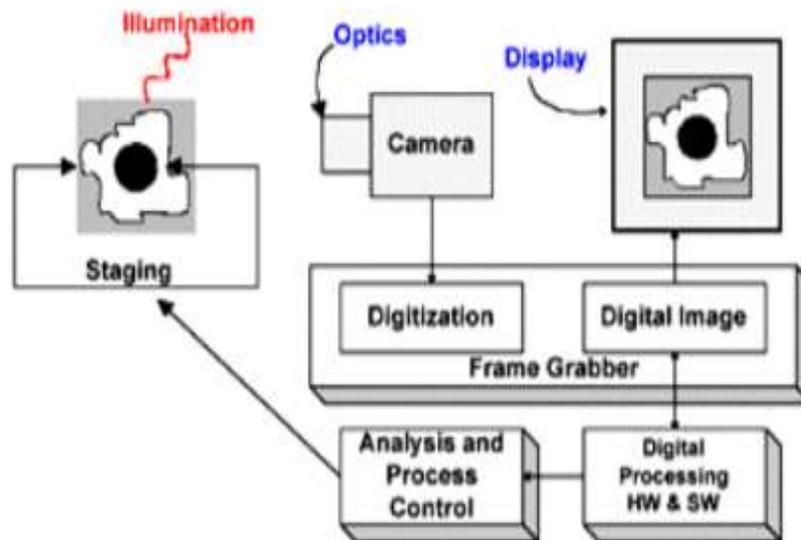


Figura 6-1. Diagrama de bloques de un Sistema de Visión Artificial

Fuente: (GONZÁLES MARCOS ET AL, 2006)

1.6.10 Sistema de reconocimiento facial

El sistema de reconocimiento facial experimentado, lleva a cabo las etapas típicas de los sistemas de visión artificial: pre proceso, segmentación de regiones y líneas de interés,

extracción de descriptores de las regiones y líneas segmentadas y clasificación. Cuando la cara test de un individuo desconocido va a ser reconocida por el sistema, dichas etapas le son aplicadas. Asimismo, estas etapas también se han aplicado en una fase previa, a las caras conocidas por él (conjunto de entrenamiento) durante la creación del mismo. (Sánchez Calle, 2005, p. 13)

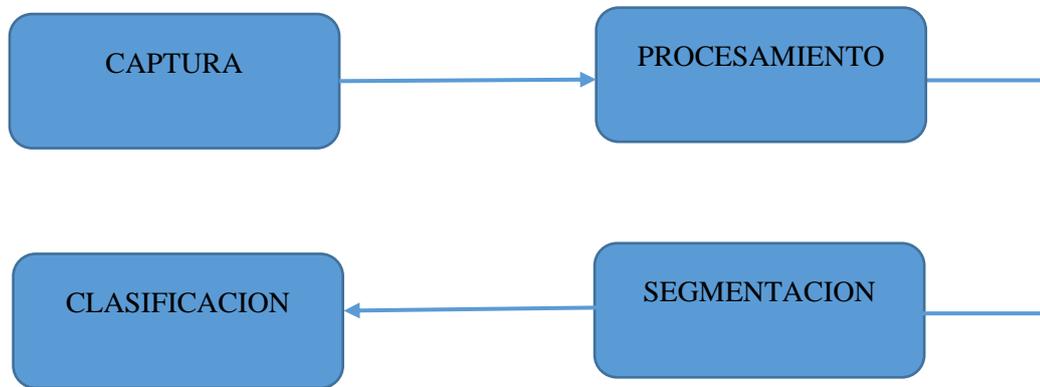


Figura 7-1. Fases de un Sistema de Visión Artificial

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

1.6.11 Métodos de segmentación- Respuesta Espectral

La segmentación de imágenes trata de descomponer una imagen en las partes que la constituyen, es decir, los objetos de interés y el fondo, esto se realiza a partir de unas características locales. La mayoría de las imágenes están constituidas por regiones o zonas que tienen características homogéneas (textura, nivel de gris, etc.). El objetivo de la segmentación es cambiar y/o simplificar la representación de la imagen de forma que la nueva imagen sea más significativa y resulte más fácil de analizar. (Riomoros Callejo, 2016, p. 24)

1.6.12 Reconocimiento Facial

Los humanos utilizan los rostros para identificar a las personas y es por ese motivo que hoy en día se han creado programas computacionales que realicen este proceso mediante la utilización de algoritmos complejos con el fin de identificar un individuo por medio de una imagen digital.

1.6.13 Librerías Java de desarrollo de aplicaciones de Visión por Computador

La librería más popular entre los desarrolladores de aplicaciones de visión por computador es OpenCV. Desde la versión 2.3.0 y hasta la última versión estable (3.0) esta librería de código abierto (open source) proporciona una API Java para su empleo en Android, trasladando a este entorno una parte muy importante de la funcionalidad presente en la librería original programada en C/C++. Existe alguna otra alternativa a OpenCV para el desarrollo sobre Android como es FastCV. Se trata de una librería de visión por computador liberada por Qualcomm optimizada para su ejecución en dispositivos Android aunque se trabaja en una versión para iOS. Esta librería se distribuye en forma de código binario con una única API que presenta dos implementaciones: una para arquitectura ARM (FastCV for ARM) y otra específicamente para SoC de Qualcomm Snapdragon (Fast for Snapdragon). (Pajares y Escalera, 2016, p. 333)

1.6.14 Entorno de LabVIEW

LabVIEW 1.0 fue lanzado al mercado en 1986 por la compañía National Instruments. Desde entonces integró características importantes que aun hoy después de dos décadas de desarrollo y mejoras siguen siendo el núcleo fundamental de este entorno de programación. La instrumentación virtual nace de la utilización del computador como una herramienta de medida y análisis de datos en el laboratorio. Aunque los programas que se desarrollan en LabVIEW se llaman instrumentos virtuales, en su forma más general un instrumento virtual es la unión de un computador con el software y el hardware especializado que le permiten realizar las mismas funciones que un instrumento real. Estas funciones son adquirir, almacenar, analizar, mostrar y comunicar datos y tomar acciones de control sobre determinado proceso cuando así se requiera. Un instrumento virtual no es una simulación, el término virtual se refiere a que el instrumento no existe como un instrumento tradicional por separado si no como parte de un sistema basado en software. Software que, en la pantalla del computador simula la apariencia de un instrumento real. Las ventajas de utilizar instrumentación virtual son esencialmente todas aquellas que se derivan de la utilización de un computador y son entre otras: alta velocidad de procesamiento de datos, capacidad prácticamente ilimitada de almacenamiento de datos, comunicación de datos a diferentes niveles y altas velocidades, toma de decisiones según argumentos lógicos preestablecidos y capacidad comunicación con instrumentos externos o interfases de adquisición de datos por diferentes tipos de interfaces o buses disponibles. (Quiñonez César et al., 2011, p. 112)

1.6.15 *OpenCV*

OpenCV es una biblioteca de visión por computadora de código abierto, la librería esta escrita en lenguaje C y C++ y se ejecuta en Linux, Windows y Mac OS X. Hay un activo desarrollo en interfaces para Python, Ruby, Matlab y otros lenguajes de programación. OpenCV fue diseñado para eficiencia computacional y con un fuerte enfoque para aplicaciones en tiempo real. OpenCV es escrito en lenguaje C optimizado aprovechando los procesadores multinúcleo. (Gary Bradski, Adrian Kaehler, 2008, p. 1)

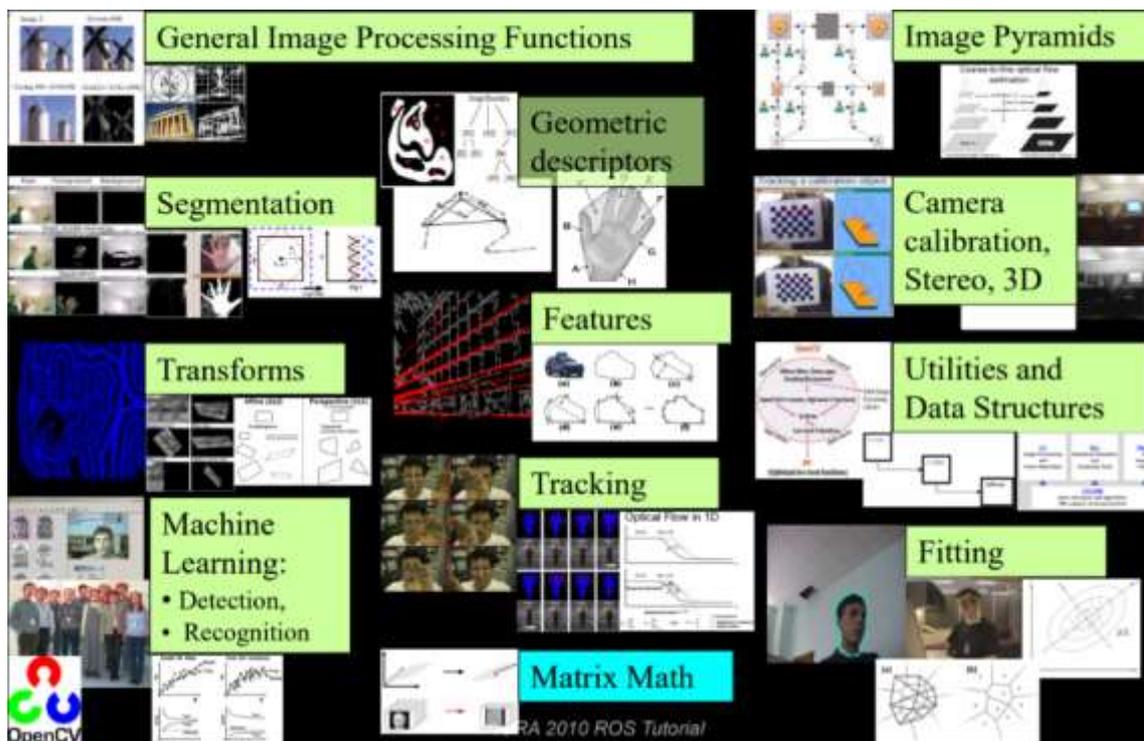


Figura 8-1. OpenCV aplicaciones

Fuente: <https://bit.ly/2TnUfzr>

1.6.16 *Conectividad EJS - Matlab*

Permite analizar y simular multitud de procesos de toda índole. Constituye una plataforma que goza de un soporte técnico y un nivel de desarrollo muy elevado, lo que ha permitido una amplia y creciente difusión en el entorno educativo y más concretamente en las asignaturas relacionadas con el control automático. Uno de los aspectos más significativos radica en la posibilidad de incorporar elementos que permiten la ejecución en tiempo real de los sistemas físicos a través de un sistema de adquisición de datos. De esta manera el tiempo de desarrollo de las aplicaciones que interactúan con los sistemas físicos se reduce considerablemente. Así,

Matlab y Simulink han sido previamente utilizados en diferentes sistemas de desarrollo de laboratorios. Sin embargo, para el acceso remoto al sistema físico, se hace uso de Matlab y más concretamente de Simulink junto con el Real Time Workshop para acceder al sistema físico de manera que se puedan modificar los parámetros de control, así como las condiciones de partida en el desarrollo de nuevos experimentos. El enlace entre Matlab y el entorno desarrollado en EJS se realiza a través de un servidor dedicado a tal efecto que se ha denominado Jim Server. (M.Vallés et al., 2010, pp. 24-25)

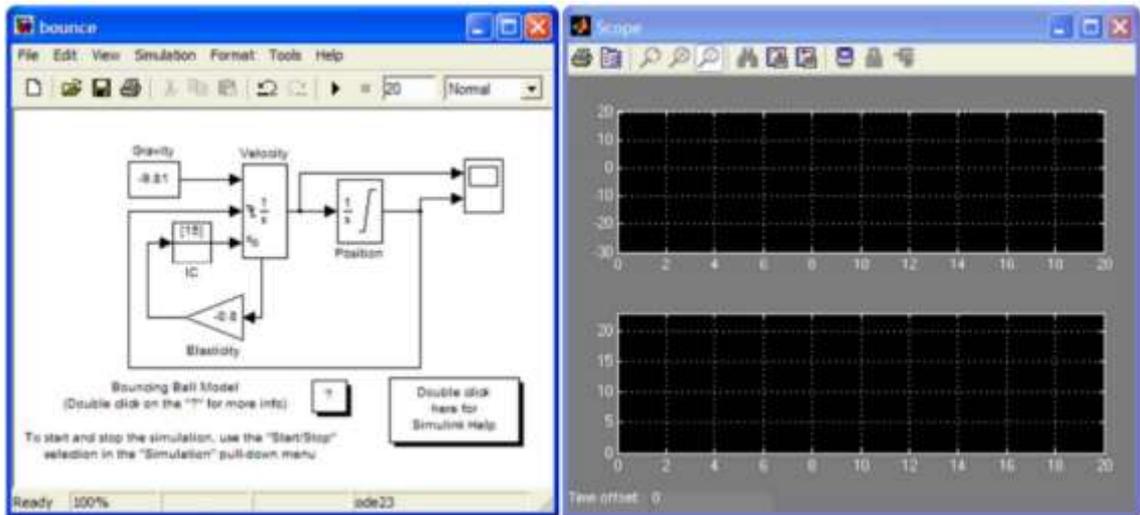


Figura 9-1. Conectividad EJS con Matlab

Fuente: <https://bit.ly/3adcU6W>

1.6.17 Herramientas en LabVIEW

Se puede acceder a las opciones de configuración de LabVIEW a través de Tools > Options . Esta ventana permite modificar muchos parámetros del programa, como:

Paths: las rutas del directorio de instalación de LabVIEW, el directorio temporal, librerías, etc.

Front Panel: varias opciones relacionadas con el Panel Frontal y los controles.

Block Diagram: manejo automático de errores y otras ayudas al programador.

Alignment Grid: permitir y configurar alineamiento a las rejillas.

Controls/Functions Palettes: formas de presentar las paletas de controles y funciones.

Source Control: selección del servidor de versiones y configuración.

Debugging: opciones para depurar errores.

Colors: cambiar el color de varios elementos de LabVIEW.

Fonts: define las fuentes de aplicación, diálogo y sistema.

Printing: opciones a la hora de imprimir.

Revision History: permite guardar información cada vez que se guarda un VI.

Menu Shortcuts: modifica y crea atajos de teclado para las opciones de los menús.

Environment: opciones de configuración misceláneas.

Security: permite restringir el acceso a LabVIEW.

Shared Variable Engine: ordenadores que ofrecen servicios para manejar variables compartidas.

VI Server: configuración del acceso a VI Server. *Web Server:* configuración del servidor web que incorpora LabVIEW. (Lajara Vizcaíno y Pelegrí Sebastián, 2007, pp. 30-31)

1.6.18 Adquisición Digital de la imagen

En el caso de las imágenes digitales, los sistemas de adquisición requieren en primer lugar elemento sensible, que generalmente está constituido por una matriz fotosensible obtenido por el sensor de imagen (CCD, CMOS, etc.). Estos dispositivos físicos dan una electricidad salida proporcional a la intensidad luminosa que recibe en su entrada. El número de elementos del sistema fotosensible de la matriz determina la resolución espacial de la imagen capturada. Además, la señal eléctrica generada por los elementos fotosensibles es muestreado y discretizado para ser almacenado en una ranura de memoria; Esto requiere el uso de un convertidor analógico-digital (ADC). El número de bits utilizados para almacenar la información de la imagen determina la resolución a la intensidad de la imagen. (Folea, 2011, p. 298)

1.6.18.1 Vision Express

Vision Acquisition Express. Este bloque se encuentra en la caja de herramientas de Vision / Vision Express y es la forma más fácil de configurar todas las características de la cámara. Dentro de este bloque hay cuatro secciones: la primera corresponde a la opción de "seleccionar fuente de adquisición" que muestra todas las cámaras conectadas en la computadora. La siguiente opción se llama "seleccionar tipo de adquisición ", que determina el modo para mostrar la imagen y hay cuatro modos: adquisición única con procesamiento, adquisición continua con procesamiento en línea, adquisición finita con procesamiento en línea y

adquisición finita con procesamiento posterior. Los la tercera sección corresponde a "configurar ajustes de adquisición" que representa el tamaño, brillo, contraste, gamma, saturación, etc. de la imagen y finalmente en la última opción es posible seleccionar controles e indicadores para controlar diferentes parámetros de la última sección durante el proceso. (Folea, 2011, p. 299)

1.6.19 Sistema de captura de imagen (cámara)

Adecuando una cámara digital compatible con LabVIEW que produce una imagen completa del objeto aplicando la técnica de iluminación direccional frontal, donde la cámara se ubica en la misma dirección que la luz y recibe la luz reflejada por el objeto. Utilizando el módulo IMAQ USB, que proporciona los instrumentos virtuales necesarios para establecer el funcionamiento y adquisición de imágenes a través de cámaras USB de bajo costo. (Prieto et al., 2010, p. 6622)



Figura 10-1. Interfaz IMAQ

Fuente: <https://bit.ly/30mulxJ>

1.6.20 Plataforma Arduino

La plataforma electrónica Arduino considerada de la gama open source, es decir de fuente libre que permite establecer una relación con su entorno adquiriendo información de su entorno por medio de su extensa gama de sensores y puede afectar a aquello que le rodea controlando diferentes tipos cargas o actuadores. (Paredes, 2014).

Se lo programa mediante el lenguaje de programación Arduino basado en Wiring y el entorno de desarrollo Arduino basado en Processing. Permite realizar la comunicación con diferentes

tipos de software y una vez cargado el programa no necesariamente debe estar conectado a un computador para ejecutar el programa cargado. (Arduino, 2013).

La descarga del software para programación del Arduino es libre de forma gratuita y existen versiones para los diferentes sistemas operativos como Windows, Mac OS X, y Linux (Arduino, 2013). Como ocurre con las distribuciones Linux, Arduino también proporciona varias ediciones pensadas en diferentes desarrolladores o para una serie de tareas específicas. (Paredes, 2014).



Figura 11-1. Modelos de Arduino

Fuente: <https://n9.cl/u1tv>

En el mercado existe una amplia gama de modelos de Arduino, en los que cada uno de ellos difieren de acuerdo al número de entradas y salidas, capacidad de memoria, entre otros recursos se destacan en el mercado el Arduino UNO y el Arduino MEGA.

Existen placas que se las puede considerar oficiales, aquellas fabricadas en empresas tales como Smart Projects de Italia y algunas que han sido diseñadas por la empresa estadounidense

SparkFun Electronics (SFE) o Gravitech como ejemplos, incluyen el logo y son las únicas que pueden llevar la marca registrada de Arduino. Pero así también existen placas no oficiales, incluyen en su nombre “duino” para identificarlas, como por ejemplo Freeduino y Funduino. (Paredes, 2014).

1.6.20.1 Características Generales



Figura 12-1. Partes fundamentales del Arduino

Fuente: <https://n9.cl/cvep3>

Entre las principales características del Arduino se citan las siguientes:

Para la selección del modelo de Arduino se debe considerar que cada uno de estos dispone de un cierto número de puertos digitales y analógicos, según el modelo las digitales pueden ser del tipo normal, actuar como entradas o salidas y también como generadoras de señales de PWM o modulación por ancho de pulso que permiten simular una salida analógica. (Paredes, 2014)

Se debe proyectar la extensión del código para considerar la cantidad de memoria flash que se requiere pues un programa muy largo, con muchas constantes y variables demandará una cantidad mayor de memoria. (Paredes, 2014)

Además, la selección de la placa se vincula con el microcontrolador que en Arduinos oficiales se pueden observar de dos tipos, los de 8 y 32 bits basados en ATmega AVR y los SMART basados en ARM de 32 bits y con un rendimiento superior, ambos creados por la compañía Atmel, estos poseen memoria RAM encargada de cargar los datos para su inmediato procesamiento, el requerimiento de velocidad de respuesta será el parámetro a considerarse para su selección. (Paredes, 2014).

CAPÍTULO II

2. PROPUESTA Y DISEÑO DEL PROTOTIPO

El presente trabajo expone el desarrollo de un sistema basado en visión artificial para asistencia a personas que sufren de cuadriplejía mediante la implementación de un prototipo para el control de ambiente de una habitación, se tiene como finalidad la inserción de plataforma de hardware y software disponibles en el medio.

Se adopta como metodología para la realización del proyecto el conjunto de pasos descritos a continuación:

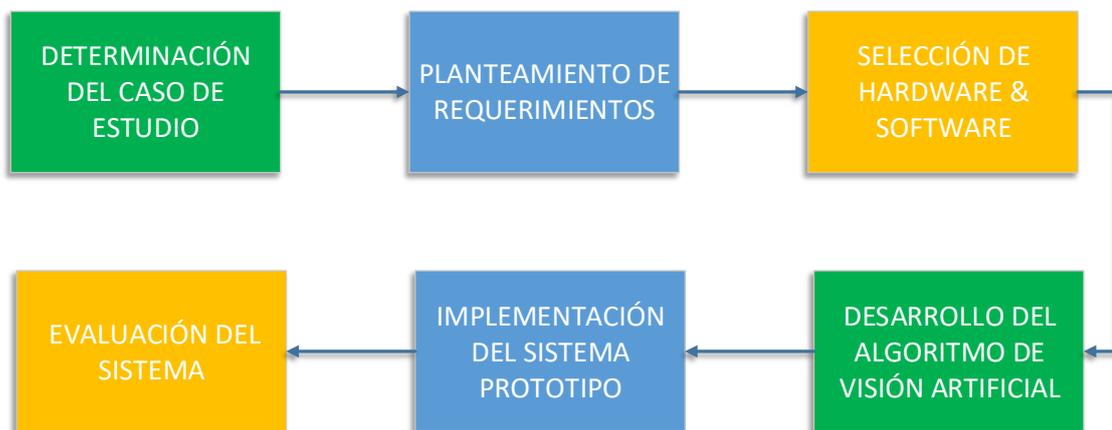


Figura 13-2. Metodología

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

El flujo de actividades planteadas están enfocadas al desarrollo segmentado del proyecto, de manera que a medida que se sigue avanzando se van cumpliendo ciertos objetivos y proyectándose al cumplimiento de otros.

2.1 Determinación del caso de estudio

Según lo establecido en el tema del proyecto el sistema está orientado a brindar asistencia a personas con un grado de cuadriplejía del 80%, de acuerdo al marco teórico expuesto en el capítulo anterior existen diferentes niveles de lesiones, este porcentaje se vincula con la lesión C4 cuadriplejía/tetraplejía que involucra parálisis completa por debajo del cuello.

Lesiones Medulares

Lesión C4

Cuadriplejia/Tetraplejia
Parálisis completa por debajo del cuello

Lesión C6

Parálisis parcial de brazos y piernas
Parálisis parte baja del cuerpo

Lesión T6

Paraplejia
Parálisis por debajo del tórax

Lesión L1

Paraplejia
Parálisis por debajo de la cintura

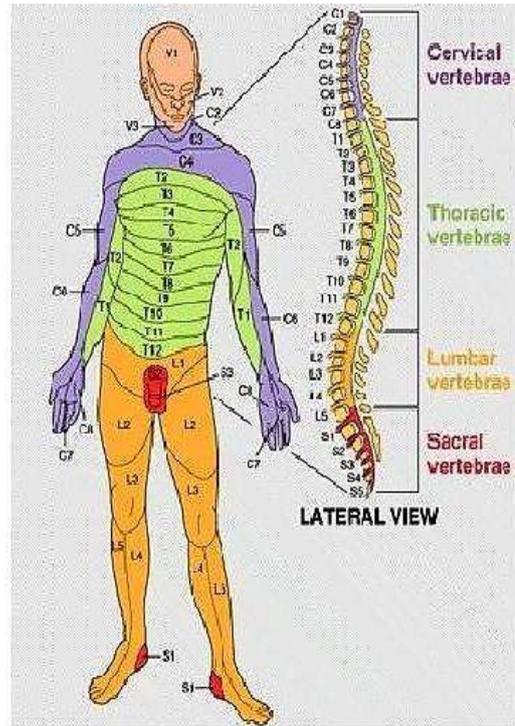


Figura 14-2. Niveles de Lesión – Caso de estudio Lesión C4

Fuente: <https://n9.cl/sms7u>

El sistema se desarrollará bajo esta consideración de caso específico de estudio tomando en cuenta que el paciente tendrá movimiento en el cuello (movimiento libre de la cabeza).

Los desplazamientos que realice el paciente a partir del cuello, servirán para determinar la forma de interacción paciente - sistema para la manipulación del ambiente de su habitación, esta información resulta importante al momento de desarrollar los algoritmos programados.



Figura 15-2. Movimientos del cuello

Fuente: <https://n9.cl/xjix>

2.2 Definición de requerimientos

El objetivo del trabajo es crear un nivel de independencia para personas con una cuadriplejía del 80%, por medio de la inserción de un sistema basado en visión artificial para el control de forma independiente cargas eléctricas vinculadas a elementos del ambiente de la habitación donde se encuentre el paciente.

De acuerdo a lo expuesto, para la presentación de una solución, una forma para brindar confort a este tipo de pacientes se plantean los siguientes requerimientos específicos por medio del siguiente diagrama de bloques:

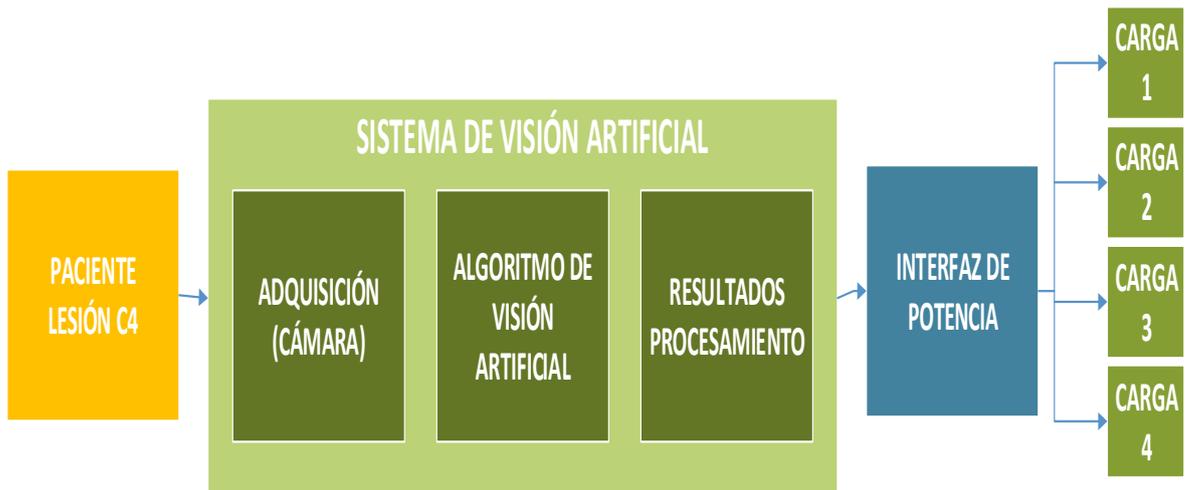


Figura 16-2. Diagrama del proceso

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

El paciente razón de ser del proyecto, resulta ser el factor esencial en este sistema, pues es la fuente de información para el diseño e implementación del sistema.

Como herramienta para el desarrollo del sistema de asistencia se ha propuesto la visión artificial por lo que el sistema plantea como requerimiento inicial para la adquisición de la información del paciente (movimientos / gestos a partir del cuello) una cámara, para que posteriormente éste elemento proporcione el flujo de imágenes requeridas por el sistema para ser procesadas mediante un algoritmo de visión artificial, resultado de la acción del algoritmo sobre la información adquirida se podrán generar señales de control para el manejo de cuatro cargas planteadas en el ambiente controlado en el que se evaluará el sistema.

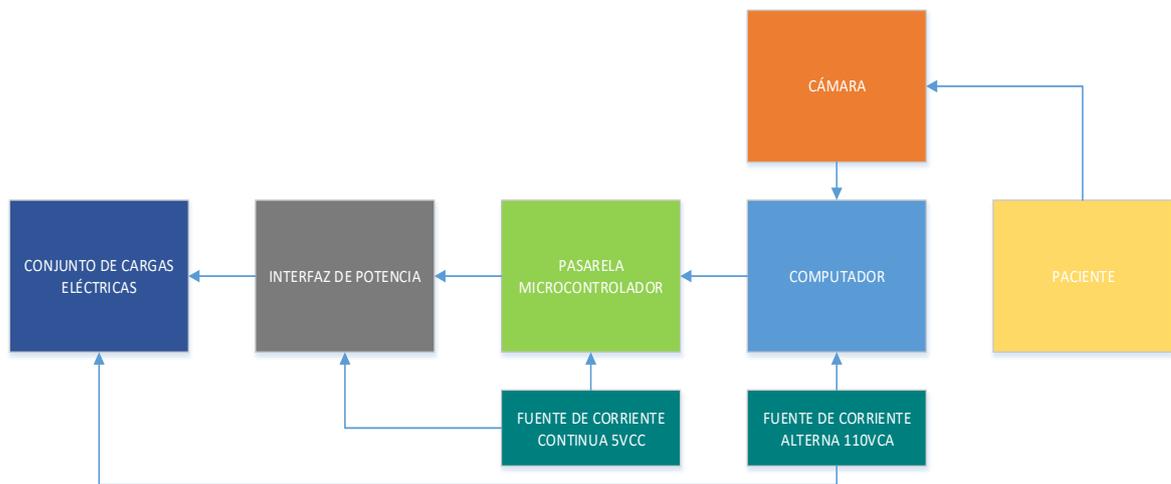


Figura 17-2. Etapas físicas del sistema

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Los requerimientos planteados generaran un proceso de selección de hardware y software que permitan cubrirlos, la figura 17-2 describe las etapas del sistema con una perspectiva de los elementos hardware necesarios para la implementación.

2.2.1 *Propósito del Sistema*

El sistema prototipo desarrollado espera mejorar las condiciones de vida de personas que padecen cuadriplejía hasta del 80%, es decir con una lesión C4 por medio de la inserción de recursos tecnológicos y procesamiento de imágenes, generando un grado de independencia para estas personas, e incursionando en el campo para servir como base de futuros proyectos que busquen brindar una vida de confort para pacientes que tal vez consideran que lo han perdido todo.

2.2.2 *Definición de variables*

En la descripción del sistema se mencionaron acciones controladas por medio de los movimientos específicos, estos se interpretan como variables de control, mientras que las cargas asignadas para la implementación serán las variables controladas.

A continuación en la tabla se definen las variables del sistema.

Tabla 2-2: Variables del sistema.

VARIABLES DE CONTROL	VARIABLES CONTROLADAS
MOVIMIENTO 1	CORTINA SUBIDA - CORTINA BAJADA
MOVIMIENTO 2	LÁMPARA
MOVIMIENTO 3	SIRENA –ALERTA
MOVIMIENTO 4	VENTILADOR

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

2.3 Selección del equipamiento

De acuerdo al antecedente planteado en la forma deseada de funcionamiento del sistema y la definición de variables del sistema en este punto se realiza la selección tanto del hardware como del software para conseguir alcanzar lo planteado.

2.3.1 Equipamiento del sistema de control

El sistema de control estará basado en un algoritmo de visión artificial, se analizó la posibilidad de usar una minicomputadora, como el caso de una Raspberry Pi por su antecedente en aplicaciones de visión artificial pero se valoró las limitaciones de la misma en el caso de desarrollar un algoritmo complejo debido a su velocidad de procesamiento. Debido a esta consideración se optó por hacer uso de un computador a disposición de las siguientes características.

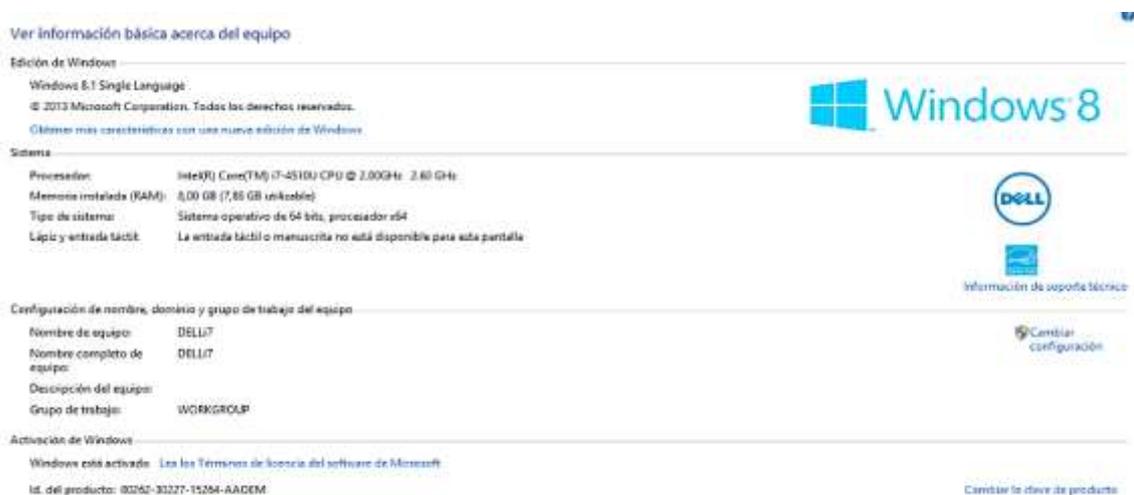


Figura 18-2. Características ordenador empleado para desarrollo del sistema.

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

2.3.1.1 Cámara adquisición de información para el sistema de control

Al haber seleccionado el ordenador con el que se va a trabajar se caracterizó a la cámara disponible mediante un test efectuado en la plataforma abierta anclada al URL <https://es.webcamtests.com/>.



Figura 19-2. Caracterización de la Cámara disponible

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Mediante esta evaluación se obtiene la Tabla 3-2 de resultados, con parámetros útiles posteriormente a ser considerados para el desarrollo del algoritmo de visión artificial. La cámara juega un papel importante dentro del sistema, pues será la que sirva para la adquisición de la información necesaria.

Tabla 3-2: Características de la cámara.

Información de la cámara web	
Nombre de la webcam:	Integrated Webcam
Calificación de calidad:	53
Micrófono incorporado:	None
Altavoz incorporado:	None
Cuadros por segundo:	30 FPS
Tipo de corriente:	Video
Modo de imagen:	RGB
Webcam Megapíxels:	0.92 MP
Resolución de cámara web:	1280x720

Video estándar:	HD
Relación de aspecto:	1.78
Tamaño de archivo PNG:	812.21 kB
Tamaño de archivo JPEG:	438.47 kB
Velocidad de bits:	12.85 MB/s
Cantidad de colores:	9885
Color RGB promedio:	
Ligereza:	23.73%
Luminosidad:	24.82%
Brillo:	24.05%
Matiz:	69°
Saturación:	5.79%

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

2.3.1.2 Selección de la pasarela del sistema de control con el entorno físico

El ordenador se enlazará a un Arduino mediante comunicación serial, habilitando de esta manera una comunicación de dos vías (RX-TX), el algoritmo de visión artificial deberá incluir sentencias de ejecución de instrucción hacia el puerto serial para ser enviadas al Arduino.

El porqué del término pasarela para el Arduino, pues se lo interpreta como un medio para relacionar la parte abstracta en este caso el algoritmo programado con el medio físico que en este caso relaciona las cargas a controlarse dentro de la habitación en la que se montará el piloto.

El sistema de control estará albergado en un ordenador, el control directo de las cargas resulta imposible por lo que se incluye el uso de un dispositivo de uso común y disponible en el mercado como es un Arduino.

Para la selección del modelo de tarjeta se parte por la revisión de la hoja de especificaciones técnicas y mediante un análisis comparativo de características de la placa versus requerimientos del sistema se determina el que se va a usar. ANEXO A, ANEXO B.

Tabla 4-2: Valoración de características placas Arduino

Requerimientos	Arduino UNO	Arduino MEGA
Comunicación Serial	CUMPLE	CUMPLE
Manejo de 4 Cargas	14 pines digitales	54 pines digitales
Precio	18usd	30usd
Control directo de cargas	No	No

Monofásicas de 110VCA	Requiere interfaz de potencia	Requiere interfaz de potencia
-----------------------	-------------------------------	-------------------------------

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

En este caso, se consideró idónea para implementación de la pasarela del sistema la tarjeta Arduino UNO, principalmente por análisis de costo y el número de pines digitales que se van a requerir pues en la definición de variables se habla de un control ON/OFF de cuatro cargas por lo que resultan suficientes y con un factor de sobredimensionamiento las de esta placa, además que permite realizar comunicación serial, a continuación se presenta una descripción más amplia de las características de esta placa.

Características Arduino UNO

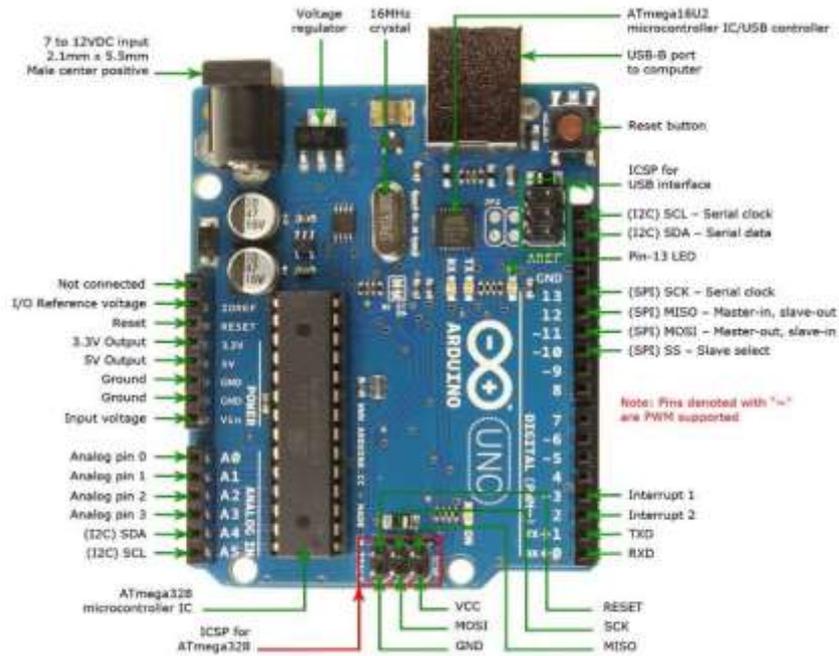


Figura 20-2. Recursos - Arduino (UNO)

Fuente: <https://n9.cl/eu8o>

- Microcontrolador: ATmega328
- Voltaje Operativo: 5V
- Voltaje de Entrada: 7-12V
- Voltaje de Entrada (límites): 6-20V
- Pines digitales de Entrada/Salida: 14 (de los cuales 6 proveen salida PWM)
- Pines analógicos de entrada: 6

- Corriente DC por cada Pin Entrada/Salida: 40 mA
- Corriente DC entregada en el Pin 3.3V: 50 mA
- Memoria Flash: 32 KB (ATmega328) de los cuales 0,5 KB es usado por Bootloader.
- Memoria SRAM: 2KB (ATmega328)
- Memoria EEPROM: 1KB (ATmega328)
- Velocidad de Reloj: 16 MHz (ATmega328)

A la placa Arduino UNO se le puede suministrar alimentación desde el puerto USB o con una fuente externa de poder. Al trabajar con una fuente externa de poder se requiere una etapa de rectificación y de regulación de dicho voltaje en el rango operativo de la placa. De igual manera se puede alimentar el micro mediante el uso de baterías. Preferiblemente el voltaje debe estar en el rango de los 7V hasta los 12V. (Sabika, 2010)

Arduino UNO posee algunos pines para la alimentación del circuito adicional al adaptador para la alimentación:

- VIN: A través de este pin es posible proporcionar alimentación a la placa.
- 5V: Podemos obtener un voltaje de 5V y una corriente de 40mA desde este pin.
- 3.3V: Podemos obtener un voltaje de 3.3V y una corriente de 50mA desde este pin.
- GND: El ground (0V) de la placa.

2.3.1.3 Definición de Actuadores

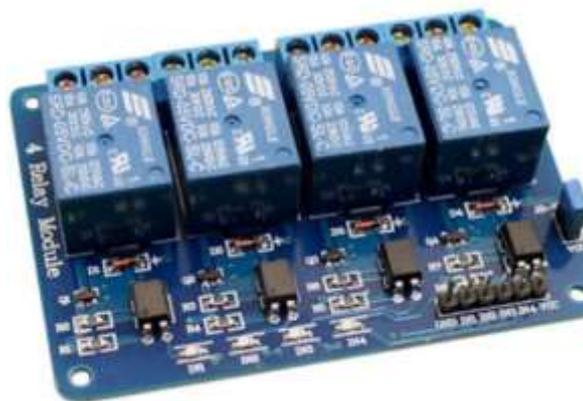


Figura 21-2. Módulo de Relés Arduino

Fuente: <https://n9.cl/eu8o>

Los actuadores se los definió en la tabla de variables controladas del sistema, en este punto se hace mención a la interfaz de potencia que será utilizada para usar las señales de bajo voltaje de los pines de salida del Arduino UNO para gestionar el control de los actuadores de corriente alterna del ambiente de la habitación del paciente.

Al tratarse de cargas monofásicas de 110 VCA se selecciona como interfaz de potencia entre el Arduino y los actuadores un módulo de relés de la gama de dispositivos para Arduino, que consiste en un sistema embebido que capta la señal digital de los puertos del Arduino y gestionan en el caso que requiera la conmutación de los contactos del o los relés correspondientes para determinar el funcionamiento de las cargas.

2.4 Algoritmo de Visión Artificial

2.4.1 Selección del Software para el desarrollo del Algoritmo de V.A.

En la actualidad la visión por computador representa un eje de desarrollo en todas las áreas de estudio, lo que generó la creación de varias plataformas o la anexión de componentes a plataformas existentes con recursos para desarrollo de algoritmos de visión artificial tales como LabVIEW, Matlab y Open CV, entre otros como se los mencionó en el marco teórico revisado anteriormente.

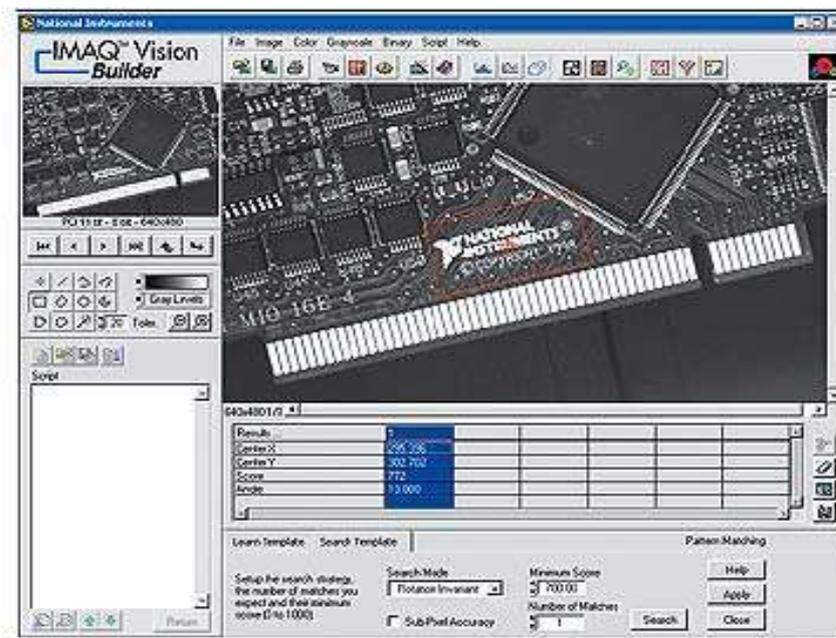


Figura 22-2. Librería IMAQ Vision

Realizado por: <https://n9.cl/eu8o>

Para el desarrollo práctico del algoritmo de visión artificial del sistema propuesto se seleccionó LabVIEW por su potencial al ofrecer herramientas para la adquisición de información, instrumentación y control.

LabVIEW incluye en su amplia gama de librerías a IMAQ Vision, una biblioteca con excelentes recursos para la detección de bordes, reconocimiento de patrones, aplicación de filtros, segmentación de imágenes, etc., es decir el procesamiento de visión o procesamiento digital de imágenes.

Además LabVIEW incluye el Vision Assistant que consiste en un entorno didáctico para el desarrollo de prototipos, en este caso se acopla a la necesidad del proyecto.

2.4.2 Desarrollo del sistema de Visión Artificial

Para el desarrollo del algoritmo de visión artificial como requerimiento indispensable se plantea la instalación de LabVIEW, esta ocasión se instaló la versión 19.0.

Adicionalmente, una vez instalado LabVIEW resulta indispensable la instalación de los complementos para el manejo de herramientas para ejecución de visión artificial. Se requiere instalar *NI Modulo Development Vision* necesario para la configuración de cámaras, personalizar análisis de imagen y generar resultados para pruebas de producción y validación.

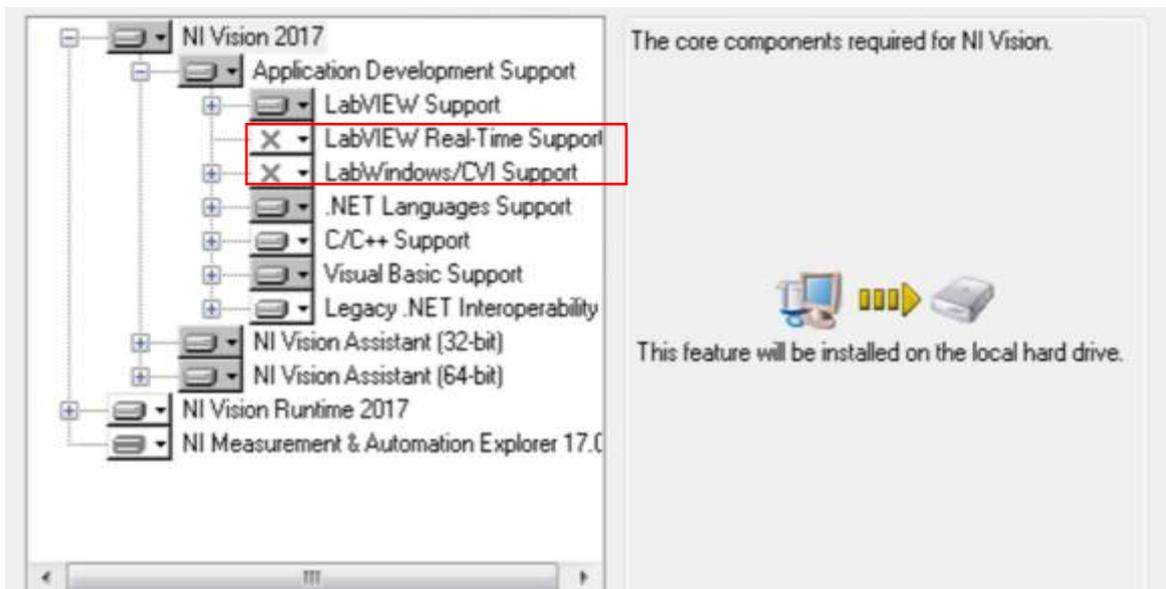


Figura 23-2. Pantalla instalación NI Modulo Development Vision

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Durante la instalación del *NI Modulo Development Vision* se recomienda habilitar dos de sus complementos que aparecen desactivados como se observa en la Figura 23-2.

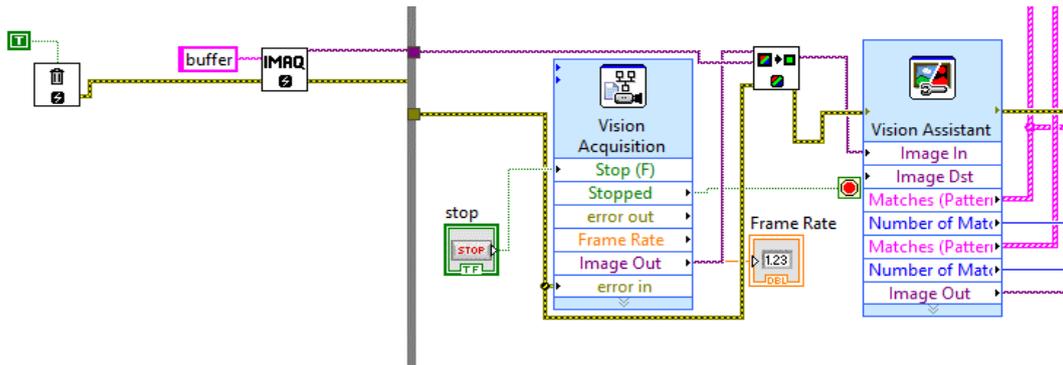


Figura 24-2. Caracterización de la Cámara disponible

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

También resulta necesaria la instalación del componente *NI Vision Acquisition Software*, es el controlador para adquirir, visualizar y guardar imágenes desde una gran variedad de tipos de cámaras.

El algoritmo de visión artificial parte de la etapa de adquisición de imágenes la figura muestra los bloques que cumplen con esta función, se parte de agregar al bloque *IMAQ* que genera un espacio de memoria temporal para el almacenamiento de las imágenes.

En la Figura 24-2 se observa el Vision Acquisition, este bloque contiene la configuración del hardware empleado para la adquisición de las imágenes, es decir se marcan parámetros para caracterizar la cámara en este caso con la información de la Tabla 3-2.

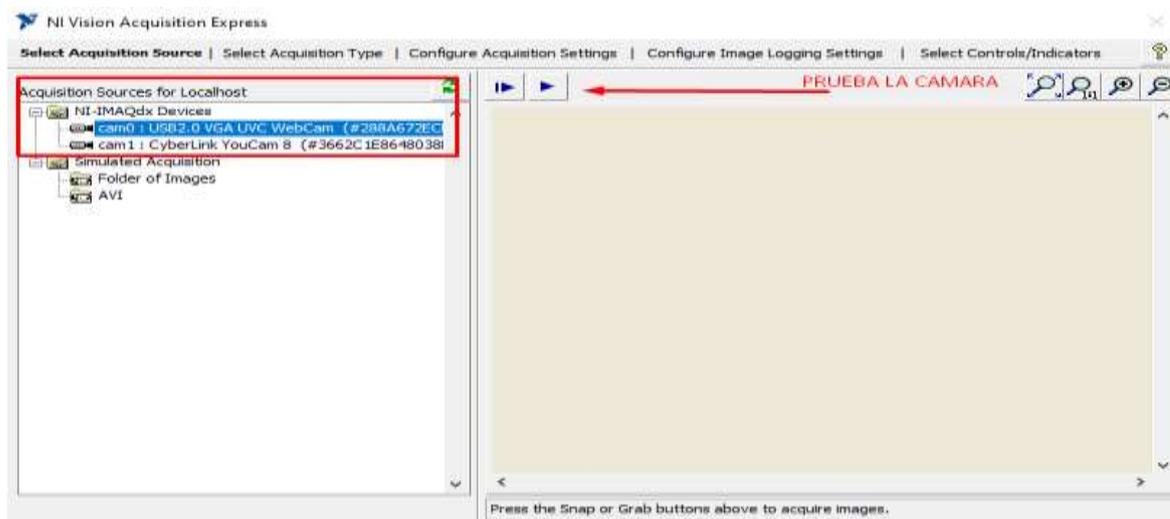


Figura 25-2. Determinación de la Cámara a emplearse

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

La Figura 25-2 es la primera ventana que se apertura al ingresar al bloque Vision Acquisition, aquí se muestran los dispositivos para adquisición de datos conectados al computador, que pueden ser internas o externas. En este caso el hardware que se dispone es una Webcam propia del ordenador asignada en el listado como *cam0*.

En esta ventana se puede realizar también un test para la cámara para verificación de su funcionalidad con la aplicación.

Establecido el hardware con el que se va a trabajar y verificada su funcionalidad en un siguiente paso aparece la ventana de la Figura 26-2 en la que se selecciona el tipo de adquisición de imagen que se desea, en el caso de estudio se emplea una adquisición continua con procesamiento en línea, lo que representa la adquisición de información para su tratamiento en forma paralela.

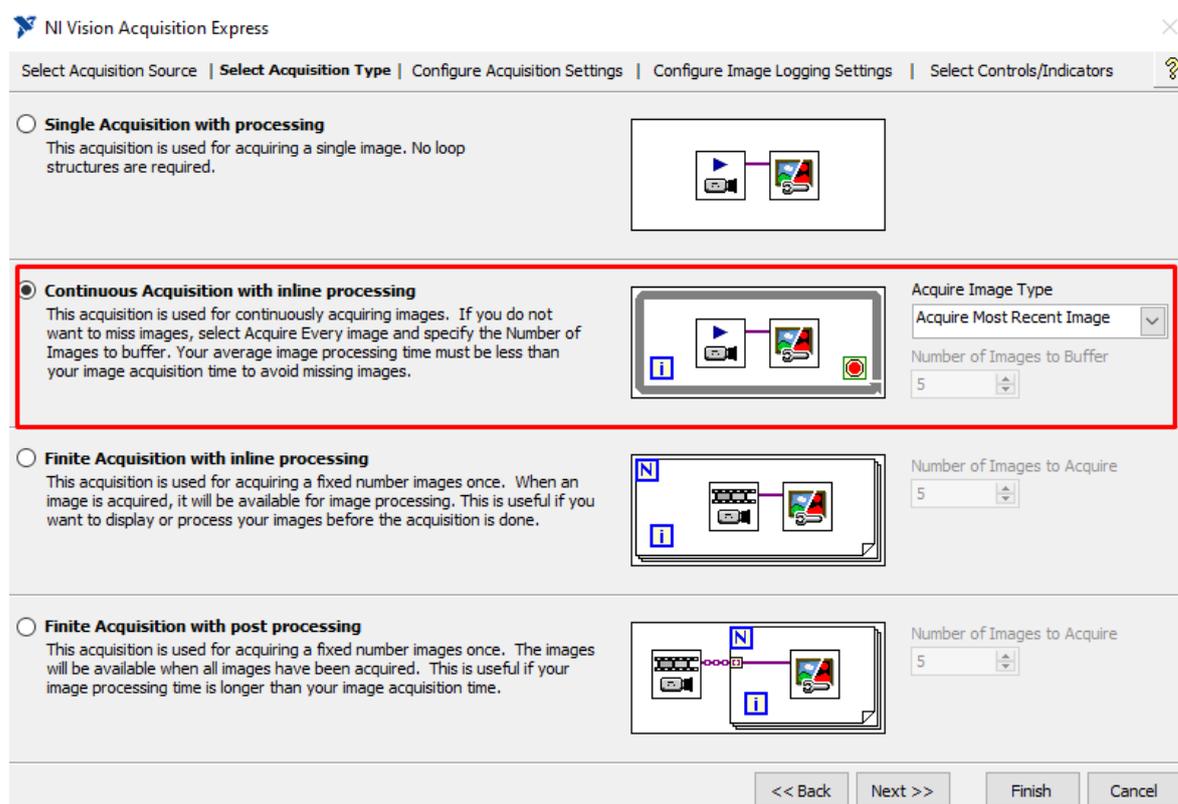


Figura 26-2. Formato de adquisición de imagen

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

En la Figura 27-2 se debe configurar parámetros de la cámara tales como, la calidad máxima de la cámara seleccionadas y los fotogramas por segundo (fps) que va a adquirir, en éste la máxima calidad es de 480p a 30 fps e igualmente permite realizar un test del hardware.

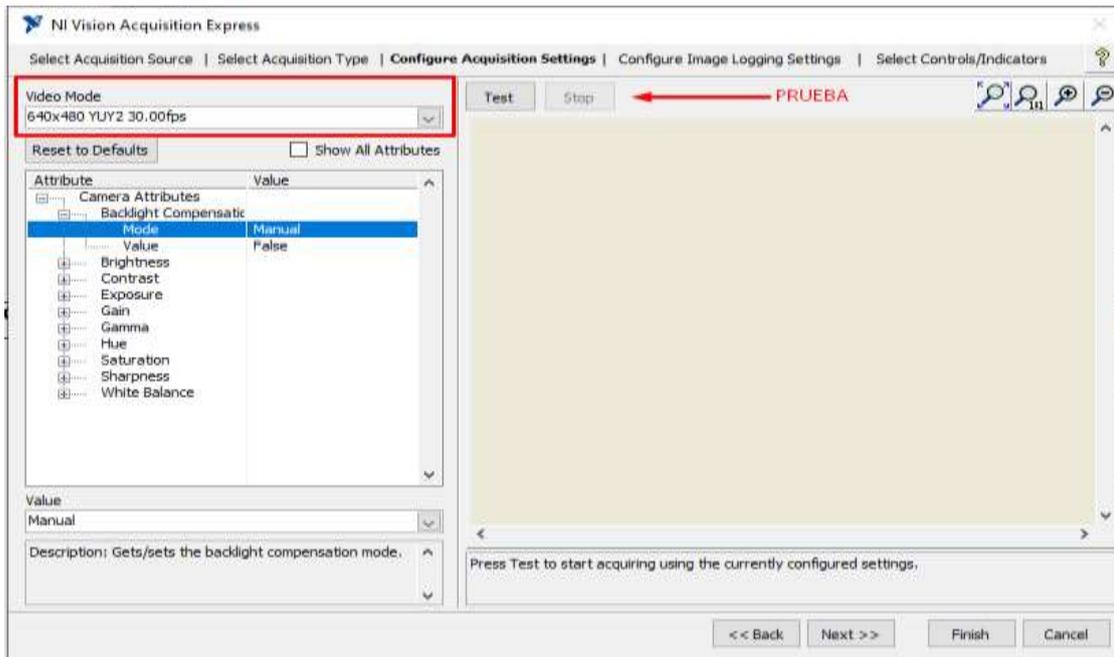


Figura 27-2. Determinación parámetros calidad de imagen

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Finalmente en la configuración de este bloque se puede establecer controles e indicadores para el control y monitoreo del funcionamiento de la cámara por ejemplo el Frame Rate (Figura 28-2) que se lo emplea para tener conocimiento del número de cuadros por segundo procesados.

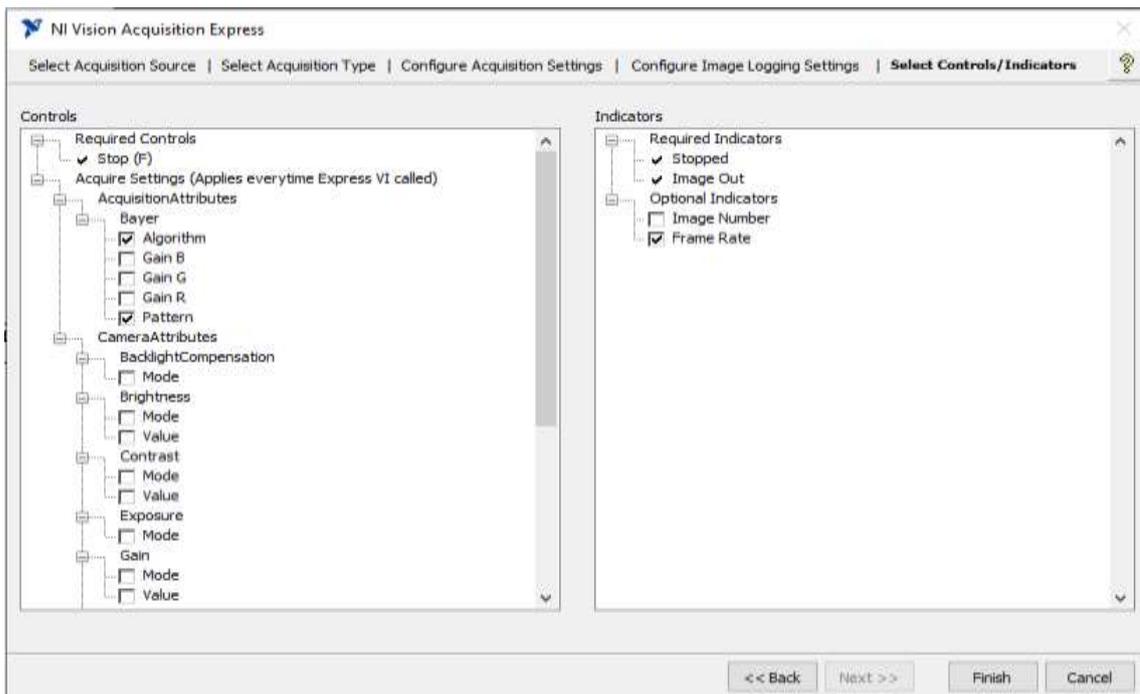


Figura 28-2. Parámetros de control e indicadores.

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

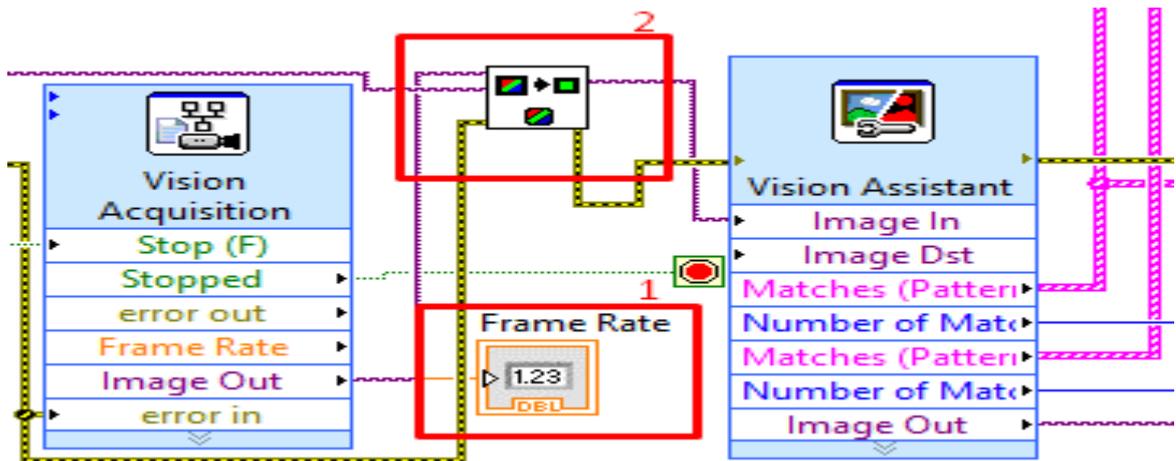


Figura 29-2. Filtro de imagen

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Posterior al bloque que relacionó toda la configuración del hardware se dispone de un filtro de imagen señalado en el recuadro rojo 2, la función de éste es convertir a la imagen en escala de grises para su procesamiento.

Este bloque representa un SubVI que al abrirlo muestra la ventana de la Figura 30-2, su función, filtrar un solo color de un plano, en este caso está seleccionado el de luminosidad que ayuda a mejorar la adquisición de datos en escala de grises o se puede interpretar como una imagen en blanco y negro.

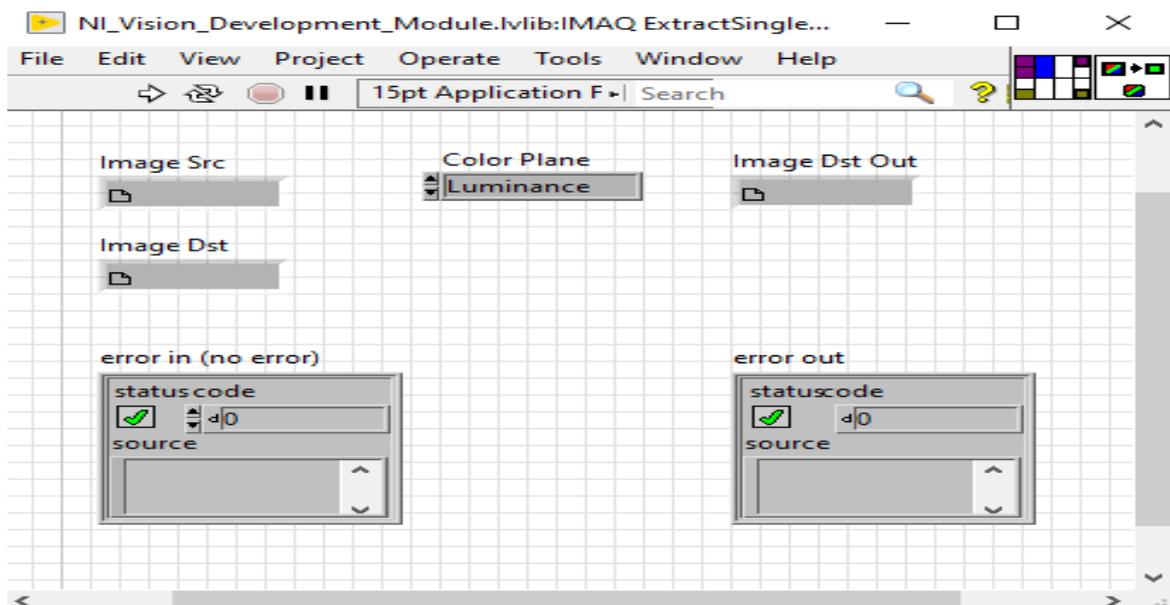


Figura 30-2. Filtro un solo color de un plano

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

En el programa continúa el bloque del Asistente de Vision el cual brinda recursos para procesar de mejor manera las imágenes, entre ellos la posibilidad de fijar puntos específicos de evaluación.

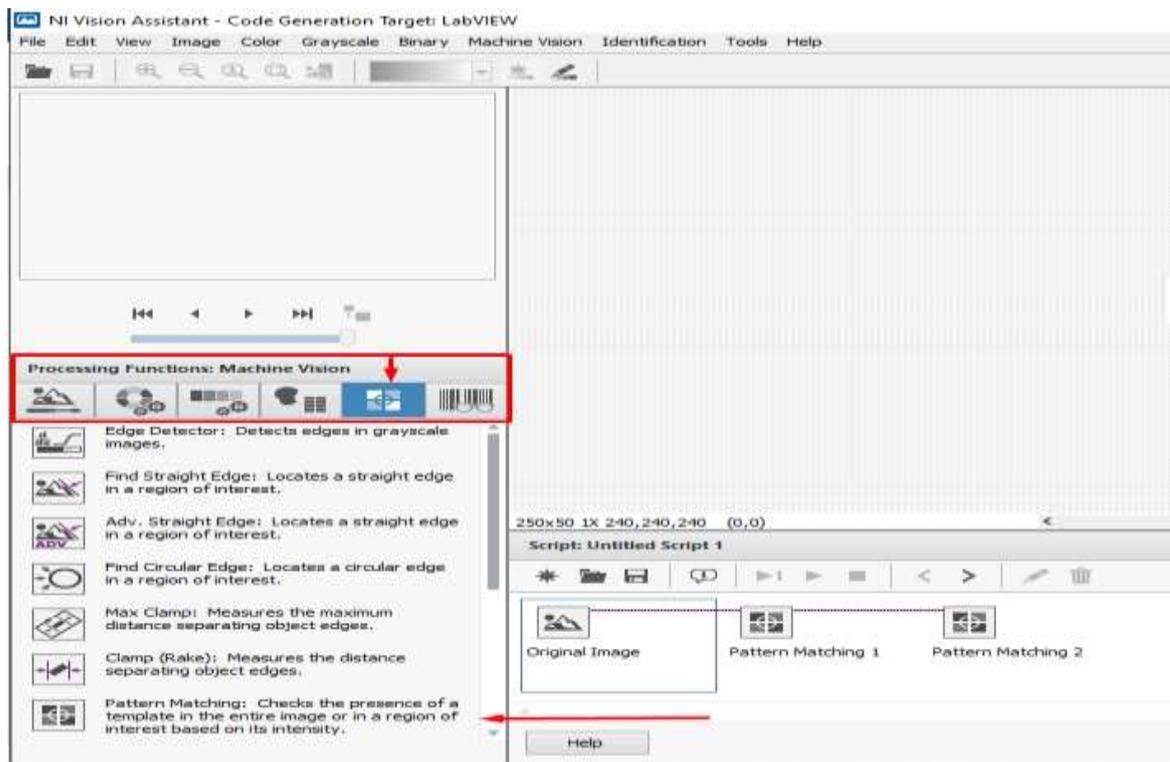


Figura 31-2. Asistente de Visión

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Para dar inicio a la configuración del asistente es necesario fijar al paciente o sujeto de prueba frente a la cámara con el fin de obtener una posición centrada del mismo en la pantalla para obtener una imagen patrón en base a la que se podrá trabajar, en el caso experimental del prototipo de sistema a implementarse se establecen condiciones ideales de luz es decir un ambiente controlado ideal para la valoración.

En la pestaña de la opción Processing Function: Machine Visión se ingresa al panel de Script el recurso Pattern Matching que hace el llamado a un proceso de reconocimiento de patrones en escala de grises, e inmediato aparecerá una nueva ventana, Figura 32-2 donde se dispone crear un New Template, que consiste en generar una plantilla nueva a partir de la imagen patrón inicial.

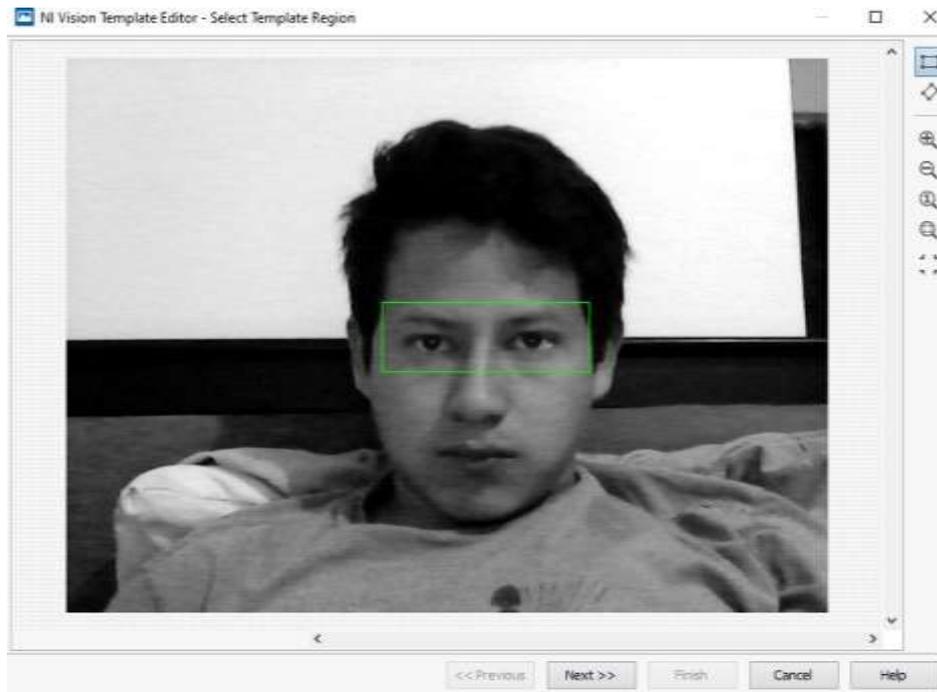


Figura 32-2. Creación New Template

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Creado el Template, especificada la zona de evaluación se define en la ventana de la figura 33-2 qué tan rotada puede estar la plantilla al momento de su evaluación, éste es el único parámetro que se edita debido a que si se mueven los otros parámetros se estaría configurando un método diferente al de encontrar patrones por escala de grises.

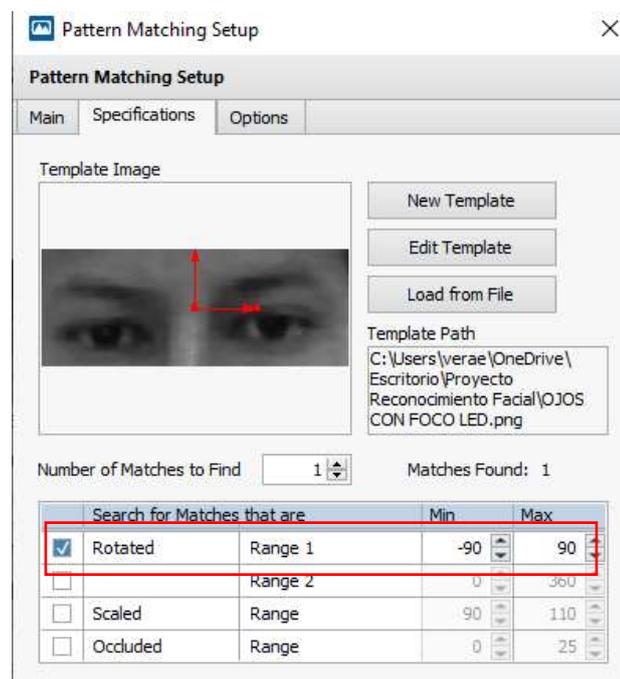


Figura 33-2. Tolerancia de rotación de la plantilla para ser identificada

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Para conseguir mayor precisión en la plantilla creada se selecciona la región que se desea ignorar, finalizando de esta manera la configuración del algoritmo de procesamiento.

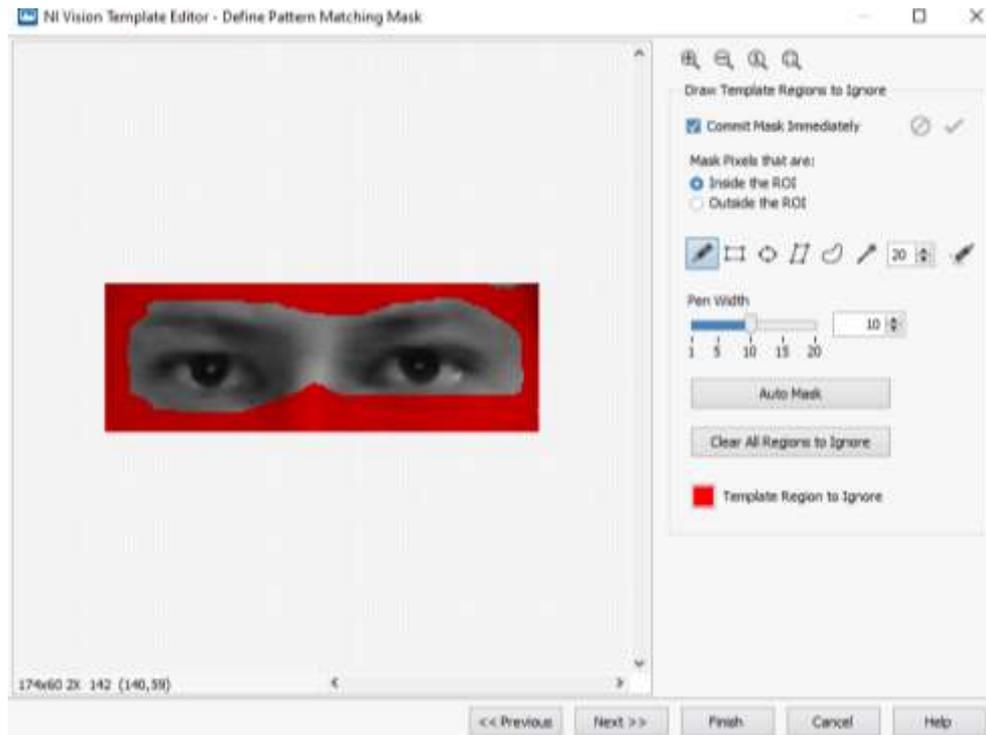


Figura 34-2. Definición área de interés

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

El proceso descrito dentro del bloque del asistente se lo repetirá las veces necesarias para la cantidad de plantillas que se deseen crear. En el caso de este sistema se crearon dos, la zona mostrada de los ojos y una adicional para la boca, Figura 22-2.

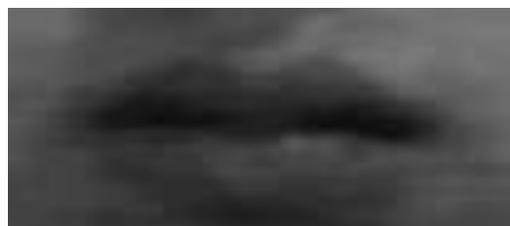


Figura 35-2. Área segunda plantilla

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

El tratamiento de la información del procesamiento de la imagen ejecutado por el asistente de visión en base a las plantillas, parte de crear una representación de los datos adquiridos y reflejarlos en un cuadro de resumen en el panel frontal del VI. Figura 36-2

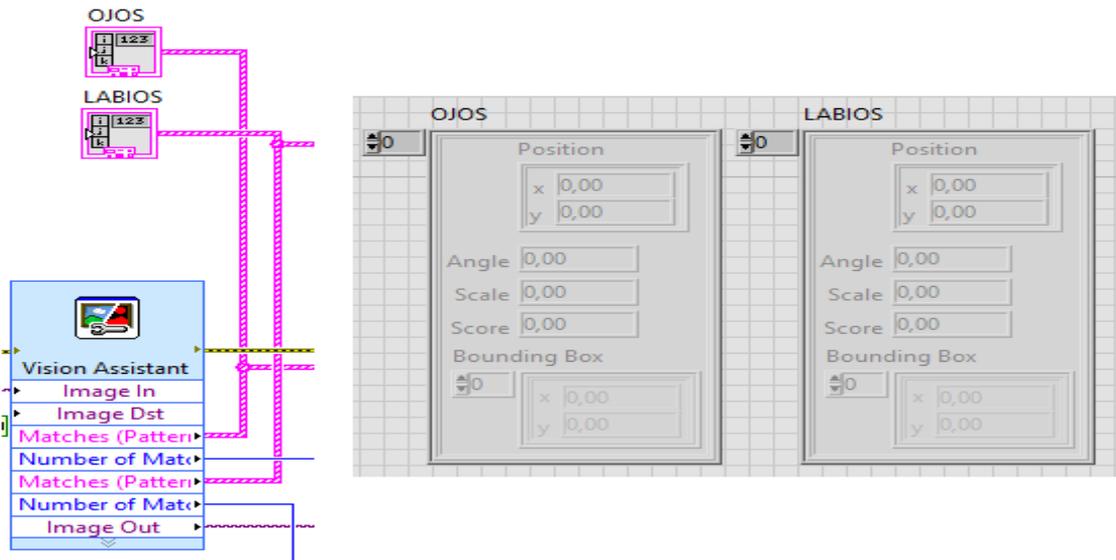


Figura 36-2. Parámetros obtenidos de las plantillas definidas

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Las plantillas reconocidas por el asistente estarán caracterizadas por valores bidimensionales de posición (x, y), un ángulo, una escala y un puntaje, éste último hace mención a los aciertos del algoritmo para la detección de la plantilla en el flujo de imágenes.

La información descrita se obtiene como un **Clúster** desde el asistente, para darle tratamiento a esos datos individualmente y adaptarlos a evaluaciones con el fin cumplir con los requerimientos planteados, se emplea el bloque **Unbundle by Name**, se observa en la figura 37-2 como a las salidas del mismo se obtienen los parámetros anclados a un bloque de visualización (recuadros rojos de la figura 37-2).

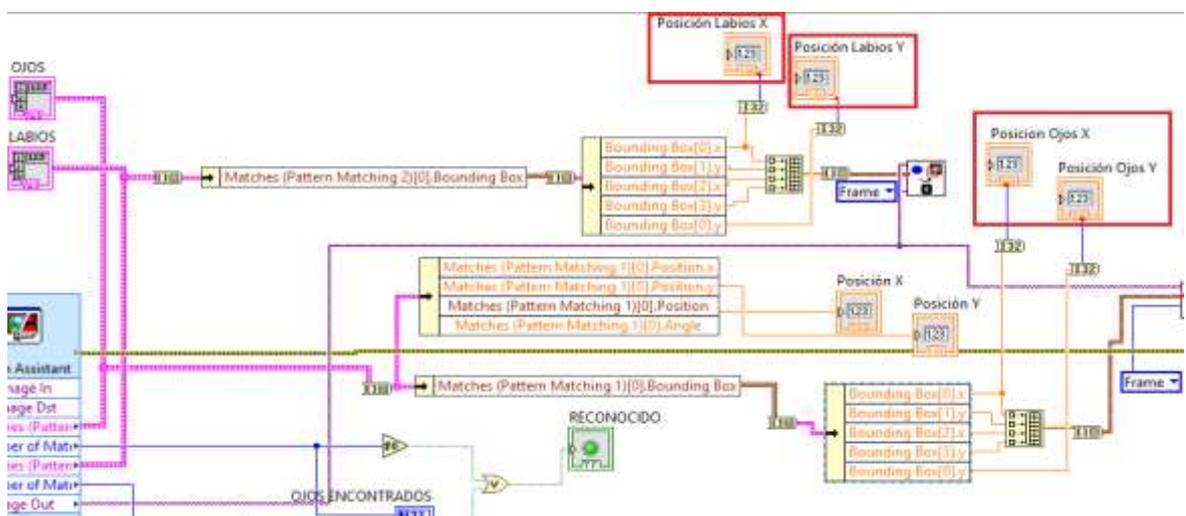


Figura 37-2. Desglose de parámetros del Clúster de las plantillas

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Adicionalmente se agregan bloques de recursos para el monitoreo del procesamiento en tiempo real, estos son el Image Out y Overlay Rectangle, el primero permite visualizar el flujo de imágenes adquiridas y procesadas mientras el que el segundo dibujará un rectángulo de color verde en este caso, para marcar el área identificada.

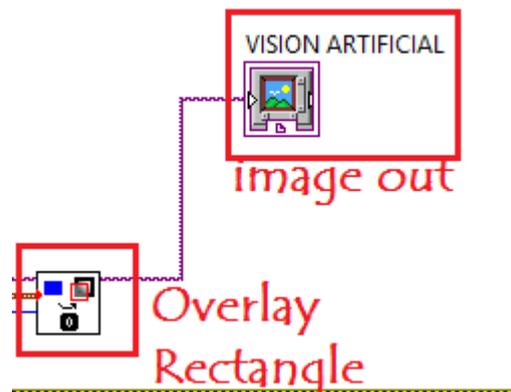


Figura 38-2. Recursos monitoreo del proceso

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

2.5 Vinculación del algoritmo de Visión Artificial con el de Control

Recordando que el caso de estudio se trata de una persona con cuádruplejía del 80% y que el propósito es controlar el ambiente de su habitación con los movimientos de la cabeza, se define que el par ordenado (x, y) obtenido de la posición para cada marcador o plantilla planteado en el algoritmo de visión resulta ser la información primordial, pues por medio de estos valores se determinarán intervalos específicos para el control.

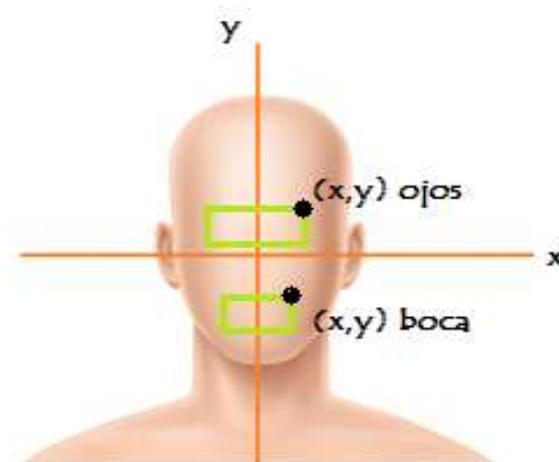


Figura 39-2. Posición de los marcadores

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Las coordenadas rectangulares de los marcadores servirán para en el algoritmo programado realizar comparaciones sobre su variación en los ejes coordenados, estas variaciones serán resultado del movimiento de la cabeza del paciente; mediante la siguiente tabla se explica la forma de asignación de los movimientos planificada para el control de las cargas de la habitación.

Tabla 5-2: Determinación de acciones según los movimientos establecidos

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	ESTADO DE LA CARGA	MARCADOR
CORTINA	Movimiento 1ª	Desplazamiento lateral de la cabeza hacia la izquierda	ON – SUBIDA PULSO	OJOS
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	ON – BAJADA PULSO	OJOS
LÁMPARA	Movimiento 2ª	Desplazamiento vertical de la cabeza hacia arriba	ON ENCLAVE	OJOS
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	OFF	OJOS
SIRENA – ALERTA	Movimiento 3A.	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	ON ENCLAVE	BOCA
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	OFF TIEMPO	BOCA
VENTILADOR	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	ON ENCLAVE	BOCA
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	OFF	BOCA

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Mediante la discriminación de movimientos planteada en la tabla se puede lograr la asignación de acciones a partir del algoritmo programado para su interacción con el ambiente de la habitación, para ello se plantea inicialmente enlazar el sistema montado en la plataforma LabVIEW y el entorno físico, en la selección de hardware se estableció el uso de un Arduino UNO por lo que se inicia por configurar su conectividad con la aplicación informática.

Se expone en el siguiente bloque de programación la herramienta VISA que ofrece LabVIEW para el tráfico de información con dispositivos anclados a un puerto COM o comunicación

serial, en este bloque se debe establecer el puerto al que se conecta el Arduino y la velocidad de transferencia de datos dual que se manejará.

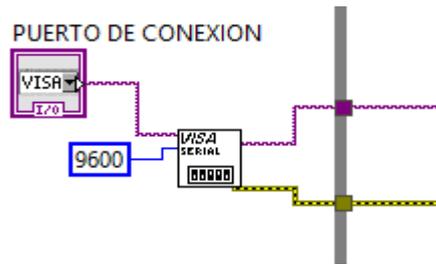


Figura 40-2. Enlace Arduino – Plataforma Informática

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Una vez establecida la conexión del microcontrolador con la plataforma informática, se complementa el algoritmo de visión artificial con los recursos para el control de las cargas del entorno establecido, se realizan comparaciones en intervalos definidos inicialmente para ensayo creando con esto tolerancias a los movimientos, que si son superadas se considerará como instrucción de ejecución para ser interpretada por el Arduino y ejecutar su acción sobre las cargas de la habitación.

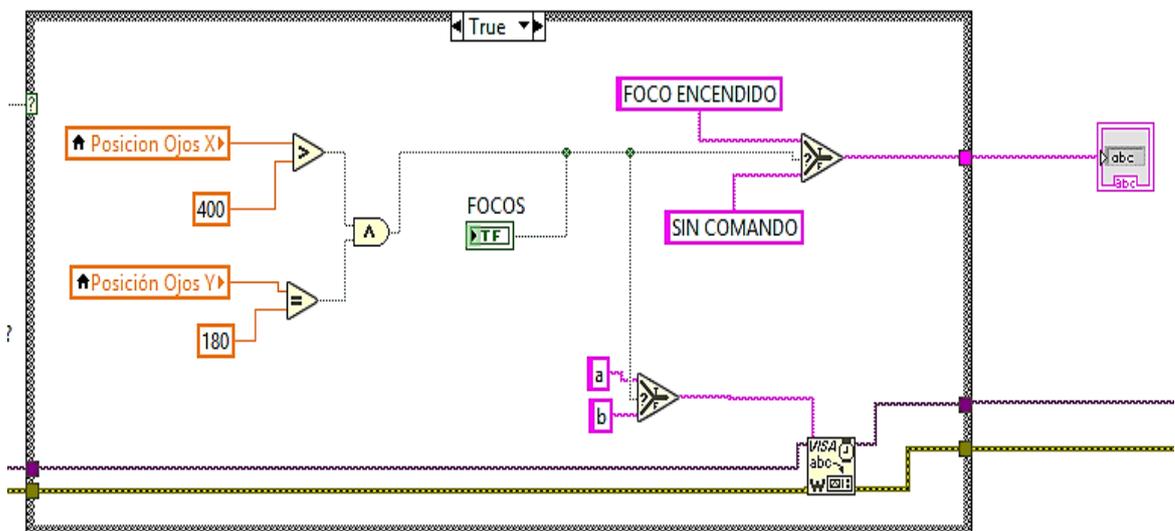


Figura 40-2. Bloques para comparación de posiciones y comunicación Serial

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

En la figura 40-2 se puede apreciar que existen bloques con operaciones de comparación y uno de evaluación del que se arrojará un dato booleano direccionado a un bloque de selección que maneja dos vías de respuesta, por verdadero y falso, los caracteres “a” y “b” según la entrada booleana del selector serán los que se escriban al puerto serial, por ende es la información que

será posteriormente procesada por el microcontrolador. Esta evaluación expuesta se la realiza para el caso de todos los movimientos que se establecieron.

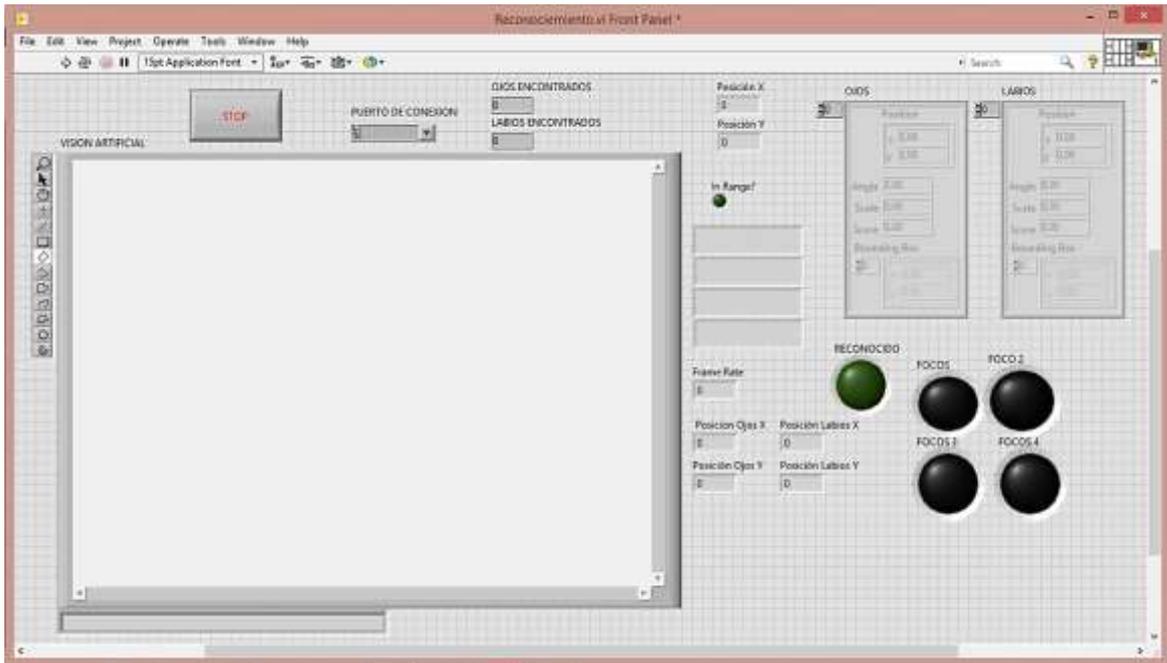


Figura 41-2. Desarrollo de la Interfaz gráfica del sistema

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Para generar un ambiente de alto nivel amigable con el usuario y poder dar seguimiento a la evaluación del algoritmo de visión artificial se diseña en el panel frontal de LabVIEW una interfaz gráfica.

2.6 Programación del microcontrolador

Establecida la comunicación entre la aplicación informática y el microcontrolador se garantiza el flujo de información para la interacción del sistema con el ambiente de la habitación.

Se debe programar el microcontrolador para la interpretación de la información que recibe, para esto se lo programa desde el IDE de Arduino.

Para el caso de estudio el programa del microcontrolador parte de la declaración de las variables, constantes e inicialización de contadores auxiliares para el control del proceso. Se le asigna un nombre a cada pin para la identificación de las variables. En el bloque de programación de la función setup () se declara el modo de funcionamiento de los pines digitales declarados y se programa el puerto serial para manejar una velocidad de comunicación de 9600 baudios.

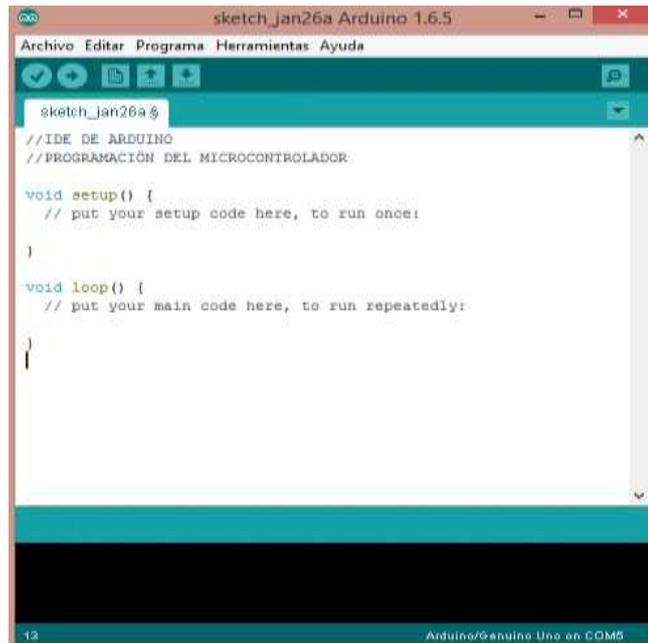


Figura 43-2. IDE de Arduino

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

```

int cortina_sub=4;
int cortina_baj=7;
int lampara=13;
int sirena=8;
int radio=9

int var1;
int cont1=0;
int cont2=0;
int cont3=0;
int cont4=0;
int cont5=0;
int cont6=0;
int cont7=0;
int cont8=0;

void setup() {
  Serial.begin(9600);
  pinMode(cortina_sub,OUTPUT);
  pinMode(cortina_baj,OUTPUT);
  pinMode(lampara,OUTPUT);
  pinMode(sirena,OUTPUT);
  pinMode(radio,OUTPUT);

```

Figura 44-2. Declaración de recursos iniciales empleados en el microcontrolador

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

De acuerdo a la Tabla 5-2 se observa en la columna “Estado de la carga” palabras claves definidas como “Pulso” y “Enclave” dependiendo de la carga, esto hace relación a que en ciertas cargas el microcontrolador debe entregar en sus salidas una señal en forma de pulso, por lo que en la programación para lograr esto se creó las sentencia para el ciclo: activación de la salida,

conteo de un tiempo y la desactivación de la salida (Figura 44-2a); en el caso de lograr un enlace se refiere a guardar un estado por acción de un suceso, hasta la conmutación de éste por otro suceso. (Figura 44-2b)

```
if (var1 == 'a'){  
    digitalWrite(cortina_sub,LOW);  
    digitalWrite(cortina_baj,HIGH);  
    delay(1000);  
    digitalWrite(cortina_sub,HIGH);  
}
```

(a)

```
if (var1 == 'e'){  
    cont3++;  
    if (cont3>0){  
        digitalWrite(foco4,LOW);  
    }  
    if (cont3=0){  
        digitalWrite(foco4,HIGH);  
    }  
}
```

(b)

Figura 42-2. Declaración de recursos iniciales empleados en el microcontrolador

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

CAPITULO III

3. VALIDACIÓN DEL PROTOTIPO

Desarrollada la aplicación informática basada en el algoritmo de visión artificial y, configurado el enlace de comunicación del software con la parte hardware que maneja el entorno físico, se procede a la implementación del sistema y la validación de su funcionalidad por medio de pruebas y análisis de resultados.

3.1 Implementación del sistema prototipo

En este punto se procede a la implementación del sistema, partiendo de la generación de un diagrama eléctrico como referencia para la instalación física en el ambiente determinado, en este caso una habitación de 14 metros cuadrados de superficie.

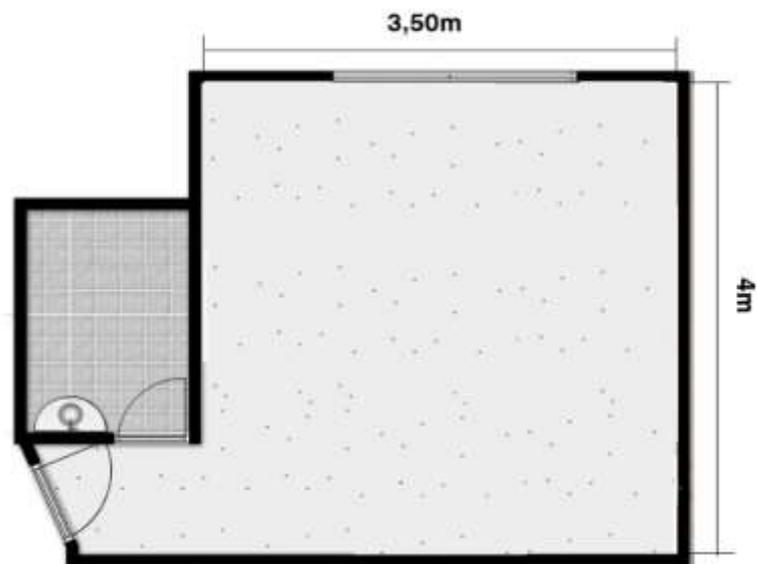


Figura 43-3. Dimensiones del ambiente para experimentación.

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

La habitación posee un acceso y un ventanal, que además de ser solo parámetros descriptivos del ambiente serán de gran influencia en la etapa de pruebas del sistema.

3.1.1 Conexión del hardware

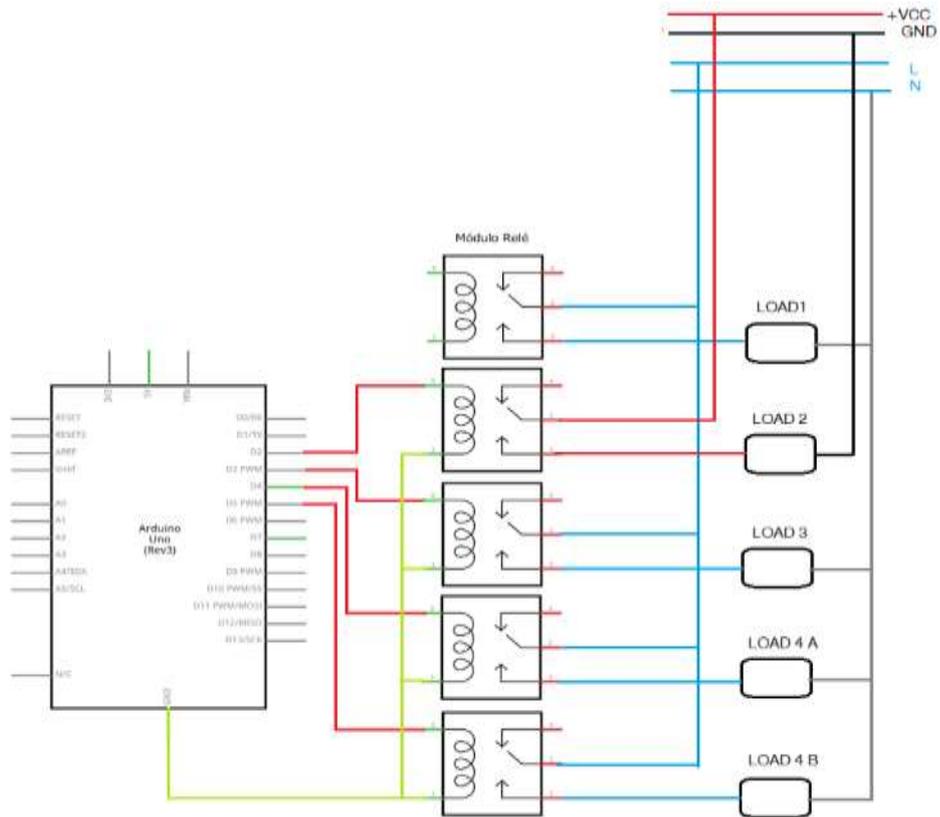


Figura 44-3. Diagrama de conexiones del sistema.

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

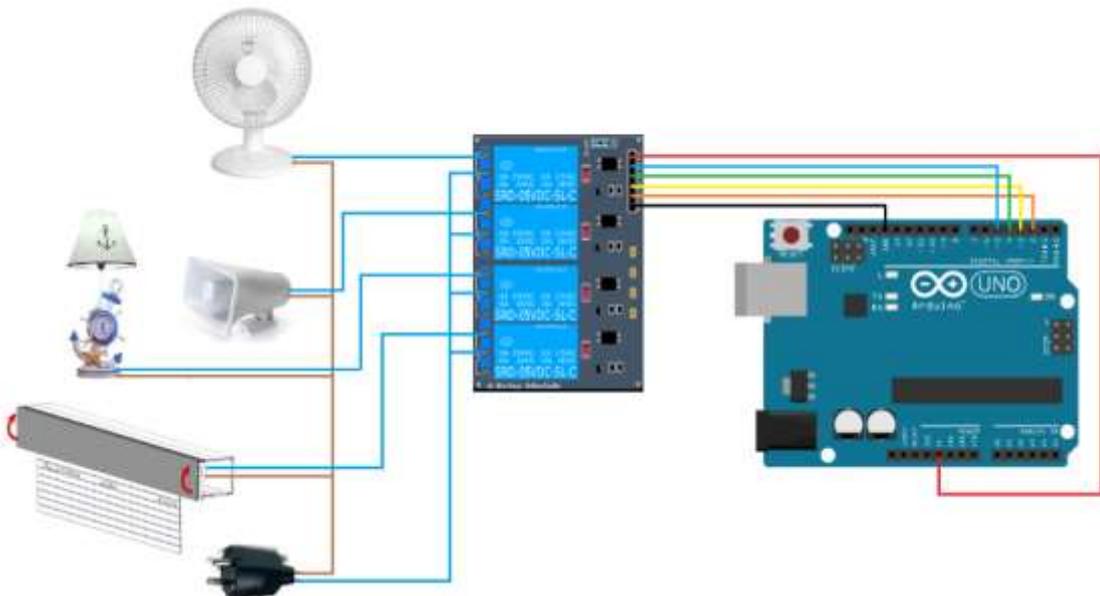


Figura 45-3. Diagrama de esquemático con cargas físicas

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

El levantamiento de un diagrama de conexiones resulta importante por lo que se plantea en la figura 47-3 las conexiones desde la pasarela, el Arduino UNO, hacia la etapa de potencia.

3.1.2 *Instalación del Hardware*



Figura 46-3. Tablero eléctrico – circuito de control.

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Para la instalación del hardware se empleó un tablero eléctrico, como medida de protección a los elementos, pues al ser un entorno familiar al que está expuesto, por desconocimiento o curiosidad podrían afectar a las conexiones del circuito.

En la figura 47-3 se puede observar el circuito de control en el que están realizadas las conexiones desde la plataforma del microcontrolador hacia el módulo de potencia o de relés de acuerdo a la codificación y orden que se planteó en la programación en la asignación de las variables. Además se observa el cable USB azul que corresponde al medio de enlace de la comunicación serial con el computador que alberga la aplicación informática.

Al ser definidas cuatro cargas para el control, se diría que se requieren únicamente cuatro relés pero se puede observar un módulo de 8 relés, el porqué, pues se ha sobredimensionado el sistema en el caso de desear añadir más cargas, lógicamente se deberían añadir marcadores y condiciones en el algoritmo de visión, pero en relación al hardware se puede expandir su utilidad.

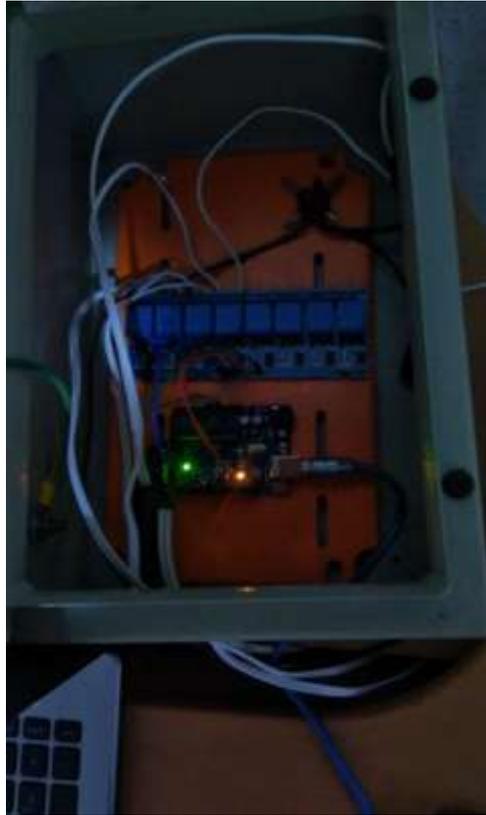


Figura 47-3. Tablero eléctrico – circuito con cargas.

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Los módulos de relé en conjunto con la programación permitirán emitir mandos a cargas que requieran de una señal de pulso o de enclave.



Figura 48-3. Cargas del ambiente controlado

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

La figura 50-3 evidencia las cargas instaladas en el ambiente de la habitación, y se puede hacer mención que cualquier tipo de carga eléctrica puede ser integrada para su control a través del prototipo de sistema desarrollado.

3.2 Interfaz Gráfica.

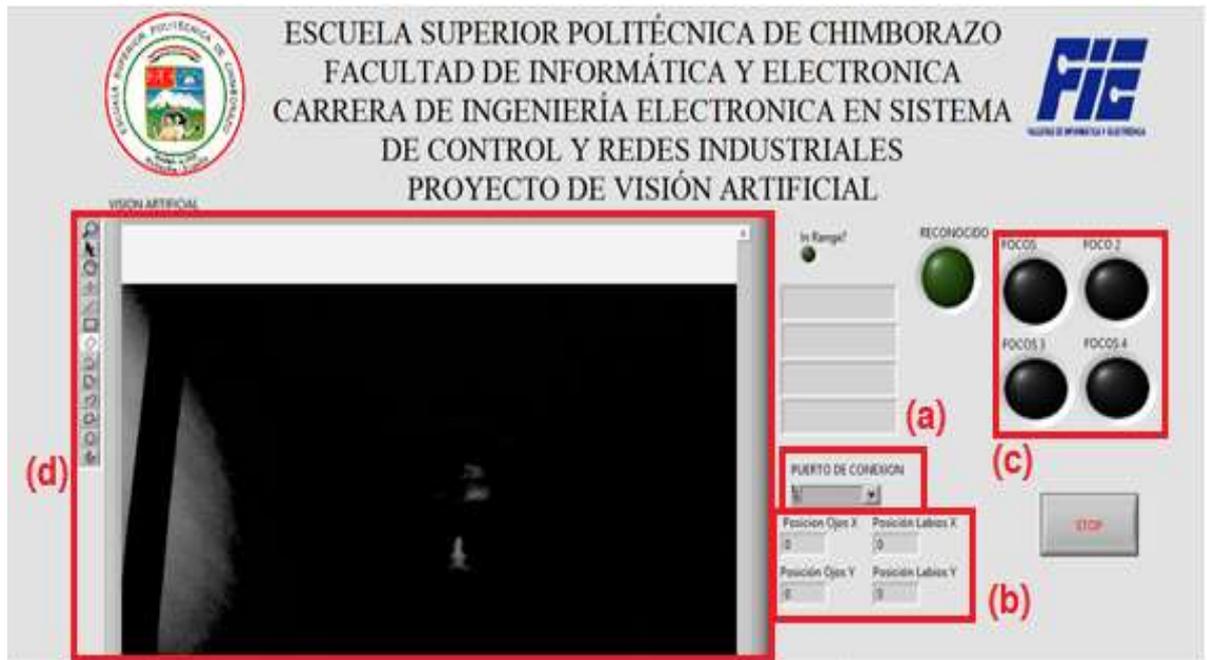


Figura 49-3. Panel frontal Interfaz gráfica

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

En el panel frontal de LabVIEW se estableció una interfaz gráfica capaz de interactuar con los entes manipuladores del sistema.

La figura 51-3 (a) muestra un selector del puerto COM al que se conectará el microcontrolador.

En la figura 51-3 (b) se muestran cuadros de texto que visualizan las coordenadas de los marcadores, definidos en el sistema como marcador de “ojos” y “labios”. En base a estos valores se podrá calibrar el sistema por lo que su presencia resulta necesaria.

En la figura 51-3 (c) muestra un conjunto de indicadores (LEDs), conmutarán a su estado de detección del movimiento pre establecido.

En la figura 51-3 (d) se observa el monitor de la imagen captada por la cámara web resulta ser útil al momento de monitorear los marcadores.

3.3 Pruebas del prototipo



Figura 50-3. Pruebas del sistema

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

La flexibilidad que presenta el sistema radica que en condiciones iniciales al paciente se lo puede colocar en cualquier posición deseada, por ejemplo acostado o en posición de descanso semi-sentado como se muestra en la figura 52-3.

Para el ensayo se consideró inicialmente usar la cámara de la computadora pero debido a sus bajas características la imagen no resultaba de buena calidad por la distancia a la que se encontraba ubicada, se optó por usar una cámara cableada de iguales características, pero ubicada en un trípode para acercarla consiguiendo de esta manera la mejora de la calidad de imágenes adquiridas.

3.3.1 Calibración del sistema

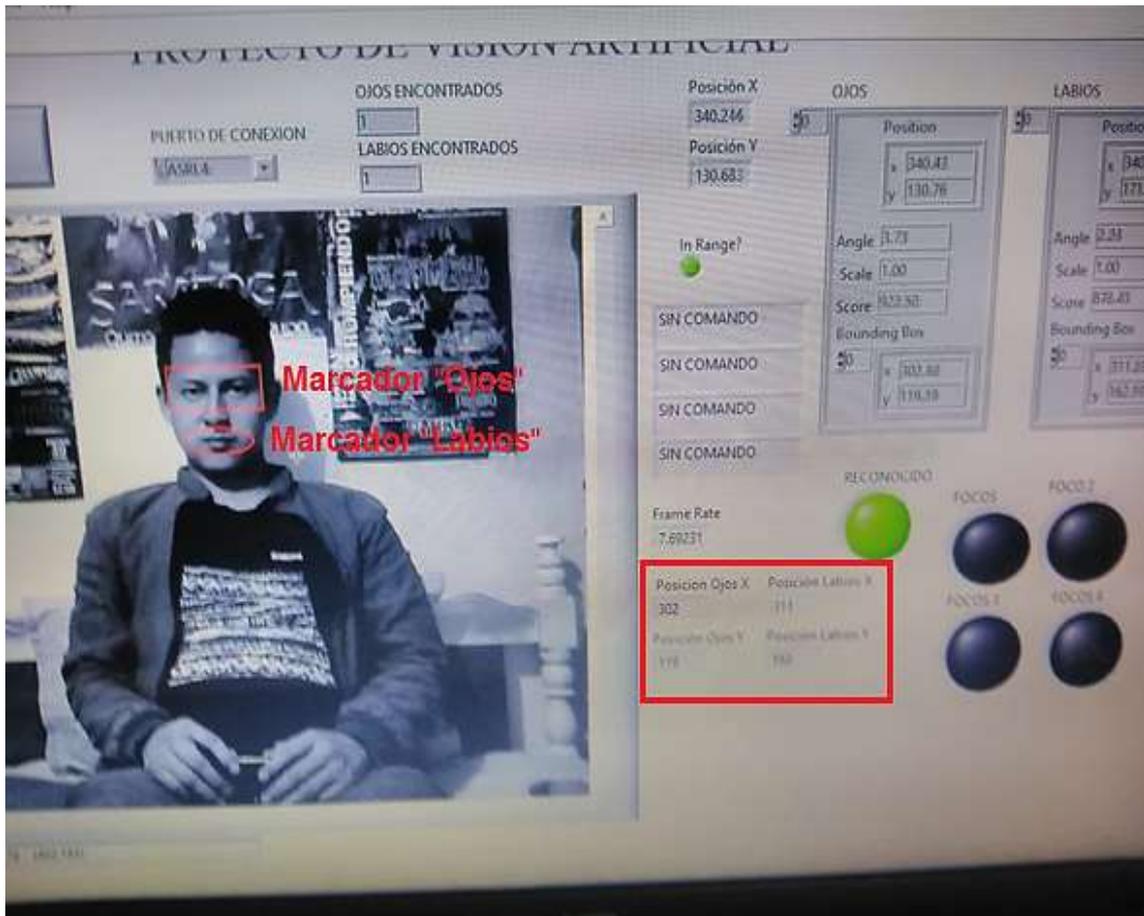


Figura 51-3. Definición posición inicial

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

La calibración del sistema consiste en establecer la coordenadas iniciales (x, y) de los marcadores, estos valores se tomarán al momento de fijar en una posición específica en la que descansará el paciente.

En la primera experimentación se realizó el ensayo y se consiguió como valores iniciales del marcador de ojos un $x=304$ y un $y=119$; para el marcador de labios un $x=311$ y un $y=162$. Estos valores son fundamentales pues se obtiene las referencias para evaluar movimiento del paciente a partir de estos valores. Por ejemplo, con el valor marcador de ojos se pretende controlar la activación de subida y bajada de la cortina, es así que en la figura 53-3 se puede observar el código con la calibración de estos valores.

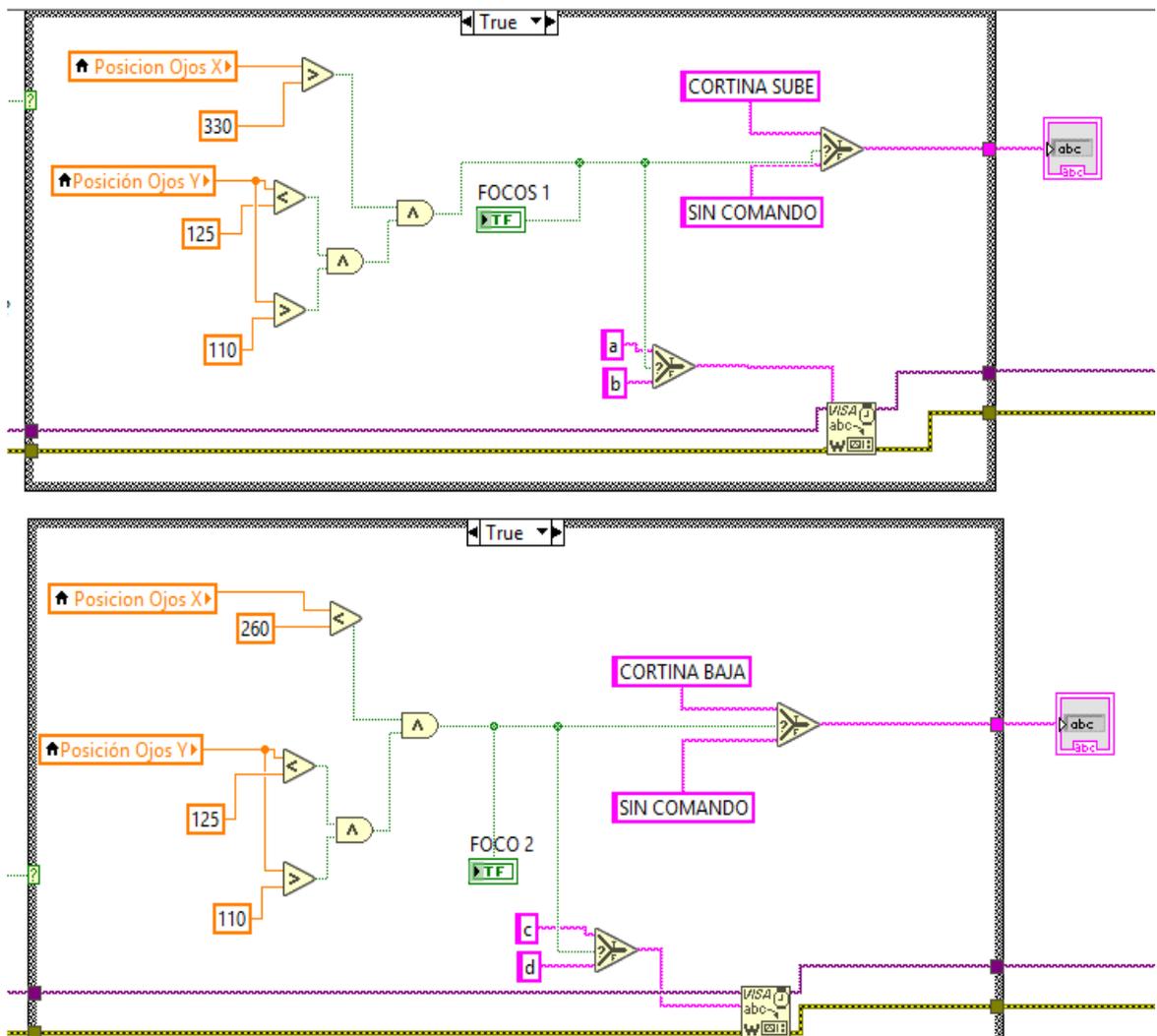


Figura 52-3. Definición de rangos

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Interpretado el código manifiesta que al ser 304 la referencia en el eje x en el marcador de ojos, en el primer caso se le da una holgura de variación hasta valores menores de 330 para no interpretar movimiento, para valores mayores de 330 se considerará un desplazamiento lateral hacia la izquierda; en el segundo caso se establece la holgura de variación hasta valores mayores de 260 para no interpretar un movimiento, para valores menores de 260 se considerará un desplazamiento lateral hacia la derecha. En el eje y se crea una ventana de tolerancia en el rango de 110 a 125 para no considerar movimiento en ambos casos.

Los valores tomados para la programación son el resultado de la evaluación del sistema en el ambiente físico, las figuras 55-3 y 56-3 evidencian las pruebas.



Figura 53-3. Desplazamiento lateral de la cabeza hacia la izquierda
Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

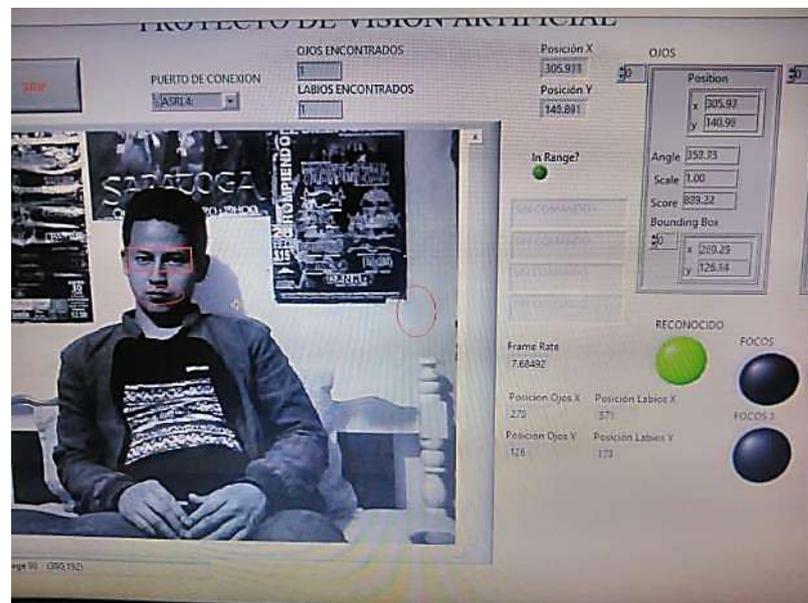


Figura 54-3. Desplazamiento lateral de la cabeza hacia la derecha
Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

3.3.2 Prueba de distribución de elementos del ambiente de la habitación

Las figuras 57-3 y 58-3 muestran las dos distribuciones efectuadas con los elementos del ambiente de la habitación para la realización de las pruebas.

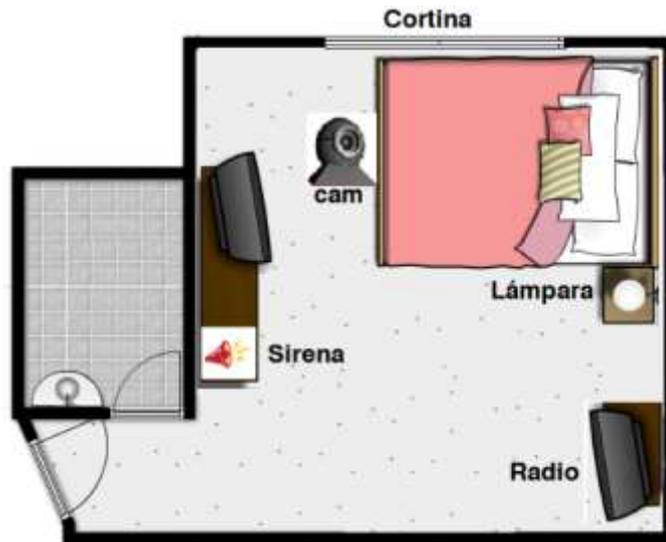


Figura 55-3. Distribución ambiente (1)

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

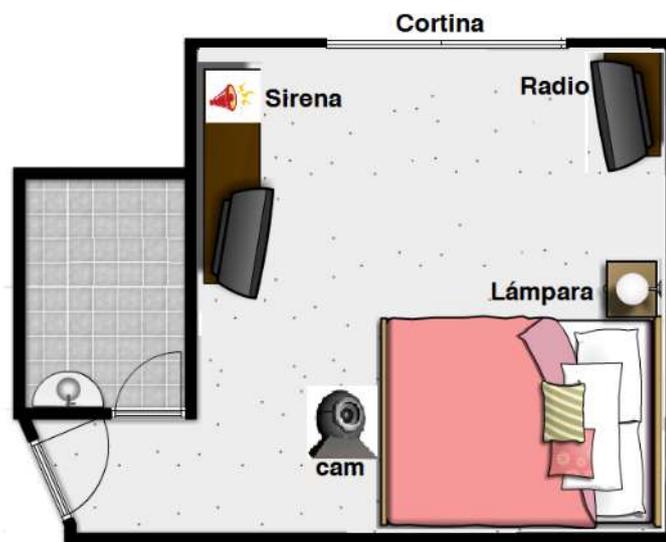


Figura 56-3. Distribución ambiente (2)

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Para este prototipo se planteó inicialmente como requerimiento un ambiente ideal o controlado, se experimentó con la distribución de los elementos de la habitación de tal forma que la luz natural que incide por la ventana en el proceso de apertura y cierre de la cortina no afecte a la evaluación de las imágenes en la aplicación desarrollada.

Al efectuar la calibración inicialmente se trabajó con la distribución 1, que con la ventana abierta resultaba todo correcto en la detección de los marcadores, pero al bajar la cortina las condiciones de luz variaban al punto de desaparecer los marcadores o dar detecciones falsas,

Figura 59-3, en vista de esta situación se optó por asumir la distribución 2 de la habitación para realizar otras pruebas, consiguiendo de esta manera un resultado óptimo a las condiciones requeridas del sistema.

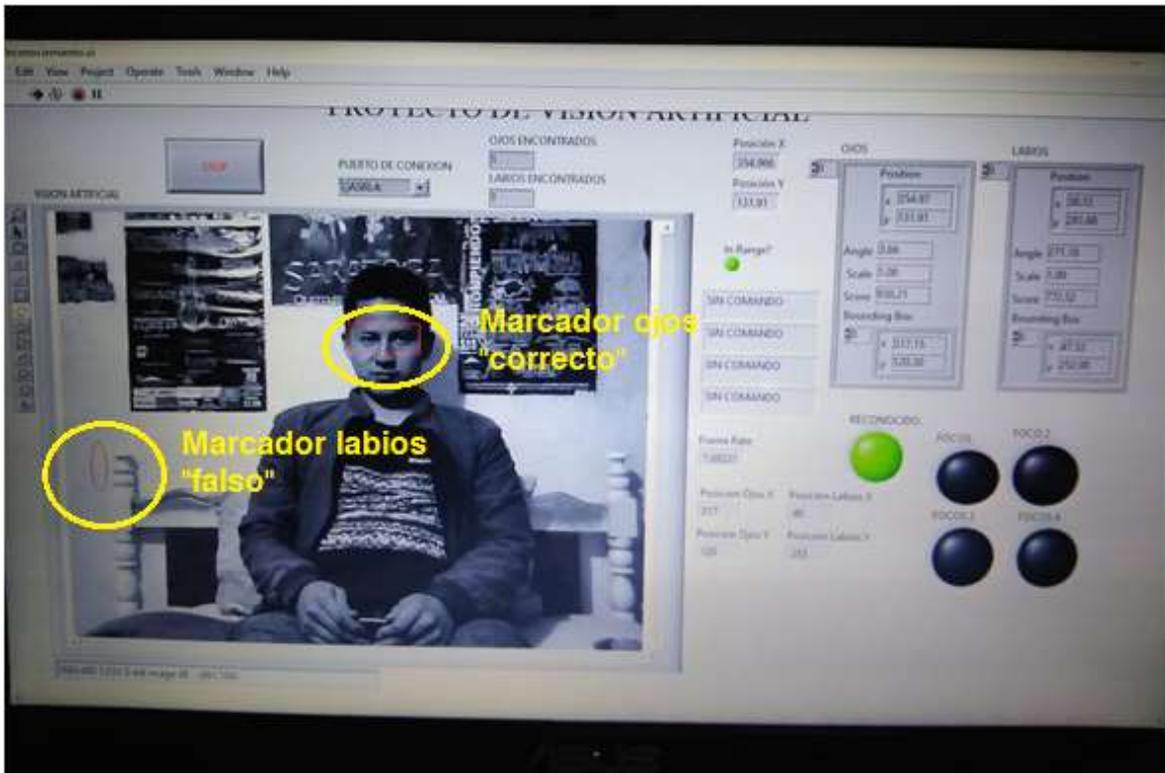


Figura 57-3. Prueba detección de marcadores – error por condiciones no ideales de luz

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

3.4 Validación de resultados obtenidos en las pruebas

Para la validación del sistema se realizaron pruebas continuas de funcionalidad, determinando su eficiencia en relación al número de ejecuciones versus el número de ejecuciones de éxito. Se tomó como muestra 10 intentos por cada instrucción y además se valoró dos posiciones del paciente.

Tabla 6-3: Evaluación del sistema “posición semi - sentado

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%

LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	9	90%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Tabla 7-3: Evaluación del sistema posición acostado

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	8	80%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	8	80%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	7	70%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Realizando el análisis de los resultados se puede establecer que en promedio la eficiencia del sistema en el caso de evaluación (paciente 1) para la posición de semi-sentado del paciente es

del 87,5%, y en la posición de acostado la eficiencia del sistema está en un 76,25%. Estos valores de eficiencia podrían ser mejorados con la mejora de la cámara para tener mayor definición en las imágenes, pues el algoritmo programado únicamente tendría que ser calibrado nuevamente.

La prueba realizada para la validación se ejecutó para diez personas experimentando en los dos entornos planteados (posición 1 semi – sentado y posición 2 acostado).

Tabla 8-3: Evaluación del sistema semi - sentado vs posición acostado pruebas experimentales

PACIENTE	MOVIMIENTO	EFICIENCIA POSICIÓN 1	EFICIENCIA POSICIÓN 2
PACIENTE 1	MOVIMIENTO 1	100	80
PACIENTE 1	MOVIMIENTO 2	90	80
PACIENTE 1	MOVIMIENTO 3	80	75
PACIENTE 1	MOVIMIENTO 4	80	70
PACIENTE 2	MOVIMIENTO 1	95	85
PACIENTE 2	MOVIMIENTO 2	85	75
PACIENTE 2	MOVIMIENTO 3	80	75
PACIENTE 2	MOVIMIENTO 4	75	70
PACIENTE 3	MOVIMIENTO 1	100	85
PACIENTE 3	MOVIMIENTO 2	90	80
PACIENTE 3	MOVIMIENTO 3	85	85
PACIENTE 3	MOVIMIENTO 4	85	75
PACIENTE 4	MOVIMIENTO 1	95	80
PACIENTE 4	MOVIMIENTO 2	100	80
PACIENTE 4	MOVIMIENTO 3	80	80
PACIENTE 4	MOVIMIENTO 4	80	70
PACIENTE 5	MOVIMIENTO 1	100	90
PACIENTE 5	MOVIMIENTO 2	90	80

PACIENTE 5	MOVIMIENTO 3	80	70
PACIENTE 5	MOVIMIENTO 4	80	70
PACIENTE 6	MOVIMIENTO 1	100	80
PACIENTE 6	MOVIMIENTO 2	100	80
PACIENTE 6	MOVIMIENTO 3	80	80
PACIENTE 6	MOVIMIENTO 4	80	70
PACIENTE 7	MOVIMIENTO 1	100	80
PACIENTE 7	MOVIMIENTO 2	85	80
PACIENTE 7	MOVIMIENTO 3	80	75
PACIENTE 7	MOVIMIENTO 4	75	70
PACIENTE 8	MOVIMIENTO 1	100	90
PACIENTE 8	MOVIMIENTO 2	100	85
PACIENTE 8	MOVIMIENTO 3	80	80
PACIENTE 8	MOVIMIENTO 4	80	80
PACIENTE 9	MOVIMIENTO 1	100	80
PACIENTE 9	MOVIMIENTO 2	90	80
PACIENTE 9	MOVIMIENTO 3	80	75
PACIENTE 9	MOVIMIENTO 4	80	70
PACIENTE 10	MOVIMIENTO 1	100	80
PACIENTE 10	MOVIMIENTO 2	90	80
PACIENTE 10	MOVIMIENTO 3	80	75
PACIENTE 10	MOVIMIENTO 4	80	70

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

La tabla 8-3 muestra los resultados porcentuales de la evaluación del sistema con las pruebas efectuadas. A partir de estos resultados se determina el movimiento con mayor eficiencia en la evaluación de algoritmo de visión artificial.

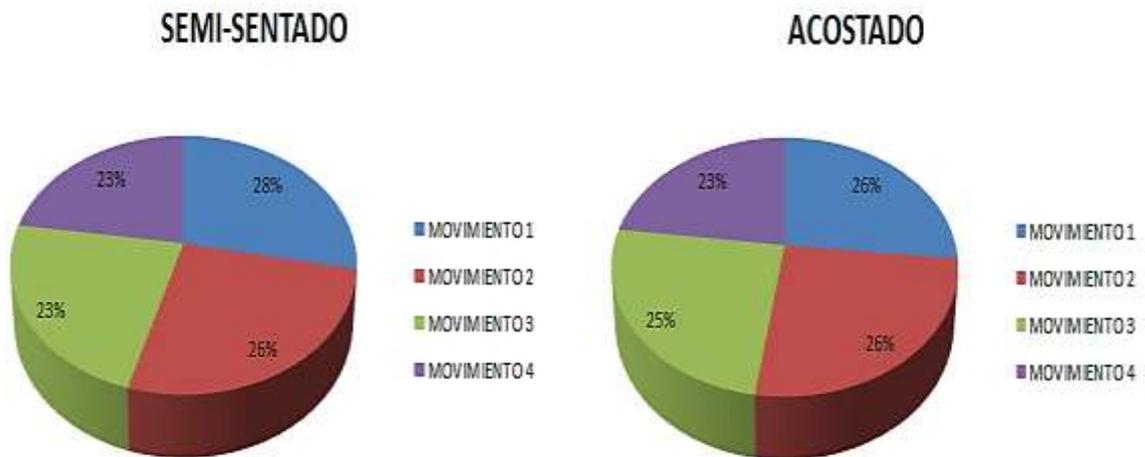


Gráfico 1-3. Porcentajes de eficiencia de movimientos en los dos casos de posición.

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

En base a la figura 60-3 y la tabla 9-3 se determina que el movimiento mejor detectado por el algoritmo de visión artificial es el movimiento 1 (ojos desplazamiento arriba / abajo), y el menos preciso el movimiento 3 (Desplazamiento lateral derecho hacia arriba / abajo referenciado con la barbilla).

Tabla 9-3: Posición y movimientos variación de eficiencia

MOVIMIENTO	SEMI-SENTADO	ACOSTADO
MOVIMIENTO 1	99	83
MOVIMIENTO 2	92	80
MOVIMIENTO 3	80,5	77
MOVIMIENTO 4	79,5	71,5

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Adicionalmente la cámara se la podría posicionar de manera fija para evitar la calibración en cada ejecución del programa.

Se aplicó una prueba estadística para tratar la información recolectada, en este caso al tratarse de pocos datos (10 pruebas) se aplica la herramienta de T-Student a la siguiente relación de variables expuestas en las tablas 10-3 y 11-3.

Se consideró evaluar como ejemplo el caso del movimiento mejor detectado y el de menor porcentaje de eficiencia. Se establece como H_0 , hipótesis nula que el sistema no detecta con eficiencia el movimiento y una H_1 , hipótesis válida que el sistema detecta eficientemente el movimiento.

Tabla 10-3: Aciertos vs Desaciertos pruebas movimiento 1B

PACIENTE	ACIERTOS	DESACIERTOS	% DE ACIERTOS
1	10	0	1
2	9	1	0,9
3	10	0	1
4	9	1	0,9
5	10	0	1
6	10	0	1
7	10	0	1
8	10	0	1
9	10	0	1
10	10	0	1

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Empleando el análisis de datos de Excel se plantea la prueba T – Student para los datos de la tabla 10-3 obteniendo los estadísticos de la tabla 11-3.

Tabla 11-3: Resultado prueba T-Student movimiento 1B

Estadístico	Variable 1	Variable 2
Media	0,98	0,02
Varianza	0,00177778	0,00177778
Observaciones	10	10
Coefficiente de correlación de Pearson	-1	
Diferencia hipotética de las medias	0	
Grados de libertad	9	
Estadístico t	36	
P(T<=t) una cola	2,4368E-11	
Valor crítico de t (una cola)	1,83311293	
P(T<=t) dos colas	4,8736E-11	
Valor crítico de t (dos colas)	2,26215716	

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Tabla 12-3: Aciertos vs Desaciertos pruebas movimiento 3A

PACIENTE	ACIERTOS	DESACIERTOS	% DE ACIERTOS
1	7	3	0,7
2	7	3	0,7
3	9	1	0,9
4	8	2	0,8
5	7	3	0,7
6	8	2	0,8
7	7	3	0,7
8	8	2	0,8
9	8	2	0,8
10	8	2	0,8

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Empleando el análisis de datos de Excel se plantea la prueba T – Student para los datos de la tabla 7-3 obteniendo los estadísticos de la tabla 13-3.

Tabla 13-3: Resultado prueba T-Student movimiento 3A

Estadístico	Variable 1	Variable 2
Media	0,77	0,077
Varianza	0,00455556	4,5556E-05
Observaciones	10	10
Coefficiente de correlación de Pearson	1	
Diferencia hipotética de las medias	0	
Grados de libertad	9	
Estadístico t	36,076139	
P(T<=t) una cola	2,3912E-11	
Valor crítico de t (una cola)	1,83311293	
P(T<=t) dos colas	4,7824E-11	
Valor crítico de t (dos colas)	2,26215716	

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

En ambos casos en el análisis de datos según los estadísticos de la T-Student se considera que la hipótesis nula no se cumple por lo tanto el sistema si detecta movimientos con eficiencia.

3.5 Consumo eléctrico

Al tratarse de un sistema conformado por elementos eléctricos y electrónicos resulta importante conocer el consumo que éste genera, para lo cual como se expresa en las tablas 5-3y 6-3 se consideró un promedio de tiempo en el que la carga está en uso y su consumo en vatios por hora (wh) por día.

La tabla 9-3 representa un consumo base del sistema, es decir, cuando quienes consumen son únicamente los elementos del bloque de control sin interacción de los actuadores, interpretando así la columna de estado, OFF – elemento sin consumo y ON- elemento consumiendo.

Tabla 14-3: Consumo base

CARGA	ESTADO	CONSUMO POR HORA (Wh)	CONSUMO CICLO (Wh)	PROMEDIO DE TIEMPO DE ACTIVACIÓN (h)
CORTINA – SUBE	OFF	80	0	0,008333333
CORTINA – BAJA	OFF	80	0	0,008333333
SIRENA	OFF	0,6	0	0,016666667
LÁMPARA	OFF	5	0	1
VENTILADOR	OFF	10	0	1
ORDENADOR	ON	110	1320	12h
MICROCONTROLADOR	ON	0,23	2,76	12h
CONSUMO BASE			1322,76	

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

El sistema sin interacción de los actuadores de acuerdo a lo elaborado en la tabla 9-3 genera un consumo de 1322,76 Wh, valor del que se puede observar es el ordenador el consumidor mayoritario.

La tabla 10-3 muestra el consumo del sistema a máxima carga, lo que significa que se consideró que todos, tanto elementos del sistema de control como actuadores se activan de forma paralela y realizan un consumo.

Tabla 15-3: Consumo a máxima carga

CARGA	ESTADO	CONSUMO POR HORA (wh)	CONSUMO CICLO (wh)	PROMEDIO DE TIEMPO DE ACTIVACIÓN (h)
CORTINA – SUBE	ON	80	0,666666667	0,008333333
CORTINA – BAJA	ON	80		0,008333333
SIRENA	ON	0,6	0,01	0,016666667
LÁMPARA	ON	5	5	1
VENTILADOR	ON	10	10	1
ORDENADOR	ON	110	1320	12
MICROCONTROLADOR	ON	0,23	2,76	12
CONSUMO MÁXIMO			1338,436667	

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

Se obtiene que el sistema a máxima carga realiza un consumo de 1338,43 Wh al día que representa un consumo moderado y al evaluar el consumo base y el consumo máximo se puede determinar que existe una diferencia de 15,67 Wh que no es considerable al ser que el uso de las cargas seleccionadas para el prototipo son de uso intermitente

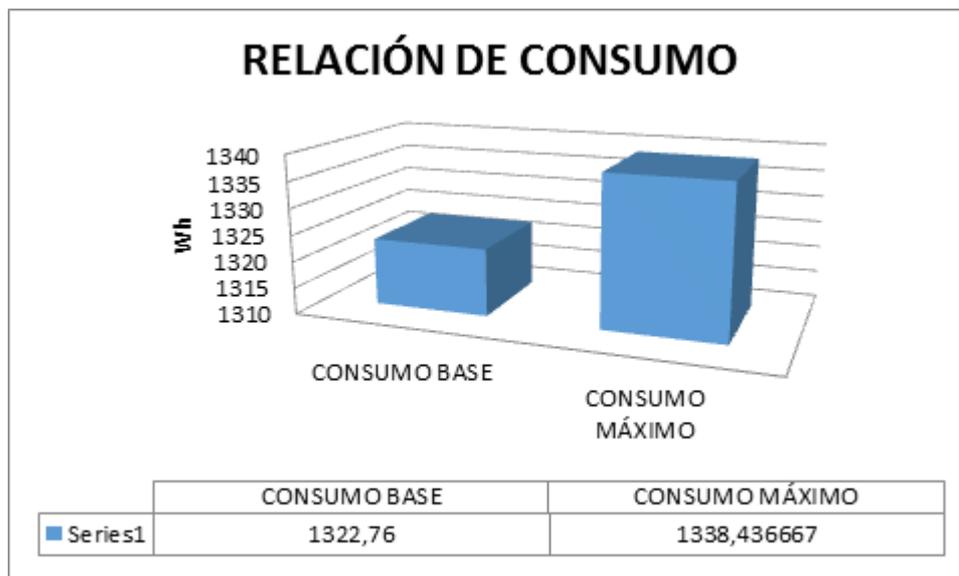


Gráfico 2-3. Consumo base vs Consumo máximo del sistema

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

CAPÍTULO IV

4. EVALUACIÓN ECONÓMICA

4.1 Costos

Se realiza un análisis general de los costos efectuados en el diseño e implementación del sistema de control del ambiente de una habitación para una persona cuadripléjica en un 80%.

4.1.1 *Costos directos*

Son los que se relacionan directamente para la realización o fabricación del modelo prototipo.

4.1.1.1 *Costos eléctricos*

Tabla 16-4: Costos eléctricos

Cantidad	Descripción	V. unitario(USD)	V. Total (USD)
1	Computador (mínimo i5)	800	800
1	Web cam (1,3 Mp)	30	30
1	Arduino UNO	15	15
1	Módulo de 8 Relés	18	18
15m	Cable gemelo 14 AWG	0,80	12
1	Cortina Automatizada 1,35mx0,95m	180	180
1	Sirena	8	8
		Total	1063

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

4.1.1.2 *Costos mecánicos*

Tabla 17-4: Costos mecánicos

Cantidad	Descripción	V. unitario(USD)	V. Total (USD)
1	Trípode de cámara	20	20
		Total	20

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

4.1.1.3 Costos de mano de obra

Tabla 18-4: Costos de mano de obra

Cantidad	Descripción	V. unitario(USD)	V. Total (USD)
1	Instalación	50	50
Total			50

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

4.1.2 Costos directos totales

Tabla 19-4: Costos directos totales

Descripción	V. Total
Costos eléctricos	1063
Costos mecánicos	20
Costo mano de obra	50
Costo directo total	1133

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

4.1.3 Costos indirectos

Tabla 20-4: Costo indirectos totales

N ^a	Descripción	V. unitario(USD)	V. Total (USD)
1	Ingenieriles (supervisión y diseño)	200	200
2	Imprevistos	50	50
Total			250

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

4.1.4 Costo total

Tabla 21-4: Costos directos totales

N ^a	Descripción	V. unitario(USD)	V. Total (USD)
1	Costos directos totales	1133,00	1133,00
2	Costos indirectos totales	250,00	250,00
Costo Total			1383,00

Realizado por: Yugcha Alomaliza, Christian Alexis, 2020

CONCLUSIONES

Se logró la implementación de un prototipo para el control del ambiente de una habitación para un paciente con lesión medular C4 calificada como persona cuadripléjicas con un 80% de discapacidad empleando recursos tecnológicos basados en la integración de hardware y software en una aplicación basada en visión artificial.

Basados en la concepción de una lesión medular C4, en la se bloquea el movimiento de los hombros hacia abajo, es decir, se considera la existencia de movimiento en el cuello y cabeza, se establecieron cinco movimientos tomados como referencias a partir de los ojos y boca para mediante su análisis en el algoritmo de visión artificial ser evaluados y vinculados con los actuadores seleccionados en el ambiente de una habitación.

Mediante la valoración del entorno para la instalación del piloto, se realizó la selección de recursos hardware, empleando como centralizador del sistema un ordenador que alberga la aplicación informática del algoritmo de visión artificial y mediante comunicación serial gestiona señales de control hacia un microcontrolador con sus respectivas interfaces de potencia para interactuar con el medio físico.

El algoritmo de visión artificial se lo realizó en la plataforma del software LabVIEW de la National Instrument empleando el asistente de visión, que a partir de la adquisición de la información por medio de una cámara procesa en tiempo real el flujo de imágenes identificando patrones previamente establecidos, brindando información de dichos patrones tales como la ubicación cartesiana, parámetros que resultaron útiles para complementar el algoritmo de visión artificial con la evaluación para la emisión de señales de control.

Al evaluar el prototipo mediante pruebas realizadas a diez personas en dos situaciones específicamente en estado de semi-sentado y acostado se logró determinar una eficiencia del 87,75% y 77,87% respectivamente, y validado estadísticamente mediante una prueba de T-Student se determinó que el sistema discrimina los movimientos establecidos con buen eficiencia.

RECOMENDACIONES

El sistema puede replicarse no solo para el control de actuadores dentro de una habitación si no en forma general donde se lo requiera tomando en cuenta las condiciones de luminosidad.

Al momento de evaluar el sistema se deben considerar las condiciones de luminosidad, evitando el contacto directo de la luz natural, en esta experiencia se debió considerar incluso la distribución de los elementos de la habitación.

Resulta importante antes de la calibración del sistema fijar una posición del paciente y de la cámara, pues el algoritmo de control maneja señales cartesianas que al momento de manipular la posición de los actores dentro del sistema este sufrirá modificaciones, provocando fallos en el mismo.

Para esta experimentación se empleó una cámara básica, la que viene por defecto con el ordenador portátil, para tener una detección más fina de los patrones se podría usar una cámara de mejores condiciones.

Se puede sugerir que a futuro en planes de seguir investigando se planteé la evaluación del sistema para otras posibles posiciones del paciente, con el fin de detectar deficiencias o tal vez aportes que permitan mejorar y ampliar el uso de la aplicación desarrolla.

GLOSARIO

LabVIEW: Es una herramienta de programación gráfica. Originalmente este programa estaba orientado para aplicaciones de control de equipos electrónicos usados en el desarrollo de sistemas de instrumentación, lo que se conoce como instrumentación virtual. (13) (Vizcaíno y Sebastián, 2011, p. 22)

Arduino: es una plataforma embebida que contiene como elemento principal un microcontrolador reprogramable, el mismo que permite la adquisición de señales, procesamiento mediante sentencias programadas y la emisión de señales de control. Este sistema embebido además se acopla a una amplia gama de sensores de la misma marca u otra que proporcionen señales del tipo digital y/o analógico para ser procesadas y según su evaluación por medio de interfaces de potencia poder realizar el control de actuadores sin importar la carga que manejen. (Arduino, 2020)

Paraplejía: El término paraplejía se utiliza para referirse a una debilidad parcial o completa de ambas piernas.

Cuadriplejía: El término cuadriplejía indica una debilidad parcial o completa de los cuatro miembros, eliminando así la necesidad de los términos paraparesia y cuadriparesia. Una misma alteración medular puede producir paraplejía o cuadriplejía, dependiendo de la localización de la lesión.(10) (Fenichel, 2006, p. 255)

La Visión Artificial o Visión por Computador: Es un área multidisciplinar que pretende, en cierta medida, reproducir artificialmente el sentido de la vista mediante el procesamiento e interpretación de imágenes, captadas con distintos tipos de sensores (fundamentalmente, cámaras), y utilizando para ello las prestaciones de los ordenadores. El desarrollo de sistemas de Visión Artificial requiere una combinación de etapas de más bajo nivel (p. ej. operaciones de eliminación de ruido o de aumento del contraste) para mejorar la calidad de las imágenes capturadas junto con otras etapas de más alto nivel (de reconocimiento de patrones y de interpretación de imágenes) para reconocer los elementos presentes en una escena. (11) (Sánchez Calle, 2003, p. 4)

BIBLIOGRAFÍA

ARDUINO. *Software Arduino.* [en línea]. Disponible en: <https://www.arduino.cc>.

ARDUINO. *¿Qué es Arduino?* [en línea]. Disponible en: <https://arduino.cl/que-es-arduino/>.

BRADSKI, G., KAEHLER, A., *Learning OpenCV: Computer Vision with the OpenCV Library* [en línea] O'Reilly Media Inc, 2008 [citado 14 de enero de 2020]. Disponible en: https://books.google.com.ec/books?hl=es&lr=&id=seAgiOfu2EIC&oi=fnd&pg=PR3&dq=open+cv&ots=hUK96icCK9&sig=jOL4DzNRIL2C7FgfwKNx9__QCmc&redir_esc=y#v=onepage&q=open%20cv&f=false

CORSINO, E., *Descripción/análisis cinemático del movimiento humano.* 1998. 38 p.

DOMINGUEZ DM., *Reconocimiento de Objetos Bidimensionales en Imágenes mediante la Transformada de Distancia utilizando Matlab* [en línea]. 2003. [citado 20 de enero del 2020]. Disponible en: http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/mendieta_d_d/portada.html

FENICHEL, GM. *Capítulo 12 - Paraplejía y cuadriplejía.* [en línea]. Madrid: Elsevier España, 2006. [citado 20 de enero del 2020]. p. 255-71. Disponible en: <http://www.sciencedirect.com/science/article/pii/B9788481749038500129>

FOLEA, S., *Practical applications and solutions using LabVIEW software* [en línea]. Rijeka: Intech, 2011 [citado 14 de enero de 2020]. Disponible en: <http://www.intechopen.com/books/practical-applications-and-solutions-using-labview-software>

GONZÁLES M., MARTÍNEZ, F., PERNÍA, V., ALBA, F., CASTEJÓN, M., *Técnicas y algoritmos básicos de visión artificial.* España: Universidad de la Rioja, 2006.

HENAO C.P., PÉREZ, J.E., *Lesiones medulares y discapacidad.* D - Universidad de La Sabana; 2010. 17 p.

KRAUSE JS., BRODERICK, L., *Outcomes after spinal cord injury: comparisons as a function of gender and race and ethnicity.* Arch Phys Med Rehabil, 2004.

LAJARA, J., & PELEGRI, J., *LabVIEW: Entorno gráfico de programación* - José Rafael Lajara Vizcaíno & José Pelegrí Sebastia - Google Libros [en línea]. Segunda. Ediciones Técnicas; 2011 [citado 29 de diciembre de 2019]. Disponible en: https://books.google.com.ec/books?hl=es&lr=&id=ZFQua3-eeQEC&oi=fnd&pg=PA21&dq=COMUNICACION+SERIAL+EN+LABVIEW&ots=qI1xU8SFhu&sig=gfJ14V93tVGnHRzA5eglxa5zsZI&redir_esc=y#v=onepage&q&f=false

- MAZAIIRA, J., et al.** *Epidemiología de la lesión medular y otros aspectos*. Rehabilitación, 1998, vol. 32, no 6, p. 365-372.
- NORMAN, E.** *Revista elementos. Nro. 1*. [en línea]. 2011 [citado 20 de enero del 2020]. Disponible en: <https://ebookcentral.proquest.com>
- PAJARES A., & ESCALERA, A.,** *Conceptos y Métodos en Visión por Computador*. CEA, 2016.
- PLAZA, M., APERADOR, W., AVILES, O.,** *Technology in Locomotion and Domotic Control for Quadriplegic*. 29th Southern Biomedical Engineering Conference. 2013. p. 99-100.
- PRIETO, C., FEBRES, J.E., CERROLAZA, M., MIQUELARENA, R.** *Sistema de Visión Artificial para el Control de Movimiento de un Asistente Robótico Médico*. 2010.
- RIOMOROS I.,** *Segmentación automática de texturas en imágenes agrícolas*. [en línea] [Tesis Doctoral]. [Madrid, España], 2016. Disponible en: <https://eprints.ucm.es/38358/1/T37480.pdf>
- SÁNCHEZ, A.,** *I Jornada sobre Visión Artificial y Biometría*. Móstoles, 2003.
- SÁNCHEZ, A.,** *Aplicaciones de la visión artificial y la biometría informática*. Madrid, España: DYKINSON, 2005.
- TORRENTO, O.,** *ARDUINO Curso básico de formación*. [en línea]. Alfaomega, 2016. [Citado 20 de enero del 2020]. Disponible en: http://cienciasvirtuales.com/wp-content/uploads/2016/11/Arduino_Curso_Practico_de_Formacion.pdf
- VARGAS, X.,** *Diseño de un prototipo de control de acceso del personal mediante reconocimiento facial en 3d para empresas públicas o privadas* (Trabajo de titulación) (Ingeniería) Escuela Superior Politécnica de Chimborazo, Riobamba – Ecuador. 2016
- WATSON, C., KAYALIOGLU. G.,** Chapter 1 - The Organization of the Spinal Cord. [en línea] San Diego: Academic Press; 2009 [citado 20 de enero del 2020]. Disponible en: <http://www.sciencedirect.com/science/article/pii/B9780123742476500055>
- WYNDAELE, M., WYNDAELE, J-J.,** *Incidence, prevalence and epidemiology of spinal cord injury: what learns a worldwide literature survey?.*, Spinal Cord, 2006.

ANEXOS

ANEXO A. DATASHETT ARDUINO UNO



INTRODUCTION

Arduino is used for building different types of electronic circuits easily using of both a physical programmable circuit board usually microcontroller and piece of code running on computer with USB connection between the computer and Arduino.

Programming language used in Arduino is just a simplified version of C++ that can easily replace thousands of wires with words:

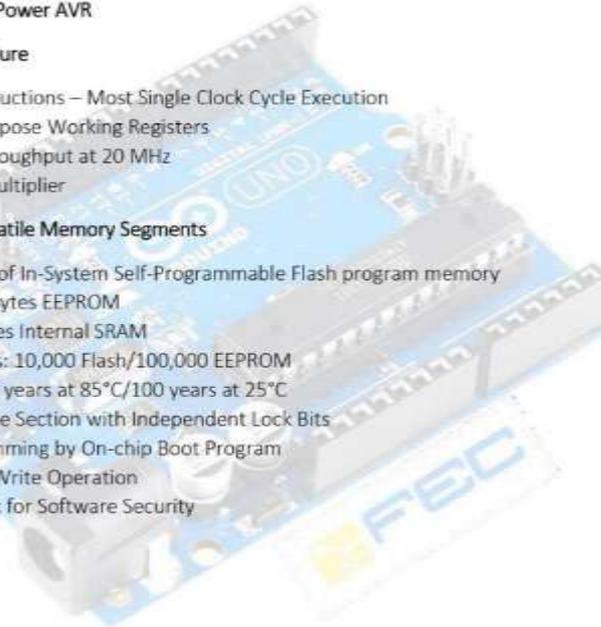


ARDUINO UNO-R3 PHYSICAL COMPONENTS

ATMEGA328P-PU microcontroller

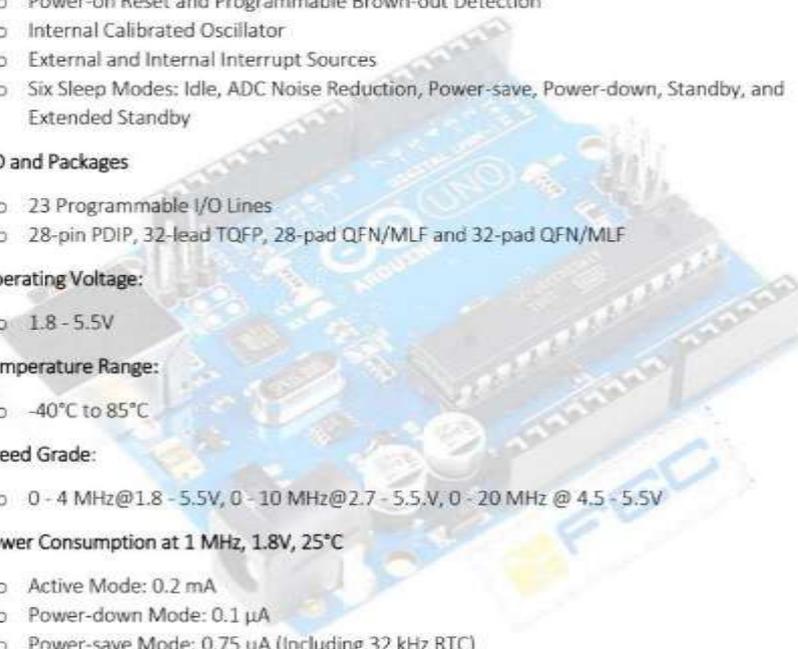
The most important element in Arduino Uno R3 is ATMEGA328P-PU is an 8-bit Microcontroller with flash memory reach to 32k bytes. It's features as follow:

- High Performance, Low Power AVR
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Up to 20 MIPS Throughput at 20 MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 4/8/16/32K Bytes of In-System Self-Programmable Flash program memory
 - 256/512/512/1K Bytes EEPROM
 - 512/1K/1K/2K Bytes Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Six PWM Channels
 - 8-channel 10-bit ADC in TQFP and QFN/MLF package
 - Temperature Measurement
 - 6-channel 10-bit ADC in PDIP Package
 - Temperature Measurement
 - Programmable Serial USART





- Master/Slave SPI Serial Interface
 - Byte-oriented 2-wire Serial Interface (Philips I2 C compatible)
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
-
- **Special Microcontroller Features**
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
-
- **I/O and Packages**
 - 23 Programmable I/O Lines
 - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
-
- **Operating Voltage:**
 - 1.8 - 5.5V
-
- **Temperature Range:**
 - -40°C to 85°C
-
- **Speed Grade:**
 - 0 - 4 MHz@1.8 - 5.5V, 0 - 10 MHz@2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V
-
- **Power Consumption at 1 MHz, 1.8V, 25°C**
 - Active Mode: 0.2 mA
 - Power-down Mode: 0.1 μ A
 - Power-save Mode: 0.75 μ A (Including 32 kHz RTC)





- Pin configuration

(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	7	22	GND
GND	8	21	AREF
(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)
(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)
(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)

ATMEGA16u2- mu microcontroller

Is a 8-bit microcontroller used as USB driver in Arduino uno R3 it's features as follow:

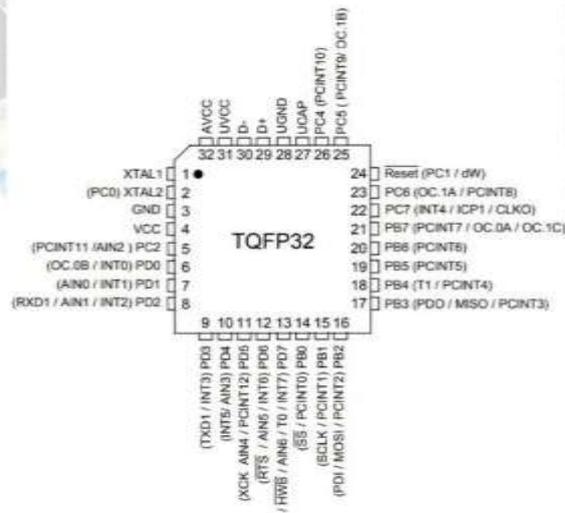
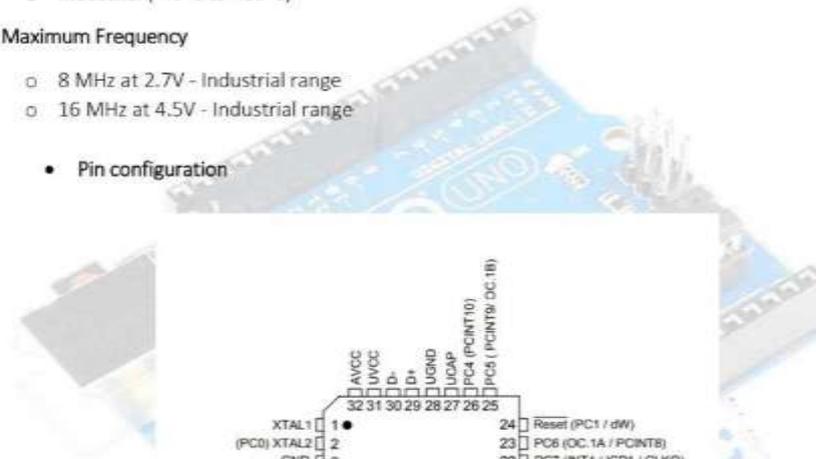
- High Performance, Low Power AVR
- Advanced RISC Architecture
 - o 125 Powerful Instructions – Most Single Clock Cycle Execution
 - o 32 x 8 General Purpose Working Registers
 - o Fully Static Operation
 - o Up to 16 MIPS Throughput at 16 MHz
- Non-volatile Program and Data Memories
 - o 8K/16K/32K Bytes of In-System Self-Programmable Flash
 - o 512/512/1024 EEPROM
 - o 512/512/1024 Internal SRAM
 - o Write/Erase Cycles: 10,000 Flash/ 100,000 EEPROM
 - o Data retention: 20 years at 85°C/ 100 years at 25°C



- Optional Boot Code Section with Independent Lock Bits
- In-System Programming by on-chip Boot Program hardware-activated after reset
- Programming Lock for Software Security
- **USB 2.0 Full-speed Device Module with Interrupt on Transfer Completion**
 - Complies fully with Universal Serial Bus Specification REV 2.0
 - 48 MHz PLL for Full-speed Bus Operation: data transfer rates at 12 Mbit/s
 - Fully independent 176 bytes USB DPRAM for endpoint memory allocation
 - Endpoint 0 for Control Transfers: from 8 up to 64-bytes
 - 4 Programmable Endpoints:
 - IN or Out Directions
 - Bulk, Interrupt and Isochronous Transfers
 - Programmable maximum packet size from 8 to 64 bytes
 - Programmable single or double buffer
 - Suspend/Resume Interrupts
 - Microcontroller reset on USB Bus Reset without detach
 - USB Bus Disconnection on Microcontroller Request
- **Peripheral Features**
 - One 8-bit Timer/Counters with Separate Prescaler and Compare Mode (two 8-bit PWM channels)
 - One 16-bit Timer/Counter with Separate Prescaler, Compare and Capture Mode(three 8-bit PWM channels)
 - USART with SPI master only mode and hardware flow control (RTS/CTS)
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- **On Chip Debug Interface (debug WIRE)**
- **Special Microcontroller Features**
 - Power-On Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Five Sleep Modes: Idle, Power-save, Power-down, Standby, and Extended Standby
- **I/O and Packages**
 - 22 Programmable I/O Lines
 - QFN32 (5x5mm) / TQFP32 packages



- Operating Voltages
 - 2.7 - 5.5V
- Operating temperature
 - Industrial (-40°C to +85°C)
- Maximum Frequency
 - 8 MHz at 2.7V - Industrial range
 - 16 MHz at 4.5V - Industrial range
- Pin configuration





OTHER ARDUINO UNO R3 PARTS

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 k Ohms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the `analogWrite()` function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function. Additionally, some pins have specialized functionality:

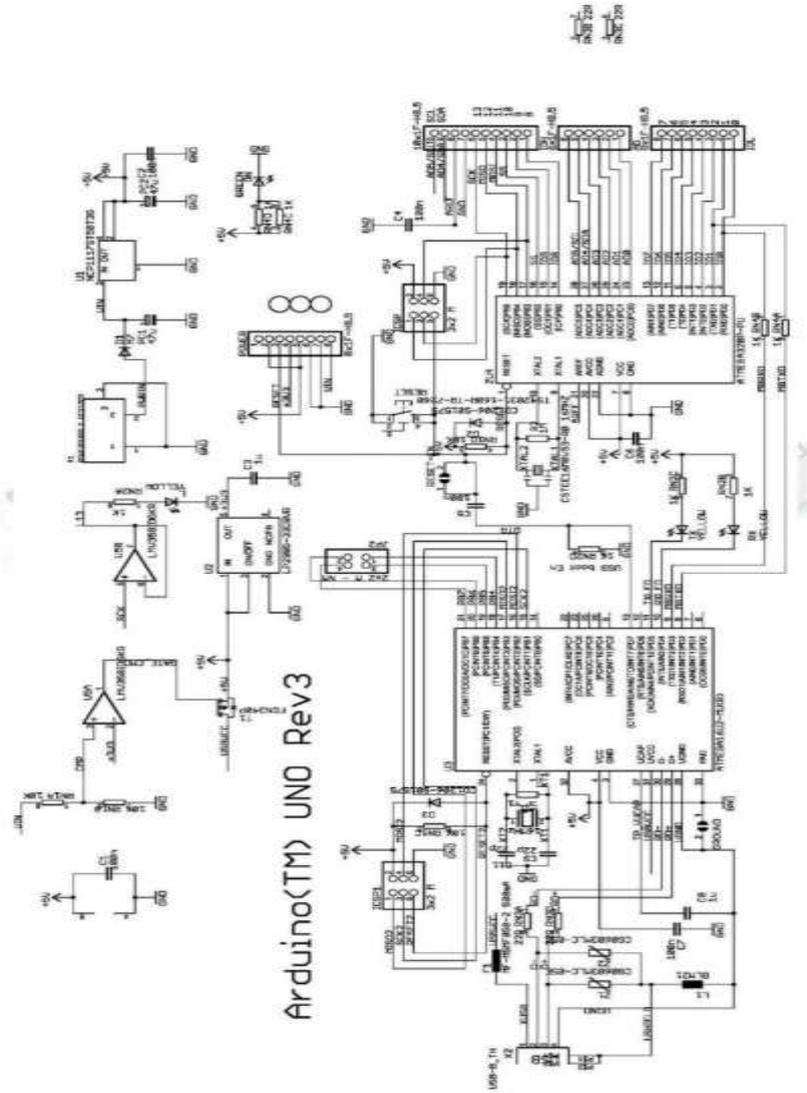
- TWI: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

There are a couple of other pins on the board:

- AREF: Reference voltage for the analog inputs. Used with `analogReference()`.
- Reset: Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.



ARDUINO UNO R3 SCHEMATIC DIAGRAM



ANEXO B. DATASHEET MEGA



Product Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Index

Technical Specifications	Page 2
How to use Arduino Programming Enviroment, Basic Tutorials	Page 6
Terms & Conditions	Page 7
Enviromental Policies half sqm of green via Impatto Zero®	Page 7



radiospares

RADIONICS



Technical Specification

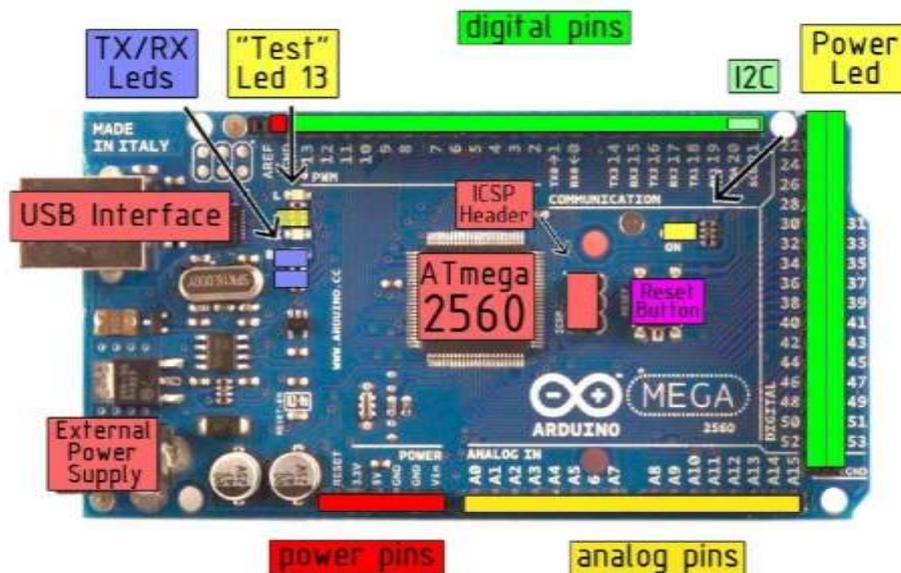


EAGLE files: [arduino-mega2560-reference-design.zip](#) Schematic: [arduino-mega2560-schematic.pdf](#)

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

the board



radiospares

RADIONICS



Power

The Arduino Mega2560 can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2).** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 0 to 13.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Duemilanove and Diecimila.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- **I²C: 20 (SDA) and 21 (SCL).** Support I²C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I²C pins on the Duemilanove.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and [analogReference\(\)](#) function.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.



radiospares

RADIONICS



Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Mega's digital pins.

The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation on the Wiring website](#) for details. To use the SPI communication, please see the ATmega2560 datasheet.

Programming

The Arduino Mega2560 can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Arduino Mega comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.



radiospares

RADIONICS



Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Mega has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics and Shield Compatibility

The maximum length and width of the Mega PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega is designed to be compatible with most shields designed for the Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega and Duemilanove / Diecimila. **Please note that I²C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).**



radiospares

RADIONICS



How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](http://arduino.cc/en/Guide/HomePage) for the latest instructions. <http://arduino.cc/en/Guide/HomePage>

Linux Install

Windows Install

Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

**File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink**

Once you have your sketch you'll see something very close to the screenshot on the right.

In **Tools>Board** select MEGA

Now you have to go to **Tools>SerialPort** and select the right serial port, the one arduino is attached to.

```
Sketch - Arduino 0017
File Edit Sketch Tools Help

Sketch

int ledPin = 13; // LED connected to digital pin 13

// The setup() method runs once, when the sketch starts

void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

void loop()
{
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(1000);                // wait for a second
  digitalWrite(ledPin, LOW);  // set the LED off
  delay(1000);                // wait for a second
}
```

Done compiling
Press Compile button (to check for errors)

Upload

TX RX Flashing

Blinking Led!

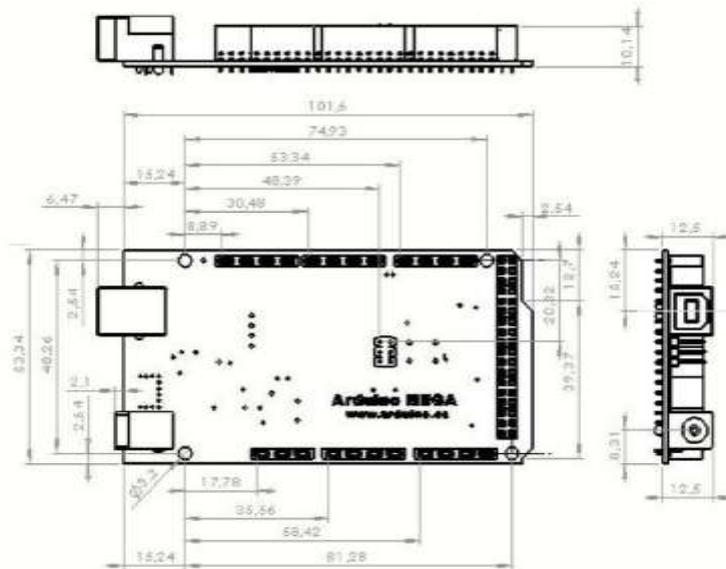
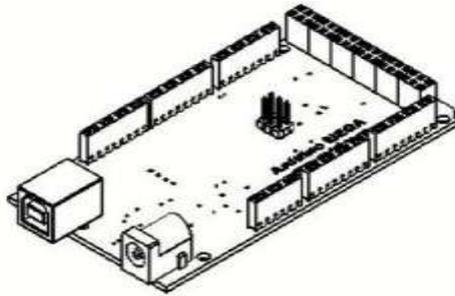


radiospares

RADIONICS



Dimensioned Drawing



radiospares RADIONICS



Terms & Conditions



1. Warranties

1.1 The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mistreatment by the Customer, including improper installation or testing, or for any products that have been altered or modified in any way by a Customer. Moreover, The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the producer elects to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

1.4 Customer agrees that prior to using any systems that include the producer products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, applications or design advice, quality characterization, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.

1.5 The Arduino® products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino® products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Arduino® products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino® products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.

2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs to the extent directly caused by: (i) an actual breach by a Customer of the representation and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

4. Changes to specifications

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.



Environmental Policies



The producer of Arduino® has joined the Impatto Zero® policy of LifeGate.it. For each Arduino board produced is created / looked after half squared Km of Costa Rica's forest's.



radiospares **RADIONICS**



ANEXO C. HOJAS DE VALORES DE PRUEBAS

PACIENTE 1

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	9	90%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	8	80%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	8	80%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%

	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	7	70%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%

DATOS PACIENTE ACOSTADO

PACIENTE 2

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	90%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	80%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	9	90%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	70%

	Movimiento 4B	Desplazamiento lateral izquierdo abajo referenciado con la barbilla	8	80%
--	---------------	---	---	-----

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	9	90%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	8	80%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	8	80%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	7	70%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	80%
	Movimiento 4B	Desplazamiento lateral izquierdo abajo referenciado con la barbilla	6	60%

DATOS PACIENTE ACOSTADO

PACIENTE 3

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%

LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	9	90%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	9	90%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	9	90%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	8	80%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	9	90%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	7	70%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	9	90%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la	7	70%

		barbilla		
	Movimiento 4B	Desplazamiento lateral izquierdo abajo referenciado con la barbilla	8	80%

DATOS PACIENTE ACOSTADO

PACIENTE 4

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	9	90%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	10	100%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	10	100%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo abajo referenciado con la barbilla	8	80%

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
-------	------------	----------------------------	-----------------------	------------

CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	8	80%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	8	80%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	8	80%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	8	80%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%

DATOS PACIENTE ACOSTADO

PACIENTE 5

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	9	90%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%

	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	9	90%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	9	90%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	7	70%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%

DATOS PACIENTE ACOSTADO

PACIENTE 6

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	10	100%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	10	100%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	8	80%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	8	80%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	7	70%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%

	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%

DATOS PACIENTE ACOSTADO

PACIENTE 7

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	8	80%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	8	80%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	8	80%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	7	70%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%

DATOS PACIENTE ACOSTADO

PACIENTE 8

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	10	100%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	10	100%

SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	9	90%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	9	90%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia abajo	8	80%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%

DATOS PACIENTE ACOSTADO

PACIENTE 9

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	9	90%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%

DATOS PACIENTE SEMI - SENTADO

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	8	80%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	8	80%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%

	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	7	70%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	7	70%
	Movimiento 4B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	7	70%

DATOS PACIENTE ACOSTADO

PACIENTE 10

CARGA	MOVIMIENTO	DESCRIPCIÓN DEL MOVIMIENTO	EJECUCIONES ACERTADAS	EFICIENCIA
CORTINA	Movimiento 1A	Desplazamiento lateral de la cabeza hacia la izquierda	10	100%
	Movimiento 1B	Desplazamiento lateral de la cabeza hacia la derecha	10	100%
LÁMPARA	Movimiento 2A	Desplazamiento vertical de la cabeza hacia arriba	9	90%
	Movimiento 2B	Desplazamiento vertical de la cabeza hacia arriba	9	90%
SIRENA – ALERTA	Movimiento 3A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%
	Movimiento 3B	Desplazamiento lateral izquierdo hacia abajo referenciado con la barbilla	8	80%
RADIO	Movimiento 4A	Desplazamiento lateral derecho hacia abajo referenciado con la barbilla	8	80%

	Movimiento 4B	Desplazamiento lateral izquierdo abajo referenciado con la barbilla	8	80%
--	---------------	---	---	-----

DATOS PACIENTE SEMI - SENTADO

ANEXO D. HOJA DE CÁLCULO DE T- STUDENT

Prueba t para medias de dos muestras emparejadas

Estadístico	Variable 1	Variable 2
Media	0,98	0,02
Varianza	0,00177778	0,00177778
Observaciones	10	10
Coefficiente de correlación de Pearson	-1	
Diferencia hipotética de las medias	0	
Grados de libertad	9	
Estadístico t	36	
P(T<=t) una cola	2,4368E-11	
Valor crítico de t (una cola)	1,83311293	
P(T<=t) dos colas	4,8736E-11	
Valor crítico de t (dos colas)	2,26215716	