



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E INDUSTRIAL

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y COMUNICACIONES

Tema:

**“SISTEMA ELECTRÓNICO DE CORRECCIÓN DEL GESTO DEPORTIVO EN
BASE A LA BIOMECÁNICA EN DEPORTISTAS DE MARCHA ATLÉTICA.”**

Trabajo de Titulación Modalidad: Proyecto de Investigación, presentado previo a la
obtención del título de Ingeniero en Electrónica y Comunicaciones.

ÁREA: Comunicaciones

LÍNEA DE INVESTIGACIÓN: Tecnologías de Comunicación

AUTOR: Fredy Bladimir Llambo Yansaguano

TUTOR: Dr. Jaime Rodrigo Guilcapi Mosquera

Ambato – Ecuador

Octubre – 2020

APROBACIÓN DEL TUTOR

En calidad de tutor del Trabajo de Titulación con el tema: SISTEMA ELECTRÓNICO DE CORRECCIÓN DEL GESTO DEPORTIVO EN BASE A LA BIOMECÁNICA EN DEPORTISTAS DE MARCHA ATLÉTICA, desarrollado bajo la modalidad Proyecto de Investigación, por el señor Llambo Yansaguano Fredy Bladimir, estudiante de la Carrera de Ingeniería en Electrónica y Comunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 15 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y el numeral 7.4 del respectivo instructivo.

Ambato, octubre del 2020



Dr. Jaime Rodrigo Guilcapi Mosquera

EL TUTOR

AUTORÍA

El presente Proyecto de Investigación titulado: SISTEMA ELECTRÓNICO DE CORRECCIÓN DEL GESTO DEPORTIVO EN BASE A LA BIOMECÁNICA EN DEPORTISTAS DE MARCHA ATLÉTICA es absolutamente original, auténtico y personal. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, octubre 2020.



Fredy Bladimir Llambo

Yansaguano C.C. 1804804779

AUTOR

APROBACIÓN TRIBUNAL DE GRADO

En calidad de par calificador del Informe Final de Trabajo de Titulación presentado por el señor Fredy Bladimir Llambo Yansaguano estudiante de la carrera de Ingeniería en Electrónica e Industrial, bajo la modalidad Proyecto de Investigación, titulado, SISTEMA ELECTRÓNICO DE CORRECCIÓN DEL GESTO DEPORTIVO EN BASE A LA BIOMECÁNICA EN DEPORTISTAS DE MARCHA ATLÉTICA, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 17 del Reglamento para obtener el Título de Tercer Nivel, de Grado de la Universidad Técnica de Ambato, y al numeral 7.6 del respectivo instructivo. Para cuya constancia suscribimos, conjuntamente con la señora presidenta del Tribunal.

Ambato, octubre 2020



Firmado electrónicamente por:
**ELSA PILAR
URRUTIA**

Ing. Pilar Urrutia, Mg.
PRESIDENTA DEL TRIBUNAL



Firmado electrónicamente por:
**MARCO ANTONIO
JURADO LOZADA**

Ing. Marco Jurado
PROFESOR CALIFICADOR



Firmado electrónicamente por:
**MARIO GEOVANNY
GARCIA CARRILLO**


Ing. Mario García
PROFESOR CALIFICADOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación en favor de la Universidad Técnica de Ambato, con fines de difusión pública. Además, autorizo su reproducción total o parcial dentro de las regulaciones de la institución.

Ambato, octubre 2020.



Fredy Bladimir Llambo

Yansagüano C.C. 1804804779

AUTOR

DEDICATORIA

A mis Padres por ser ejemplo de superación, por el apoyo incondicional para no decaer en momentos de angustia, por su frase de ánimo “Todo trabajo difícil bien realizado siempre trae sus recompensas”, que me ayudo a seguir en adelante, levantándome con más ganas cuando se presenta alguna dificultad, en pocas palabras por ser pilar fundamental en mi vida. A mis hermanos por sus ánimos para seguir en adelante en mi carrera profesional.

A mi hijo Elián que a pesar de la distancia es mi fuente de motivación para ser un ejemplo de superación en su vida, y que recuerde que lo que uno se propone hay que lograrlo.

A mis familiares y amigos, que estuvieron presente en el transcurso de mi carrera profesional, por los deseos de superación y apoyo.

Fredy Bladimir Llambo Yansaguano

AGRADECIMIENTO

Primeramente, agradezco a Dios, por brindarme la sabiduría y el don de no dejarme vencer por cosas difíciles, porque “podrá ser difícil pero jamás imposible”, por las fuerzas, perseverancia y valentía para llegar a conseguir una meta más en mi vida, a toda mi familia que siempre me apoyaron y me supieron aconsejar para no decaer ni desviar mi camino.

A mis docentes por dar lo mejor de ellos para ser un gran profesional, por brindar su ayuda y amistad, en especial a mi tutor Doctor Jaime Guilcapi por su apoyo en la realización de esta investigación y al Ing. Santiago Manzano por brindarme todo su ayuda para que sea posible la realización del proyecto.

A mis compañeros que siempre me alentaron para salir juntos en adelante, a mis grandes amigos Javier T. y Gabriel T. por su apoyo y colaboración. A la mujer que me amó y dedico su vida a mi lado para apoyarme en todo momento difícil S.E.

Fredy Bladimir Llambo Yansaguano

ÍNDICE GENERAL DE CONTENIDOS

APROBACIÓN DEL TUTOR	II
AUTORÍA	III
APROBACIÓN TRIBUNAL DE GRADO	IV
DERECHOS DE AUTOR.....	V
DEDICATORIA	VI
AGRADECIMIENTO.....	VII
ÍNDICE GENERAL DE CONTENIDOS	VIII
ÍNDICE DE TABLAS.....	XI
ÍNDICE DE FIGURAS	XIII
RESUMEN EJECUTIVO	XVI
ABSTRACT	XVII
Glosario de términos y acrónimos	XVIII
CAPÍTULO I.- MARCO TEÓRICO	1
1.1 Tema de investigación.....	1
1.2 Antecedentes investigativos.....	1
1.2.1 Contextualización del problema	3
1.2.2 Fundamentación teórica	5
1.2.2.1 Inteligencia Artificial.....	5
1.2.2.2 Python.....	11
1.2.2.3 OPENCV	14
1.2.2.4 MICROCONTROLADORES.....	18
1.2.2.5 SENSORES.....	22
1.2.2.6 CAMARA DIGITAL	26
1.2.2.7 PROCESAMIENTO DIGITAL DE IMÁGENES	27
1.2.2.8 REDES NEURONALES CONVOLUCIONALES	29

1.2.2.9	BIOMECÁNICA.....	42
1.2.2.10	MARCHA ATLÉTICA	44
1.3	Objetivos.....	49
1.3.1	Objetivo general.....	49
1.3.2	Objetivos específicos.....	49
CAPÍTULO II.- METODOLOGÍA		50
2.1	Materiales	50
2.2	Métodos	50
2.2.1	Modalidad de Investigación.....	50
2.2.2	Recolección de Información.....	51
2.2.3	Procesamiento y análisis de datos.....	51
CAPÍTULO III.- RESULTADOS Y DISCUSIÓN		52
3.1	Análisis y discusión de los resultados	52
3.2	Desarrollo de la propuesta	52
3.2.1	Estudio de Factibilidad.....	52
3.2.1.1	Factibilidad Técnica	52
3.2.1.2	Factibilidad Económica	53
3.2.1.3	Factibilidad Bibliográfica	53
3.2.2	Requerimientos para el desarrollo del sistema	53
3.2.2.1	Selección de elementos para la implementación del sistema	54
3.2.2.2	Cámara web	55
3.2.2.3	Sensor de detección de proximidad.....	57
3.2.2.4	Motor	59
3.2.2.5	Microcontrolador y pantalla LCD	61
3.2.2.6	Procesador.....	65
3.2.2.7	Tarjeta de video.....	69
3.2.2.8	Lenguaje de programación.....	70
3.2.2.9	Software IDE Arduino.....	71

3.2.2.10	Servidor XAMPP	71
3.2.3	Diseño del prototipo	72
3.2.3.1	Implementación de Fuente de alimentación	72
3.2.3.2	Estructura del prototipo	74
3.2.3.3	Sistema de procesamiento de video.....	76
3.2.3.4	Interfaz HMI	97
3.2.3.5	Recolección de imágenes.....	100
3.2.3.6	Creación de base datos	104
3.2.3.7	Pruebas de funcionamiento.....	109
3.2.4	Presupuesto	124
CAPITULO IV.- CONCLUSIONES Y RECOMENDACIONES		125
4.1	Conclusiones	125
4.2	Recomendaciones.....	126
REFERENCIAS BIBLIOGRÁFICAS		127
ANEXOS		136
ANEXO A		137
ANEXO B.....		140
ANEXO C.....		141
Anexo D.....		146
ANEXO E.....		151
ANEXO F		156
ANEXO G		157
ANEXO H		176
ANEXO I.....		181

ÍNDICE DE TABLAS

Tabla 1: Métodos de IA [14], [16].	6
Tabla 2: Ventajas y desventajas de la IA [13], [14], [16].	8
Tabla 3: Módulos OpenCV [22].	15
Tabla 4: Ventajas y desventajas de los microcontroladores [25]–[27].	21
Tabla 5: Ventajas y desventajas de los sensores [30], [32].	24
Tabla 6: Ventajas y desventajas de las cámaras digitales [34], [35].	27
Tabla 7: Matriz de Kernel.	30
Tabla 8: Matriz de entrada.	31
Tabla 9: Resultado de la convolución.	33
Tabla 10: Imagen de entrada.	33
Tabla 11: Capa Convolutiva.	35
Tabla 12: Mapas de características con ReLU.	36
Tabla 13: Max-Pooling.	37
Tabla 14: Funciones de activación [39].	39
Tabla 15: Comparación de cámaras web.	56
Tabla 16: Comparación de las características de sensores de proximidad.	57
Tabla 17: Comparación de motor a pasos.	60
Tabla 18: Comparación de microcontroladores.	62
Tabla 19: Características de la LCD.	63
Tabla 20: Final de carrera SPFT largo.	64
Tabla 21: Comparación de procesadores de gama alta.	66
Tabla 22: Comparación de procesadores Intel Core.	67
Tabla 23: Características de Nvidia GeForce MX130.	69
Tabla 24: Características de los lenguaje de programación.	70
Tabla 25: Especificaciones generales de laptop.	77
Tabla 26: Versiones compatibles de XAMPP.	105
Tabla 27: Prueba de sensor ultrasónico.	114
Tabla 28: Tiempo de desplazamiento del soporte de la cámara.	115
Tabla 29: FPS y tiempo de reproducción.	116
Tabla 30: Análisis de modelos del aprendizaje automático.	118
Tabla 31: Ángulos en fase de apoyo simple.	120
Tabla 32: Fase de apoyo doble.	122

Tabla 33: Presupuesto del sistema.....124

ÍNDICE DE FIGURAS

Figura 1: Áreas Tecnológicas [14], [15].	5
Figura 2: Aplicaciones de la IA [15]	7
Figura 3: Tipos de aprendizaje de IA [17].	9
Figura 4: Esquema general del aprendizaje supervisado [17].	10
Figura 5: Imagen a procesar.	16
Figura 6: Pixelado de Figura 5.	17
Figura 7: Filtrado de Figura 5.	17
Figura 8: Elementos Básicos de un microcontrolador. [25].	19
Figura 9: Comportamiento de la Brújula. [31]	22
Figura 10: Clasificación de sensores [33].	25
Figura 11: Sensores. [33]	25
Figura 12: Arquitectura de las CNN [40].	30
Figura 13: Capa de entrada de la RNA.	37
Figura 14: Red Neural Artificial [40].	38
Figura 15: CNN descripción gráfica [40].	41
Figura 16: Representación gráfica del estudio de la biomecánica.	43
Figura 17: Ejercicio 1. [45]	44
Figura 18: Ejercicio 2. [45]	44
Figura 19: Ejercicio 3. [45]	45
Figura 20: Ejercicio 4. [45]	45
Figura 21: Ejercicio 5. [45]	45
Figura 22: Ejercicio 6. [45]	45
Figura 23: Ejercicio 7. [45]	46
Figura 24: Ejercicio 8. [45]	46
Figura 25: Ejercicio 9. [45]	46
Figura 26: Ejercicio 10. [45]	46
Figura 27: Ejercicio 11. [45]	47
Figura 28: Participación muscular en las fases de impulso y apoyo. [46]	48
Figura 29: Diagrama de bloques del sistema electrónico.	54
Figura 30: Descripción grafica de polea.	65
Figura 31: Esquema de conexión del prototipo.	72
Figura 32: Diseño de fuente de alimentación.	73
Figura 33: Fuente de alimentación.	73

Figura 34: Pata lateral.	74
Figura 35: Pata del medio y soldadura de varilla.	74
Figura 36: Estructura finalizada.	75
Figura 37: Esquema de polea y banda para desplazamiento del soporte de la cámara.	75
Figura 38: Soporte de cámara con polea.	76
Figura 39: Diagrama de implementación del sistema electrónico de biomecánica de marcha atlética.	78
Figura 40: Descarga de entorno de programación.	79
Figura 41: Métodos de instalación de python.	80
Figura 42: Opción de Instalación.	80
Figura 43: Características opcionales de python.	81
Figura 44: Opciones avanzadas de instalación.	81
Figura 45: Código de ejemplo.	82
Figura 46: Descarga de OpenCV.	82
Figura 47: Extracción de archivos OpenCV.	83
Figura 48: Extracción y OpenCV en versiones de python.	83
Figura 49: Finalización de instalación de librería OpenCV.	84
Figura 50: Instalación de OpenCV mediante consola.	84
Figura 51: Verificación de instalación OpenCV.	85
Figura 52: Diagrama de Flujo captura de video OpenCV.	85
Figura 53: Resultado captura de video con OpenCV.	86
Figura 54: Diagrama de flujo de estimación humana.	87
Figura 55: Proceso de estimación humana.	88
Figura 56: Arquitectura multietapa de CNN.	89
Figura 57: Uso de CPU python y OpenCV.	90
Figura 58: FPS de video capturado.	90
Figura 59: Requerimientos para el aprendizaje automático.	91
Figura 60: Características de tarjeta de video.	91
Figura 61: Descarga de driver NVIDIA.	92
Figura 62: Comprobación del sistema.	92
Figura 63: Elección de elementos.	93
Figura 64: Descarga de CUDA.	94
Figura 65: a) Comprobación de compatibilidad. b) Opciones de Instalación.	94

Figura 66: a) Verificación de versiones. b) Ubicación de instalación.	95
Figura 67: CUDA a) Requerimientos. b) Verificación de requerimiento.	95
Figura 68: Documentación de herramienta CUDA.	96
Figura 69: a) VS 201.7 b) Instalación Finalizada.	96
Figura 70: Pruebas satisfactorias de estimación humana.	97
Figura 71: GUI de control de ingreso del sistema electrónico de biomecánica.	98
Figura 72: Alerta de seguridad del sistema.	98
Figura 73: Panel de calibración del sistema.	99
Figura 74: Panel de control.	99
Figura 75: Ayuda del sistema electrónico.	100
Figura 76: Ayuda de aplicación.	101
Figura 77: Aplicación celular.	101
Figura 78: Programa Laptop.	102
Figura 79: Resultado de instalación DroidCam.	103
Figura 80 Reconocimiento cámara web.	104
Figura 81: Descarga XAMPP.	105
Figura 82: Selección de servidor y lenguaje de programación.	106
Figura 83: Selección de ubicación y lenguaje.	106
Figura 84: Instalación XAMPP.	107
Figura 85: Inicio de módulos para uso de XAMPP.	107
Figura 86: Base de datos en el servidor.	108
Figura 87: Funcionamiento de motor.	109
Figura 88: Poleas y banda.	109
Figura 89: Servomotor, ultrasónico y cámara.	110
Figura 90: Soporte de cámara.	110
Figura 91: Verificación de salidas de voltaje.	111
Figura 92: Circuito de prototipo.	111
Figura 93: a) HMI Contraseña. b) Control Sistema.	112
Figura 94: Calibración posición cámara.	112
Figura 95: Funcionamiento de sensor.	113
Figura 96: Análisis de resultados del voltaje DC y tiempo de activación.	115
Figura 97: Ángulos en la estimación humana.	116
Figura 98: Ángulo en el cuerpo humano.	117
Figura 99: Ingreso de base de datos.	117

RESUMEN EJECUTIVO

En el presente proyecto de investigación se describe el diseño e implementación del sistema electrónico de corrección del gesto deportivo en base a la biomecánica en deportistas de marcha atlética mediante machine learning, deep learning y aprendizaje automático profundo, estas bibliotecas y modelos que están dirigidos a este campo ayudan a la implementación del prototipo de seguimiento de los practicantes de marcha atlética mediante la sensorización y detección, esta estructura es apta para todo tipo de practicantes, ya sea de diferente edad, estatura o condición física. Para el funcionamiento del procesamiento de las imágenes recolectadas requiere de un sistema de visión artificial que depende de las redes neuronales convolucionales (CNN) que ayuda con la detección de puntos claves para dibujar un esqueleto sobre el practicante de marcha atlética y proceder al análisis de cada movimiento.

El prototipo del sistema de seguimiento consta de un sensor de presencia para la activación de un motor a pasos que sirve para desplazar el soporte de la cámara en el riel electrónico con el objetivo de recolectar imágenes, el soporte consta de servomotores que sirve para colocar la cámara a un punto fijo del practicante de marcha atlética, el prototipo también tiene de finales de carrera que sirve tanto para finalizar y resetear el funcionamiento del sistema. La adquisición de las imágenes es para realizar el análisis respectivo de los movimientos que realizó el marchista tanto en la fase de apoyo simple y la de apoyo doble, este análisis se realiza mediante el cálculo de ángulos en todas las etapas de las fases, ángulos que son enviados a una base de datos.

Todo este procedimiento y gracias al conjunto de pruebas dio como resultado la obtención de un sistema con alta eficiencia para la detección y análisis de las diferentes etapas de las fases de la marcha atlética.

Palabras claves: Redes neuronales convolucionales (CNN), Procesamiento y análisis de imágenes, Visión artificial, OpenCV, modelos de aprendizaje profundo automático.

ABSTRACT

This research project describes the design and implementation of the electronic sports gesture correction system based on biomechanics in athletic walking athletes using machine learning, deep learning and deep machine learning, these libraries and models that are aimed at this field help the implementation of the prototype for monitoring the practitioners of athletic walking through sensorization and detection, this structure is suitable for all types of practitioners, whether of different age, height or physical condition. For the operation of the processing of the collected images requires an artificial vision system that depends on convolutional neural networks (CNN) that helps with the detection of key points to draw a skeleton on the practitioner of athletic walking and proceed to the analysis of each movement.

The prototype of the monitoring system consists of a presence sensor for the activation of a stepper motor that serves to move the camera support on the electronic rail in order to collect images, the support consists of servomotors that serve to position the camera at a fixed point of the athletic gait practitioner, the prototype also has limit switches that serve both to end and reset the system's operation. The acquisition of the images is to carry out the respective analysis of the movements that the marcher made both in the single support phase and the double support phase, this analysis is carried out by calculating the angles in all the stages of the phases, angles that they are sent to a database.

All this procedure and thanks to the set of tests resulted in obtaining a highly efficient system for the detection and analysis of the different stages of the phases of athletic walking.

Keywords: Convolutional Neural Networks (CNN), Image Processing and Analysis, Computer Vision, OpenCV, Machine Deep Learning Models.

Glosario de términos y acrónimos

CUDA: La arquitectura unificada de dispositivos de computo es un modelo de programación para computación general en unidades de procesamiento gráfico.

GPU: Las unidades de procesamiento gráfico es un coprocesador que se encarga de acelerar la carga del trabajo del procesador central.

CNN: Las redes neuronales convolucionales es un algoritmo de Deep Learning diseñado para trabajar en el campo de la visión por computadora.

Deep Learning: El Deep Learning es un tipo de aprendizaje de computadora para que realice tareas similares a los humanos.

Machine Learning: Machine Learning es una rama de la inteligencia artificial usada para el análisis de datos que son recolectados en la automatización de modelos analíticos.

RGB: Es un modelo de color que consiste en representar distintos colores a partir de los colores primarios.

PAF: Un campo de afinidad de partes es un conjunto de campos de flujo que codifica relaciones de pares no estructuradas entre partes del cuerpo.

IA: La inteligencia artificial es una rama de la informática dedicada a la creación de sistemas informáticos inteligentes.

CI: Un circuito integrado es un chip donde se integra varios componentes electrónicos.

IDE: El entorno de desarrollo integrado es una herramienta utilizado por los programadores para desarrollar software.

VS: Visual Studio es un entorno de desarrollo integrado.

FPS: Los fotogramas por segundo es una unidad de medida de la velocidad de cuadros que muestra cuántos cuadros de video cambian en la pantalla en un segundo.

HMI: Una interfaz hombre máquina es un método y medio de proporcionar interacción directa entre un operador y un sistema técnico, proporcionando al operador la capacidad de controlar este sistema y monitorear su operación.

GUI: Una interfaz gráfica de usuario es un tipo de interfaz de usuario que permite a los usuarios navegar por una computadora o dispositivo y realizar acciones utilizando indicadores visuales e íconos gráficos.

ReLU: La unidad lineal rectificadora es la función de activación más utilizada en los modelos de aprendizaje profundo.

CAPÍTULO I.- MARCO TEÓRICO

1.1 Tema de investigación

“SISTEMA ELECTRÓNICO DE CORRECCIÓN DEL GESTO DEPORTIVO EN BASE A LA BIOMECÁNICA EN DEPORTISTAS DE MARCHA ATLÉTICA.”

1.2 Antecedentes investigativos

Mediante la investigación realizada en repositorios de diferentes Universidades, artículos publicados en revistas de investigación médicas, reglamentos de competición de deportistas y libros, se ha encontrado investigaciones que, mediante su enfoque y resultados, tienen relación a la biomecánica de los atletas practicantes de diferentes disciplinas del deporte.

En China en el año 2017 se presenta el artículo publicado en la IEEE con el tema “REAL-TIME HUMAN GESTURE GRADING BASED ON OPENPOSE” a cargo de los investigadores Sen Qiao, Yilin Wang y Jian Li, ellos realizaron un sistema de clasificación de gestos humanos 2D en tiempo real a partir de imágenes monoculares basadas en OpenPose, una biblioteca para la detección de puntos clave de múltiples personas en tiempo real, logrando resultados sobre el sistema que funcionaba eficientemente con un alto rendimiento en tiempo real, un bajo costo de equipo y una gran robustez ante la interferencia del ruido. Para la implementación de este sistema se utilizó el sistema operativo de Linux con la ayuda del lenguaje de programación de Python y la biblioteca libre de visión artificial OpenCV.[1]

En el año 2018 en Stanford University California, Gabriela Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Srinath Sridhar, Gerard Pons-Moll y Christian Theobalt, publicaron en la IEEE su estudio con el tema “SINGLE-SHOT MULTI-PERSON 3D POSE ESTIMATION FROM MONOCULAR RGB”, realizaron un método diferente de estimación humana utilizando una cámara monocular RGB(Reed Green Blue) el cual permite la inferencia de la pose del cuerpo completo incluso si hay fuertes oclusiones parciales de otras personas, hasta objetos que estén en la escena de prueba del proyecto, cumpliendo el objetivo de obtener la estimación humana en 3D para

múltiples personas con la ayuda del modelo pre entrenado robusto llamado MuPoTs-3D realizado en Ubuntu, Python, con la biblioteca de aprendizaje PyTorch y un conjunto de datos de imágenes diseñado para la detección de objetos(COCO), que sirve para lograr un rendimiento óptimo del proyecto realizado.[2]

Rishabh Dabral, Nitesh B Gundavarapu, Rahul Mitra, Abhishek Sharma, Ganesh Ramakrishnan y Arjun Jain, en su artículo publicado en la IEEE con el tema “MULTI-PERSON 3D HUMAN POSE ESTIMATION FROM MONOCULAR IMAGES” en India, realizaron el proceso de estimación humana en 3D de múltiples personas a partir de imágenes monoculares, ayudándose de una máscara de red neuronal que es conocida como máscara RCNN esta máscara que se utilizó se la llama como HG-RCNN, que aprovecha la arquitectura Hourglass para la estimación de pose humana 3D en múltiples personas, logrando el objetivo mediante dos etapas la primera fue mediante la estimación de puntos clave en 2D y luego llevar a puntos clave en estimación 3D, logrando una red simple y modular para estimación 3D de múltiples personas que no requiere ningunos datos de pose 3D.[3]

En el año 2019 en Universidad de California Berkeley, Zhe Cao EECS, Gines Hidalgo Martinez, Tomás Simón, Shih-En Wei y Yaser A. Sheikh, presentan el artículo en la IEEE con el tema “OPENPOSE: REALTIME MULTI-PERSON 2D POSE ESTIMATION USING PART AFFINITY FIELDS”, realizaron la estimación humana en 2D en tiempo real para la detección de varias personas en una sola imagen, para lo cual se utilizó detección de puntos clave del cuerpo y pie combinado, basado en conjuntos de datos pre entrenados(COCO), este método utiliza una representación no paramétrica, a las que se las define como PAF(campos de afinada de parte), para la asociación de partes del cuerpo. Este sistema ascendente logra una alta precisión y un rendimiento en tiempo real independiente del número de personas actuales en la imagen, demostrando así que el uso de un refinamiento solo de PAF es capaz de lograr un aumento sustancial en el rendimiento y precisión del tiempo de ejecución. Openpose fue realizado tanto en C++ como en Python, con la librería de OpenCV.[4]

1.2.1 Contextualización del problema

En Ecuador, las Federaciones Deportivas se rigen en el reglamento de la Asociación Internacional de Federación de Atletismo artículo 230.[5] A lo largo del tiempo, han notado los principales errores en la práctica de la marcha atlética los cuales son: posición del tronco, posición de la cadera, movimiento de los hombros, movimiento de los brazos, y posición de la cabeza, estos errores son los causantes de que haya lesiones como tendinitis aquilea, bursitis retroaquilea y periostitis tibial. Estos errores en niños entre 14 a 17 años representan un 57%, en adolescentes 18 a 25 años un 43% y entre 26 a 30 un 38%.[6] Pese a que los practicantes de la marcha atlética destaquen o se preparen bien tanto en resistencia, fuerza, flexibilidad y velocidad, están propensos a cometer los errores anteriormente descritos,[7] por lo tanto, es necesario realizar un estudio sobre el análisis de movimiento de los marchistas, grabación de video y digitalización por computadora, utilizados para medir la locomoción humana y mejora del rendimiento del practicante de marcha atlética.[8]

A lo largo del tiempo, se han realizados estudios en biomecánica por parte de la “Revista Cubana de Investigaciones Biomédicas” de la actividad del cuerpo humano para solucionar problemas que ocurren debido a distintas condiciones a las que están expuestas el cuerpo, lo cual abre el campo para el desarrollo de sistemas que permite la detección de puntos porque se determinan geométricamente ángulos formados entre las articulaciones. La biomecánica en la marcha atlética es importante ya que aporta con distintos conocimientos, en esta investigación se destaca una de ellas como: ingeniería, anatomía y mecánica. Se destaca estas disciplinas porque ayuda en el diseño de sistemas electrónicos para la rehabilitación y corrección de la movilidad de los practicantes, imparten información de las partes del cuerpo, como están interrelacionadas entre si y las fuerzas que interactúan en el movimiento del practicante. [9], [10]

Hay fases importantes en la marcha atlética: la fase de apoyo, describe el intervalo de tiempo donde el suelo y el pie tiene contacto, la fase de oscilación describe el intervalo de tiempo en el que el pie está en vuelo, es decir suspendido en el aire, en el intervalo que el pie está en vuelo algunos autores aseguran una pérdida de contacto de 0,005 a 0,05 milésimas de segundo, esta pérdida de contacto depende

de algunos factores se destaca la edad y la velocidad del practicante, edad, sexo, aspectos biológicos y antropológicos. Por lo tanto, tomando a consideración estas fases se puede solucionar los errores descritos, mediante la biomecánica basado en la detección conjunta de puntos claves que se puede utilizar para analizar la inclinación máxima entre dichos puntos en la parte inferior de las extremidades.[11]

Los practicantes de marcha atlética con el pasar del tiempo, deben estar preparados para las competencias, no se registra una tasa de descalificaciones a nivel nacional, pero diario El Comercio supo manifestar que Andrés Chocho registra 11 descalificaciones en 13 años en competencias mundiales, de igual manera Jefferson Pérez en los juegos olímpicos de Atlanta de 1996 tuvo dos amonestaciones que era el límite en el reglamento, existía el riesgo de una descalificación, si un profesional tiene errores en la marcha atlética un practicante también lo tiene, estas descalificaciones a nivel mundial en las federaciones deportivas presentan una pérdida del 25 al 30% del monto de ingreso a las competencias, estas federaciones facturan un aproximado de 150 millones de dólares americanos, en relación a las descalificaciones se considera un porcentaje alto de pérdida, por lo tanto es necesario corregir todas las falencias en los practicantes con ayuda de un sistema electrónico que permita analizar detenidamente todos los movimientos para comparar con ejercicios que ayudan a la similitud de la técnica, ayudando al practicante a no pasar por un reaprendizaje de las técnicas. [12]

1.2.2 Fundamentación teórica

1.2.2.1 Inteligencia Artificial

La inteligencia artificial (IA) es un área amplia y compleja de la informática que se dedica a la creación de sistemas informáticos inteligentes para aumentar la eficiencia y la productividad al automatizar procesos que generalmente necesitan de inteligencia humana, es decir enfocada en el desarrollo de máquinas inteligentes que tiene la capacidad de simular el comportamiento rutinario de un ser humano para llevar a cabo tareas complejas y aprender gradualmente, utilizando una serie de algoritmos, sistema de software e información recopilada. La IA es el conjunto de tecnologías que se desarrollan de manera eficiente y rápida, las propiedades principales son la comprensión de lenguaje, el aprendizaje, las funciones de percepción, razonamiento y actuación que pueden llevarse a cabo bajo el control del dispositivo computacional.[13]

Algunos de los ejemplos de la IA son:[13]

- Procesamiento natural del lenguaje.
- Aprendizaje automático.
- Sistemas expertos.
- Agentes virtuales (chatbots y asistentes virtuales).
- Sistemas de recomendación.

La IA abarca varias áreas tecnológicas como se visualiza en la figura 1.



Figura 1: Áreas Tecnológicas [14], [15].

Elaborado por: Fredy Llambo.

El desarrollo del uso de IA conduce a la adaptación de tecnologías en sectores clásicos de la economía a lo largo de toda la cadena de valor y las transforma en herramientas de predicción, lo que lleva a la algoritmización de casi toda la funcionalidad, desde la logística hasta la gestión de la empresa. En la tabla 1 se presenta a modo de resumen los métodos que se usan en la IA.[14], [16]

Tabla 1: Métodos de IA [14], [16].

Elaborado por: Fredy Llambo

Lenguaje natural. Tecnología del habla	Visión por computadora (Redes Neuronales)	Análisis de los datos
<ul style="list-style-type: none"> • Textos: reconocer, traducir automáticamente • Discurso: reconocer, generar 	<ul style="list-style-type: none"> • Encontrar, rastrear, clasificar, identificar objetos • Extraer datos de imágenes • Analizar la información recibida <p>Se aplica para</p> <ul style="list-style-type: none"> • Reconocimiento de objetos • Análisis de video • Descripciones de contenido de imagen y video • Reconocimiento de gestos y escritura a mano • Procesamiento de imagen intelectual 	<ul style="list-style-type: none"> • Extraer conocimiento • Encontrar patrones en los datos • Predecir <p>Utilizar métodos</p> <ul style="list-style-type: none"> • Estadísticas • Econometría • Aprendizaje automático , aprendizaje profundo

Aplicaciones

El desarrollo de productos y servicios basados en inteligencia artificial requiere una interpretación inequívoca, el alcance de la IA es lo suficientemente amplio y cubre tantas tecnologías de mecanismos físicos completos destinados a realizar determinadas tareas y para implementar aplicaciones inteligentes, desde el aprendizaje profundo de aspiradoras hasta nuevos experimentos para examinar las condiciones de microgravedad. Con el pasar del tiempo llegaron a la conclusión de

que la única forma de implementar la inteligencia artificial es desarrollar sistemas de IA por sí mismas. [16]

Sin embargo, hoy en día la mayoría de las organizaciones prefieren no desarrollar tales soluciones y ni siquiera comprar sistemas de inteligencia artificial independientes, sino recibir herramientas de inteligencia artificial como parte de las aplicaciones corporativas. Toda su diversidad puede dividirse según el criterio de los puntos clave de desarrollo, algunas de las aplicaciones se las detalla a continuación.[16]

- IA para defensa militar, negocios, educación.
- IA antifraude.
- IA en la energía eléctrica.
- IA en bancos, comercio, bienes y servicios.
- IA en el deporte, medicina.

En la figura 2 se muestra de manera gráfica las aplicaciones de la IA que se pueden incorporar en varias empresas para desarrollar relaciones con los consumidores, mejorar los recursos humanos, optimizar todos los procesos, transformar productos en servicios e incluso cambiar el modelo de negocio. [15]



Figura 2: Aplicaciones de la IA [15]

Las ventajas y desventajas de la IA se los describe de manera general a continuación en la tabla 2.[13], [14], [16]

Tabla 2: Ventajas y desventajas de la IA [13], [14], [16].

Elaborado por: Fredy Llambo

Ventajas	Desventajas
<ul style="list-style-type: none"> • Los sistemas expertos ayudan a desarrollar soluciones efectivas donde no hay suficientes especialistas altamente calificados. • Los dispositivos autónomos examinan ambientes donde una persona no puede estar sin daño a la salud o peligro de vida. • La IA ayuda a minimizar el factor humano en áreas críticas de actividad en operaciones monótona, donde se requieren muchas horas de actividad focalizada. • En el transporte, la IA se utiliza para el control no tripulado de trenes, automóviles, barcos, lo que le permite moverse durante mucho tiempo sin interrupciones para el descanso del conductor. • AI proporciona a los usuarios de computadoras interfaces convenientes que le permiten emitir comandos y percibir información de forma natural. 	<ul style="list-style-type: none"> • La acumulación de una gran cantidad de datos personales sobre los usuarios de sistemas informáticos conduce a la restricción de las libertades personales, en una situación en la que las decisiones se tomarán por automatización, las personas enfrentarán fuerzas no susceptibles a la influencia humana ordinaria. • Los dispositivos alimentados por IA pueden no tener en cuenta todos los factores ambientales, ya que son menos exigentes en un entorno limpio. • El uso de IA puede afectar negativamente a la economía, los recortes de empleos debido a la automatización pueden conducir a un fuerte aumento del desempleo, ya hay muchas profesiones en peligro.

Tipos de aprendizaje

Una característica importante de la IA es aprender de manera automática y los tipos de aprendizaje que utiliza la inteligencia artificial son machine learning (aprendizaje automático) y deep learning (aprendizaje profundo). En la figura 3 se puede visualizar como la IA es un campo general que contiene a estos tipos de aprendizaje. [17]

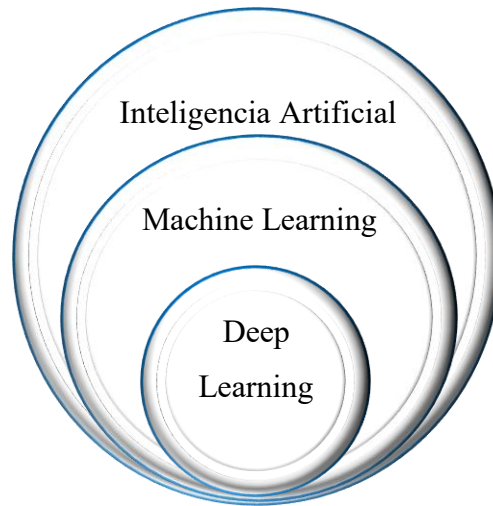


Figura 3: Tipos de aprendizaje de IA [17].

Elaborado por: Fredy Llambo.

Machine Learning (Aprendizaje Automático)

El aprendizaje automático es una de las ramas de la inteligencia artificial. El principio básico es que las máquinas reciben datos y aprenden de ellos. Los sistemas de aprendizaje automático permiten aplicar rápidamente el conocimiento adquirido en la capacitación, en grandes conjuntos de datos permitiendo que el sistema aprenda a reconocer patrones y tomar decisiones con una participación humana mínima.[18]

El aprendizaje profundo consta de 3 tipos de aprendizaje principales: [17], [18]

- Aprendizaje automático supervisado.
- Aprendizaje automático no supervisado.
- Aprendizaje automático reforzado.

El aprendizaje automático supervisado es la forma más común de aprendizaje automático, significa que refinas un algoritmo, entrenándolo en base a un conjunto

de datos y una respuesta correcta previamente conocida. En la figura 4 se muestra de manera gráfica el concepto del aprendizaje supervisado. [17], [18]

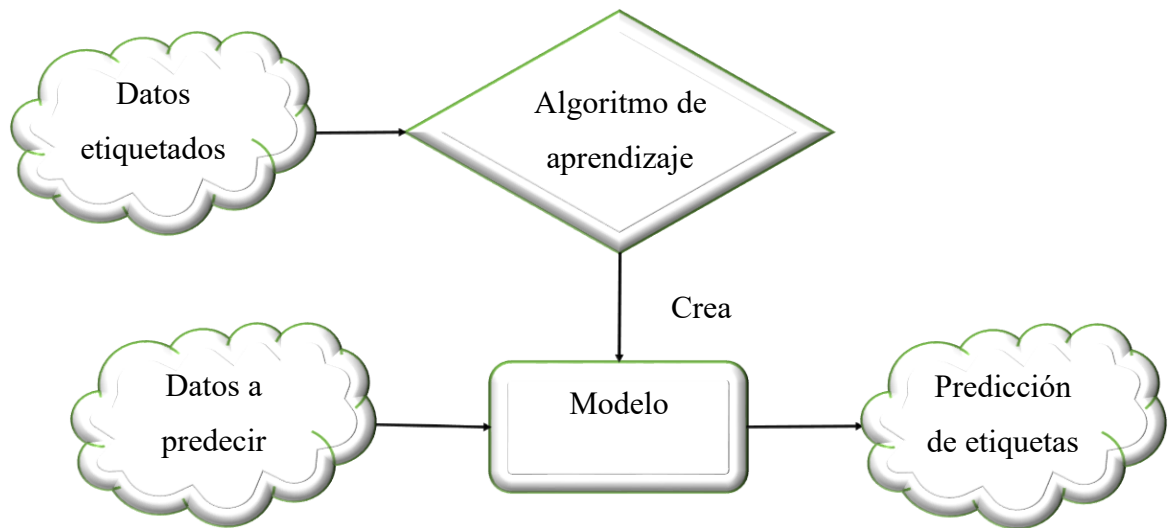


Figura 4: Esquema general del aprendizaje supervisado [17].

Elaborado por: Fredy Llambo.

El aprendizaje automático no supervisado examina los datos para identificar patrones, sin tener ninguna respuesta conocida como resultado. Y no puede comparar la función resultante con una respuesta conocida. [17], [18]

El aprendizaje automático reforzado se centra en los procesos de aprendizaje regulares que incluyen un conjunto de acciones, parámetros y valores finales. Al establecer las reglas, el algoritmo de aprendizaje automático intenta averiguar mediante varias opciones y evaluaciones cada resultado para determinar cuál es el más adecuado. [17], [18]

Deep Learning (Aprendizaje Profundo)

El aprendizaje profundo es un subconjunto del aprendizaje automático. Utiliza algunas técnicas de aprendizaje automático para resolver problemas del mundo real utilizando redes neuronales que pueden simular la toma de decisiones humanas y requiere una gran cantidad de datos de entrenamiento. Esto se debe a que hay una gran cantidad de parámetros que deben ajustarse para que los algoritmos de aprendizaje eviten falsos positivos. Por ejemplo, para descubrir cómo es un gato, se necesita una gran cantidad de imágenes para aprender a distinguir entre los detalles más pequeños que permiten distinguir a un gato de otro animal.[18]

1.2.2.2 Python

Python es un popular lenguaje de programación de alto nivel orientado a objetos con semántica dinámica integrada que admite estilos de programación orientados a objetos, funcionales e imperativos, capaz de desarrollar aplicaciones de sistema GUI, aplicaciones científicas, juegos, aplicaciones web, etc. Y lo hace de manera rápida y eficiente, lo que significa una mayor productividad del desarrollador y menores costos de soporte. En otras palabras, es un lenguaje simple pero lo suficientemente poderoso como para simplificar tareas en prácticamente cualquier ciencia o industria. Python ha evolucionado de manera rápida, actualmente está en uso activo, dos versiones principales: Python 2.x y Python 3.x. Aquí, la letra x se refiere a un lanzamiento específico de cada versión. [19]

El lenguaje de programación Python fue creado en 1991 por el holandés Guido van Rossum. Después de que Rossum desarrolló el lenguaje, lo publicó en Internet, donde la comunidad de programadores se unió a su mejora, logrando aplicaciones de escritorio gui, desarrollo de software, científico y numérico, aplicaciones comerciales, aplicación basada en consola, aplicaciones basadas en audio o video, aplicaciones cad 3d, aplicaciones empresariales, aplicaciones para imágenes.[20]

Características clave del lenguaje de programación: [19], [20]

- El código del programa se define como scripts.
- Soporte para una amplia variedad de paradigmas de programación, incluidos los paradigmas funcionales y orientados a objetos.
- Para trabajar con scripts, necesita un intérprete que ejecute y ejecute el script.
- No importa el sistema operativo: Windows, Mac OS, Linux, es suficiente para nosotros escribir un script que se ejecute en todos estos sistemas operativos con un intérprete.
- Gestión automática de la memoria.
- Escritura dinámica.

Algunas de las ventajas de Python son:[19], [20]

- En comparación con otros lenguajes de programación, python es el más aplicado por los desarrolladores últimamente porque es muy sencillo, fácil y dispone de varios recursos para principiantes. En los próximos párrafos, se describe las ventajas del lenguaje de programación para desarrolladores en contraste con otros lenguajes.
- Las principales ventajas del lenguaje Python son que es fácil de leer y fácil de aprender. Es más fácil escribir un programa en Python que en C o C ++. Con este lenguaje, tiene la posibilidad de pensar con claridad mientras codifica, lo que también hace que el código sea más fácil de mantener. Lo que reduce el costo de mantenimiento del programa y es visto como una de las ventajas de programación de Python.
- Python tiene algunas características únicas que son valiosas para los programadores porque facilitan la codificación y ningún error puede originar una falla de segmentación.
- Una ventaja importante del lenguaje Python es que tiene una amplia aplicabilidad y es ampliamente utilizado por científicos, ingenieros y matemáticos. Es por esta razón que Python es tan útil para la creación de prototipos y todo tipo de experimentos. Se utiliza en muchos campos innovadores. También se usa cuando se produce animación para películas y en aprendizaje automático.
- El lenguaje incluye una gran biblioteca con administración de memoria, que es otra de las ventajas de la programación Python.

Algunas de las desventajas de Python son: [19], [20]

- Como lenguaje interpretado, Python tiene una velocidad de ejecución lenta. Es más lento que C y C ++ porque funciona con un intérprete, no con el compilador.
- El lenguaje se considera menos adecuado para el desarrollo móvil y el desarrollo de juegos. A menudo se usa en computadoras de escritorio y servidores, pero solo hay varias aplicaciones móviles que se desarrollaron

con Python. Otra desventaja que tiene Python es el error de tiempo de ejecución. El lenguaje tiene muchos límites de diseño y necesita más tiempo de prueba. El programador tiene la posibilidad de ver errores solo durante el tiempo de ejecución.

- Python tiene un alto consumo de memoria y no se usa en navegadores web porque no es seguro. La flexibilidad del lenguaje se considera entre las ventajas y desventajas de Python.
- A los desarrolladores les gusta Python por su simplicidad en el aprendizaje y la codificación, tanto que puede ser difícil para algunos de ellos aprender y usar otros idiomas.

Principales usos de Python: [19], [20]

- Desarrollo de páginas web: Existe dos frameworks web Python más populares para principiantes que son Django y Flask. Útiles para principiantes si el objetivo del desarrollador es oportunidades de aprendizaje y escoger independientemente componentes a usar.
- Procesamientos de datos: Incluye aprendizaje automático, análisis y visualización de datos, usando un algoritmo que busca automáticamente un patrón determinado en los datos de entrada. La flexibilidad del lenguaje permite no limitarse a ningún paradigma de desarrollo, el programador puede escribir tanto programas orientados a objetos como procedimientos
- Aplicaciones integradas: Muchos desarrolladores utilizan este lenguaje para Raspberry Pi con la finalidad de conectar su proyecto al mundo real. Se usa a menudo para desarrollar partes individuales de una aplicación, le permite crear un sistema de modificación simple. Debido al alto grado de modularidad, cambiar una parte del programa puede no afectar a otra.
- Juegos: La biblioteca PyGame se puede usar para la creación de juegos, aunque hay disponibles desarrolladores de juegos más populares. Se puede crear proyectos de aficionados en él, pero para el desarrollo de juegos de alta gama es necesario mejores desarrolladores.

1.2.2.3 OPENCV

OpenCV es una biblioteca que hasta la primera versión fue desarrollada en el Centro de Desarrollo de Software Intel y por un equipo ruso en Nizhny Novgorod. Se la define como una colección de tipos de datos, funciones y clases para el procesamiento de imágenes con algoritmos de visión por computadora. OpenCV proviene de las siglas en inglés Open Source Computer Vision Library, esta biblioteca es muy popular debido a su apertura y la capacidad de usarla de forma gratuita con fines educativos y comerciales. OpenCV está escrito en un lenguaje de alto nivel C / C ++ y contiene algoritmos para:[21]

- Interpretar imágenes
- Calibrar una cámara contra una referencia
- Eliminar las distorsiones ópticas
- Determinar la similitud
- Analizar el movimiento del objeto
- Determinar la forma de un objeto y rastrear un objeto
- Reconstrucción 3d
- Segmentación de objetos
- Reconocimiento de gestos

En la versión 3, la estructura de la biblioteca se ha reorganizado, ahora en lugar de grandes módulos universales como cxcore (estructuras y algoritmos de datos básicos, operaciones en matrices numéricas multidimensionales, álgebra matricial, funciones matemáticas, generadores de números aleatorios, estructuras de datos de XML), Cvaux (búsqueda y descripción de características faciales, calibración estéreo, autocalibración, búsqueda de coincidencia estéreo y clics en gráficos) y Highgui (creación de una interfaz de usuario, funciones para organizar una interfaz de usuario simple, captura de video de cámaras y archivos de video), la biblioteca OpenCV se divide en pequeños módulos de acuerdo con el uso funcional en la tabla 3 se puede observar cuales son:[22]

Tabla 3: Módulos OpenCV [22].

Elaborado por: Fredy Llambo

Módulos OpenCV	Descripción
opencv_core	Núcleo: estructuras básicas, cálculos como funciones matemáticas, generación de números pseudoaleatorios, DFT, DCT, entrada / salida XML, etc.
opencv_imgproc	Procesamiento de imágenes como filtros, transformaciones, etc.
opencv_highgui	Interfaz de usuario simple, cargando / guardando imágenes y videos.
opencv_ml	Métodos y modelos de aprendizaje automático como SVM, árboles de decisión, etc.
opencv_features2d	Clase base abstracta para detectores de características de imágenes 2D y extractores de descriptores.
opencv_video	Análisis de movimiento y seguimiento de objetos como flujo óptico, patrones de movimiento, eliminación de fondo.
opencv_objdetect	Detección de objetos en la imagen, wavelets de Haar, HOG, etc.
opencv_calib3d	Calibración de la cámara, búsqueda de coincidencia estéreo y elementos de procesamiento de datos 3D.
opencv_flann	Biblioteca de búsqueda rápida de vecinos más cercanos
opencv_contrib	Módulos extras de OpenCV para CUDA
opencv_legacy	Código heredado retenido para compatibilidad con versiones anteriores
opencv_gpu	Aceleración de algunas funciones de OpenCV debido a CUDA (nvidia)

En la figura 5 muestra la imagen original que será procesada mediante funciones que están disponibles en la biblioteca de OpenCV. El código realizado en python y la librería de OpenCV que sirve para la presentación de la imagen es la siguiente:

```
import cv2 # import cv2 #Librería OpenCV  
img = cv2.imread('lena.jpg') #Lectura de la imagen  
cv2.imshow("Imagen a procesar", img) #Presentación de Imagen
```



Figura 5: Imagen a procesar.

Elaborado por: Fredy Llambo en base a [23]

En la figura 6 se logra el pixelado de la imagen original, analizando cada uno de los pixeles que integra la imagen y realizando un cálculo del promedio de pixeles vecinos y el código realizado en python para pixelar es:

```
import cv2 #Librería OpenCV  
img = cv2.imread('lena.jpg') #Lectura de la imagen  
alto, ancho = img.shape[:2] #Forma de imagen(Numero filas y columnas)  
alto1, ancho1 = (100, 100) #Nuevo Pixel  
nuevo = cv2.resize(img, (alto1, ancho1), interpolation=cv2.INTER_LINEAR) #  
una interpolación bilineal  
salida = cv2.resize(nuevo, (ancho, alto), interpolation=cv2.INTER_NEAREST) #  
una interpolación de vecino más cercano  
cv2.imshow('Output', salida)
```

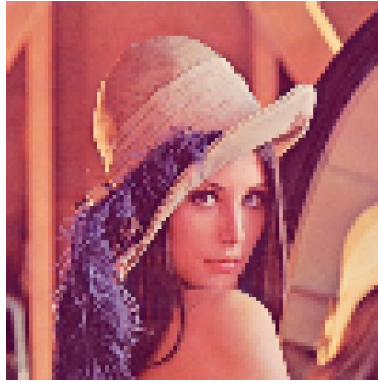


Figura 6: Pixelado de Figura 5.

Elaborado por: Fredy Llambo en base a [24]

En la figura 7 se logra el filtrado de la imagen original, realizando un algoritmo para la variación de ángulos de cada pixel, es decir la rotación de los pixeles integrada en la imagen y se logra mediante el código:

```
import cv2 #Librería OpenCV
import math #Librería para cálculos matemáticos
img = cv2.imread('lena.jpg') # Lectura de la imagen
ax,bx=50,100
ay,by=1,1
rows,cols=img.shape[:2]
for i in range(rows):
    for j in range(cols):
        x=int(ax*math.cos(2*math.pi*i/bx))
        y=int(ay*math.cos(2*math.pi*j/by))
        if i+ y<rows and j+x<cols:
            img[i,j]=img[(i+ y)%rows,(j+x)%cols]
cv2.imshow('Warped',img)
```

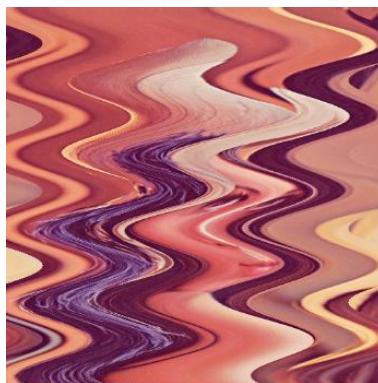


Figura 7: Filtrado de Figura 5.

Elaborado por: Fredy Llambo en base a [24]

1.2.2.4 MICROCONTROLADORES

Un microcontrolador es un circuito integrado(CI) programable o circuito electrónico, en cuyo chip se ensambla un microordenador real encargado de realizar diferentes instrucciones de forma secuencial con el objetivo de controlar o automatizar cualquier proceso, se puede configurar la cantidad específica de memorias y dispositivo auxiliar, estos dispositivos tienen los mismos bloques de funcionamiento que una computadora básica, por eso “Algirdas Baskys” lo denomino como una computadora de un solo chip de bajo costo.[25], [26]

Los microcontroladores están presentes en la mayoría de bienes de consumo, se puede encontrar en los microcontroladores en la mayoría de los dispositivos que controlan, miden, calculan o muestran información, son diseñados para reducir costos y consumo de energía, se utilizan como soluciones simples con respecto a la lógica transistor-transistor, se las abrevia con las siguientes letras MCU provenientes de las siglas en inglés “microcontroller unit”. A pesar del complejo dispositivo, el principio de funcionamiento del microcontrolador es muy simple porque se basa en un principio de funcionamiento analógico, es decir que pueden tomar cualquier valor de la señal de entrada dentro de cierto rango. [25], [26]

Un microcontrolador está diseñado para ser dependientes del control y recolección de datos, está integrado por varios componentes electrónicos para que puedan lograr muchos procesos diferentes por sí mismo, contiene varios componentes adicionales como microprocesador, temporizadores, conversores A/D(Análogo/Digital) y D/A(Digital/Análogo), referencias de voltaje, generadores PWM, interfaces seriales UART y USB. Se clasifican por el ancho del bus de datos de la CPU (4, 8, 16 y 32 bits), por sus funciones (universal y especializado) y por arquitectura del sistema informático como: CISC (computadoras con un conjunto de instrucciones complejas) y RISC (computadoras con un conjunto de instrucciones reducidas). El rendimiento y el consumo de energía del microcontrolador aumentan al aumentar la frecuencia del reloj. La frecuencia del reloj o velocidad del bus, determina cuántos cálculos se pueden realizar por unidad de tiempo. [25], [26]

Los dispositivos integrados más comunes en microcontroladores son: dispositivos de memoria, puertos de Entrada/Salida (E/S), interfaces de comunicación,

temporizadores y relojes del sistema. Los dispositivos de memoria incluyen RAM (memoria de acceso aleatorio), ROM (memoria solo de lectura), EPROM (ROM flash), EEPROM (ROM flash eléctrica). Los temporizadores incluyen tanto relojes en tiempo real como temporizadores de interrupción. Las herramientas de entrada/salida incluyen puertos de comunicación serie, puertos paralelos, convertidor analógico/digital y digital/analógico, LCD (controladores de pantalla de cristal líquido) o VFD (controladores de pantalla fluorescente de vacío). Los dispositivos integrados son altamente confiables ya que no requieren ningún circuito eléctrico externo. Los elementos básicos de un microcontrolador se puede observar en la figura 8 y son las siguientes:[27]

- Unidad de proceso.
- Memoria de datos.
- Puertos de Entradas y Salidas.

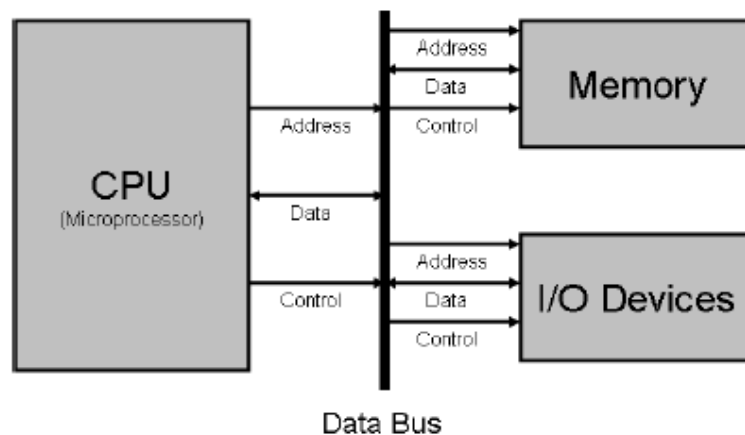


Figura 8: Elementos Básicos de un microcontrolador. [25]

En un microcomputador, estos elementos están separados como circuitos integrados, están conectados mediante un bus de datos en un circuito impreso y son controlados por un microprocesador (CPU proveniente de las siglas en inglés Central Processing Unit), que es considerado como el elemento más importante porque se encarga de almacenar, procesar y controlar todas las instrucciones o información recibida. El funcionamiento del microcontrolador y su periferia se lleva a cabo de acuerdo con un programa que se graba en la memoria interna y puede almacenarse en dicha memoria durante un período suficientemente largo. El

sistema puede ser diseñado para adaptarse a aplicaciones particulares, según el tipo de CPU, capacidad de memoria y la selección de unidades de entrada/salida para el control de elementos externos, lectura de señales y comunicación entre otros microcontroladores.[27]

Los microcontroladores actualmente tienen muchas aplicaciones increíbles. Desde automatización industrial hasta electrodomésticos, desde la gestión de estaciones nucleares hasta juguetes para niños, desde sistemas militares secretos hasta cambio de canales en su radio. En una palabra, es más fácil enumerar dónde no se aplican. La invención y el desarrollo posterior de los microcontroladores han revolucionado la electrónica digital. No solo los circuitos y la base del elemento han cambiado, sino también los principios de los sistemas de construcción. El ciclo de desarrollo ha sufrido cambios significativos. Han aparecido clases enteras de dispositivos, cuya existencia hubiera sido imposible sin controladores.[28]

Es económicamente factible producir varias docenas de tipos de controladores en lugar de cientos de variedades de circuitos lógicos. Un microcontrolador es mucho más sensible a los errores de programador que las computadoras grandes. La cantidad limitada de memoria, los requisitos de velocidad por pasos y la ausencia casi completa de infalible requieren desarrolladores altamente calificados. El programador que trabaja con controladores está mucho más cerca del hardware que el programador de aplicaciones. En consecuencia, no puede prescindir de los principios de funcionamiento de este hierro.[29]

La separación de software y hardware permitió unificar la base del elemento. Se puede usar el mismo controlador para crear muchos dispositivos diferentes. La unificación conduce a menores costos de producción. Hay una gran variedad de dispositivos en esta clase. Todos estos dispositivos se pueden dividir en los siguientes tipos principales: [29]

- Microcontroladores integrados de 8 bits
- Microcontroladores de 16 y 32 bits
- Procesadores de señal digital

Al igual que otros circuitos electrónicos los microcontroladores también tienen sus ventajas y desventajas a continuación en la tabla 4 se describe algunas de ellas.[25]–[27]

Tabla 4: Ventajas y desventajas de los microcontroladores [25]–[27].

Elaborado por: Fredy Llambo

Ventajas	Desventajas
<ul style="list-style-type: none"> • Se requiere poco tiempo para realizar la operación. • Los chips del procesador son muy pequeños y se produce flexibilidad. • Debido a su mayor integración, el costo y el tamaño del sistema se reducen. • El microcontrolador puede conectar fácilmente RAM, ROM y puertos de E / S adicionales. • Una vez que los microcontroladores están programados, no se pueden reprogramar. • Sin piezas digitales, puede actuar como microordenador. • Es fácil de usar, la solución de problemas y el mantenimiento de los sistemas es simple. 	<ul style="list-style-type: none"> • El microcontrolador no puede conectar dispositivos de alta potencia directamente. • Tiene una estructura más compleja en comparación con el microprocesador. • Solo realizó un número limitado de ejecuciones simultáneamente. • Generalmente se usa en micro equipos.

1.2.2.5 SENSORES

Los sensores son dispositivos, módulos, máquinas o subsistemas cuyo propósito es detectar eventos o cambios en su entorno y enviar la información a otros dispositivos electrónicos. Los sensores están en todas partes, uno de los primeros sensores conocido por la humanidad es la brújula, es la herramienta de navegación más antigua, este fue creado entre los tiempos romanos, la composición básica es de material magnético, una aguja fina que siempre rota, es considerada como un sensor porque detecta campos magnéticos muy leves, en la figura 9 muestra de manera gráfica cómo funciona la brújula. [30], [31]

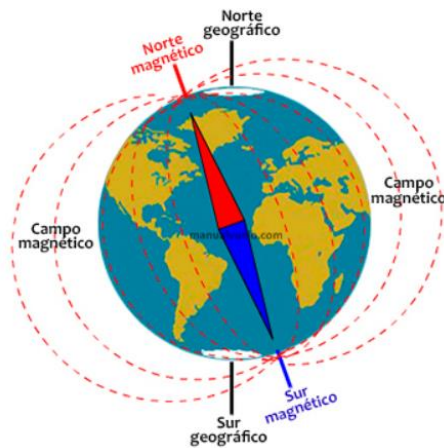


Figura 9: Comportamiento de la Brújula. [31]

Elaborado por: Fredy Llambo.

El principio del funcionamiento de la brújula se basa en la atracción de los extremos de una flecha hecha de un imán permanente a los polos magnéticos del planeta. Este mecanismo está ajustado para detectar los campos magnéticos naturales del planeta. La Tierra tiene un núcleo de hierro, que es parte de un cristal líquido y parcialmente sólido creado por la presión de la gravedad. Se cree que son los movimientos en el núcleo externo los que crean el campo magnético. El campo magnético de la tierra es paralelo a la superficie de la tierra y se dirige estrictamente al polo norte magnético. [31]

Los sensores varían en precisión, linealidad, resolución, respuesta de frecuencia, respuesta de ruido, impedancia de entrada y salida, y muchos otros parámetros. Según la estructura de construcción y según el método de conexión de los elementos, los sensores se pueden convertir secuencialmente, diferencial y

compensación. Juegan un papel importante en la sociedad por eso deben cumplir con ciertas propiedades que son indispensables para una correcta medición las cuales son: repetitividad, estabilidad, sensibilidad y resolución, estas propiedades son descritas a continuación. [30]

- La repetitividad describe la habilidad del sensor para producir la misma signatura de medida expuesta repetidamente, al no tener esta propiedad el sensor tendría algunas degradaciones y presentaría medidas erróneas.
- La estabilidad de un sensor se puede deducir examinando la forma general de la signatura del sensor. A corto plazo, a uno le gustaría minimizar o eliminar cualquier deriva, fluctuación o variaciones repentinas en una sola respuesta del sensor.
- La sensibilidad se define entonces como la relación entre la señal de salida y la propiedad medida. Por ejemplo, si un sensor mide la temperatura y tiene una salida de voltaje, la sensibilidad es una constante con las unidades $[V / ^\circ C]$
- La resolución es el cambio mínimo que se puede detectar en la cantidad que está midiendo.

Con el pasar del tiempo, en las industrias los sensores llegaron a formar partes de equipos industriales, disminuyendo considerablemente los accidentes en los operarios. El uso de varios sensores para monitorear el funcionamiento de los mecanismos se usa cada vez más. Los requisitos previos para el desarrollo de estos dispositivos es que el mejor control proporcionará productos de la mejor calidad. A los sensores a veces se denominan transductores de entrada y los actuadores como transductores de salida. Transductor es un término muy amplio que se refiere a cualquier dispositivo que convierte una magnitud física en eléctrica. [32]

Las ventajas y desventajas de los sensores se los describe en manera general a continuación en la tabla 5. [30], [32]

Tabla 5: Ventajas y desventajas de los sensores [30], [32].

Elaborado por: Fredy Llambo

Ventajas	Desventajas
<ul style="list-style-type: none">• Respuesta rápida.• Alta sensibilidad con gran respuesta de la señal.• Alta selectividad para reconocer específicas especies químicas.• Su tamaño es pequeño, ligero, compacto y conveniente para usar.• Rango de medida.• Precisión.	<ul style="list-style-type: none">• Linealidad.• Baja temperatura de operación.• Baja resolución• Vida útil limitada

Clasificación de sensores

Los sensores utilizados son muy diversos en la actualidad y se pueden clasificar según varios criterios que son: [33]

- Según el principio de funcionamiento.
- Según el tipo de señal de salida.
- Según el tipo de parámetro.
- Según el tipo de variable física.

En la figura 10 se muestra como se divide cada clasificación.[33]

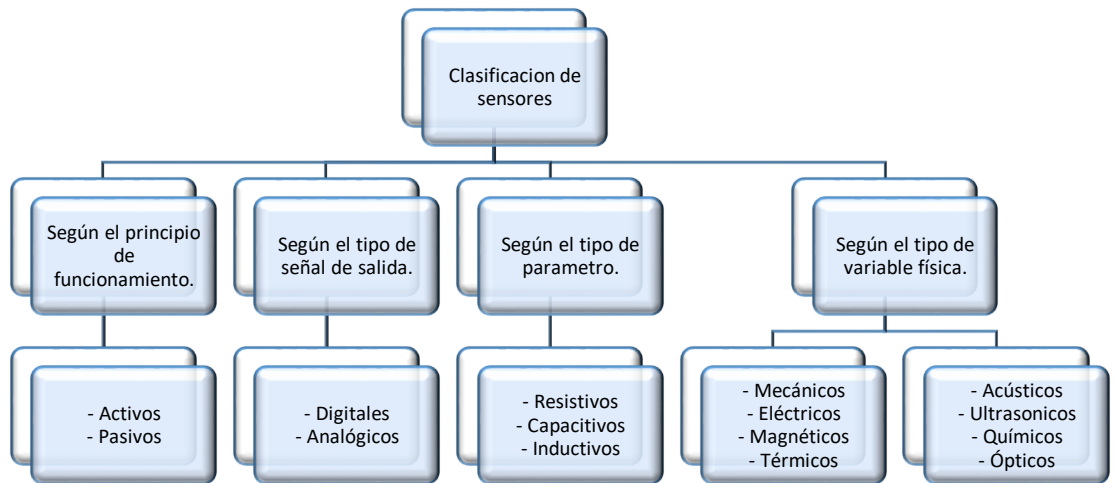


Figura 10: Clasificación de sensores [33].

Elaborado por: Fredy Llambo.

Existe una variedad de sensores en la figura 11 se muestra algunos de los más conocidos que son: de temperatura, luz, distancia, color, humedad, velocidad, sonido, ópticos y magnéticos. [33]

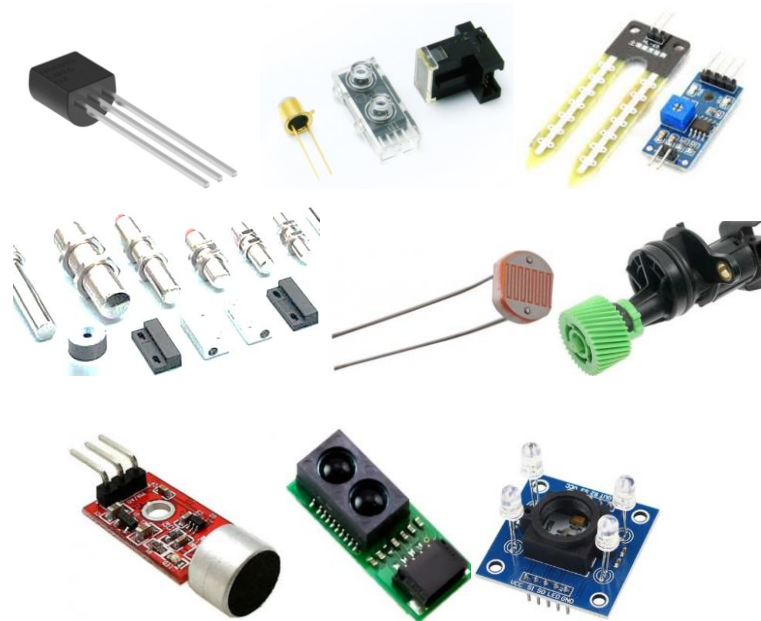


Figura 11: Sensores. [33]

Elaborado por: Fredy Llambo.

1.2.2.6 CAMARA DIGITAL

Una cámara digital es un dispositivo de hardware que toma fotografías y almacena la imagen como datos en una tarjeta de memoria. A diferencia de una cámara analógica, que expone los químicos en la película a la luz, una cámara digital utiliza componentes ópticos digitales para registrar la intensidad y el color de la luz, y la convierte en datos de píxeles. Una cámara que siempre está conectada a su computadora sin almacenamiento también podría denominarse cámara digital. Sin embargo, es más apropiado referirse a este dispositivo como una cámara web. Una cámara digital puede considerarse un dispositivo de entrada y salida ya que puede tomar fotografías (entrada) y enviarlas a su computadora (salida).[34], [35]

La cámara de video es una herramienta que está profundamente entrelazada en la vida cotidiana. Ya sea en un teléfono inteligente, tableta, computadora portátil u otro dispositivo, la mayoría de nosotros tienen la tecnología de video y rara vez se piensa en ello. La cámara es utilizada para la adquisición electrónica de imágenes en movimiento, desarrollada inicialmente para la industria de la televisión, pero ahora también es común en otras aplicaciones.[34]

La cámara recolecta imágenes en tiempo real directamente a una pantalla para observación inmediata. Algunas cámaras son para operaciones de seguridad, militares e industriales donde se requiere una visión subrepticia o remota. Las imágenes se graban en un dispositivo de almacenamiento para su archivo o procesamiento posterior. Durante muchos años, la cinta de video fue el formato principal utilizado para este propósito, pero fue reemplazado gradualmente por un disco óptico, disco duro y luego memoria flash. Las imágenes digitales en blanco o negro, es una matriz de n filas y m columnas, cada elemento de la matriz representa un pixel. [34]

Las cámaras de video modernas tienen numerosos diseños y uso, la cámara de video profesionales, como las que se usan en la producción de usualmente usan tres sensores para registrar por separado el rojo, el verde y el azul, la televisión de circuito cerrado (CCTV) generalmente usa cámaras de panorámica, inclinación y zoom (PTZ), para fines de seguridad y vigilancia, las cámaras web son cámaras de video que transmiten una transmisión de video en vivo a una computadora.[35]

Una característica interesante de las cámaras digitales es el ajuste automático del balance de blancos, que le permite compensar las distorsiones de color causadas por diferentes fuentes de luz (luz solar, luz incandescente o luz fluorescente). Pero como otros dispositivos también tiene ventajas y desventajas como se describe en la tabla 6. [34], [35]

Tabla 6:Ventajas y desventajas de las cámaras digitales [34], [35].

Elaborado por: Fredy Llambo

Ventajas	Desventajas
<ul style="list-style-type: none"> • Rango dinámico más alto, lo que lo hace mejor para capturar detalles en blancos y negros. • Es ligera. • Tiene una resolución alta. • Su almacenamiento. • Filtros incorporados. 	<ul style="list-style-type: none"> • Costo. • Enfoque. • Duración de batería.

1.2.2.7 PROCESAMIENTO DIGITAL DE IMÁGENES

El procesamiento de imágenes es un método para realizar análisis en una imagen, con el fin de obtener una imagen mejorada u obtener información útil de la imagen. Hoy en día, el procesamiento de imágenes se encuentra entre las tecnologías de rápido crecimiento. También forma un área central de investigación dentro de las disciplinas de ingeniería e informática.[36]

El procesamiento de imágenes básicamente incluye los siguientes tres pasos:[37]

- Importar la imagen a través de herramientas de adquisición de imágenes.
- Análisis y manipulación de imágenes.
- Salida de imágenes procesadas.

Para el procesamiento de imágenes existe dos métodos, el procesamiento de imágenes analógicas y digitales. El procesamiento de imágenes analógicas se aplica a señales analógicas y procesa solo señales bidimensionales. Las imágenes son manipuladas por señales eléctricas. En el procesamiento de imágenes

analógicas, las señales analógicas pueden ser periódicas o no periódicas. Se aplica un procesamiento de imagen digital a las imágenes digitales (una matriz de pequeños píxeles y elementos). Para manipular las imágenes, hay una serie de software y algoritmos que se aplican para realizar cambios. El procesamiento de imágenes digitales es una de las industrias de más rápido crecimiento que afecta la vida de todos.[36]

Para el procesamiento de imágenes se debe tener claro los conceptos entre píxel y resolución, porque son parámetros que se considera para el proceso.[33] Un píxel es una palabra compuesta de dos palabras en inglés pi (picture) y el (element), es por eso que se la define como el elemento más pequeño e indivisible, tiene un color y una ubicación específica en una imagen. Mientras que la resolución es la cantidad de píxeles tanto en alto y ancho que tiene una imagen. Cuantos más píxeles haya por unidad de área, mejor será la imagen.[33]

FASES DE PROCESAMIENTO DE IMAGEN.[36], [37]

1. **ADQUISICIÓN:** La adquisición es el proceso de recibir una imagen en formato digital. El trabajo principal implica en la conversión de color (RGB a escala de grises o viceversa)
2. **MEJORA DE IMÁGENES:** La mejora de imagen se utiliza para extraer algunos detalles ocultos de una imagen y es subjetiva.
3. **RESTAURACIÓN DE IMÁGENES:** La restauración de imagen se basa en un modelo matemático o probabilístico, para la recuperación de una imagen procesada.
4. **PROCESAMIENTO DE IMÁGENES EN COLOR:** Este procesamiento tiene como objetivo cambiar el color de un píxel o un conjunto de píxeles.
5. **PROCESAMIENTO DE RESOLUCIÓN MÚLTIPLE:** El procesamiento de resolución múltiple es la base para representar imágenes de diferente resolución en imágenes de una determinada resolución.
6. **COMPRESIÓN DE IMÁGENES:** La compresión de imágenes implica desarrollar algunas funciones para disminuir el tamaño de una imagen, pero sin perder su calidad.

7. PROCESAMIENTO MORFOLÓGICO: El procesamiento morfológico se trata de herramientas para extraer componentes de imágenes que son útiles en la representación y descripción de formas.
8. PROCEDIMIENTO DE SEGMENTACIÓN: El procesamiento de segmentación se trata de la división de una imagen en partes o separación de objetos constituyentes. La segmentación autónoma es la tarea más difícil en el procesamiento de imágenes.
9. REPRESENTACIÓN Y DESCRIPCIÓN: La representación y descripción sigue la salida de la etapa de segmentación, elegir una representación es solo la parte de la solución para transformar los datos sin procesar en datos procesados.
10. DETECCIÓN Y RECONOCIMIENTO DE OBJETOS: La detección y reconocimiento de objetos es un proceso que asigna una etiqueta a un objeto en función de su descriptor.

1.2.2.8 REDES NEURONALES CONVOLUCIONALES

Las redes neuronales convolucionales (CNN) es un algoritmo de aprendizaje profundo (Deep Learning) utilizada para la clasificación e identificación de objetos y de personas en imágenes, al hablar de CNN se podría decir que es la mezcla entre la matemática y biología, innovada para la visión por computadora, la CNN tuvo impacto en el año 2012 cuando Alex Krizhevsky ganó la competencia ImageNet que es la olimpiada de visión por computadora. Actualmente muchas empresas hacen uso de las redes neuronales y son conocidas por todo el mundo unas de ellas son: [38], [39]

- Facebook: Hace uso de redes neuronales para algoritmos de etiquetado automático.
- Google: Son utilizadas para buscar fotos de usuarios
- Amazon: Genera recomendaciones para los productos.
- Pinterest: Utilizado para personalizar la página de inicio de un usuario.
- Instagram: Sirve para la infraestructura de búsqueda.

Las CNN para obtener valores de entrada para la red neuronal artificial (RNA) se basa en la siguiente estructura que se muestra en la figura 12.

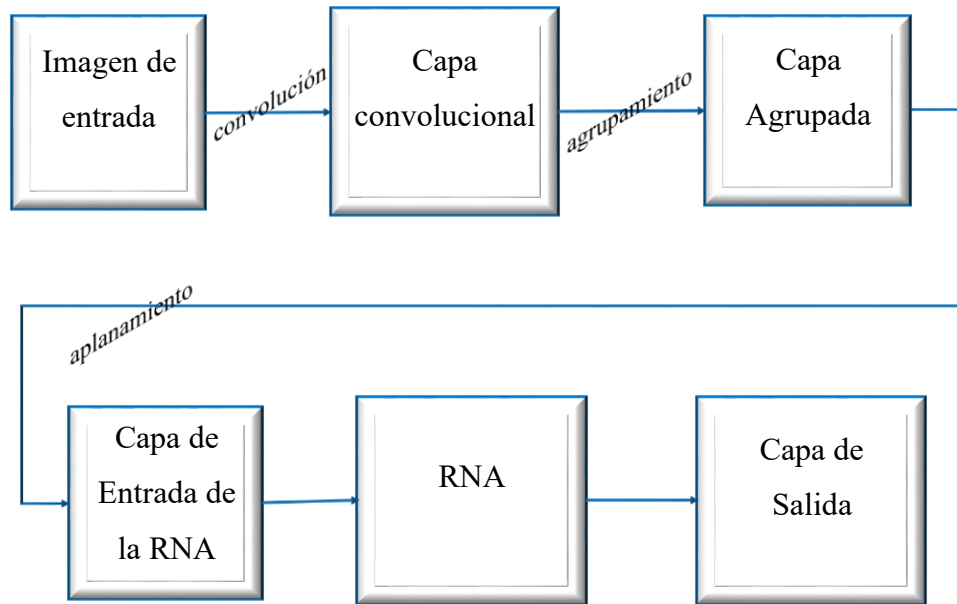


Figura 12: Arquitectura de las CNN [40].

Elaborado por: Fredy Llambo.

Para la comprensión de la arquitectura de las CNN se describe a continuación paso a paso con ejemplos.

Capa convolucional

Para obtener una capa convolucional se debe realizar el proceso de convolución en la imagen de entrada, este proceso es la forma de filtrar características necesarias para el procesamiento de la imagen, para lo cual se utiliza la matriz de convolución llamada matriz de Kernel, esta matriz es el filtro que se va a utilizar en la imagen, existen un sin número de filtros ya sea para bordes, contrastes, sombras, etc. Para el ejemplo en la tabla 7 se muestra la matriz de 3x3 a utilizar, este filtro es para líneas verticales.[40]

Tabla 7: Matriz de Kernel.

Elaborado por: Fredy Llambo.

$G[3,3] =$	-1	0	-1
	-1	0	1
	-1	0	1

Cuando se tiene una imagen de 1 solo color (escala de grises) de 16x16 de ancho y alto, significa que se utilizara 256 neuronas, pero si fuese una imagen a color se necesita 3 canales RGB, es decir se utilizara 16x16x3=768 neuronas. Estas neuronas forman nuestra capa de entrada. Para el ejemplo se utilizar una matriz de 6x6 con valores cualesquiera como se observa en la tabla 8.

Tabla 8: Matriz de entrada.

Elaborado por: Fredy Llambo.

F[6,6]=	3	5	2	2	7	1
	1	7	3	1	8	9
	5	4	8	4	2	6
	5	4	8	4	2	6
	9	6	1	8	7	1
	7	1	3	6	1	9

Se realiza la convolución 2D entre la matriz de Kernel y la matriz de entrada desplazando horizontal y verticalmente. Para el cálculo de la convolución en señales discretas (SD): [41]

$$f(n) * g(n) = \sum_{k=-\infty}^{\infty} f(k)g(n - k) \quad (1)$$

Para el cálculo de la convolución 2D se basa en la teoría del cálculo de la convolución entre SD. La solución es la sumatoria de la convolución entre la imagen de entrada y la matriz Kernel.

$$H[x, y] = F[x, y] * G[x, y]$$

$$H[x, y] = \sum_{n1} \sum_{n2} F[n1, n2]G[x - n1, y - n2] \quad (2)$$

Dónde:

F[x,y] : matriz de entrada.

G[x,y] : matriz Kernel.

Solución:

$F[6,6] * G[3,3]$

3	5	2	2	7	1
1	7	3	1	8	9
5	4	8	4	2	6
5	4	8	4	2	6
9	6	1	8	7	1
7	1	3	6	1	9



-1	0	-1
-1	0	1
-1	0	1

$$H[1,1] = F[1,1] * H[1,1] + F[1,2] * H[1,2] + F[1,3] * H[1,3] + F[2,1] * H[2,1] + F[2,2] * H[2,2] + F[2,3] * H[2,3] + F[3,1] * H[3,1] + F[3,2] * H[3,2] + F[3,3] * H[3,3]$$

$$H[1,1] = (3 * -1) + (5 * 0) + (2 * -1) + (1 * -1) + (7 * 0) + (3 * 1) + (5 * -1) + (4 * 0) + (8 * 1)$$

$$H[1,1] = -3 + 0 - 2 - 1 + 0 + 3 - 5 + 0 + 8$$

$$\mathbf{H[1,1] = 0}$$

$$H[2,1] = F[2,1] * H[1,1] + F[2,2] * H[1,2] + F[2,3] * H[1,3] + F[3,1] * H[2,1] + F[3,2] * H[2,2] + F[3,3] * H[2,3] + F[4,1] * H[3,1] + F[4,2] * H[3,2] + F[4,3] * H[3,3]$$

$$H[2,1] = (1 * -1) + (7 * 0) + (3 * -1) + (5 * -1) + (4 * 0) + (8 * 1) + (5 * -1) + (4 * 0) + (8 * 1)$$

$$H[2,1] = -1 + 0 - 3 - 5 + 0 + 8 - 5 + 0 + 8$$

$$\mathbf{H[2,1] = 2}$$

$$H[1,2] = F[1,2] * H[1,1] + F[1,3] * H[1,2] + F[1,4] * H[1,3] + F[2,2] * H[2,1] + F[2,3] * H[2,2] + F[2,4] * H[2,3] + F[3,2] * H[3,1] + F[3,3] * H[3,2] + F[3,4] * H[3,3]$$

$$H[1,2] = (5 * -1) + (2 * 0) + (2 * -1) + (7 * -1) + (3 * 0) + (1 * 1) + (4 * -1) + (8 * 0) + (4 * 1)$$

$$H[1,2] = -5 + 0 - 2 - 7 + 0 + 1 - 4 + 0 + 4$$

$$\mathbf{H[1,2] = -13}$$

Al realizar la convolución en toda la matriz de entrada con el filtro Kernel se obtiene los siguientes resultados como se muestra en la tabla 9.

Tabla 9: Resultado de la convolución.

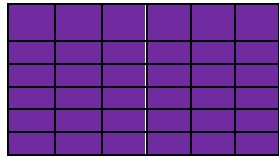
Elaborado por: Fredy Llambo.

H[4,4]=	0	-13	-10	7
	2	-8	-23	-6
	-18	-6	-10	-15
	-25	-1	-6	-14

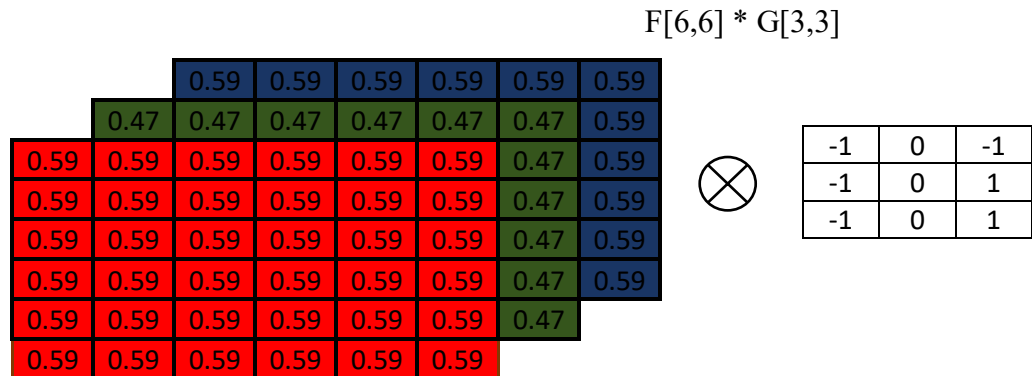
De la misma manera se realiza para las imágenes a color en las redes neuronales, las imágenes están formadas por los colores primarios que son las Rojo(R), Verde(G) y Azul(B). En la tabla 10 se puede observar la separación de los colores de la imagen de entrada, estas son denominadas capas.[40]

Tabla 10: Imagen de entrada.

Elaborado por: Fredy Llambo.

Imagen de entrada		} Capas	0.59	0.59	0.59	0.59	0.59
			0.59	0.59	0.59	0.59	0.59
			0.59	0.59	0.59	0.59	0.59
			0.59	0.59	0.59	0.59	0.59
			0.59	0.59	0.59	0.59	0.59
			0.59	0.59	0.59	0.59	0.59
			0.47	0.47	0.47	0.47	0.47
			0.47	0.47	0.47	0.47	0.47
			0.47	0.47	0.47	0.47	0.47
			0.47	0.47	0.47	0.47	0.47
			0.47	0.47	0.47	0.47	0.47
			0.47	0.47	0.47	0.47	0.47
			0.59	0.59	0.59	0.59	0.59
			0.59	0.59	0.59	0.59	0.59
			0.59	0.59	0.59	0.59	0.59

Para la convolución de la imagen se utiliza la misma matriz de Kernel de la tabla 7. Los colores varían de entre 0 a 255, pero en CNN se debe transformar de 0 a 1. Para cada color de debe realizar la convolución con la matriz de Kernel.[39]



$$H[1,1] = F[1,1] * H[1,1] + F[1,2] * H[1,2] + F[1,3] * H[1,3] + F[2,1] * H[2,1] + F[2,2] * H[2,2] + F[2,3] * H[2,3] + F[3,1] * H[3,1] + F[3,2] * H[3,2] + F[3,3] * H[3,3]$$

$$H[1,1] = (0.59 * -1) + (0.59 * 0) + (0.59 * -1) + (0.59 * -1) + (0.59 * 0) + (0.59 * 1) + (0.59 * -1) + (0.59 * 0) + (0.59 * 1)$$

$$H[1,1] = -0.59 + 0 - 0.59 - 0.59 + 0 + 0.59 - 0.59 + 0 + 0.59$$

$$\mathbf{H[1,1] = -1,18}$$

$$K[1,1] = F[1,1] * H[1,1] + F[1,2] * H[1,2] + F[1,3] * H[1,3] + F[2,1] * H[2,1] + F[2,2] * H[2,2] + F[2,3] * H[2,3] + F[3,1] * H[3,1] + F[3,2] * H[3,2] + F[3,3] * H[3,3]$$

$$K[1,1] = (0.47 * -1) + (0.47 * 0) + (0.47 * -1) + (0.47 * -1) + (0.47 * 0) + (0.47 * 1) + (0.47 * -1) + (0.47 * 0) + (0.47 * 1)$$

$$K[1,1] = -0.47 + 0 - 0.47 - 0.47 + 0 + 0.47 - 0.47 + 0 + 0.47$$

$$\mathbf{K[1,1] = -0,94}$$

$$M[1,1] = F[1,1] * H[1,1] + F[1,2] * H[1,2] + F[1,3] * H[1,3] + F[2,1] * H[2,1] + F[2,2] * H[2,2] + F[2,3] * H[2,3] + F[3,1] * H[3,1] + F[3,2] * H[3,2] + F[3,3] * H[3,3]$$

$$\begin{aligned}
 M[1,1] &= (0.59 * -1) + (0.59 * 0) + (0.59 * -1) + (0.59 * -1) \\
 &\quad + (0.59 * 0) + (0.59 * 1) + (0.59 * -1) + (0.59 * 0) \\
 &\quad + (0.59 * 1)
 \end{aligned}$$

$$M[1,1] = -0.59 + 0 - 0.59 - 0.59 + 0 + 0.59 - 0.59 + 0 + 0.59$$

$$M[1,1] = -1,18$$

Como la imagen es de un solo color, se realiza la convolución una sola vez para cada capa, en la tabla 11 se visualiza el resultado. A cada resultado de la convolución se la conoce como mapa de característica.

Tabla 11: Capa Convolutional.

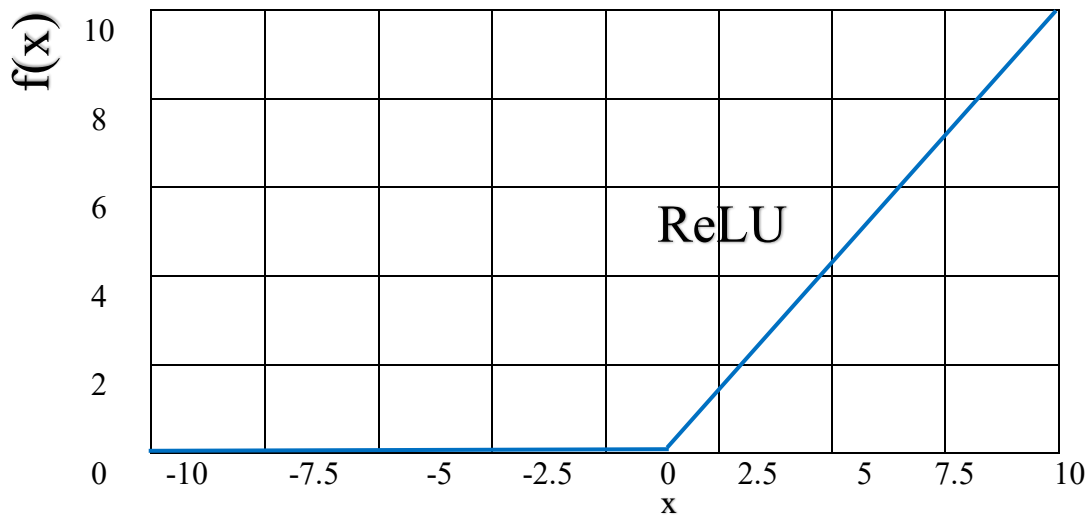
Elaborado por: Fredy Llambo.

H[4,4]=	-1,18	-1,18	-1,18	-1,18
	-1,18	-1,18	-1,18	-1,18
	-1,18	-1,18	-1,18	-1,18
	-1,18	-1,18	-1,18	-1,18
K[4,4]=	-0,94	-0,94	-0,94	-0,94
	-0,94	-0,94	-0,94	-0,94
	-0,94	-0,94	-0,94	-0,94
	-0,94	-0,94	-0,94	-0,94
M[4,4]=	-1,18	-1,18	-1,18	-1,18
	-1,18	-1,18	-1,18	-1,18
	-1,18	-1,18	-1,18	-1,18
	-1,18	-1,18	-1,18	-1,18

Los valores de la matriz Kernel de la CNN determina el número de características que se combinarán para obtener una nueva característica en la salida. El proceso de la convolución hace la imagen más pequeña sin perder información necesaria y ayuda al computador a procesarla. [39]

La explicación de la convolución lleva a la definición de la capa convolucional que es el conjunto de todos los mapas de características. En la capa convolucional se debe aplicar la función rectificadora ReLU (Unidad Lineal Rectificada) conocida como la función de activación el cual sirve para aumentar la no linealidad de la red neuronal y se expresa de la siguiente manera con su respectiva representación gráfica: [39]

$$f(x) = x^+ = \max(0, x) \quad (3)$$



El ReLU iguala todos los valores negativos a 0 y conserva los valores mayores a cero de cada píxel. Al aplicar ReLU a 2 mapas de características de la tabla 9 y 11 se obtiene los resultados de la tabla 12.

Tabla 12: Mapas de características con ReLU.

Elaborado por: Fredy Llambo.

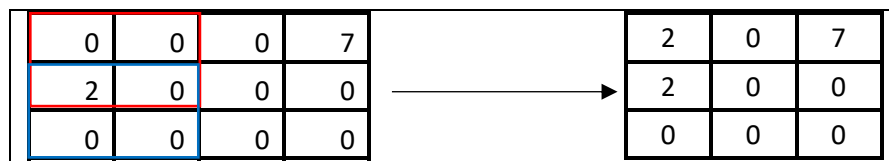
Tabla 9. H[4,4]				Tabla 11. M[4,4]			
0	0	0	7	0	0	0	0
2	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Capa agrupada

Para obtener una capa agrupada se debe realizar una aproximación máxima en cada mapa de característica, en las CNN se la conoce como Max-Pooling, esta consiste en aplicar una agrupación de 2x2, aplicar saltos horizontales o verticales y escoger el valor mayor de la matriz, mientras más cortos sean los saltos más detalles se adoptan, en la tabla 13 se da a conocer como se realiza el Max-Pooling.[3]

Tabla 13: Max-Pooling.

Elaborado por: Fredy Llambo.



Realizar repetidas veces el mismo proceso de agrupación da como resultado un conjunto de matrices agrupadas, conjunto llamado capa de agrupaciones.

Capa de entrada de la RNA

Para obtener la capa de entrada se procede a aplanar la capa agrupada de características, es decir a colocar todos los valores de manera vertical o en columna, tal cual se muestra en la figura 13.[40]

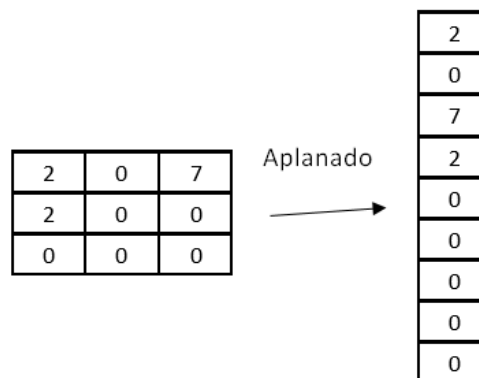


Figura 13: Capa de entrada de la RNA.

Elaborado por: Fredy Llambo.

RNA (Red Neuronal Artificial)

La RNA se encarga de aumentar más características a la capa de entrada para obtener mejores predicciones, estas características están añadidas en las neuronas de la capa completamente conectada. Para conseguir se debe enlazar todas las neuronas de entrada con todas las neuronas de la capa completamente conectada para obtener la capa de salida, como muestra la figura 14. [40]

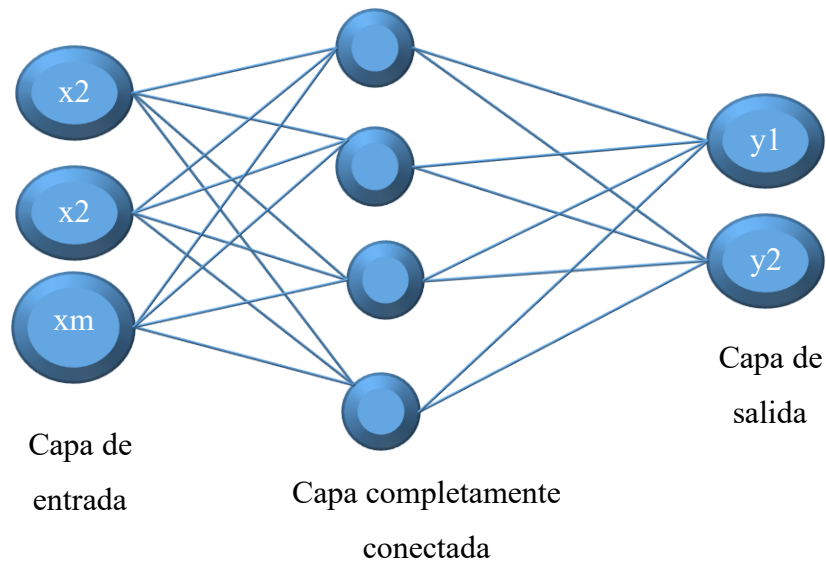


Figura 14: Red Neuronal Artificial [40].

Elaborado por: Fredy Llambo.

Matemáticamente para obtener el mapa de características se hace uso de la siguiente expresión: [39]

$$z_{i,j,k}^l = w_k^{lT} x_{i,j}^l + b_k^l \quad (4)$$

Dónde: [39]

k, l : Filtro k -enésimo y capa l -enésimo respectivamente.

$z_{i,j,k}^l (i, j)$: Mapa de característica y posición del píxel.

$x_{i,j}^l, w_k^l$: Peso o valor del píxel (i, j) de entrada y peso de la matriz Kernel.

b_k^l : sesgo o valor único de la neurona, sirve para determinar la probabilidad de si la neurona corresponde a un determinado valor de píxel.

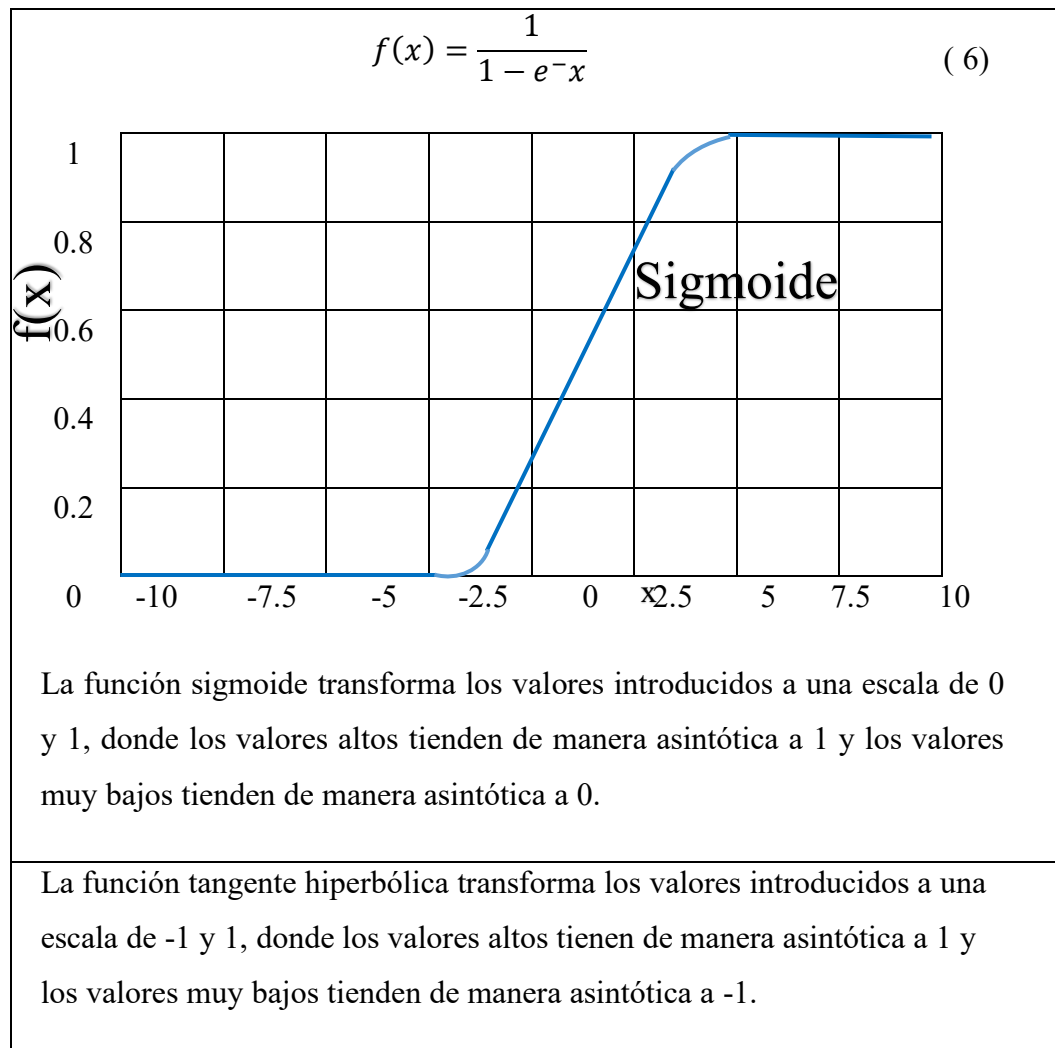
La función de activación $a_{i,j,k}^l$ de los mapas de características $z_{i,j,k}^l$ puede ser definido como: [39]

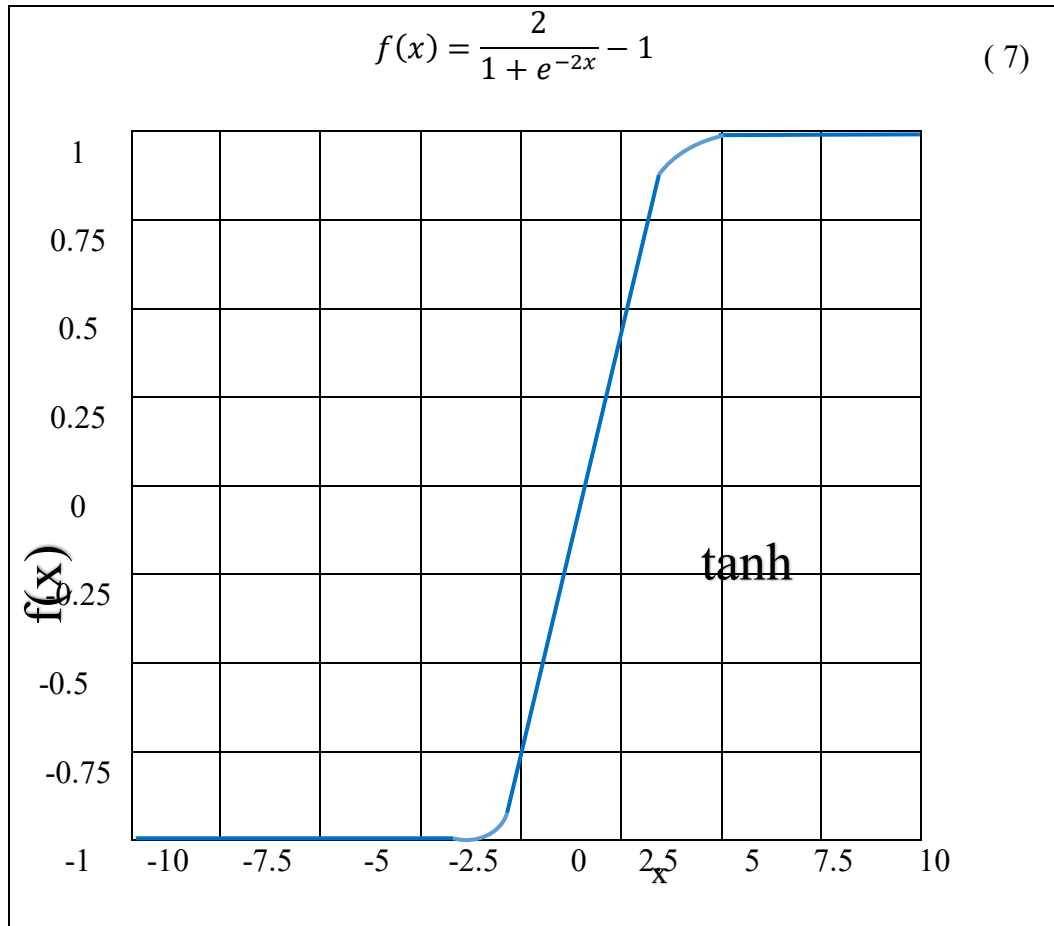
$$a_{i,j,k}^l = a(z_{i,j,k}^l) \quad (5)$$

Hay 3 funciones de activación utilizadas en las CNN que son la sigmoide, tanh y ReLU que sirven para aumentar la no linealidad. En la tabla 14 se muestra a detalle estas funciones. [39]

Tabla 14: Funciones de activación [39].

Elaborado por: Fredy Llambo.





La capa de agrupación tiene como objetivo reducir la resolución de los mapas de características, generalmente se coloca entre dos capas convolucionales. Cada mapa de características de una capa de agrupación está conectado a su mapa de características correspondiente de la capa convolucional anterior. La función de agrupación para cada mapa $a_{i,j,k}^l$ es: [39]

$$y_{i,j,k}^l = pool(a_{m,n,j,k}^l), \forall (m, n) \in R_{ij} \quad (8)$$

Dónde: [39]

(i, j), (m, n): ubicación de píxel vecino y de mapa de características.

R_{ij} : Vecino local alrededor de su ubicación

A veces en el procesamiento de las imágenes ocurren pérdidas de parámetros en los píxeles y pueden ser calculadas de la siguiente manera:

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^N l(\theta; y^n, o^n) \quad (9)$$

Dónde: [39]

θ denota los parámetros de CNN por ejemplo vectores de peso y términos de sesgo.

y^n corresponde a la etiqueta de destino (nodos).

o^n es la salida de CNN.

En base a la explicación de las CNN se puede describir de manera gráfica como se muestra en la figura 15.

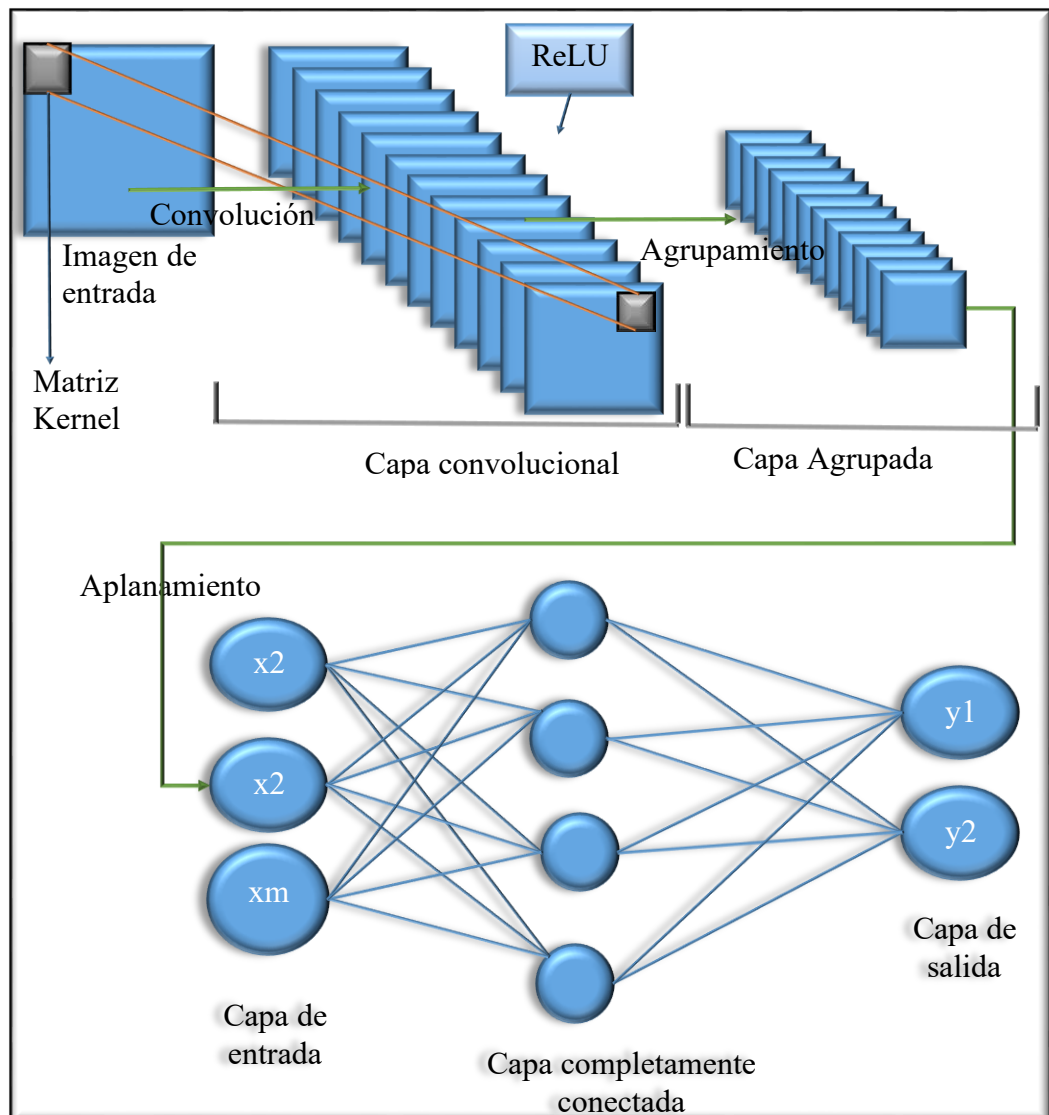


Figura 15: CNN descripción gráfica [40].

Elaborado por: Fredy Llambo

1.2.2.9 BIOMECÁNICA

La Biomecánica es el estudio del movimiento de los seres vivos utilizando la ciencia de la mecánica, que proporciona herramientas tanto matemáticas y también conceptuales necesarias para entender cómo se mueven los seres vivos y como los kinesiólogos pueden mejorar la actividad del movimiento y el tratamiento de problemas del sistema neuromuscular del ser humano. Cualquier persona que use correctamente el conocimiento de la biomecánica, ya sea un atleta profesional o un atleta aficionado que se esfuerza por mejorar la forma física y el bienestar diario, puede revelar completamente su potencial, al tiempo que minimiza la probabilidad de lesiones. [42]

Muchas áreas se interesan en la biomecánica, algunas de ellas son, Kinesiología, Ingeniería, Física, Biología y Zoología, este interés por la biomecánica es debido a la maravilla del movimiento tanto humano como animal, el interés del movimiento humano es por las aplicaciones que desglosa la biomecánica, la kinesiólogía de entre estas aplicaciones hace enfoque al deporte y al ejercicio, debido a varias aplicaciones de la biomecánica en el movimiento del ser humano para la mejora del rendimiento y el tratamiento de lesiones.[42]

La razón por el cual se ha desarrollado la biomecánica teórica, se basa en principios de ingeniería y modelado físico y matemático, el estudio de la biomecánica ha venido creciendo de manera multidisciplinaria, por lo que ha impulsado a los ingenieros a aumentar su nivel de conocimiento en el campo de la Biología, Medicina y Educación Física, actualmente la ciencia de la biomecánica está adherida en Ingeniería, Medicina, Deporte, movimientos fundamentales y problemas generales relacionados con la Biónica, la Ingeniería Biomédica y la Ingeniería de rehabilitación. Pero dentro de todo este estudio hay expertos del tema que se enfocan en:[43]

- ✓ Problemas de identificación de diferentes propiedades de la actividad del movimiento animal y humano.
- ✓ Modelado matemático y simulación de locomoción humana.
- ✓ Mediciones y control del movimiento de las extremidades inferiores. Metodología e investigación de la actividad de movimientos en el deporte.

- ✓ Problemas relacionados con la ingeniería de rehabilitación.
- ✓ Técnicas de medición necesarias para probar dichos modelos.

Al evaluar la capacidad humana ante lesiones causadas por alguna actividad física, se puede asegurar que mediante modelados matemáticos se puede obtener información sobre toda la anatomía del ser humano, es por eso que la biomecánica y la Ingeniería ayudaran hacer de este mundo un lugar seguro para vivir, con la unión de estas dos ramas se puede promover hábitos seguros de trabajo, mejores técnicas deportivas, para corrección de prueba de deporte y prevención de lesiones.[44]

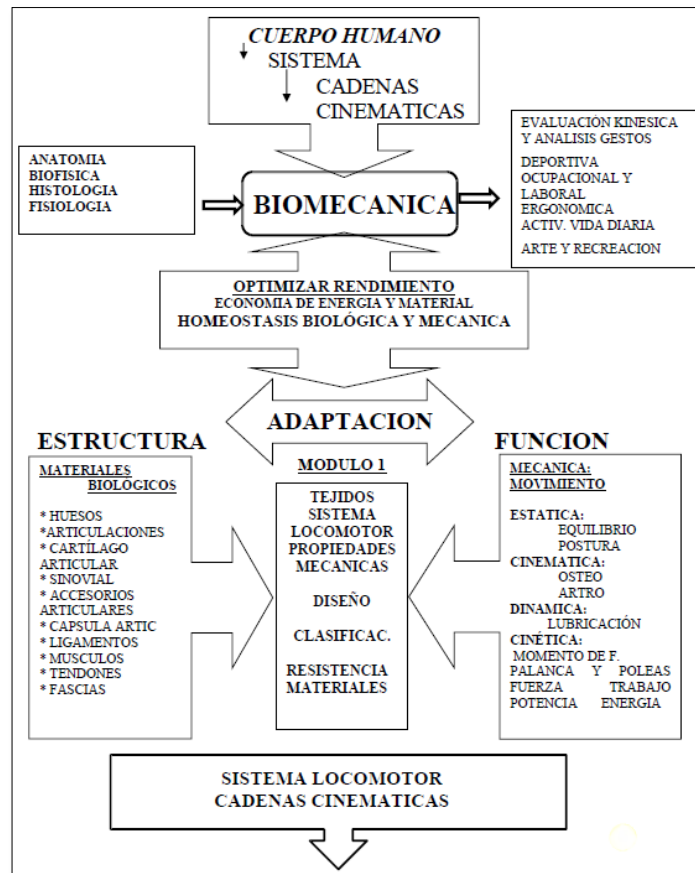


Figura 16: Representación gráfica del estudio de la biomecánica.

Fuente: [9]

En la figura 16 se puede observar las características principales de la biomecánica, como la estructura, sus funcionalidades, objetivos principales y las ramas que se necesita saber para el estudio a profundidad.

1.2.2.10 MARCHA ATLÉTICA

La marcha atlética es una disciplina del deporte que tiene como objetivo realizar series de paso diferente a la marcha normal, es decir que el contacto con el piso se conserve sin ninguna interrupción.[12]

- Durante la etapa de cada paso, el pie que traslada tiene que realizar contacto con el piso antes de que el otro pie haya perdido contacto al piso.
- La pierna de apoyo siempre tiene que estar derecha, es decir sin doblar la rodilla, hasta que llegue a posición vertical.
- Los jueces tienen la potestad de elegir un juez principal.
- En las competencias todos los jueces actúan de manera individual y sus calificaciones se basarán en observaciones realizadas a simple vista.

Ejercicios de iniciación que ayudan a la asimilación de la técnica de la marcha:

1. En el ejercicio 1 de la figura 17 se observa la marcha efectuando pasos largos en línea recta, moviéndose los brazos hacia adelante y atrás.[45]



Figura 17: Ejercicio 1. [45]

2. En el ejercicio 2 de la figura 18 se observa el mismo ejercicio 1, pero con la pierna derecha colocada hacia la izquierda de la línea trazada en el suelo y la pierna izquierda, a la derecha de la línea.[45]



Figura 18: Ejercicio 2. [45]

3. En el ejercicio 3 de la figura 19 se observa la marcha efectuando pasos largos girando simultáneamente los hombros en dirección al paso de la pierna contraria.[45]



Figura 19: Ejercicio 3. [45]

4. En el ejercicio 4 de la figura 20 se observa el mismo ejercicio 3, pero con un palo gimnástico colocado encima de los hombros.[45]



Figura 20: Ejercicio 4. [45]

5. En el ejercicio 5 de la figura 21 se observa el mismo ejercicio 4, pero con el palo sujeto detrás de la espalda, con los brazos doblados por las articulaciones cubitales.[45]



Figura 21: Ejercicio 5. [45]

6. Para realizar el ejercicio 6 de la figura 22 se puede visualizar que se debe pasar todo el peso del cuerpo de una pierna a la otra teniendo los músculos de la articulación coxofemoral relajados al máximo sin causar tensión, porque puede causar lesiones al musculo.[45]



Figura 22: Ejercicio 6. [45]

7. En el ejercicio 7 de la figura 23 se observa el mismo ejercicio 6, pero con apoyo de los brazos abre la valla.[45]



Figura 23: Ejercicio 7. [45]

8. En el ejercicio 8 de la figura 24 se observa la posición del pie al subir el muslo. Los brazos están doblados per la articulación cubital.[45]



Figura 24: Ejercicio 8. [45]

9. En el ejercicio 9 de la figura 25 se observa el mismo ejercicio 8, pero en el apoyo abre la valla.[45]



Figura 25: Ejercicio 9. [45]

10. En el ejercicio 10 de la figura 26 se observa la separación lateral de la pierna doblada por la rodilla utilizando el apoyo o sin éste.[45]



Figura 26: Ejercicio 10. [45]

11. En el ejercicio 11 de la figura 27 se observa pasar desde la media punta hacia el apoyo sobre el talón. Los pies colocados paralelamente al nivel de anchura los hombros.[45]

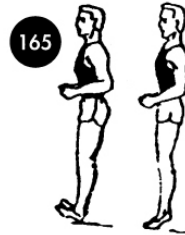


Figura 27: Ejercicio 11. [45]

En la fase de la pierna libre actúan algunos músculos a continuación se describe cuáles son los que intervienen en cada fase:

Fase de impulso Posterior

En esta fase la pierna posterior ayuda como palanca a la pierna de apoyo anterior, inicia la fase de impulso de la pierna libre y el musculo que interviene es el tríceps sural como muestra la figura 28.a. En la fase de impulso posterior la contracción del musculo isquiocrurales aporta a la extensión de la cadera en figura 28.b representa de manera gráfica. En la fase de impulso posterior el muslo oscila en vertical, la pierna es elevada por los músculos isquiocrurales para realizar la oscilación, esta oscilación anterior del pie tiene lugar en la articulación superior del tobillo realizada por el músculo tibial anterior. [46]

Fase de impulso anterior

En la fase de impulso anterior mientras va aumentando la flexión de la cadera, ejecutada por el musculo recto femoral, psoasíaco y tensor de la fascia lata como se visualiza en la figura 28.c, los isquiocrurales son más estirados, al final de la fase los isquiocrurales son los encargados de frenar la fase, e inicia la fase de apoyo anterior de la pierna de apoyo.[46]

Fase de apoyo anterior

Para la fase de apoyo anterior inicia la contracción del musculo glúteo mayor, en el momento de contacto del pie en el suelo. En la figura 28.d se muestra como

cumple la función de extensor de cadera, durante la fase de apoyo anterior recibe ayuda de los músculos aductores y isquiocrurales. Cuando la pierna de apoyo pasa de la posición vertical descrito gráficamente en la figura 28.e, empieza la fase de apoyo posterior.[46]

Fase de apoyo posterior

Los músculos glúteo mayor e isquiocrurales en la fase de apoyo posterior son los causantes de complementar la extensión de la cadera hasta que el pie despegue del piso. En el instante de elevación del pie, absolutamente todos los tejidos contráctiles que participan en la extensión de cadera, extensión de rodilla y doblamiento plantar, alcanzan su segundo y más alto grado de contracción como se observa en la figura 28.a.[46]

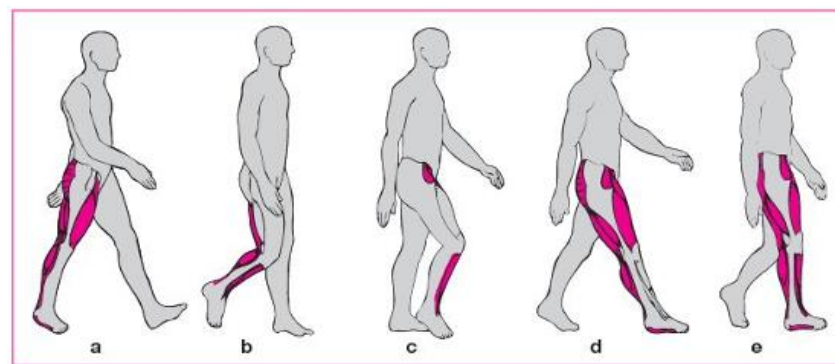


Figura 28: Participación muscular en las fases de impulso y apoyo. [46]

1.3 Objetivos

1.3.1 Objetivo general

Implementar un sistema electrónico de corrección del gesto deportivo en base a la biomecánica en deportistas de marcha atlética.

1.3.2 Objetivos específicos

- Identificar los factores principales que causan que los practicantes de marcha atlética cometan errores.
- Desarrollar una aplicación que me permita visualizar la pose del practicante de marcha atlética.
- Diseñar un sistema electrónico de seguimiento de la práctica de marcha atlética para la recolección de imágenes biomecánicas.

CAPÍTULO II.- METODOLOGÍA

2.1 Materiales

Para llevar a cabo el desarrollo de la metodología para el diseño e implementación del presente proyecto de investigación se emplearon los siguientes materiales como: artículos publicados en revistas científicas, proyectos de investigación relacionado al tema del proyecto, libros tanto físicos como virtuales, referencias de prácticas de marcha atlética.

2.2 Métodos

2.2.1 Modalidad de Investigación

El proyecto de investigación se sujetó a la modalidad de investigación y desarrollo (I+D) porque enfoca al campo de aplicación para defensa de conocimientos científicos previamente adquiridos para el desarrollo del sistema electrónico de captura y análisis de video enfocado en la biomecánica del marchista, su procedimiento será justificado mediante diferentes modalidades de investigación.

Modalidad Aplicada

Para el desarrollo del proyecto de investigación se empleó los conocimientos adquirido en el transcurso de la carrera en la Facultad de Ingeniería en Electrónica Sistemas e Industrial, especialmente los módulos de Comunicaciones Avanzadas, VLSI, Procesamiento Digital de Señales y programación, para la solución del sistema electrónico de corrección de la biomecánica de los marchistas.

Modalidad Bibliográfica

El desarrollo del proyecto de Investigación se basó en información confiable provenientes de fuentes científicas como: artículos científicos, publicaciones de internet, libros físicos y virtuales, revistas técnicas y proyectos de tesis que tengan similitud con el tema del sistema electrónica de corrección de la biomecánica de los marchistas.

Investigación Experimental

En el proyecto de investigación se realizó una serie de pruebas para el correcto funcionamiento y la calibración del sistema electrónico.

2.2.2 Recolección de Información

La información necesaria para el desarrollo del proyecto de tesis, tuvo origen de diferentes estudios, revistas técnicas, artículos científicos, proyectos ya realizadas, libros y fuentes proporcionadas de páginas web, que proporcionen información para el sistema electrónico de seguimiento, captura y análisis de video en relación al gesto deportivo, basado en la biomecánica en deportistas practicantes de la prueba de marcha atlética.

2.2.3 Procesamiento y análisis de datos

Para el procesamiento y análisis de datos del presente proyecto de investigación se realizó mediante constantes pruebas de prácticas con personas que estén orientadas en la marcha atlética de la siguiente manera:

- Analizar el seguimiento de la práctica de marcha atlética para determinar errores en el sistema.
- Comprobar la eficacia de la práctica de marcha atlética mediante la comprobación de los movimientos realizados por el practicante y los movimientos realizados por un profesional en el campo de la marcha.
- Elaborar tablas y diagramas de los datos en cada una de las pruebas para el análisis de la información adquirida.

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN

3.1 Análisis y discusión de los resultados

La implementación de un sistema electrónico de corrección del gesto deportivo, en base a la biomecánica en deportistas practicantes de la prueba de marcha atlética, permite mejorar las técnicas básicas y avanzadas de las prácticas de la marcha atlética, modernizando de esta manera los ejercicios simples y complejos de la marcha atlética, así se evita lesiones por sobre esfuerzo físico por largas practicas debido al fallo continuo en la misma práctica, al enfocarnos en ese punto se puede agilizar el tiempo que pasan en la marcha hasta obtener un eficiente desplazamiento y los practicantes no sean corregidos a simple vista.

El libre acceso a las bibliotecas de visión artificial, conocimiento del aprendizaje automático por computadora y el uso del lenguaje de programación interpretado hace posible conseguir los propósitos del proyecto de investigación, estos se logra mediante la recolección, procesamiento y análisis de imágenes de manera profunda, esto permite que detecte puntos claves desarrollados por neuronas artificiales en el practicante de la marcha atlética y se pueda observar de manera repetitiva la práctica realizada.

3.2 Desarrollo de la propuesta

3.2.1 Estudio de Factibilidad

Los detalles del análisis de la factibilidad del presente proyecto de investigación se las describen a continuación, tomando en consideración todo lo necesario para la implementación del prototipo y software.

3.2.1.1 Factibilidad Técnica

El presente proyecto de investigación es factible técnicamente en razón a que los materiales para la construcción de la estructura y dispositivos electrónicos requeridos están disponibles en el mercado nacional e internacional.

3.2.1.2 Factibilidad Económica

Se analizó la factibilidad económica del presente proyecto de investigación y se determinó su accesibilidad debido que los gastos fueron financiados por el investigador.

3.2.1.3 Factibilidad Bibliográfica

La información requerida para este proyecto de investigación se la puede encontrar en variedad de libros, artículos científicos, repositorios, tesis, etc. Por lo tanto, tiene factibilidad bibliográfica.

3.2.2 Requerimientos para el desarrollo del sistema

Para el desarrollo del proyecto de investigación fue necesario que el sistema electrónico cumpla con todas las exigencias necesarias en cada proceso. El enfoque del sistema electrónico es detectar al practicante de marcha atlética al pasar por frente de la cámara que está sobre el soporte deslizante e inicia el funcionamiento del riel motorizado para la captura y análisis del video, para el cual el sistema debe cumplir óptimamente con todas las funciones como sensorización, detección y captura de video inmediato sin presentar ningún tipo de fallo o retardo en tiempo real. El sistema necesita algunos requerimientos que son:

- El sistema debe ser calibrado de manera exacta al punto central del practicante para iniciar con la activación del sensor de presencia y evitar errores en el procesamiento de video.
- El sistema de detección debe funcionar de manera exacta para que no presente ningún retardo o fallo en la activación del motor para el desplazamiento del soporte de la cámara en el riel motorizado.
- El proceso de detección no debe presentar ningún fallo, porque es necesario para la recolección de video de toda la práctica e inicio del análisis de la recolección.
- El sistema de reinicio y paro debe actuar de manera instantánea sin presentar ninguna falla o retardo, porque puede llegar a ocurrir grandes problemas tanto en software como en hardware.

En el diagrama de bloques que se muestra en la figura 29 se puede observar las características básicas del sistema electrónico donde se especifica los componentes principales que se utilizó en la implementación. El sistema electrónico para corrección en relación al gesto deportivo de la biomecánica del practicante, está constituido por cuatro partes: La primera consta de un panel de control para la calibración y orientación de cámara hacia el practicante. La segunda es la determinación de la presencia o ausencia del marchista para la activación del sistema de movimiento del riel electrónico. La tercera es la de adquisición y procesamiento de los datos a través del sensor de detección de proximidad, el cual envía esta señal mediante un microcontrolador para el funcionamiento del motor que mediante la interacción con el software se pueda realizar en análisis de video. Y la cuarta es el envío de resultados a la base de datos.

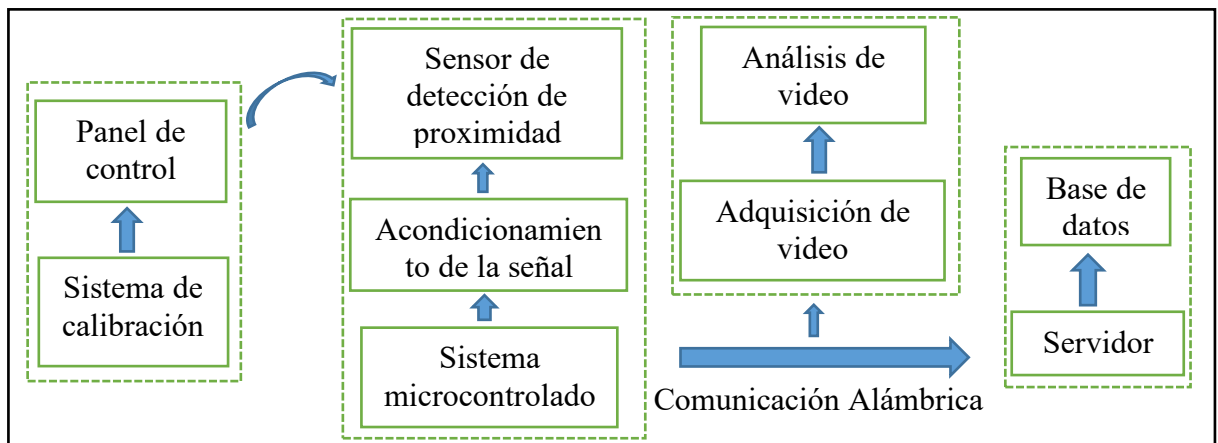


Figura 29: Diagrama de bloques del sistema electrónico.

Elaborado por: Fredy Llambo.

3.2.2.1 Selección de elementos para la implementación del sistema

De acuerdo al diagrama de bloques de la figura 29 se va a seleccionar los elementos de todo el sistema electrónico, para el cual se realiza un análisis de cada uno de los elementos para la selección óptima y adecuada que necesite la implementación del sistema. A continuación, se describe los dispositivos que son necesarios para el desarrollo del sistema:

- Sensor de detección de proximidad
- Motor y driver
- Microcontrolador y pantalla LCD
- Cámara web

- Procesador
- Tarjeta de video
- Lenguaje de programación

3.2.2.2 Cámara web



La función principal de la cámara web es la recolección de imágenes en la implementación del sistema electrónico, para el análisis y recolección de datos en los modelos preentrenados que posee las bibliotecas de python. No importa cuán trivial pueda sonar, debe verse como un nuevo dispositivo. La forma más fácil de determinar esto es por el cuerpo. Los plásticos, que ya han sido manipulados, generalmente son un poco brillantes y resbaladizos: las huellas digitales contienen grasa, que se mancha en una capa uniforme sobre el cuerpo cuando se limpia. El nuevo plástico se adhiere un poco a la piel. Es una sensación agradable cuando tienes una nueva técnica en tus manos recién sacada de la caja. Los paquetes que contienen los cables para la transmisión de señales y datos, el disco con el software debe estar en paquetes completos. Si los paquetes están rotos, la cámara estaba en uso y, posiblemente, regresó a la tienda después de la reparación.

En la tabla 15 se describe algunas características a tomar en cuenta para hacer una selección y sea acorde a las necesidades del sistema electrónico, para la elección de una cámara web la marca y modelo son más importantes gracias a sus reconocimientos durante la gama que ha venido ofreciendo en el campo de las ventas, actualmente hay cámaras web que vienen con alta resolución (HD), pocas marcas siguen con resoluciones bajas, los drivers de instalación son automáticos, sus actualizaciones de igual manera pero la compatibilidad no hay que dejar de tomar en cuenta por que el prototipo a implementarse se realiza para compatibilidad con todo sistema operativo que permita el uso de python.

A simple vista las cámaras escogidas para la comparación son de buena calidad, pero se considera como mejor opción a la cámara web Adesso modelo H2 ya que cumple con los requerimientos tanto de resolución como driver de instalación, esencialmente su tamaño y peso que son necesario para que se pueda desplazar en los rieles y el motor no sufra una sobrecarga.

Tabla 15: Comparación de cámaras web.

Elaborado por: Fredy Llambo en base a [47]–[50].

Cámaras Web				
Descripción gráfica				
Marca	Mecatech	Adesso	Altek	Genius
Modelo	001	H2	Xmk-82183	FACECAM 1000X
Sensor de imagen	10,0 MP	GC2053	-----	CMOS
Interfaces	USB2.0	USB2.0	USB 2.0	USB 2.0
Drivers de instalación	Incluye en CD	Incluye en CD	Incluye en CD	Incluye en CD
Resolución de imagen	5 Mpx	6 Mpx	720 Mpx	1MP
Resolución de video	640x480.	Full HD 1080p	VGA 480p	HD 720p
Peso	55g aprox	40g aprox	45g aprox	50g aprox
Compatibilidad	Windows, Linux, Mac	Windows, Linux	Windows, Linux, Mac	Windows, Mac OS, Linux 2.6.21
Costo	\$ 16.48	\$ 34.99	\$ 25	\$ 23.99

3.2.2.3 Sensor de detección de proximidad

El sensor de detección de proximidad es el dispositivo que se va encargar de determinar la presencia o ausencia del practicante, este sensor debe medir la distancia en la que se encuentra el practicante, existen una variedad de estos sensores cada uno con características y funcionamientos diferentes, se puede decir que uno mejor que otro. En la tabla 16 se describe algunos sensores de proximidad con sus respectivas características.

Tabla 16: Comparación de las características de sensores de proximidad.

Elaborado por Fredy Llambo en base a [51]–[56].

Sensor de proximidad	Fotoeléctrico Infrarrojo	Fotoeléctrico	Ultrasónico	Infrarrojo
Descripción gráfica				
Nombre/Referencia	E18-D80NK	E3F-R2N12	HC-SR04	IR FC-51
Voltaje de funcionamiento [V]	+5V DC	6~36VDC	+5V DC	3.3~5VDC
Corriente de trabajo	>25mA(min) ~100mA(máx)	300mA	15mA	18mA~20mA (5V)
Ángulo para detección	15°	-----	30°	35°
Temperatura ambiente	-25°C~55°C	-25°C~70°C	-15°C~70°C	-----
Rango de detección	3cm ~ 80 cm	2m (±10%)	2cm~4m	2~30cm
Dimensión	1.7 x 4.5cm	18 x 75mm	45x20x15mm	3.2 x 1.4cm
Costo	\$ 25	\$ 66,24	\$ 4	\$4,25

Para la selección del sensor de proximidad que cumpla con los requerimientos técnicos y electrónicos para el desarrollo del sistema, se realizó una comparación entre los sensores descritos en la tabla 16, se puede observar mediante las características que los sensores son de mucha utilidad para la medición y detección del marchista, pero se debe tomar en cuenta parámetros importantes para el desarrollo y necesidad del sistema los cuales se describen a continuación:

- Angulo de detección se considera porque los marchistas tienen diferente estatura y cambia el punto de detección.
- Rango de detección para enfocar al marchista todo el cuerpo o simplemente la parte inferior.
- Dimensión debido a que la estructura del riel no debe ser pesada para el movimiento del motor.

Al tomar como referencia los parámetros descritos el sensor fotoeléctrico infrarrojo E18-D80NK se puede considerar como buena opción porque el ángulo de detección es considerable y el costo no es alto porque es un sensor industrial, pero en relación al rango de detección es muy corta para enfoque de la cámara y la dimensión de igual manera es una desventaja para el movimiento del motor.

El sensor de proximidad fotoeléctrico E3F-R2N12 es mejor opción que el sensor fotoeléctrico infrarrojo, porque el rango de detección es de 2 metros, es adecuado para el enfoque de la cámara al marchista y se puede visualizar de manera exacta un punto en específico, pero las desventajas son muchas del sensor, la corriente de trabajo es muy alta para las fuentes de alimentación que se está usando en la implementación, la dimensión porque resulta difícil al motor el movimiento, y el costo por ser industrial es alto.

El sensor infrarrojo IR FC-51 se acopla bien a los parámetros de la implementación del sistema, el ángulo para la detección es correcto y no habrá problema para el funcionamiento del riel electrónico, su dimensión es perfecta para el movimiento del correcto del motor, y su costo es accesible para el investigador, pero la desventaja está en el rango de detección que es muy corta y es difícil el enfoque del marchista peor aún la activación del sistema.

Y por último está el sensor ultrasónico HC-SR04 que cumple con todos los parámetros que se toma a consideración por el enfoque de la cámara y son necesarios para la implementación del sistema, el ángulo de detección es el correcto para diferentes estaturas de los marchistas, el rango de detección es el indicado para poder enfocar tanto la parte inferior o todo el cuerpo de marchista, la dimensión es perfecta para el desplazamiento de la cámara y el costo es accesible para el desarrollo del sistema.

3.2.2.4 Motor


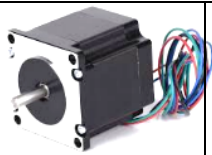


El motor es el dispositivo de vital importancia porque ayudara con el desplazamiento para el seguimiento del practicante de la marcha y posteriormente para la captura de las imágenes de la práctica realizada, el cual debe ser captadas de manera correcta para la facilidad de análisis del mismo, en la tabla 17 se describe características principales de algunos motores a pasos, para la selección del motor que requiere el sistema se debe priorizar mayor precisión al desplazar el peso de la base corrediza de la cámara, es decir para accionamientos de movimientos controlados y precisos, con la finalidad de evitar fallos en el arranque y parada del motor.

La correcta selección del motor es con la finalidad de evitar ruptura de la banda, tirones de la base de la cámara y principalmente errores en la detección de imágenes. A continuación, se describe a detalle las especificaciones que requiere el sistema para obtener resultados precisos y eficientes.

- El voltaje de trabajo debe ser de 12V (a mayor voltaje más velocidad) y la corriente de 0.5A porque la fuente de alimentación suministra este valor.
- El torque por la fuerza que debe ejercer al momento de desplazar el peso de la base de la cámara.
- El ángulo de paso debe ser de 1.8° porque gira 200 pasos en una vuelta y el sistema requiere para velocidad de rotación y seguimiento del marchista.

Tabla 17: Comparación de motor a pasos.

Elaborado por: Fredy Llambo en base a [57]–[62].

Motor	Motor a pasos			
Descripción gráfica				
Nombre/Referencia	Nema 17	Nema 23	Nema 34	28BYJ-48
Tipo de motor	Paso unipolar	Paso bipolar	Paso bipolar	Paso unipolar
Tamaño	42x42x39mm	57x57mm	114x37x34mm	35x15mm
Peso	290gr	350gr	3.8Kg	30gr
Corriente	0.4A	2A	5A	55 mA
Tensión	12V	4.5V	24V~60V	5V
Resistencia	30 Ω /bobina	2.25(\pm 10% Ω)/fase	1 Ω /fase	50 (\pm 7% Ω)
Angulo de paso	1.8°(200 pasos/vuelta)	1.8°(200 pasos/vuelta)	1.8°(200 pasos/vuelta)	5.625°(64 pasos por vuelta)
Numero de fases	4	2	2	4
Diámetro de eje	5mm	6.35mm	14mm	5mm
Inductancia por fase	30(\pm 20% mH)	3.6(\pm 20% mH)	11(\pm 20% mH)	-----
Torque	3.2Kg/cm	1.2N.m	8.63Nm	34 N/m
Costo	\$19	\$45	\$220	\$8

Tomando como referencia las especificaciones que requiere el sistema, el motor a pasos unipolar 28BYJ-48 se puede considerar como una opción viable porque el voltaje de trabajo, la corriente está en el rango de la fuente de alimentación y el costo es accesible para el investigador, la desventaja de esta opción es el torque y el ángulo de paso por que se requiere desplazar una base pesada y a mayor velocidad. Una mejor elección es el motor a pasos unipolar NEMA 17 por que cumple con todas las especificaciones para el sistema, el voltaje de trabajo, la corriente, el torque, el ángulo de paso y el costo es accesible para el investigador.

El motor a pasos bipolar nema 23 es mejor opción en relación al 28BYJ-48 porque el voltaje de trabajo, el torque y el ángulo de paso, son acordes a lo que se requiere para la implementación, pero el precio es muy elevado a comparación del motor a pasos unipolar nema 17, se hace comparación en precio porque realizan la misma funcionalidad con la diferencia del torque mas no de la velocidad que es primordial para el sistema. A simple vista se puede decir que el motor a pasos bipolar nema 34 es una gran opción, pero la desventaja es la escasas en el mercado y su precio es muy elevado para el investigador.

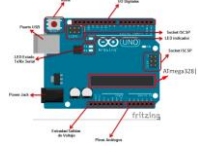



3.2.2.5 Microcontrolador y pantalla LCD

El objetivo principal del microcontrolador es la adquisición y procesamiento de los datos obtenidos por el sensor y los interruptores, ejecutara todas las instrucciones que requiere el sistema electrónico, en la tabla 18 se realizó un análisis comparativo con las características principales de los microcontroladores más conocidos y utilizados en la mayoría de implementaciones para la selección del microcontrolador idóneo que necesita el sistema electrónico.

Con la descripción de las características de los microcontroladores y los requerimientos del sistema electrónico se decidió utilizar el microcontrolador ATmega2560, que es una plataforma de hardware libre y entorno fácil de programación, el cual dispone para la utilización 54 pines digitales de I/O, de los cuales 15 pines son de salidas PWM y 16 pines de entradas analógicas, que son de gran ayuda para la convergencia de todos los dispositivos electrónicos del sistema, y para evitar vibraciones, temperaturas o humedad antes de su instalación se colocó una protección.

Tabla 18: Comparación de microcontroladores.

Elaborado por: Fredy Llambo en base a [63]–[68].


Microcontrolador	ATmega328	ATmega2560	ATmega32 U4	Atmel ATmega328
Descripción gráfica				
Voltaje Operativo	5V	5V	5V	5V
Voltaje de entrada recomendado	7~12V	7~12V	7~12V	7~12V
Voltaje límite máximo de entrada	6~20V	6~20V	6~20V	6~20V
Pines digitales I/O	14 (6 salidas PWM)	54 (15 salidas PWM)	20 (7 salidas PWM)	14 (6 salidas PWM)
Pines analógicos Inputs	6	16	12	8
Corriente por pines I/O	20mA	40mA	40mA	40mA
Corriente para 3.3V	50mA	50mA	50mA	-----
Velocidad de reloj	16MHz	16MHz	16MHz	16MHz
Memoria Flash	32KB (0.5KB para bootloader)	256KB (8KB para bootloader)	32KB (4KB para bootloader)	32KB (2KB para bootloader)
Memoria SRAM	2KB	8KB	2.5KB	2KB

Memoria EEPROM	1KB	4KB	1KB	1KB
Costo	\$17	\$25	\$15	\$10

Según las necesidades del prototipo se debe utilizar una pantalla para visualizar el funcionamiento o trabajo del sistema electrónico, para el cual se hace utilizara una pantalla LCD 16x2 que cumple con las características que son necesarias para el prototipo, en la tabla 19 se describe los parámetros de la LCD.

Tabla 19: Características de la LCD.


Elaborado por: Fredy Llambo en base a [69].

Pantalla LCD 16 x 2	
Descripción gráfica	
Voltaje de operación	(4.7V – 5.3V)
Ajuste de contrastes	Mediante una resistencia variable
Selector de registro	0 (comando) 1(datos)
Estados de registro	Bajo (escribir) Alto (leer)
Envío de datos	Enable
Pines de datos	DB0-DB1-DB2-DB3-DB4-DB5-DB6-DB7
Costo	\$ 7

Para el frenado y reinicio de todo el sistema electrónico es necesario la utilización de los sensores de detección de posición del soporte de la cámara, que realiza trabajo mediante el accionamiento mecánico, es decir el motor empieza su trabajo que desplaza mediante la banda y las poleas el soporte de la cámara, cuando llega al punto de finalización del desplazamiento se detiene y regresa a su posición original todo el soporte, para el desarrollo se utiliza un final de carrera SPDT largo de la marca omron serie 1185RE8, en la siguiente tabla 20 se puede observar las características del elemento anteriormente nombrado.

Tabla 20: Final de carrera SPFT largo.

Elaborado por: Fredy Llambo en base a [70], [71].

Sensor de detección de posición	Final de carrera SPFT
Descripción gráfica	
Corriente y voltaje de trabajo	15 A - 250VAC 0,6 A - 125 VDC 0,3 A – a 250 VDC
Polos	SPDT
Pines	3(No/NC/COM)
Longitud palanca	53 mm

La polea para el desarrollo del prototipo se describe a continuación.

- 20 dientes
- 0,7 cm de diámetro del eje
- 0,8 cm longitud de dientes por la medida de la banda
- 1,8 cm diámetro total

En la figura 30 se muestra la descripción grafica de las características descritas de la polea del motor a pasos.

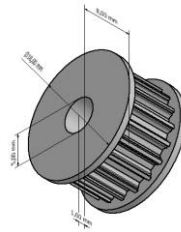


Figura 30: Descripción grafica de polea.

Elaborado por: Fredy Llambo en Autodesk Inventor Profesional.

3.2.2.6 Procesador

Para el análisis de video es esencial que la laptop tenga un procesador o CPU de gama media o alta por las funciones que debe realizar al momento de ejecución de las redes neuronales, para el procesamiento de información utilizando operaciones aritméticas y lógicas, también para controlar el funcionamiento de todo el hardware de la computadora, las operaciones individuales pueden tomar desde varias fracciones de una medida hasta decenas de medidas. Medido en megahercios o gigahercios cuanto mayor sea la velocidad del reloj, más rápido procesará la CPU la información entrante y es lo que se necesita para el funcionamiento de esta etapa.



Cabe recalcar que la elección de un procesador siempre va depender de cómo y para qué van a utilizarse. A continuación, en la tabla 21 se describe los procesadores de gama alta más conocidos y utilizados en el campo de la informática en la actualidad con sus respectivas características enfocando en las principales y necesarias para el funcionamiento del prototipo del sistema electrónico.

Haciendo una comparación entre estas dos compañías que tienen un gran nombramiento con gamas altas se puede observar en la tabla 21 que no hay mucha diferencia entre ellas, porque hace poco AMD ha trabajado en características que la ubicaba por debajo de Intel y de esa manera ha llegado casi a estar en el mismo nivel, pero la diferencia entre estas son las aplicaciones a las que se orienta cada una de ellas, por lo cual se selecciona a Intel porque está acorde a las necesidades

y especificaciones de toda la implementación del prototipo tanto software como hardware.

Tabla 21: Comparación de procesadores de gama alta.





Elaborado por: Fredy Llambo basado en a [72], [73].

Procesador	AMD	INTEL
Descripción gráfica		
Consumo de energía	Alto	Bajo
Precio	Bajo	Alto
Refrigeración	Picos altos y bajos	Larga duración
Velocidad	4.6 Ghz	5Ghz
Núcleos/hilos	16/32	8/16
Overclocking	Si	No
TDP	65W	127W
Caché	L3: 32MB	12 MB Intel® Smart Cache
PCI Express	PCIe 4.0 x16	PCIe 3.0 x16
Memoria	DDR4	DDR4-2933
Temperatura máxima	95°C	100°C
Aplicaciones	Gaming	Ingeniería

Hay una gran variedad de procesadores en Intel pero sus características dependen de la generación que se describa, en la tabla 22 se detalla las características en una generación específica para realizar una comparación no desequilibrada para que no haya diferencia en el desempeño, optimización o tecnología que posea y con respecto a esas características se efectuara una selección adecuada para el sistema electrónico.

Tabla 22: Comparación de procesadores Intel Core.

Elaborado por: Fredy Llambo en base a [72].

Procesador	I3	I5	I7	I9
Descripción gráfica				
Número del procesador	i3-8130U	i5-8250U	i7-8550U	i9-8950HK
Núcleos/hilos	2/4	4/8	4/8	6/12
Frecuencia básica del procesador	2,20 GHz	1,60 GHz	1,80 GHz	2,90 GHz
Frecuencia turbo máxima	3,40 GHz	3,40 GHz	4,00 GHz	4,80 GHz
TDP	15W	15 W	15 W	45W
Caché	4 MB Intel® Smart Cache	6 MB Intel® Smart Cache	8 MB Intel® Smart Cache	12 MB Intel® Smart Cache
Gráficos de procesador	Gráficos Intel® Iris® Plus 620	Gráficos HD Intel® 620	Gráficos Intel® UHD 620	Gráficos HD Intel® 630
Memoria	32 GB	32 GB	32 GB	64 GB

Tipo de memoria	DDR4-2400	DDR4-2400	DDR4-2400	DDR4-2666
Costo	\$ 250	\$ 320	\$ 480	\$690

Para la selección del procesador que cumpla con las especificaciones técnicas para el desarrollo del sistema, se realizó una comparación entre los procesadores descritos en la tabla 22, se puede observar mediante las características que los procesadores son de mucha utilidad para el procesamiento de información utilizando operaciones aritméticas y lógicas, pero se debe tomar en cuenta parámetros importantes para el desarrollo y necesidad del sistema los cuales se describe a continuación:

- Núcleos e hilos por el funcionamiento de varios programas simultáneamente sin ningún problema ni bug.
- Frecuencia turbo máxima debido a la velocidad de gestión de los programas en funcionamiento.
- TDP importante debido al disipador incluido en la laptop.
- Cache para su rápido acceso a datos almacenado.

Al tomar como referencia los parámetros descritos el procesador Intel Core I3 (I3-8130U) se puede considerar como buena opción porque la frecuencia turbo máxima es normal para el gestionamiento de los programas, el TDP es el indicado para el disipador que posee la laptop y el costo es accesible para el investigador, pero tiene una gran desventaja en relación a la cache y núcleos por qué se va a presentarse bugs en la ejecución del software.

El procesador Intel Core I5 (I5-8250U) es mejor opción que el procesador Intel Core I3 (I3-8130U) por el número de núcleos que posee es específica para el software a utilizar, la frecuencia es normal el gestionamiento de los programas, el TDP es el indicado para el disipador que posee la laptop y el costo es accesible para el investigador, pero tiene una gran desventaja en relación a la cache porque se va a presentarse bugs en la ejecución del software.

Se acopla bien al a los parámetros de la implementación del sistema el procesador Intel Core I9 (I9-8950HK), por el número de núcleos que posee es específica para el software a utilizar, la frecuencia es perfecta el gestionamiento de los programas

y el cache supera las especificaciones para el sistema, pero tiene desventajas el TDP no es el indicado para el disipador que posee la laptop y el costo es alto para el investigador.

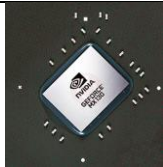
Y por último está el procesador Intel Core I7 (I7-8550U) que cumple con todos los parámetros necesarios para esta implementación del sistema, el número de núcleos es correcto para para el software a utilizar, la frecuencia es perfecta porque el sistema requiere que la velocidad de la CPU en el procesamiento sea rápida, el TDP es el indicado para el disipador que posee la laptop y el cache es recomendable para el software, y el costo es accesible para el desarrollo del sistema.

3.2.2.7 Tarjeta de video

Para el análisis de video también es primordial la tarjeta de video ya sea de media o alta gama para el análisis de video mediante los modelos preentrenados de Caffe Deep Learning. En la tabla 23 se describe las características de la tarjeta de video incorporada en la laptop y que ayuda de manera eficiente con el software implementado.

Tabla 23: Características de Nvidia GeForce MX130.

Elaborado por: Fredy Llambo en base a [74].

Tarjeta de video	NVIDIA® GeForce® MX130
Descripción gráfica	
Arquitectura	Maxwell
Motor GPU	Hasta 2.5x
Tipo de memoria	DDR3, GDDR5
Soporte Optimus	si

GPU Boost	2.0
Soporte GameWorks	Si
DirectX	12 API
OpenGL	4.5
Cuda	Si
Costo	\$78

3.2.2.8 Lenguaje de programación

Los lenguajes de programación que se empleara en el transcurso de la implementación son los descritos en la tabla 24, se hace énfasis por ser la herramienta primordial para crear varias instrucciones que permite acciones consecutivas para el control físico y lógico del sistema.

Tabla 24: Características de los lenguaje de programación.

Elaborado por: Fredy Llambo en base a [20], [75].

Lenguaje de Programación	C++	Python
Origen	Años 80 por Bjarne Stroustrup	Años 80 por Guido Van Rossum
Significado	Incremento de C	Lenguaje interpretado
Características	<ul style="list-style-type: none"> • Compatibilidad con diferentes bibliotecas • Orientado a objetos • Compilación de bajo nivel para ejecución • Didáctico 	<ul style="list-style-type: none"> • Código definido como scripts. • No importa el sistema operativo. • Orientado a objetos. • Escritura dinámica.

Ventajas	<ul style="list-style-type: none"> • Alto rendimiento • Multiplataforma • Extendido • Lenguaje actualizado 	<ul style="list-style-type: none"> • Rápido y simplificado • Interpretable • Productividad • Legible y organizado • Portable
Aplicaciones y usos	<ul style="list-style-type: none"> • Base de datos • Navegadores web • Videojuegos • Compiladores • Sistemas Operativos 	<ul style="list-style-type: none"> • IA • Análisis de datos • Desarrollo web

3.2.2.9 Software IDE Arduino

IDE arduino proveniente del inglés IDE (Integrated development environment) entorno de desarrollo integrado, es una plataforma que permite escribir código de manera fácil y permite subir a cualquier placa arduino. El software es de código abierto que sirve para la creación fácil y rápida de prototipos electrónicos, es decir basada en software y hardware fácil de utilizar, se hace uso de este IDE para la programación de nuestro microcontrolador.[76], [77]

3.2.2.10 Servidor XAMPP

Para la base de datos se requiere de un servidor local y el software acorde es XAMPP, ya que es un software gratuito y sobre todo de código abierto. XAMPP es acrónimo para: X (L (Linux), M(Mac) o W(Windows)), Apache, MySQL, PHP y Perl. [78]

Las funciones principales de XAMPP son las que se describe a continuación: [78]

- X la compatibilidad con sistemas operativos como: Linux, Mac o Windows.
- Apache es servidor web.
- MySQL es el gestor de base de datos.
- PHP y Perl son lenguajes de programación.

3.2.3 Diseño del prototipo

El prototipo del sistema electrónico está basado en etapas y subsistemas complementarios que requieren de algunos elementos, con su respectivo análisis y procesos para la implementación, los cuales se describen a detalle a continuación en la figura 31. El esquema esta realizado para el movimiento del riel electrónico, los servomotores son los que ayuda para el enfoque del marchista mediante la manipulación de los potenciómetros, el sensor ultrasónico es para la detección del marchista e inicie el seguimiento, el motor a pasos servirá para el desplazamiento del soporte para la cámara con ayuda de la banda y las poleas.

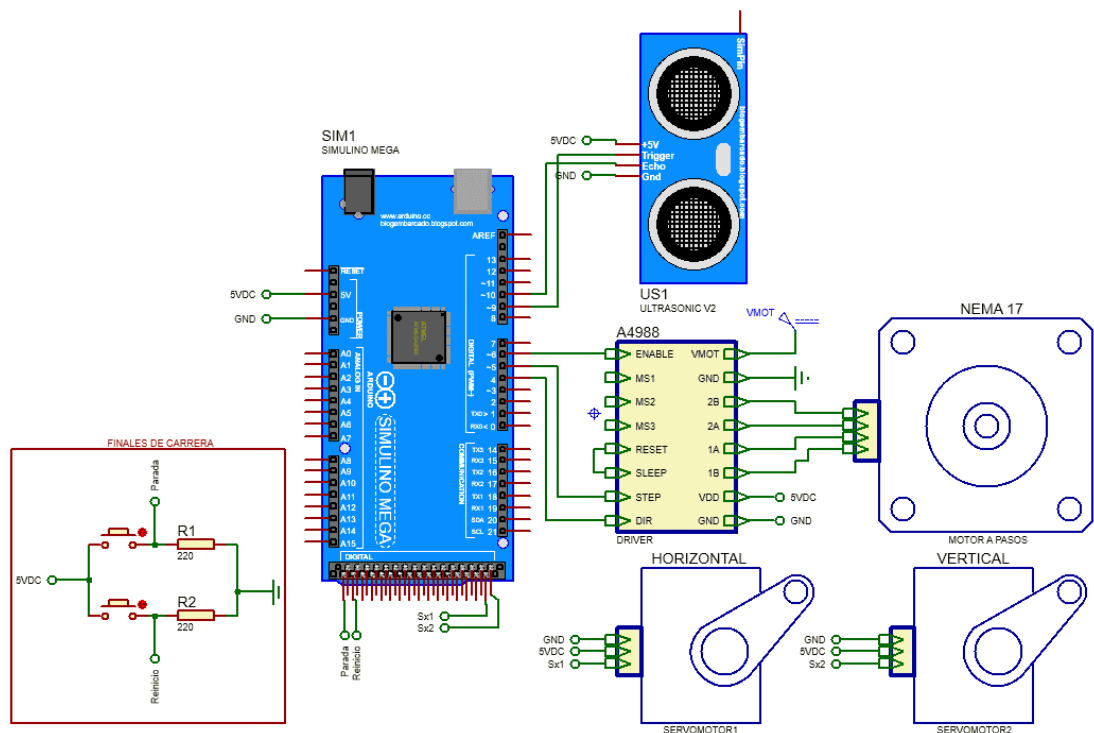


Figura 31: Esquema de conexión del prototipo.

Elaborado por: Fredy Llambo en Proteus.

3.2.3.1 Implementación de Fuente de alimentación

Para el funcionamiento del motor, del servomotor y los finales de carrera se utiliza una fuente de 0.5A con salidas de voltaje de 5, 10 y 15 V. El diseño de la fuente se puede observar en la figura 32.

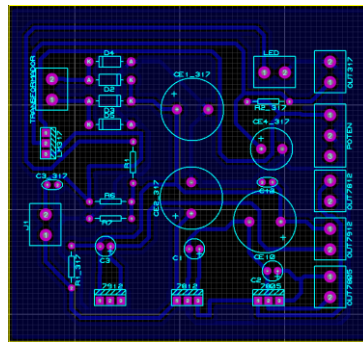
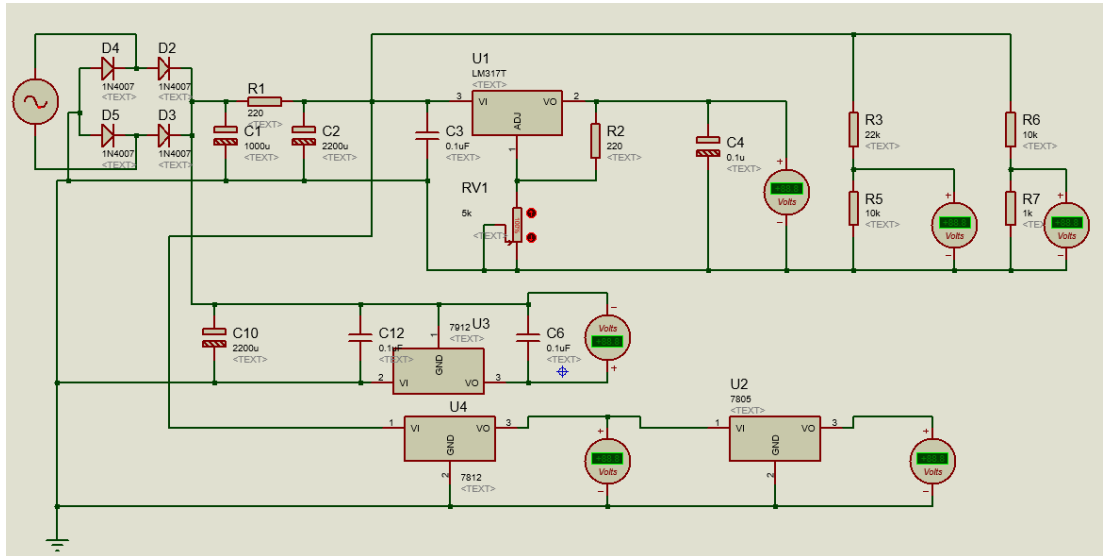


Figura 32: Diseño de fuente de alimentación.

Elaborado por: Fredy Llambo.

Para que el circuito no sea de gran tamaño se procede a la impresión a los dos lados. En la figura 33 se puede observar el armado de la fuente que servirá para todo el sistema electrónico.



Figura 33: Fuente de alimentación.

Elaborado por: Fredy Llambo.

3.2.3.2 Estructura del prototipo

Para la elaboración de la estructura primeramente se construyó las patas laterales y del medio o centro para que se puede llevar fácilmente a cualquier lugar, en la figura 34 se puede visualizar el armado de la base con la pata lateral y el soporte con huecos para que se puede colocar las varillas, en la pata lateral se realiza un corte para el travesaño, la construcción es a escuadra y nivel para que se quede fijo al sentarlo y no tienda a caerse.



Figura 34: Pata lateral.

Elaborado por: Fredy Llambo.

En la figura 35 se observa de la misma manera el armado de la base con la pata del centro que lleva cortes a ciertas distancias para la colocación de los travesaños, se realizó unos agujeros en la parte de arriba de la pata para colocar las varilla que tiene una varilla pequeña soldada en la mitad de la varilla grande con la finalidad de que no tiendan a colgarse y dificulte el desplazamiento del soporte de la cámara.



Figura 35: Pata del medio y soldadura de varilla.

Elaborado por: Fredy Llambo.

La estructura del prototipo está hecha a base de madera, las patas, los travesaños y el soporte de la cámara, para el desplazamiento del soporte de la cámara con ayuda de las poleas de soporte y los rieles son varillas de acero inoxidable, toda la construcción está a nivel para evitar cualquier accidente, en la figura 36 se muestra el prototipo de la estructura finalizada.



Figura 36: Estructura finalizada.

Elaborado por: Fredy Llambo

Para la implementación de la estructura se realizó un diseño con todos los detalles necesario para que no tenga inestabilidad, ni desnivel para la recolección de imágenes, en el ANEXO A se detalla todas las dimensiones para el prototipo. En la figura 37 se presenta el diseño de las poleas y la banda que se necesita para el desplazamiento del soporte con polea para la cámara.

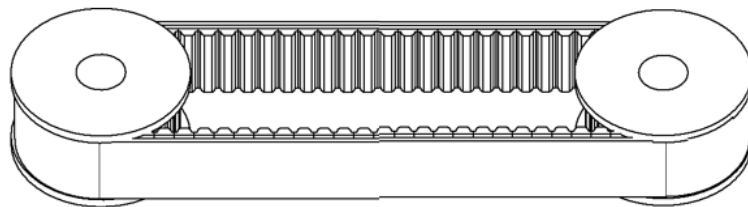


Figura 37: Esquema de polea y banda para desplazamiento del soporte de la cámara.

Elaborado por: Fredy Llambo Autodesk Inventor Profesional.

En la figura 38 se describe gráficamente el soporte de la cámara con poleas, las poleas se colocan para el desplazamiento en la varilla, estas poleas están construidas en torno de acuerdo al diseño elaborado en autodesk inventor, este diseño cumple con las especificaciones exactas para el desplazamiento correcto en la varilla, en el ANEXO A se describe a detalle las medidas para la construcción de las poleas con rodamiento, estas son de acero y para que giren libremente se las

colocó rodamientos en el centro de las poleas, tiene tuercas libres para el sostén de los extremos de la banda y se pueda desarmar para poder transportar a lugar que se desee.

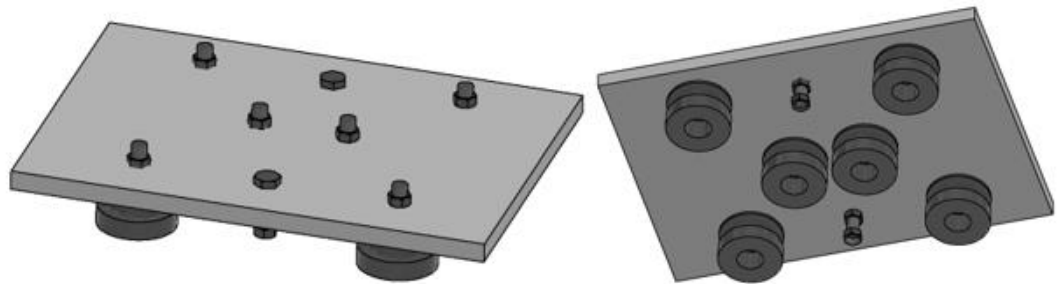


Figura 38: Soporte de cámara con polea.

Elaborado por: Fredy Llambo en Autodesk Inventor Profesional.

3.2.3.3 Sistema de procesamiento de video

Para el procesamiento de video el sistema encargado de realizar esta función es una maquina electrónica portable capaz de la recolección y procesamiento de video, mediante la detección del practicante de marcha atlética el sistema inicia la recolección de imágenes por medio de la configuración de algoritmos y parámetros permitiendo el análisis y almacenamiento en una base de datos de la información recolectada.

Especificaciones de la maquina electrónica portable

Al hablar de una maquina electrónica portable hace referencia a una laptop, en la tabla 25 se describe de manera detallada las especificaciones principales de la laptop que se utilizara en la implementación y son de vital importancia para el desarrollo del software debido a los requerimientos que se necesita para las bibliotecas de la visión artificial, de no cumplir con esos requerimientos el software tendrá muchos errores al compilar cualquier programa y en los peores de los casos puede dañar la máquina.

Tabla 25: Especificaciones generales de laptop.

Elaborado por: Fredy Llambo

Laptop HP	
Sistema Operativo	Windows 10 Home Single Language 64 bits
Microprocesador	Intel® Core™ i7-8550U CPU @ 1.8GHz
Memoria del sistema	12 GB
Ranura de memoria 1	8GM Samsung 2400MHz
Ranura de memoria 2	4GM Samsung 2400MHz
Placa del sistema	84A7 80.43
BIOS del sistema	F.24
Fuente de alimentación	Alimentación por batería
Unidades de disco duro	2
Unidad 1	GIGABYTE GP-GSM2NE8256GNTD – 238,47 GB
Unidad 2	HGST HTS541010B7E610 – 931,51 GB
Dispositivo grafico 1	NVIDIA GeForce MX130, resolución 1920x1080, frecuencia de actualización 59, versión 26.21.14.4575
Dispositivo grafico 1	Intel® UHD Graphics 620, resolución 1920x1080, frecuencia de actualización 60, versión 26.20.100.7986
Memoria de gráficos	2.132 MB
Estado Aero	Activado
Costo	\$1300

Implementación del sistema de biomecánica

Se procede a realizar la implementación del sistema electrónico de corrección del gesto deportivo en base a la biomecánica en deportistas de marcha atlética, siguiendo la estructura del diagrama que se plantea en la figura 39.

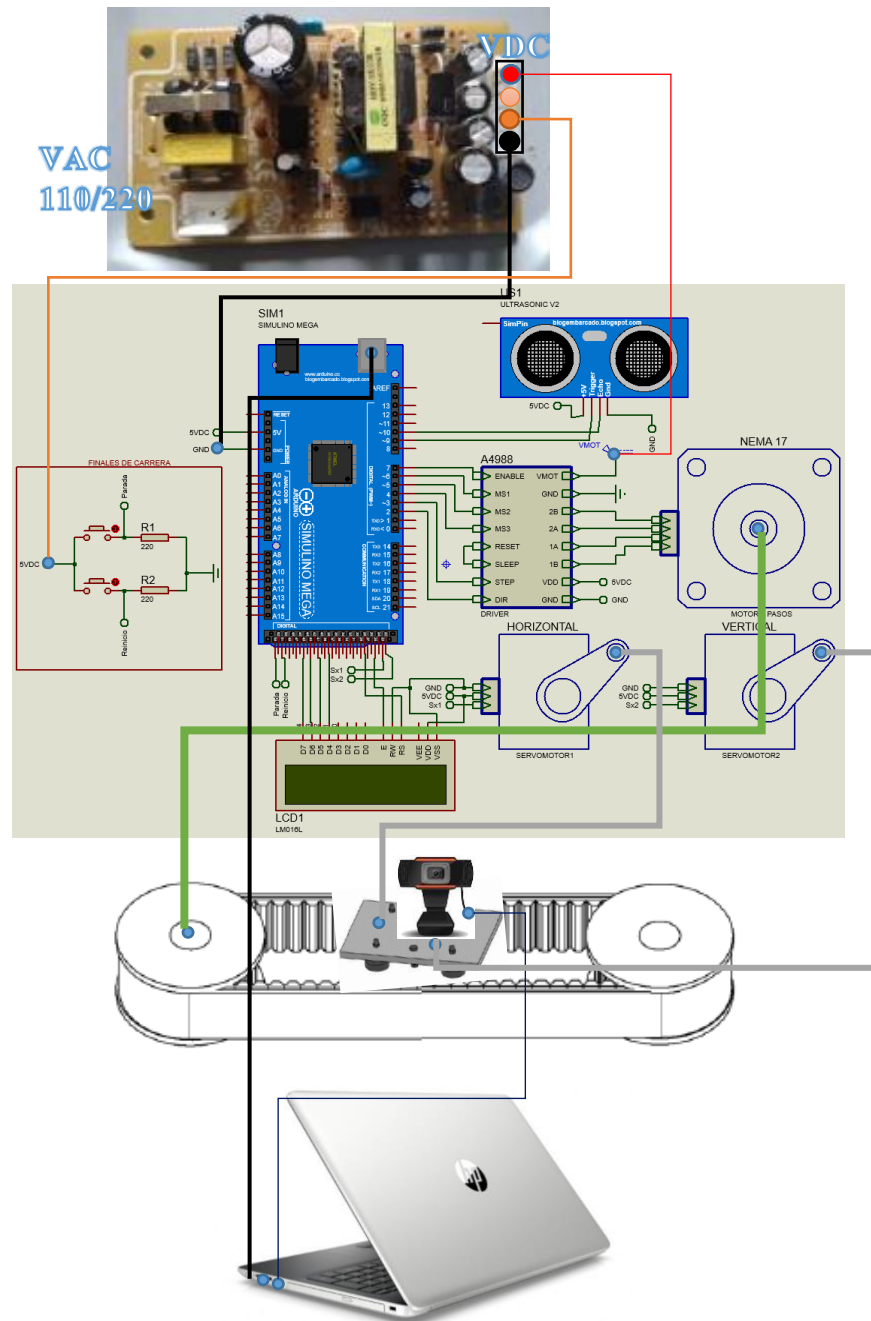


Figura 39: Diagrama de implementación del sistema electrónico de biomecánica de marcha atlética.

Elaborado por: Fredy Llambo.

Instalación de entorno de programación

Para la programación del software a implementarse es necesario la instalación de un editor de código fuente, para el desarrollo del proyecto se seleccionó sublime text porque es portable y está escrito en C++ y Python, en la figura 40 se muestra las versiones disponibles para descargar e instalar de la página oficial para darle uso, el entorno de programación brinda la posibilidad de descargar un ejecutable o la versión portable para simplemente ejecutar la aplicación y usarla, por lo tanto se escogió como opción para el proyecto.



Figura 40: Descarga de entorno de programación.

Elaborado por: Fredy Llambo en base a [79].

Instalación del lenguaje de programación

Python se utilizará como el lenguaje de programación porque es fácil de usar en relación a otros lenguajes, existe dos maneras de instalar en el sistema operativo de Windows en la figura 41 muestra que es mediante la tienda de Microsoft que es gratuita y se puede verificar ingresando al símbolo del sistema e ingresando la palabra python para la redirección a la página de instalación de la tienda y el otro método es descargando de la página oficial de python escogiendo la versión que desee, el descargar de la tienda de Microsoft tiene una desventaja a descargar de la página oficial que no se tiene acceso a las carpetas de instalación, el cual si se requiere.

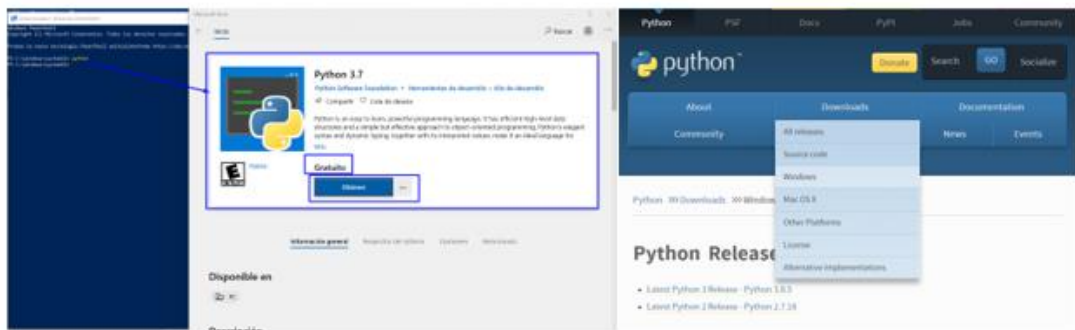


Figura 41: Métodos de instalación de python.

Elaborado por: Fredy Llambo en base a [80].

Una vez descargado la aplicación se procedió con la instalación en la figura 42 se describe gráficamente los pasos que se debe seguir para la correcta instalación, se escoge la opción de instalación marcada por que permite elegir las características a instalarse, se procede a seleccionar en añadir PATH para evitar editar las variables de entorno del sistema que permite a otros programas saber las rutas de instalación.

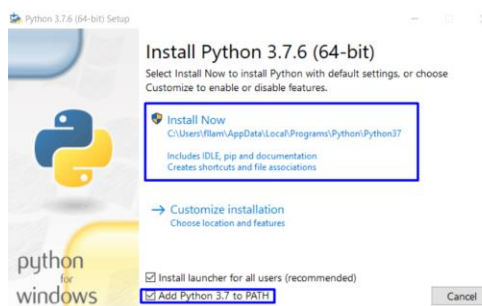


Figura 42: Opción de Instalación.

Elaborado por: Fredy Llambo.

En la figura 43 se puede observar las características que se va instalar, la documentación es para ayuda al programador para mejor entendimiento del lenguaje, pip es el instalador de todos los paquetes que se necesite, tcl/tk and IDLE hace referencia a la instalación de librerías para la creación de interfaz gráfica y el entorno de desarrollo, py launcher es la creación de una variable global para la compilación de códigos, de todos estos puedes seleccionar el que mejor te parezca y continuar con la instalación.

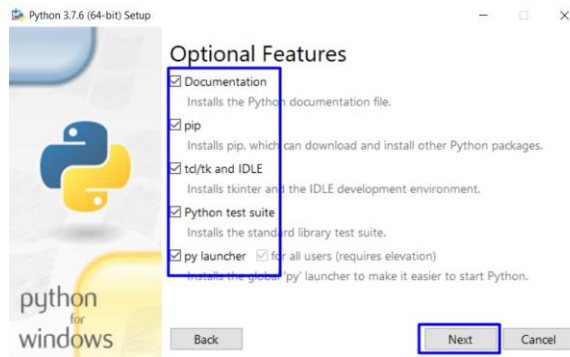


Figura 43: Características opcionales de python.

Elaborado por: Fredy Llambo.

Las opciones avanzadas de igual manera deben entender para que sirve de manera de no instalar erróneamente el lenguaje, en la figura 44 se observa las opciones según las necesidades del programador, la pestaña de asociar archivos con python y requiere de py launcher es para que al abrir el archivo *.py siempre se abra con un entorno de programación detectado automáticamente por la laptop, la creación de atajos para la instalación de aplicaciones, añadir python al entorno de variables, descarga de depuración de símbolos y descarga de depurador de binarios, los nombres son claros y no hay necesidad de explicar para que sirve cada uno.

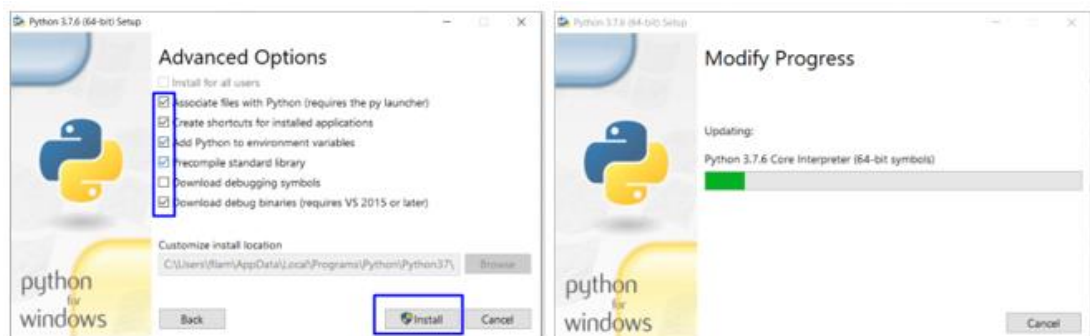


Figura 44: Opciones avanzadas de instalación.

Elaborado por: Fredy Llambo.

Para verificar la correcta instalacion de python se abre el simbolo de sistema y se ejecuta la palabra python o py, como se visualiza en la figura 45 para ver el funcionamiento del lenguaje se puede crear el tipico “Hola Mundo” que simplemente se lo crea con la linea de codigo `<<print(“Hola Mundo”)>>` y para ver la version instalada de python la linea de codigo es `<<print(sys.version) >>`

importando la librería sys, de esta manera se debe verificar la instalación y funcionamiento del lenguaje.

```
Anaconda Prompt (Anaconda3) - py
(base) C:\Users\Fredy>py
Python 3.7.4 (default, Aug 9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hola MUndo")
Hola MUndo
>>> import sys
>>> print(sys.version)
3.7.4 (default, Aug 9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)]
>>>
```

Figura 45: Código de ejemplo.

Elaborado por: Fredy Llambo.

Para la instalación de librerías que sea necesario para la implementación se hace uso del comando pip install *librería*, si desconoce las librerías que tiene python se puede guiar de la página oficial de pypi.org el cual describe la funcionalidad de las librerías.

Instalación de librería multiplataforma de visión artificial

Para este proyecto la visión artificial cumple un papel importante en el análisis del video recolectado, con la ayuda de la indagación de tesis, libros y revistas científicas la librería que cumple esta función es OpenCV. Para la instalación existe 2 métodos, el primer método es descargar de SOURCEFORGE la versión 4.2.0. como se observa en la figura 46.

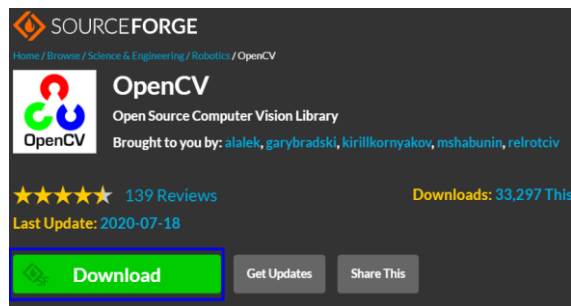


Figura 46: Descarga de OpenCV.

Elaborado por: Fredy Llambo en base a [81].

Ya descargado la biblioteca se procedió a ejecutar la descarga con permisos de administrador, se extrajo los archivos en el disco local del sistema operativo en nuestro caso Disco Local C como muestra la figura 47 y se esperó el tiempo de extracción, el tiempo va depender mucho de su sistema operativo.

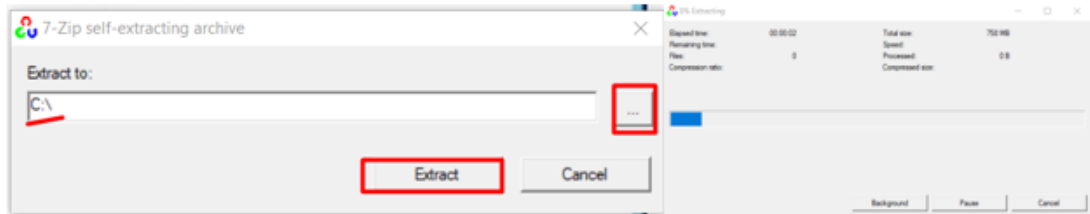


Figura 47: Extracción de archivos OpenCV.

Elaborado por: Fredy Llambo.

Al finalizar la extracción se observó los archivos de OpenCV descargados en la ruta especificada en la figura 47, en el siguiente directorio C:\opencv\build\python\cv2\python-3.8 se copió el archivo que se encuentra en la carpeta de python tomando a consideración la versión instalada como muestra la figura 48.

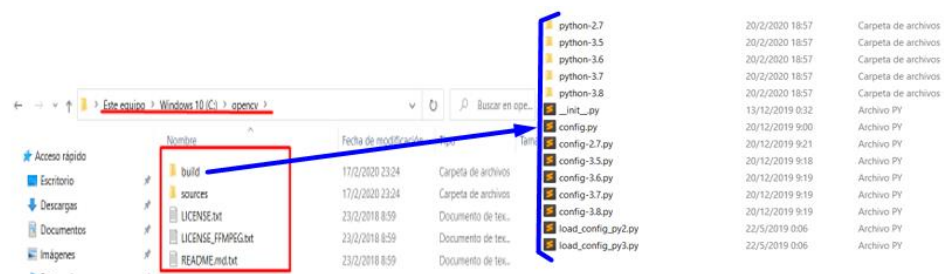


Figura 48: Extracción y OpenCV en versiones de python.

Elaborado por: Fredy Llambo.

Dentro de las carpetas esta un archivo con el siguiente nombre cv2.cp36-win_amd64.pyd el numero 36 va cambiar porque hace referencia a la versión de python 36 (python3.6). Una vez que se copió el archivo se pegó en la carpeta C:\Program Files\Python38\Lib\site-packages donde se ubica todas las librerías instaladas por defecto, se cambió de nombre al archivo a cv2 como muestra la figura 49, para poder importar en la programación la librería de la siguiente manera <<import cv2>>.

__pycache__	21/2/2020 14:05	Carpeta de archivos
pip	21/2/2020 14:05	Carpeta de archivos
pip-19.2.3.dist-info	21/2/2020 14:05	Carpeta de archivos
pkg_resources	21/2/2020 14:05	Carpeta de archivos
setuptools	21/2/2020 14:05	Carpeta de archivos
setuptools-41.2.0.dist-info	21/2/2020 14:05	Carpeta de archivos
cv2	20/12/2019 9:20	Python Extension ...
easy_install	21/2/2020 14:05	Archivo PY
README	18/12/2019 23:26	Documento de tex...

Figura 49: Finalización de instalación de librería OpenCV.

Elaborado por: Fredy Llambo

El otro método es mediante comandos en el símbolo del sistema con ayuda del comando `pip install *libreria*` como se describe a continuación y se muestra en la figura 50 se puede instalar la librería, en caso de algún error por administrador puede intentar el primer método.

```
>> pip install opencv-python
```

opencv-python if you need only main modules

```
>> pip install opencv-contrib-python
```

En el caso de requerir otra versión de cualquier librería deben seguir el siguiente formato `<<pip install librería==version>>`

```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> cd ..
PS C:\Windows> cd ..
PS C:\> cd ..\Program Files\
PS C:\Program Files> cd ..\Python37\
PS C:\Program Files\Python37> pip install opencv-contrib-python
Requirement already satisfied: opencv-contrib-python in c:\program files\python37\lib\site-packages (4.2.0.32)
Requirement already satisfied: numpy>=1.14.5 in c:\program files\python37\lib\site-packages (from opencv-contrib-python)
(1.18.1)
PS C:\Program Files\Python37> _
```

Figura 50: Instalación de OpenCV mediante consola.

Elaborado por: Fredy Llambo

Para verificar la correcta instalación de la librería se ejecuta las siguientes líneas de código. Como se observa en la figura 51 se debe importar la librería de OpenCV (`import cv2`) y para verificar la versión (`print(cv2.__version__)`).

```
Anaconda Prompt (Anaconda3) - python
(base) C:\Users\Fredy>python
Python 3.7.4 (default, Aug 9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import cv2
>>> print(cv2.__version__)
4.3.0
>>> _
```

Figura 51: Verificación de instalación OpenCV.

Elaborado por: Fredy Llambo.

Funcionamiento OpenCV en la adquisición de imágenes.

En la figura 52 se muestra el diagrama de flujo que describe cómo funciona la librería OpenCV en python.

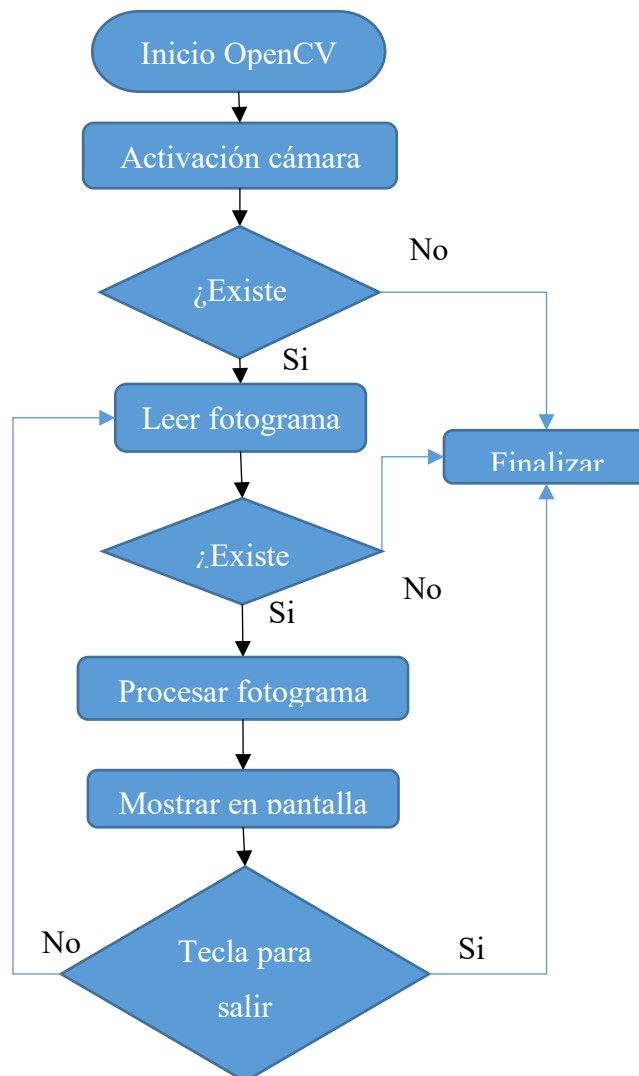


Figura 52: Diagrama de Flujo captura de video OpenCV.

Elaborado por: Fredy Llambo.

Este diagrama de flujo representa el modo de capturar y presentación de video con ayuda de OpenCV, en la figura 52 muestra en el terminador el inicio OpenCV el cual hace referencia a la importación de la librería al editor de código, con la ayuda de esta librería se procede a la activación de la cámara, en caso de no presentar el dispositivo ninguna cámara el proceso finaliza, caso contrario inicia la lectura de los fotogramas, al no detectar los fotogramas finaliza el proceso o inicia el procesamiento de los fotogramas con la finalidad de presentarlas en el monitor una secuencia de imágenes conocida como video, para finalizar este proceso se presiona una tecla de no ser así el programa presenta continuamente la recolección de las imágenes.

En el **ANEXO B** se muestra el código que representa el diagrama de flujo para la captura de video, en la figura 53 se puede observar el resultado de la compilación del código.



Figura 53: Resultado captura de video con OpenCV.

Elaborado por: Fredy Llambo.

Una vez conseguido el funcionamiento de captura de video mediante OpenCV, se debe proseguir a la estimación de la postura del cuerpo humano con ayuda de modelos de aprendizaje automático que son COCO, MPI, BODY_25 y redes neuronales,[4] en la figura 54 se describe mediante un diagrama de flujo los pasos para la creación de la estimación de postura humana y en la figura 55 se muestra de manera gráfica. En el desarrollo de la estimación cabe recalcar que se hace uso del marco de Caffè Deep Learning, para uso del marco se debe especificar la arquitectura de red neuronal y el almacenamiento de pesos de los modelos de

aprendizaje, para el uso de marco de caffe la entrada ya sea imagen o video debe ser convertido en blob de entrada con la finalidad de ser alimentado a la red.

OpenCV tiene formato diferente a Caffe, convertido la imagen se pasa al modelos de prediccion o modelos de aprendizaje automatico de las redes neuronales profundas, la salida del modelo tiene una matriz de cuatro dimensiones, la primera es el ID de la imagen o video, la segunda indica los puntos clave de la imagen es decir crea mapas de concatenados para la imagen o video, la tercera y cuarta son altura y ancho simultaneamente del mapa de salida. [82]

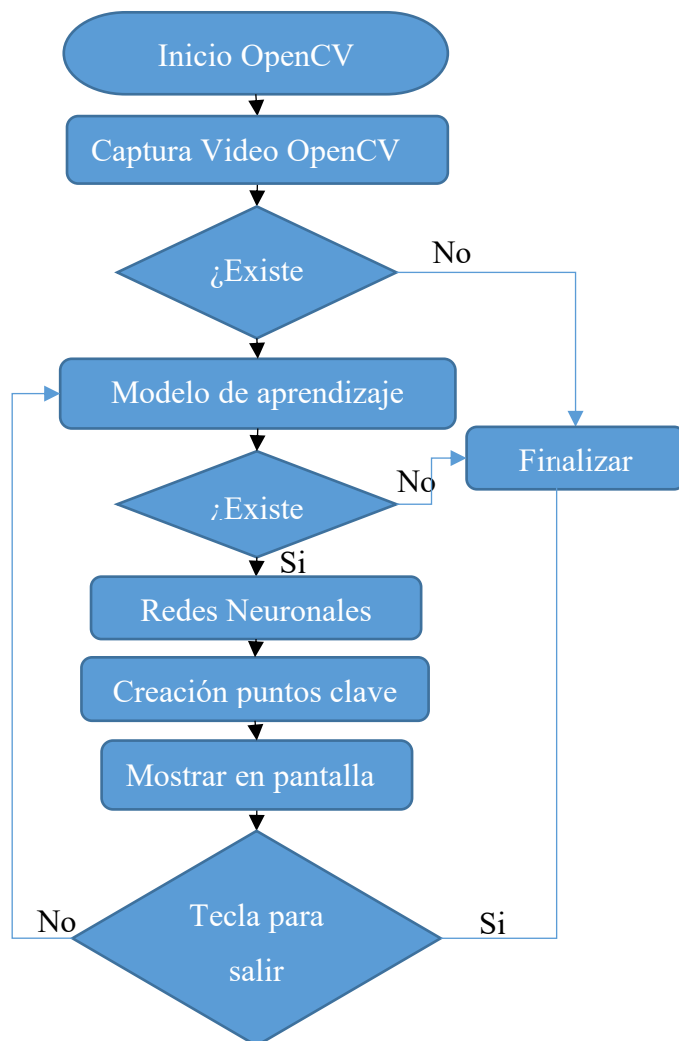


Figura 54: Diagrama de flujo de estimación humana.

Elaborado por: Fredy Llambo

Para la creacion de puntos en el cuerpo humano o llamado estimacion humana se basa en los modelos de aprendizaje automatico el cual esta desarrollado en base a la arquitectura multietapa de CNN, ingresa una imagen al metodo de convolucion en la figura 55.a muestra el resultado de la convolucion de la imagen original y el filtro introducido, el mapa de confianza o mapa de caracteristica son representadas en la figura 55.b que son el conjunto de imágenes convolucionadas, para la asociacion de partes se usan PAFs que son los encargados de la estimacion de poses en base a los modelos que se usan en el procesamiento, en la figura 55.c se presenta el resultado de aplicar las neuronas necesarias para deteccion de puntos, para obtener la capa de asociacion de partes con los puntos clave que se asocia al cuerpo humano como muestra la figura 55.d se realiza un procesamiento continuo y repetitivo con las neuronas necesarias, de esta manera se obtiene los puntos clave en el lugar exacto que se desea la figura 55.e muestra los resultados del procesamiento con las neuronas, para la creacion del esqueleto como se presenta en la figura 55.f se procede a la union de cada punto clave determinado.

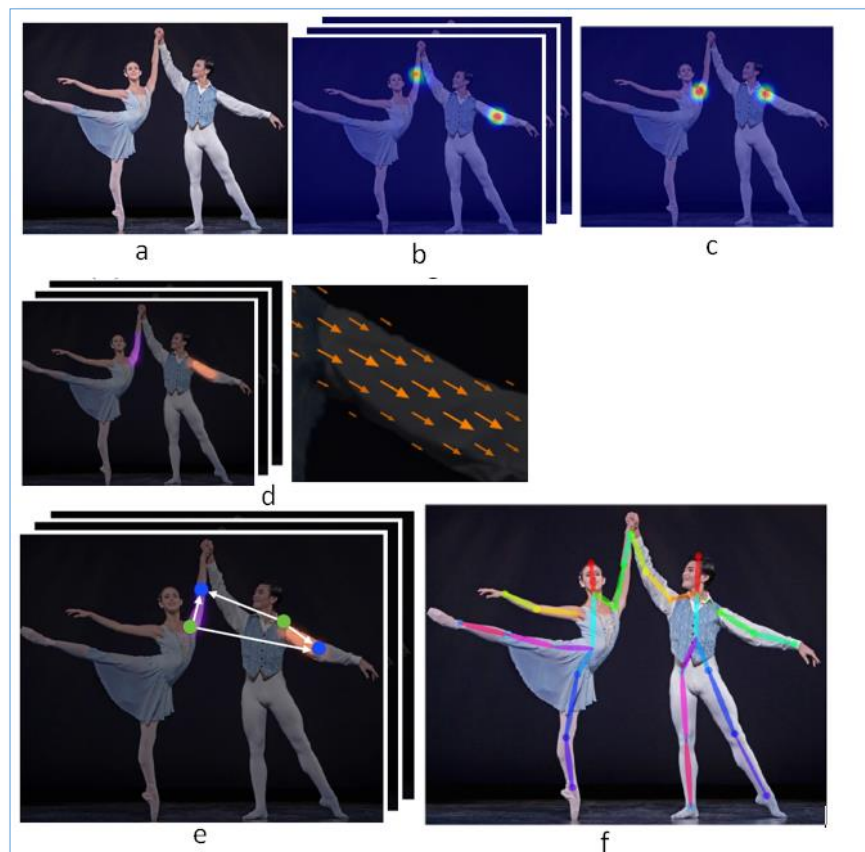


Figura 55: Proceso de estimación humana.

Fuente: [4]

Explicado de forma de sistemas en la figura 56 se explica de manera gráfica el procesamiento de una imagen, donde F es la imagen original de $h \times w$, el cual ingresa al cálculo de la convolución mediante la matriz de kernel logrando así obtener varias imágenes de cada pixel calculado obteniendo de esta manera nuevas imágenes de $h' \times w'$, para que sea procesado por un modelo de aprendizaje y sea el enlace o la unión de todos los puntos procesados y la creación de la estimación humana.

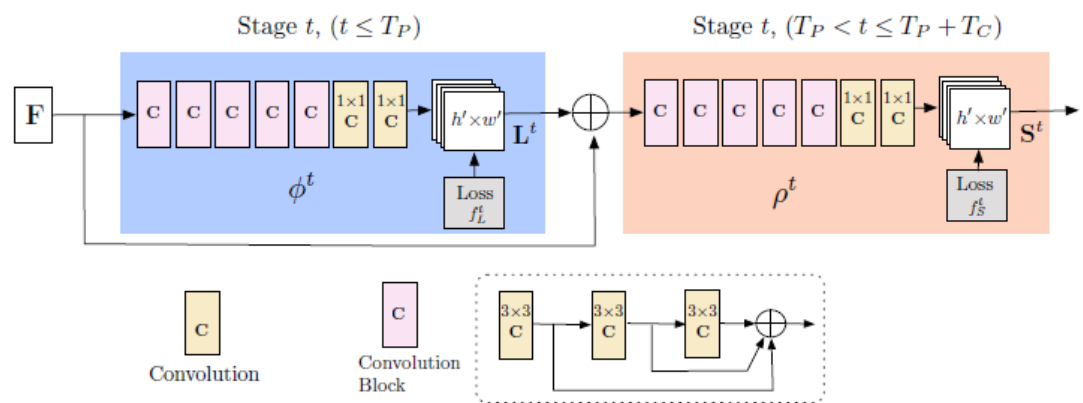


Figura 56: Arquitectura multietapa de CNN.

Fuente: [4]

El procesamiento de imágenes con estas herramientas tiene un alto consumo en video, en la figura 57 se muestra el exceso de uso de la tarjeta de video Intel propia del procesador, se puede verificar de manera breve en el administrador de tareas del sistema operativo y se puede visualizar que incrementa con el tiempo de compilación del software el porcentaje de uso del CPU y la memoria del ordenador, alterando de esta manera el funcionamiento de los ventiladores, que viene a provocar el recalentamiento del procesador, la congelación del ordenador y bugs en el sistema operativo.

En la figura 58 mediante el código de cálculo de fps se puede notar la cantidad baja de fps, indagando información de estimación del cuerpo humano se puede corregir el error, con ayuda de los requerimientos para aprendizaje automático se evita el recalentamiento del procesador de la laptop y daño en la mainboard, por lo que se requiere la instalación de arquitectura unificada de dispositivos de cómputo (CUDA) que es la programación que ayudara con el procesamiento gráfico, para aceleración de aplicaciones y aprovechamiento de las GPU (Unidades de

procesamiento grafico). En el ANEXO C se detalla el codigo utilizado para la estimacion humana.

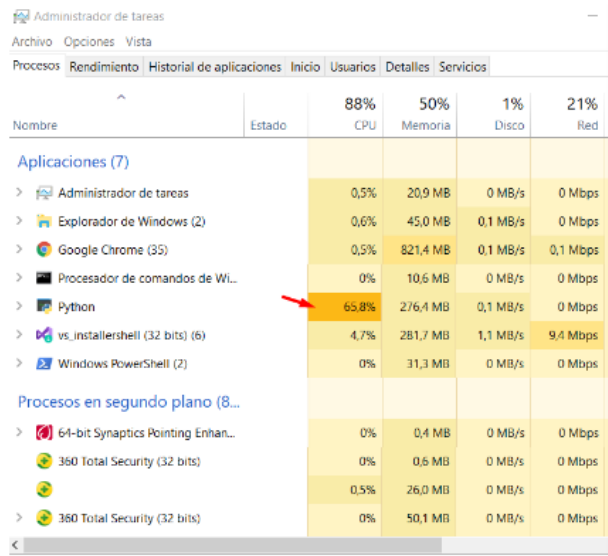


Figura 57: Uso de CPU python y OpenCV.

Elaborado por: Fredy Llambo.

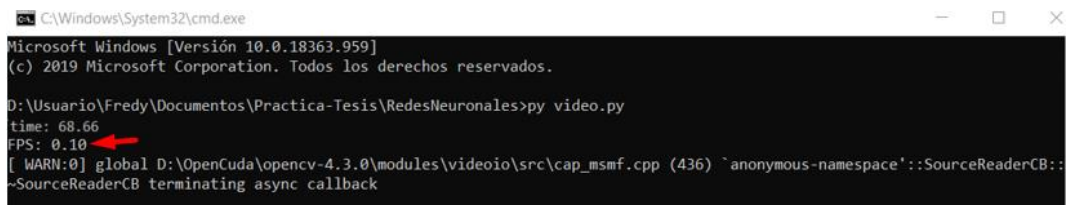


Figura 58: FPS de video capturado.

Elaborado por: Fredy Llambo.

Instalación requerimientos para aprendizaje automático

No basta con la incorporación de la tarjeta de video en la laptop, porque se debe instalar los controladores para su uso, el software tiene errores solo con la tarjeta de video de Intel que viene internamente con el procesador, pero la página oficial para aprendizaje automático especifica los requerimientos que se puede visualizar en la figura 59 y con la ayuda de la tarjeta de video se corrige esos errores y se obtiene resultados efectivos, por lo tanto se debe instalar los controladores de NVIDIA que proporciona la página oficial.

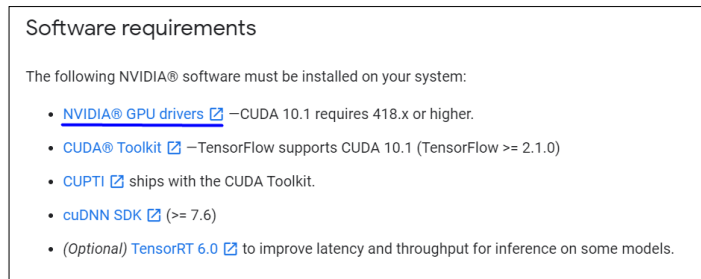


Figura 59: Requerimientos para el aprendizaje automático.

Elaborado por: Fredy Llambo en base a [83]

Para instalar el driver NVIDIA® GPU se descarga de la página oficial de NVIDIA, para elegir el correcto driver se debe ingresar los datos de la tarjeta de video instalada, para saber las especificaciones de la tarjeta en la figura 60 se puede observar los pasos para saber que tarjeta de video está instalada.

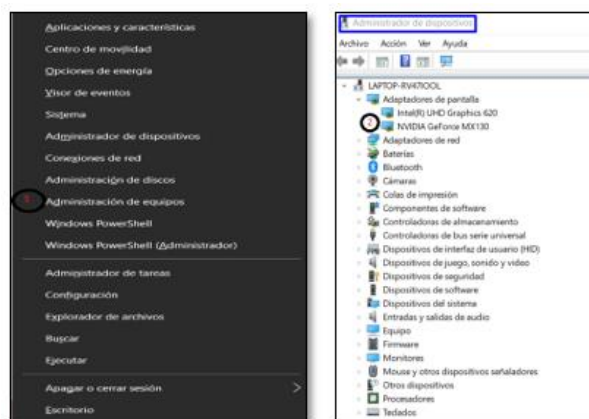


Figura 60: Características de tarjeta de video.

Elaborado por: Fredy Llambo.

Una vez conocido las características de la tarjeta de video se coloca en la página oficial NVIDIA como muestra la figura 61, selecciona el tipo, serie y producto de la tarjeta de video, el sistema operativo de la laptop, el tipo de controlador que necesite o esté disponible y el lenguaje que desee, luego de la descarga del controlador se procede a la instalación, si el driver está instalado el ejecutable simplemente lo actualiza.

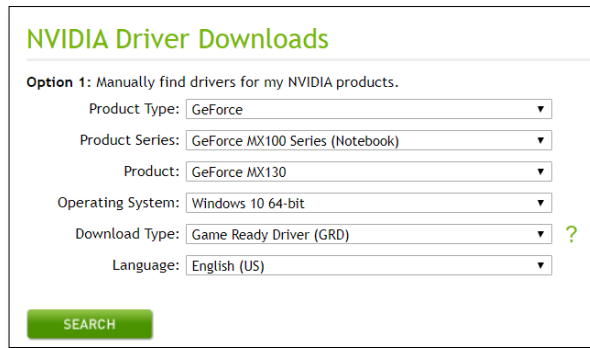


Figura 61: Descarga de driver NVIDIA.

Fuente: [84]

En la figura 62 el controlador comprueba que sea compatible y la tarjeta de video este instalada en la laptop para continuar a los siguientes pasos de instalación, en caso de no tener una tarjeta de video instalada o no sea compatible el controlador se observara una X dando a conocer de esa manera que no se puede continuar con la instalación del controlador.

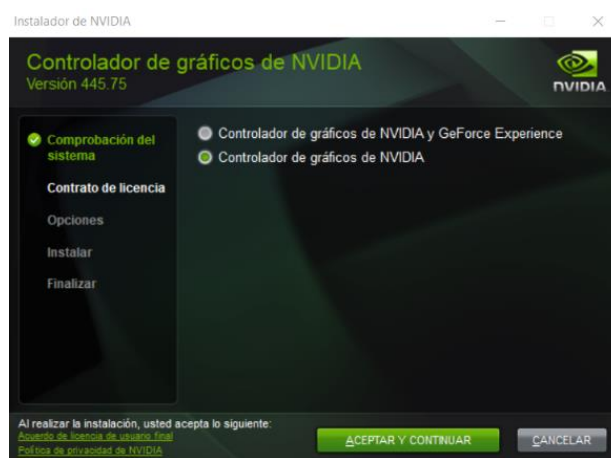


Figura 62: Comprobación del sistema.

En la figura 63 se muestra el siguiente paso que es la opción de instalación, es recomendable la instalación personalizada para seleccionar solo los componentes a instalar, la opción rápida actualiza todos los componentes algunos de ellos no son compatibles con algunos controladores de la maquina por lo que dan errores en el funcionamiento, al seguir las instrucciones el instalador de abra finalizado.



Figura 63: Elección de elementos.

OpenCV en el módulo aprendizaje de redes neuronales enumera las tarjetas compatibles con las que funciona sin ningún error la estimación humana, [85] haciendo referencia a la tarjeta de video NVIDIA y la compatibilidad con el sistema operativo en uso, CUDA es la herramienta que necesita la tarjeta por el procesamiento de video y en la figura 59 se observa que es un requerimiento para el aprendizaje.

Instalación CUDA

CUDA® está desarrollado para NVIDIA como modelo de programación, que ayuda con la aceleración de aplicaciones que tienen problemas con el desarrollo o reproducción de videos y aprovecha de manera eficiente las unidades de procesamiento gráfico(GPU). [74] Y se la puede encontrar en la página oficial de NVIDIA, en la figura 64 se puede observar que para descargar hay que especificar la arquitectura y versión del Sistema Operativo.

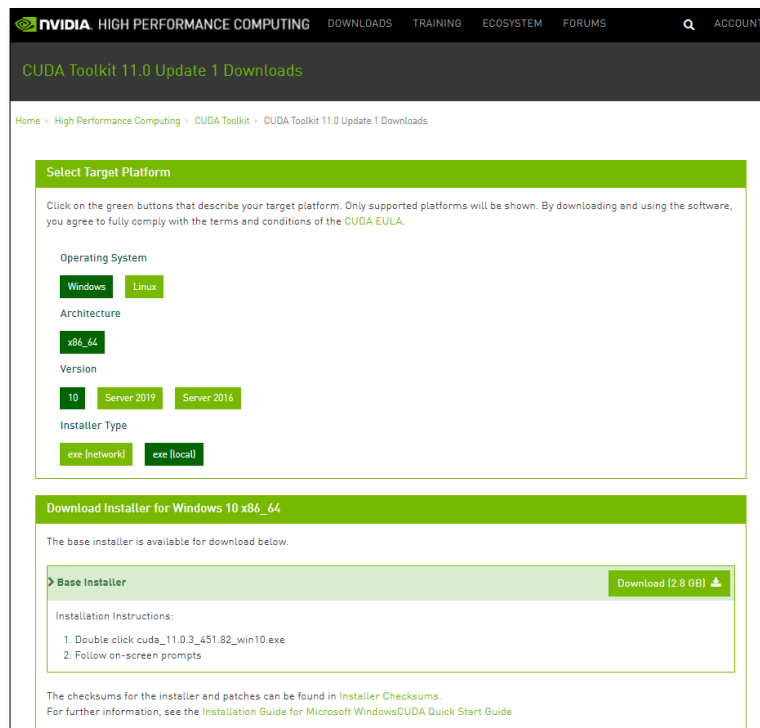


Figura 64: Descarga de CUDA.

Elaborado por: Fredy Llambo en base a [74].

Se procedió a la descarga e instalación, la herramienta comprueba la compatibilidad del sistema operativo instalado en la figura 65.a muestra cómo se debe visualizar si es compatible, las opciones de instalación son recomendado la rápida, pero hay tarjetas que vienen instaladas con versiones actuales y no existe en las páginas oficiales, por eso se recomienda la instalación avanzada la figura 65.b da a conocer las opciones de instalación, para verificar las versiones instaladas.



Figura 65: a) Comprobación de compatibilidad. b) Opciones de Instalación.

Elaborado por: Fredy Llambo.

Las versiones de los componentes vienen instaladas versiones por defecto es decir versiones de fábrica en algunos casos no existe en la página oficial, en la figura 66.a se puede observar las versiones actuales instaladas de fábrica y las versiones actuales que posee el controlador, si posee varios discos duros la mejor opción es instalar en el que tenga más almacenamiento y no en las redireccionadas por defecto como en la figura 66.b.



Figura 66: a) Verificación de versiones. b) Ubicación de instalación.

CUDA verifica la compatibilidad de todas las aplicaciones que requiere para el procesamiento del video, en la figura 67.a se observa que se requiere de Visual Studio, pero en la mayoría de laptops viene instaladas por defecto y en la figura 67.b se verifico que está instalado.

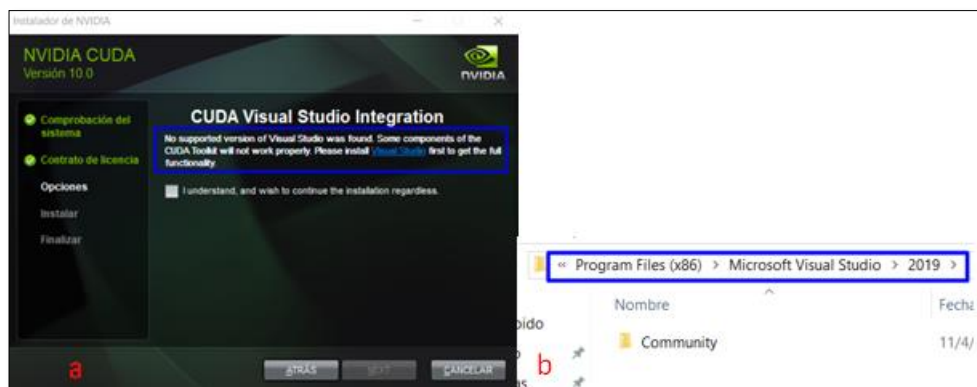


Figura 67: CUDA a) Requerimientos. b) Verificación de requerimiento.

Elaborado por: Fredy Llambo.

En la documentación de la versión de CUDA muestra las versiones compatibles de Visual Studio (VS), [86] en la figura 68 se puede visualizar que la versión VS 2019

no es compatible con la versión de CUDA, sino la versión VS 2017 por lo tanto se procedió a instalarla.

1.1. System Requirements

To use CUDA on your system, you will need the following installed:

- A CUDA-capable GPU
- A supported version of Microsoft Windows
- A supported version of Microsoft Visual Studio
- the NVIDIA CUDA Toolkit (available at <http://developer.nvidia.com/cuda-downloads>)

The next two tables list the currently supported Windows operating systems and compilers.

Table 1. Windows Operating System Support in CUDA 10.0

Operating System	Native x86_64	Cross (x86_32 on x86_64)
Windows 10	YES	YES
Windows 8.1	YES	YES
Windows 7	YES	YES
Windows Server 2016	YES	NO
Windows Server 2012 R2	YES	NO

Table 2. Windows Compiler Support in CUDA 10.0

Compiler	IDE	Native x86_64	Cross (x86_32 on x86_64)
Visual C++ 15.x	Visual Studio 2017 (RTW and all updates)	YES	NO
Visual C++ 14.0	Visual Studio 2015	YES	NO
	Visual Studio Community 2015	YES	NO
Visual C++ 12.0	Visual Studio 2013	YES	YES
Visual C++ 11.0	Visual Studio 2012	YES	YES

Figura 68: Documentación de herramienta CUDA.

Elaborado por: Fredy Llambo.

Verificado la versión que se requiere, se aceptó las condiciones y se prosiguió a instalar la versión 2017 como se observa en la figura 69.a, al finalizar la instalación los componentes de CUDA y VS (2017) y no exista errores, para verificar que todo este correcto se debe observar tal cual muestra la figura 69.b.

Figura 69: a) VS 2017 b) Instalacion Finalizada.

Elaborado por: Fredy Llambo.

La instalación de CUDA fue exitoso para la estimación humana, luego de la programación en el software en desarrollo incluida CUDA como se observa en el

ANEXO D, hay cambio satisfactorio, el rendimiento del procesador es adecuado y se puede verificar con los FPS que se muestra en la figura 70.a y el rendimiento del CPU como se da a conocer en la figura 70.b. En el ANEXOS E se muestra las pruebas realizadas de estimación humana.

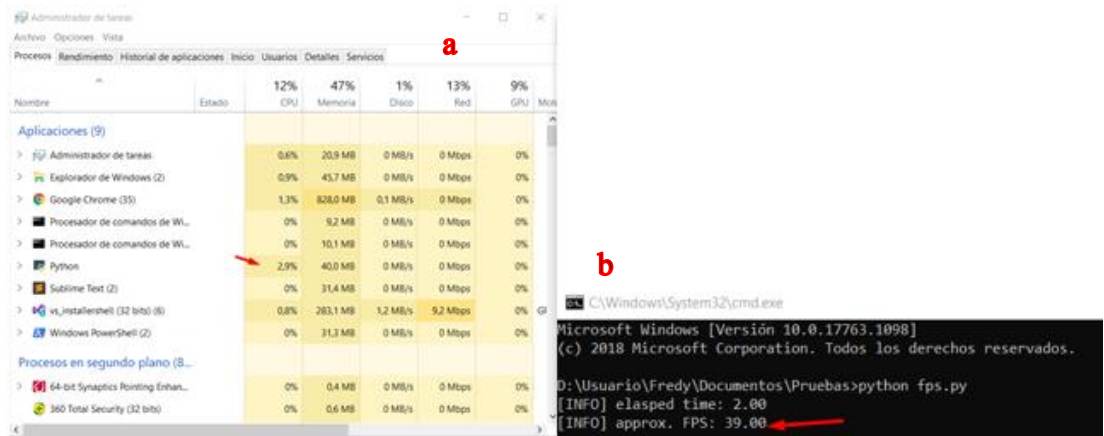


Figura 70: Pruebas satisfactorias de estimación humana.

Elaborado por: Fredy Llambo.

3.2.3.4 Interfaz HMI

La interfaz gráfica es desarrollada para la visualización y control del sistema electrónico del profesional encargado del software, para la creación de la interfaz hombre maquina(HMI), se requiere de una librería que permita la creación de interfaz gráfica de usuario (GUI) en python la más utilizada para el tipo de software que se desarrollara es TKINTER. En la figura 71 se puede apreciar GUI creada en python que consta del contraseña y usuario por seguridad porque se puede perder o eliminar fácilmente la base de datos, evitar el mal uso del sistema electrónico o causar daños severos en los dispositivos de todo el sistema. La seguridad que tiene el sistema es denegar el acceso a personas que no sepan de la manipulación del prototipo, en la figura 72 se puede observar la seguridad.

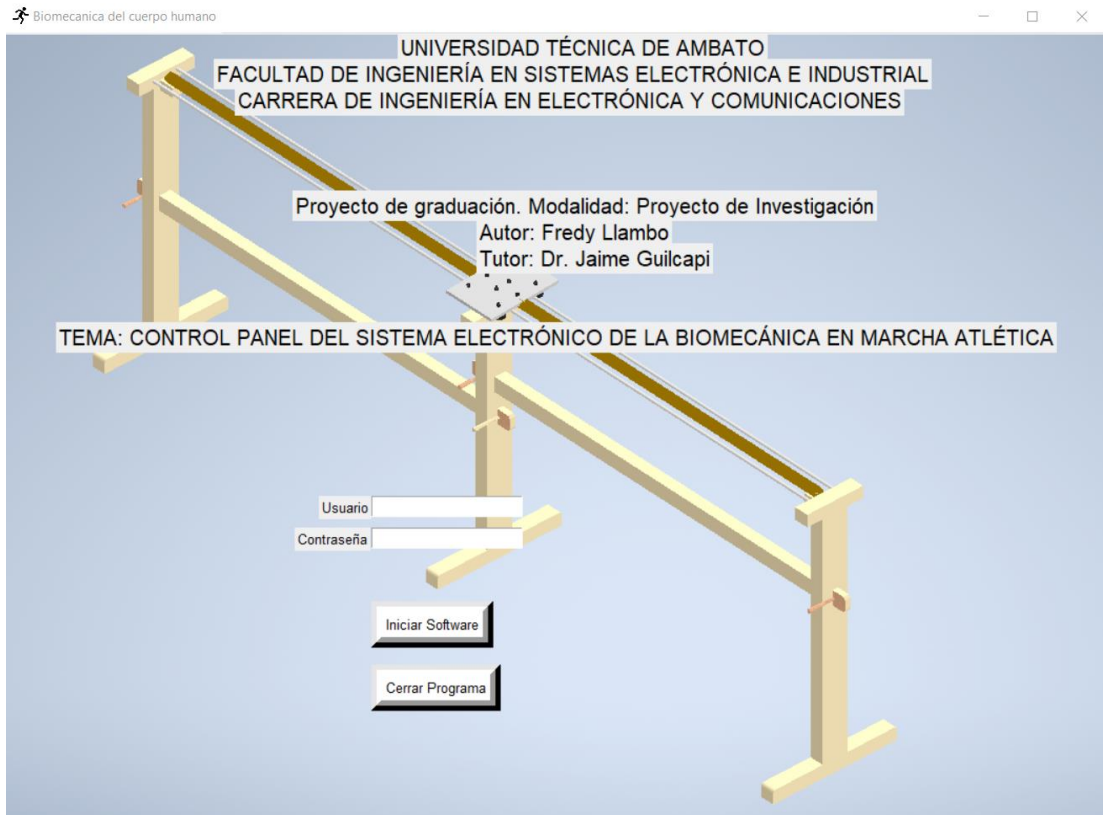


Figura 71: GUI de control de ingreso del sistema electrónico de biomecánica.

Elaborado por: Fredy Llambo.

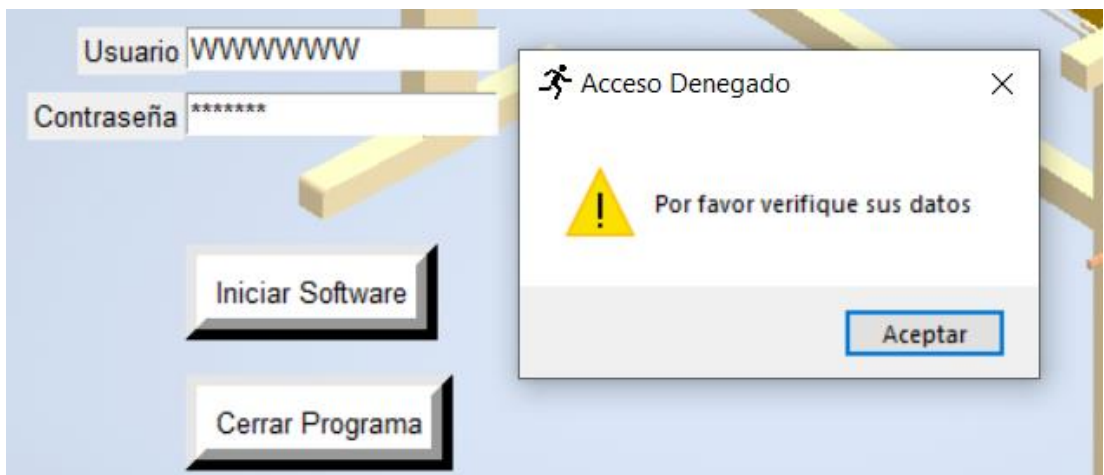


Figura 72: Alerta de seguridad del sistema.

Elaborado por: Fredy Llambo.

Ingresado tanto usuario como contraseña y sean correctos, enlaza a otra pantalla para el control del sistema, en la figura 73 se puede observar en la primera pestaña de la GUI que sirve para la calibración de posición de la cámara, para ubicar la posición correcta hacia el practicante, se podrá posicionarla tanto vertical como horizontalmente.

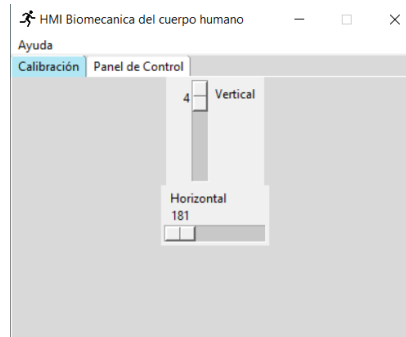


Figura 73: Panel de calibración del sistema.

Elaborado por: Fredy Llambo.

En la pestaña 2 como se puede visualizar en la figura 74, encuentra los botones de control del sistema, el botón de “Iniciar Programa” permite el funcionamiento del seguimiento y grabación de video del practicante, el botón de “Parar Programa” que permite el paro de emergencia del sistema en caso de presentar problemas físicos, en caso de realizar el paro de emergencia el botón de “Reiniciar Programa” ayuda al posicionamiento original de todo el sistema y el botón de “Base de Datos” sirve para ingresar todos los valores que se requiera de la práctica del marchista.

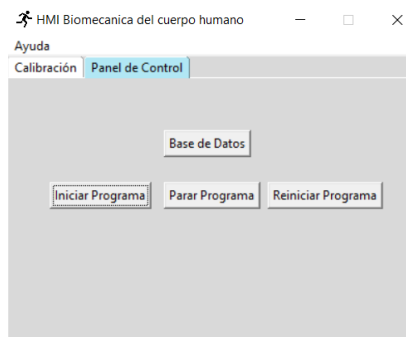


Figura 74: Panel de control.

Elaborado por: Fredy Llambo.

En la figura 75 se muestra lo que contiene la pestaña Ayuda, donde se podrá acceder a los pasos que se debe realizar en caso del desconocimiento de utilización del software, que será descrita en formato pdf, para acceder a cualquier archivo hay que hacer uso de la librería <<webbrowser>>.

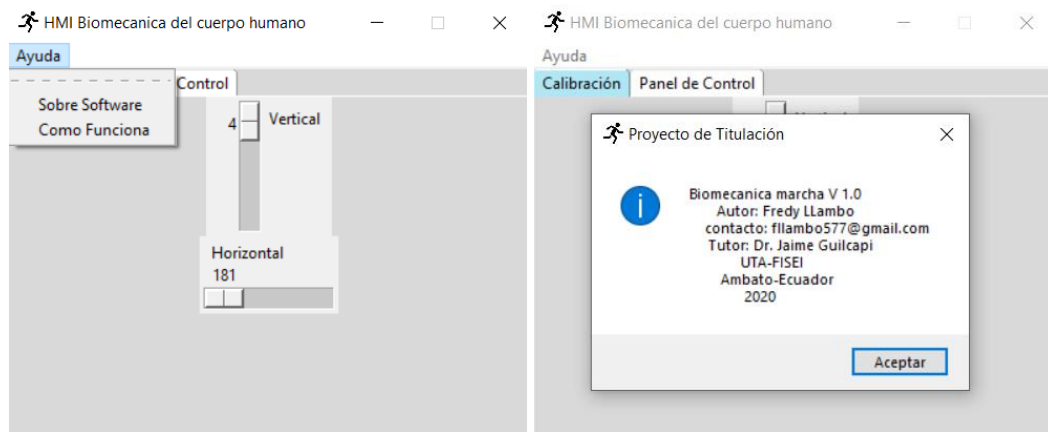


Figura 75: Ayuda del sistema electrónico.

Elaborado por: Fredy Llambo

En el ANEXO G se describe a detalle y por partes toda la programación del software. Para la comunicación entre el software y el prototipo, python y arduino ayudaran a realizar este proceso, la comunicación serial sirve para la implementación del sistema electrónico de biomecánica. En el ANEXO G y H se describe todas las librerías de python y arduino que se utiliza. El ANEXO H detalla la programación de la parte electrónica del sistema.

3.2.3.5 Recolección de imágenes

Para la recopilación de video se utilizará una cámara web y cámara IP, para hacer uso de una cámara IP portable se utilizará la aplicación de DroidCam el cual permite convertir la cámara de un celular en una cámara IP, que servirá para la recolección de video. A continuación, se describe los pasos para la ejecución de la aplicación y uso de cámara IP.

Descargado la aplicación de Google Play Store, se procede abrir e inicia con el cuadro de ayuda de la descripción gráfica de la figura 76, el cual es la manera de comunicación inalámbrica del celular a la laptop, de esta manera se visualiza todo lo que capta la cámara del celular en la pantalla de la laptop.

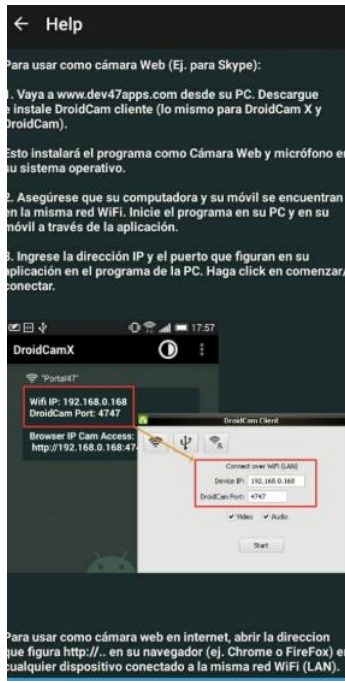


Figura 76: Ayuda de aplicación.

Elaborado por: Fredy Llambo

En la figura 77 se muestra la aplicación abierta y con los datos que se necesita para la conectividad, entre la computadora y el dispositivo móvil.



Figura 77: Aplicación celular.

Elaborado por: Fredy Llambo.

Siguiendo la ayuda de la aplicación redirecciona a la página oficial de DroidCam, el cual es el cliente DroidCam que se debe instalar en la laptop o computadora en donde este la implementación del software. Los requisitos del sistema son Windows 7, 8 y 10, ya sea de 32 o 64 bits. [87] En la figura 78 se puede visualizar ya instalado el programa en la laptop, tiene tres maneras de conexión para la comunicación, dos de ellas son para la comunicación inalámbrica, para la vinculación se debe llenar la dirección del dispositivo y el puerto.

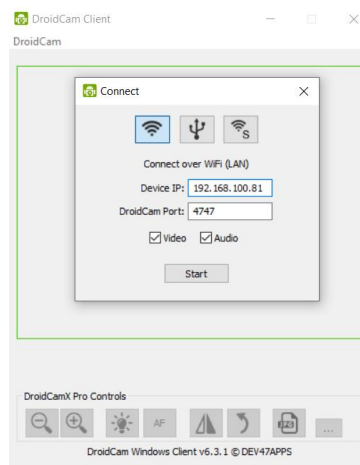


Figura 78: Programa Laptop.

Elaborado por: Fredy Llambo.

En la programación de elección de cámara debe especificar de 0 a 100 su variable de capture dependiendo de su laptop y reconozca la ubicación, en algunas laptops es consecutiva en otras se establece randomicamente.

```
video = (1)
```

```
camara = cv2.VideoCapture(video)
```

En la figura 79 se puede visualizar el resultado de la instalación y vinculación entre el celular y la laptop para el uso de la recolección de video.

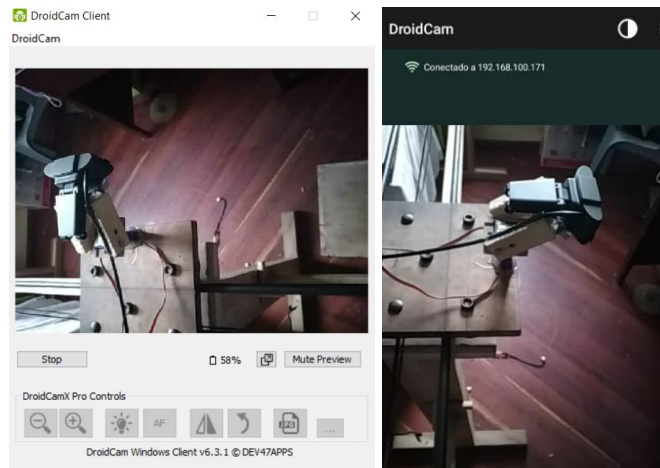


Figura 79: Resultado de instalación DroidCam

Elaborado por: Fredy Llambo

La utilización de una cámara web es la forma de comunicación alámbrica para este sistema, esta forma de recolección de video no necesita de ninguna aplicación o instalación para hacer uso del dispositivo, simplemente se debe conectar el cable USB a la laptop y verificar que se reconozca con el sistema operativo, en raros casos y por problemas de los drivers de la laptop se debe actualizarlo, si en caso de detectar la cámara y no se visualice en el software video, el problema es la versión del controlador de la cámara en compatibilidad con el sistema operativo, de ser el caso se debe desinstalar el controlador del sistema operativo y actualizarlo con el disco de instalación que viene en las cajas de la cámara. En la figura 80 se puede observar de manera breve la forma de reconocimiento de la cámara web y las soluciones si presentan errores.

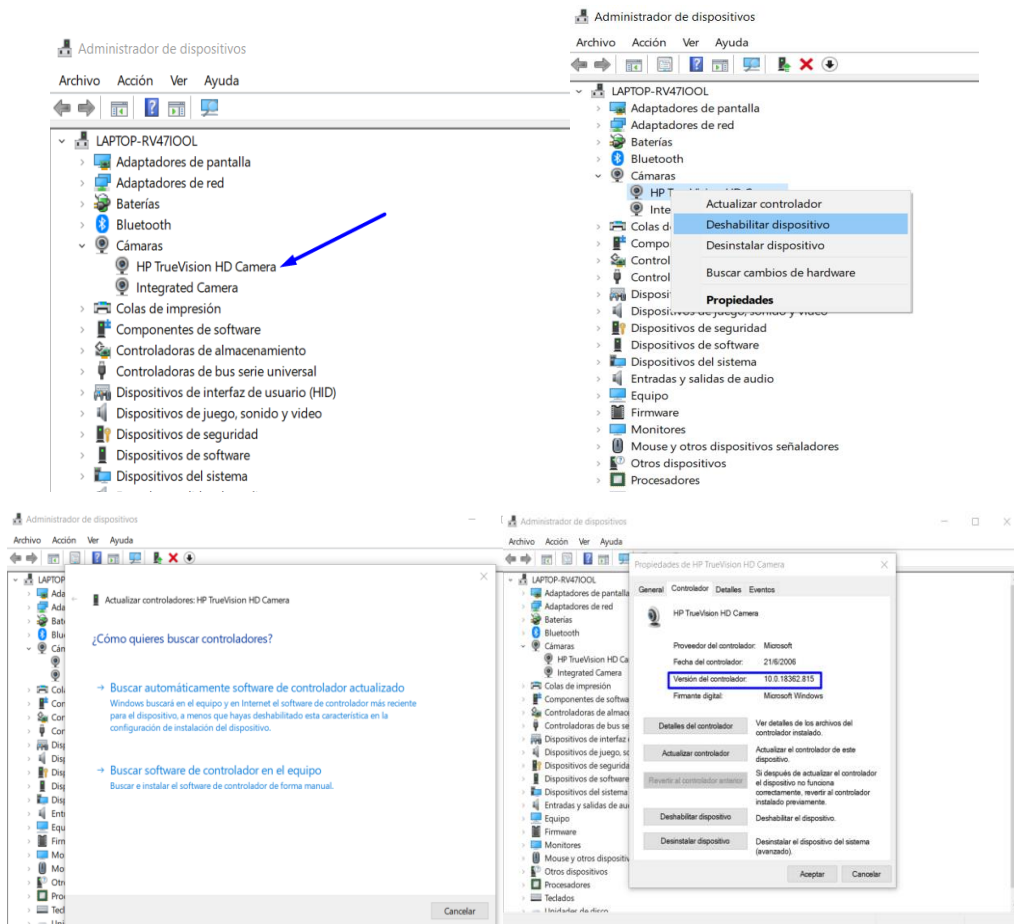


Figura 80 Reconocimiento camara web.

Elaborado por: Fredy Llambo.

3.2.3.6 Creación de base datos

El sistema operativo que se utilizó es el proyecto de investigación es Windows 10 y la distribución de compatibilidad es XAMPP.

Instalación de XAMPP

Se utilizará XAMPP porque es una distribución gratuita y no hay dificultad para instalar. Para descargar esta distribución se buscó la página oficial de APACHE FRIENDS y se procedió a la descarga para el sistema operativo en uso. En la tabla 26 se puede ver tanto las versiones compatibles de los sistemas operativos que soporta XAMPP como la cantidad de memoria que utiliza el programa en su instalación.

Tabla 26: Versiones compatibles de XAMPP

Elaborado por: Fredy Llambo en base a [88]

XAMPP			
SO	WINDOWS	LINUX	OS X
Versión y Tamaño	7.2.32 – 154 Mb	7.2.32 - 149 Mb	7.2.32 - 161 Mb, 7.3.20 - 162 Mb
	7.3.20 – 154Mb	7.3.20 - 150 Mb	7.4.8 - 162 Mb, 7.2.32 - 357 Mb
	7.4.8 – 155Mb	7.4.8 - 150 Mb	7.3.20 - 357 Mb, 7.4.8 - 357 Mb

En la figura 81 se muestra las versiones compatibles con XAMPP y el tamaño del archivo, como nuestro proyecto de investigación está desarrollado en el sistema operativo(SO) de Windows, se procedió a descargar la versión de este SO.

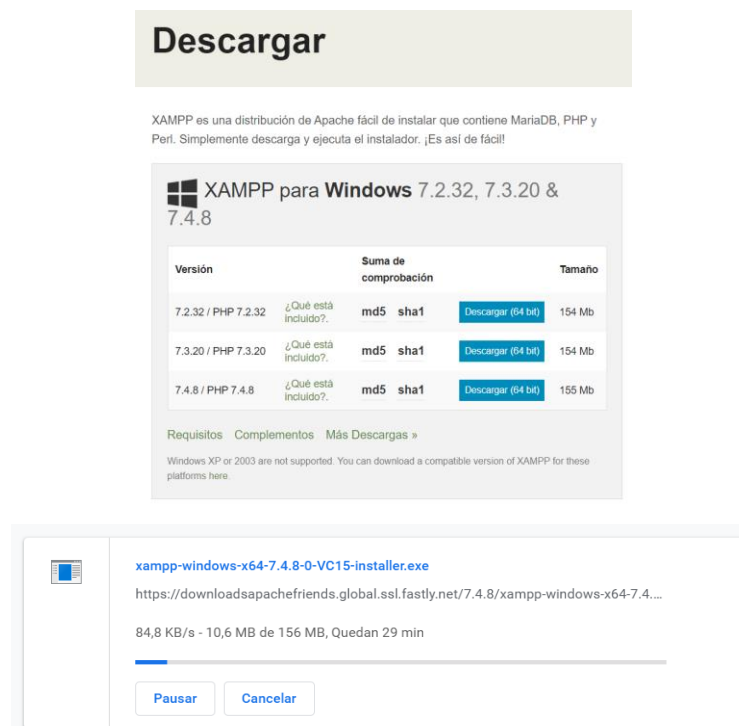


Figura 81: Descarga XAMPP .

Elaborado por: Fredy Llambo.

Para la instalación se puede seleccionar el servidor y el lenguaje de programación según desee el programador como se muestra en la figura 82.

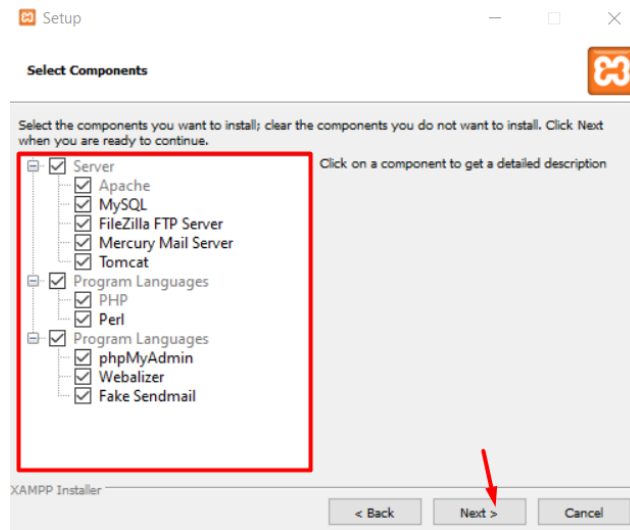


Figura 82: Selección de servidor y lenguaje de programación.

Elaborado por: Fredy Llambo.

Se eligió el disco donde se va a instalar y el lenguaje del programa la figura 83 da a conocer estos pasos, en caso de requerir de programas recomendados por la aplicación lo puede instalar o no.

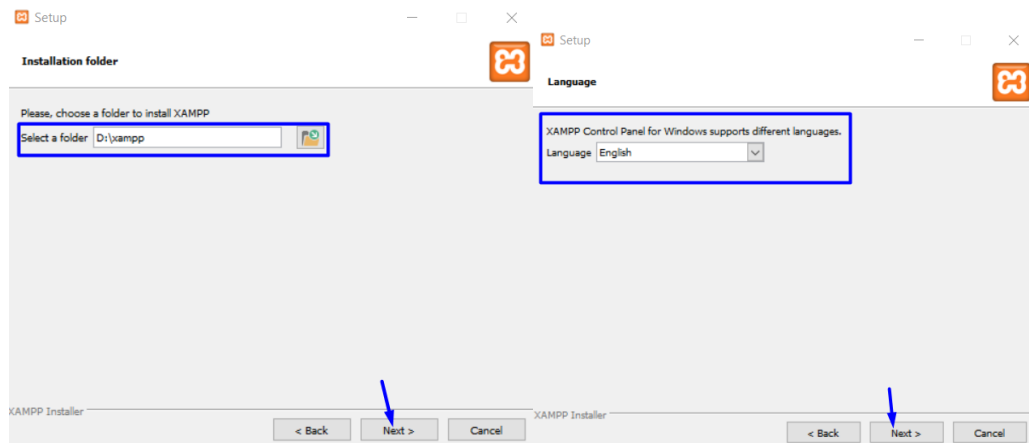


Figura 83: Selección de ubicación y lenguaje.

Elaborado por: Fredy Llambo.

Finalizado la instalación de XAMPP correctamente se debe visualizar en su monitor la ventana que muestra la figura 84.

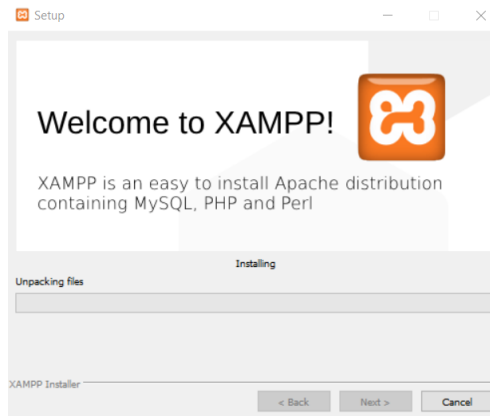


Figura 84: Instalación XAMPP.

Elaborado por: Fredy Llambo.

Para hacer uso de XAMPP se abrió el programa y se observó los módulos y servicios que ofrece para la creación de la base de datos, en el proyecto de investigación se hizo uso de Apache y MySQL por lo tanto se inició y abrió los módulos como se muestra en la figura 85.

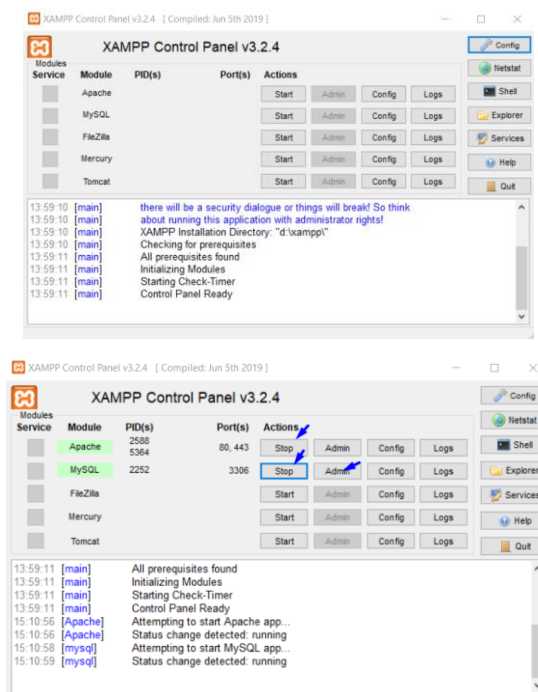


Figura 85: Inicio de módulos para uso de XAMPP.

Elaborado por: Fredy Llambo.

Automáticamente abrirá phpMyAdmin si se realizó correctamente los pasos anteriores de la figura 85, se procedió a la creación de la base de datos para la vinculación con el software en la figura 86 se muestra los valores a considerar para nuestro proyecto de investigación. En el Anexo F se puede observar la programación base del envío de datos.

Mostrando filas 0 - 5 (total de 6, La consulta tardó 0,0006 segundos.)

```
SELECT * FROM `datos`
```

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Nii

+ Opciones

	id	BrazoDerecho	Brazolzquierdo	PiernaDerecha	Piernalzquierda
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	55	34	66	43
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	12	23	34	53
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	3	12	23	34	53
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	4	2	83	38	59
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	5	1	93	37	73
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	6	120	73	74	88

Seleccionar todo | Para los elementos que están marcados: Editar Copiar Borrar

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Sort by key: Nii

Operaciones sobre los resultados de la consulta

Imprimir Copiar al portapapeles Exportar Mostrar gráfico Crear vista

Guardar esta consulta en favoritos

Figura 86: Base de datos en el servidor.

Elaborado por: Fredy Llambo.

3.2.3.7 Pruebas de funcionamiento

Pruebas del sistema electrónico

Una vez que se ha implementado el prototipo se procede a probar el funcionamiento del sistema electrónico, primero el desplazamiento de la cámara para lo cual se necesita del correcto funcionamiento del motor. En la figura 87 se observa una descripción grafica del funcionamiento.

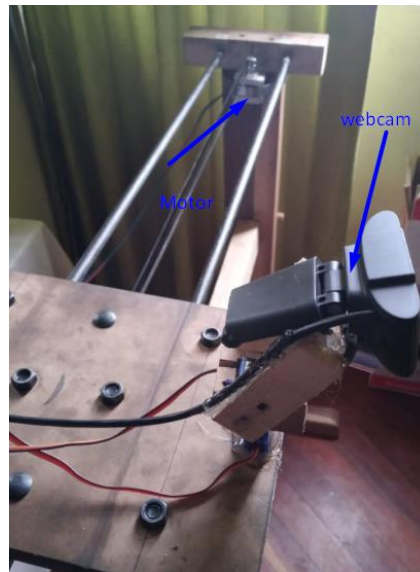


Figura 87: Funcionamiento de motor.

Elaborado por: Fredy Llambo.

Para el desplazamiento se utiliza poleas de 20 dientes y una banda de la medida de la polea, en la gráfica 88 se observa el armado de las poleas y banda.



Figura 88: Poleas y banda.

Elaborado por: Fredy Llambo.

Para la calibración de la posición de la cámara se utiliza servomotores en la figura 89 se observa el armado de la cámara y los servomotores.

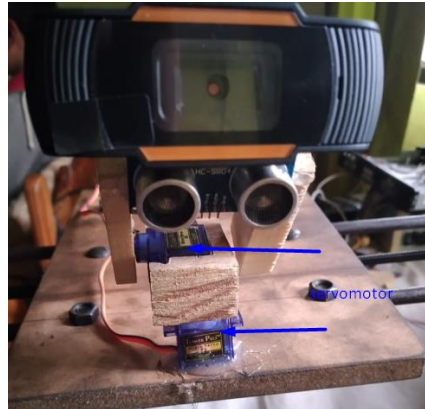


Figura 89: Servomotor, ultrasónico y cámara.

Elaborado por: Fredy Llambo.

Para el desplazamiento de la cámara el soporte que está construida con rodamientos y poleas a torno desde ser del mismo material que los carriles como se muestra en la figura 90 el armado del soporte de la cámara y colocación en las barrillas que son los rieles para el desplazamiento.



Figura 90: Soporte de cámara.

Elaborado por: Fredy Llambo.

En la figura 91 se puede observar la verificación de las salidas de voltajes los 5 voltios serán utilizados para los dispositivos que requieren de este voltaje, los 15 voltio se utilizara para el funcionamiento del motor.

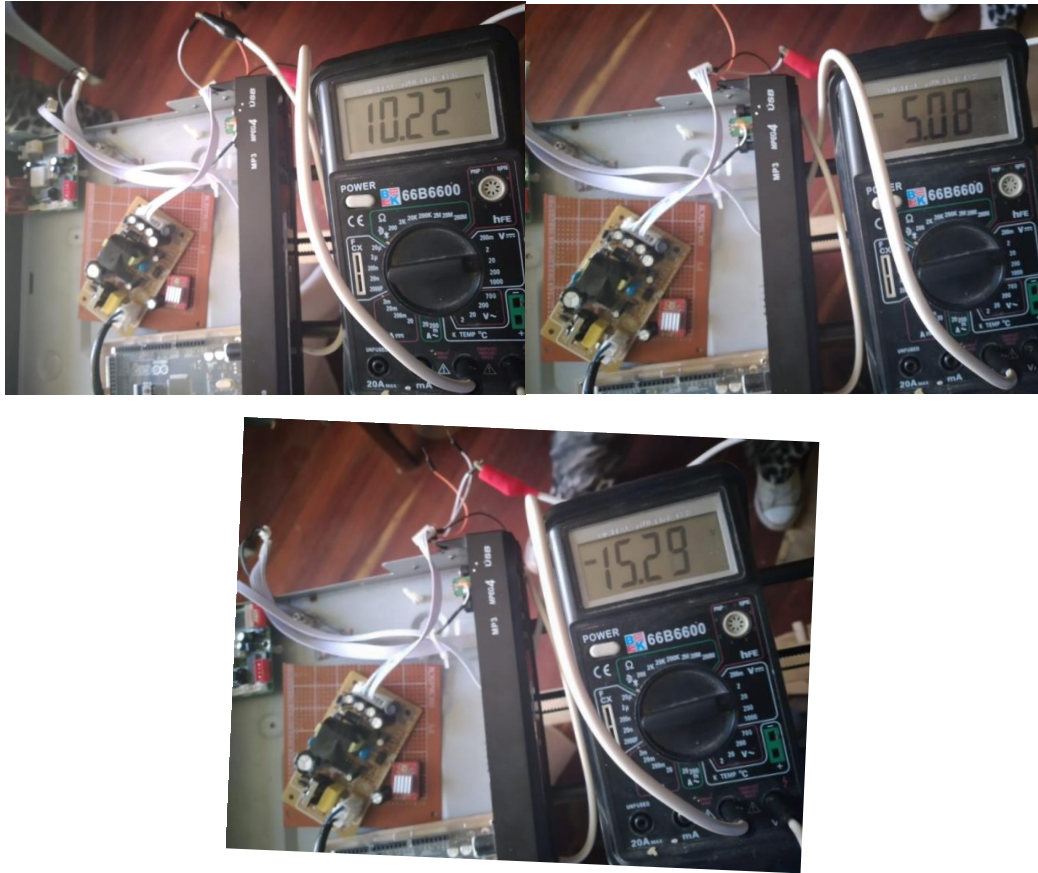


Figura 91: Verificación de salidas de voltaje.

Elaborado por: Fredy Llambo.

Para el funcionamiento de los dispositivos mencionados anteriormente se utilizó arduino mega en la figura 92 se puede observar como procede el armado de la parte de control del sistema electrónico.

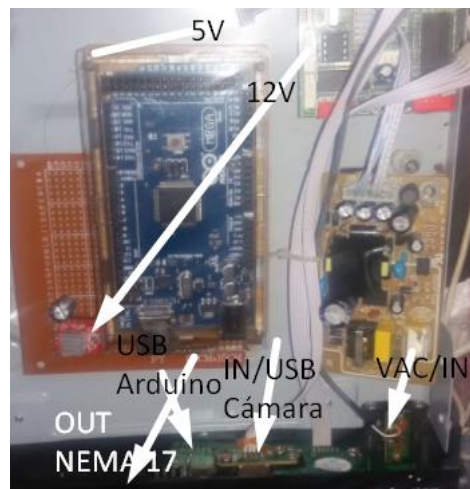


Figura 92: Circuito de prototipo.

Elaborado por: Fredy Llambo.

Pruebas del software

El software inicia con un usuario y contraseña que es un medio de protección de los datos que se guardaran de las prácticas de los marchistas. En la figura 93.a se puede observar el correcto funcionamiento de la seguridad del software y en la figura 93.b se muestra el ingreso al control del sistema con la contraseña correcta del usuario.

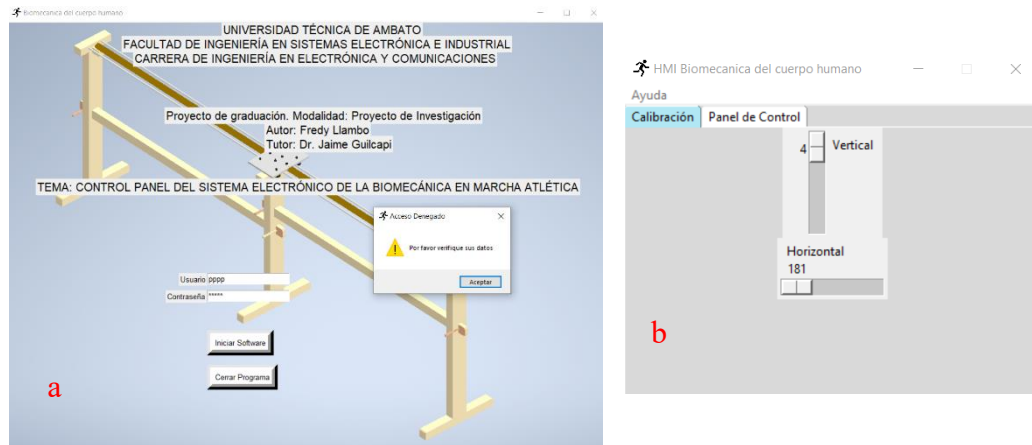


Figura 93: a) HMI Contraseña. b) Control Sistema
Elaborado por: Fredy Llambo.

El funcionamiento desde el software para la calibración se la describe gráficamente el funcionamiento en la figura 94, cuando se mueve verticalmente el slider el servo que está en conexión se moverá el ángulo definido en el slider.

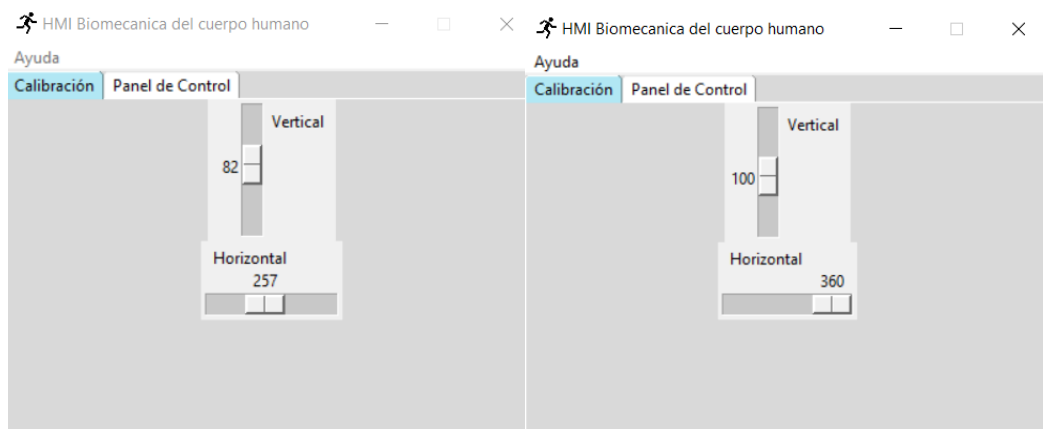


Figura 94: Calibración posición cámara.
Elaborado por: Fredy Llambo.

En el panel de control del software al presionar iniciar programa, el sensor ultrasónico se pone en funcionamiento para detectar e iniciar la activación del motor. En la figura 95 se puede observar el seguimiento del practicante con la detección del sensor de distancia. En la tabla 27 se indica los datos de las distancias que se recolecta para la estimación y activación del motor, las pruebas son realizadas con diferentes personas tanto en edad y estatura.



Figura 95: Funcionamiento de sensor.

Elaborado por: Fredy Llambo.

Para cada estatura la detección del practicante es diferente y el tiempo de respuesta no es alto para la detección, mediante las pruebas la distancia es correcta en la medición y el tiempo es regular para la activación del motor.

Tabla 27: Prueba de sensor ultrasónico.

Elaborado por: Fredy Llambo.

Prueba	Estatura baja (cm)	Estatura media (cm)	Estatura alta (cm)	Tiempo de respuesta (s)
1	87	119	360	4
2	83	121	260	2
3	84	121	269	4
4	83	120	362	4
5	81	120	298	1
6	85	122	369	3
7	86	122	287	1
8	88	122	355	4
9	89	120	318	3
10	88	120	259	1
11	86	119	269	2
12	85	118	250	3
13	85	118	285	3
14	85	120	357	4
15	84	121	300	1

En la tabla 27 se puede observar todas las distancias a la que fue detectado el practicante, estas distancias son expresadas en centímetros, la columna 1 muestra el número de pruebas que se realizó, la columna 2 da a conocer la distancia de detección a practicantes de estatura baja, es decir practicantes de entre 1.10 a 1.30 metros, la columna 2 para practicantes de estatura mediana, es decir practicantes de entre 1.31 a 1.50 metros, la columna 3 para practicantes de estatura alta, es decir practicantes de 1.65 metros en adelante.

Caída de voltaje de la fuente

Para la activación del motor con todo el peso del soporte de la cámara y los servomotores, se debe hacer varias pruebas para que no haya caída de tensión y el motor no deje de funcionar en el desplazamiento, en la tabla 28 se puede observar el tiempo que se demora en desplazar el soporte todo el riel y la caída de tensión no es alta debido a la estructura de poleas.

En la figura 96 se puede observar que no hay mucha pérdida de voltaje en el tiempo máximo de activación, de esta manera se verifica que la fuente funciona perfectamente para el sistema.

Tabla 28: Tiempo de desplazamiento del soporte de la cámara.

Elaborado por: Fredy Llambo.

Prueba	Voltaje DC [v]	Tiempo de desplazamiento [s]
1	12,45	36
2	12,57	22
3	12,72	23
4	13,37	22
5	13,52	41
6	13,65	21
7	13,40	32
8	13,11	31
9	13,18	44
10	12,59	36
11	13,03	30
12	13,06	38
13	13,02	32
14	13,00	32
15	13,00	30

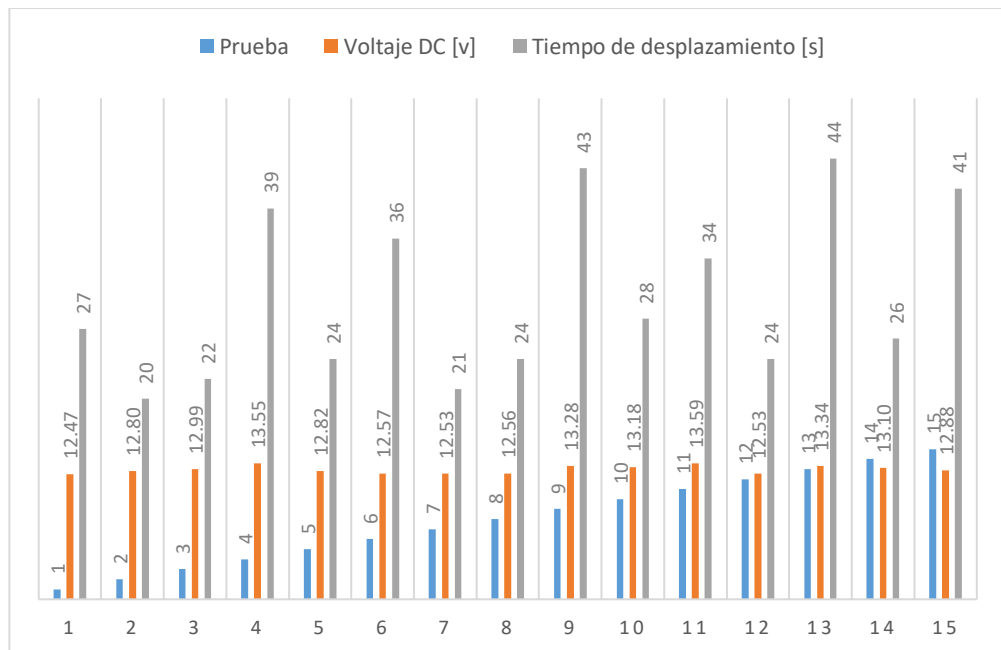


Figura 96: Análisis de resultados del voltaje DC y tiempo de activación.

Elaborado por: Fredy Llambo.

FPS de reproducción de video

Los fotogramas por segundo del video procesado con las CNN son importantes para el análisis del video en el tiempo transcurrido de reproducción. En la tabla 29 se puede observar los fotogramas por segundo del proyecto de investigación, estos FPS se lograron gracias a la tarjeta de video NVIDIA GEFORCE MX130

Tabla 29: FPS y tiempo de reproducción.

Elaborado por: Fredy Llambo.

Prueba	FPS	Tiempo (s)
1	33	28
2	23	41
3	31	28
4	30	35
5	22	33
6	20	33
7	25	30
8	31	45
9	28	31
10	30	24
11	20	44
12	24	30
13	24	25
14	30	22
15	29	27

La estimación humana del cuerpo humano se puede observar en el ANEXO E, con respuestas de FPS adecuadas.

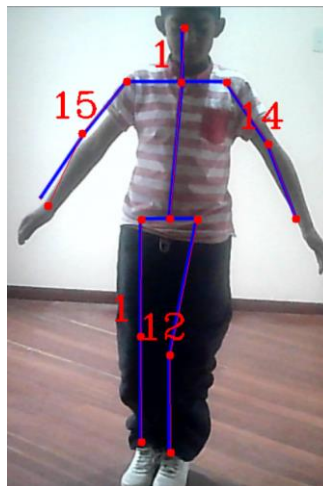


Figura 97: Ángulos en la estimación humana.

Elaborado por: Fredy Llambo.

La determinación de los ángulos de cada parte del cuerpo se muestra en la figura 97 sin instalación de CUDA y bajo FPS, es por eso que el punto clave de la rodilla derecha tiene una leve inclinación de 12 grados.

En la figura 98 se observa la respuesta con el controlador de NVIDIA, en donde se puede ver claramente todos los puntos clave bien establecidos según nuestra RNA.

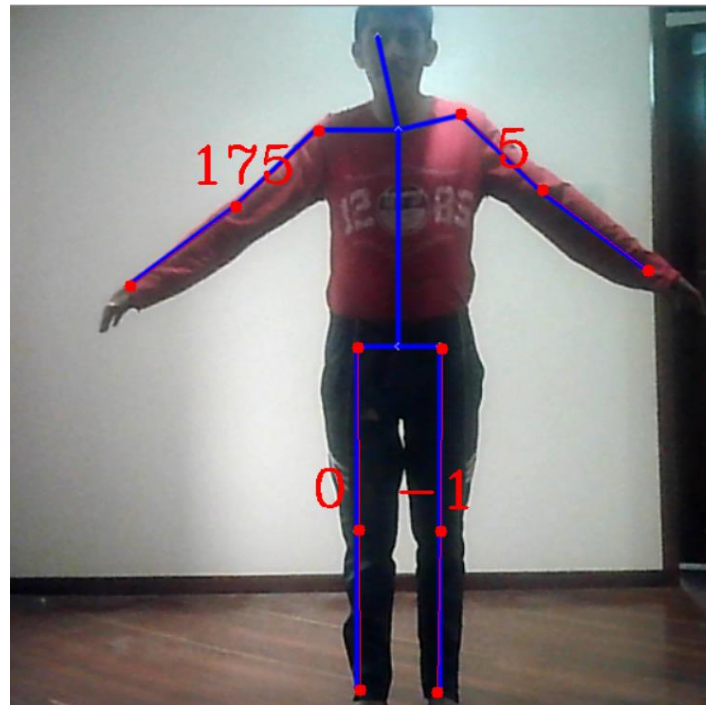


Figura 98: Angulo en el cuerpo humano.

Elaborado por: Fredy Llambo.

En la figura 99 se muestra el envío de datos (ángulos) a la base de datos que se recolecto en la figura 98.

	id	Brazo derecho	Brazo izquierdo	pierna izquierda	Pierna derecha
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	24.12	23.12	13	0
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	7	5	175	0	-1
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	8	15	14	1	12
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	9	5	175	0	-1
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	10	15	14	1	12

Figura 99: Ingreso de base de datos.

Elaborado por: Fredy Llambo.

Pruebas de modelo para el análisis profundo.

Se realizaron diferentes pruebas con tres modelos más utilizados en el aprendizaje profundo automático, para determinar cuál es el adecuado para el proyecto de investigación, en la tabla 30 se muestra el valor de los FPS en un intervalo de tiempo, si hay o no recalentamiento en la CPU y GPU. En las pruebas realizadas el modelo COCO presento FPS de 1.20, el modelo MPI 1.74 FPS y el modelo BODY_25 presento 3,36 FPS. Estos datos confirmaron que para el proyecto el mejor modelo es BODY_25, porque los FPS son el doble en comparación a los otros modelos, en el Anexo E se muestra las pruebas realizadas con el modelo de mayor FPS.

Tabla 30: Análisis de modelos del aprendizaje automatico.

Elaborado por: Fredy Llambo

Prueba	FPS COCO	FPS MPI	FPS BODY25	Tiempo (s)	CPU (%)	GPU (%)
1	1,22	1,76	3,38	18,36	12,49	0,43
2	1,19	1,77	3,36	25,26	12,87	0,43
3	1,18	1,74	3,36	30,08	12,47	0,44
4	1,22	1,75	3,35	23,44	12,37	0,5
5	1,19	1,76	3,34	18,9	12,85	0,49
6	1,22	1,77	3,34	29,38	12,23	0,43
7	1,18	1,75	3,34	28,58	12,52	0,45
8	1,21	1,77	3,35	25,38	12,44	0,5
9	1,20	1,74	3,34	30,18	12,87	0,4
10	1,19	1,75	3,36	27,37	12,78	0,41
11	1,18	1,76	3,34	26,99	12,32	0,5
12	1,19	1,74	3,35	21,02	12,76	0,47
13	1,18	1,77	3,35	19,16	12,78	0,5
14	1,19	1,73	3,35	22,05	12,47	0,46
15	1,22	1,73	3,36	23,66	12,88	0,45
16	1,18	1,76	3,38	18,91	12,32	0,46
17	1,21	1,75	3,35	22,09	12,52	0,45
18	1,20	1,77	3,33	25,35	12,45	0,48

19	1,20	1,78	3,34	18,55	12,71	0,44
20	1,21	1,77	3,35	26,37	12,66	0,45
21	1,20	1,76	3,33	27,94	12,26	0,45
22	1,22	1,73	3,33	17,03	12,8	0,47
23	1,19	1,74	3,38	29,73	12,54	0,46
24	1,19	1,77	3,36	25,76	12,85	0,48
25	1,19	1,77	3,37	21,05	12,8	0,42
26	1,22	1,78	3,38	23,46	12,42	0,48
27	1,18	1,74	3,36	28,69	12,22	0,43
28	1,22	1,74	3,35	26,98	12,88	0,47
29	1,20	1,78	3,38	27,19	12,23	0,45
30	1,18	1,74	3,33	31,06	12,44	0,45
31	1,22	1,76	3,34	23,38	12,56	0,43
32	1,19	1,74	3,38	22,92	12,63	0,46
33	1,21	1,74	3,34	22,16	12,27	0,46
34	1,18	1,76	3,33	30,18	12,29	0,42
35	1,19	1,76	3,38	22,77	12,43	0,43
36	1,19	1,75	3,36	24,31	12,31	0,45
37	1,20	1,77	3,34	21,28	12,24	0,42
38	1,20	1,78	3,36	23,39	12,29	0,44
39	1,19	1,76	3,35	28,48	12,32	0,49
40	1,18	1,78	3,33	30,06	12,27	0,5
41	1,18	1,78	3,36	20,98	12,87	0,43
42	1,21	1,77	3,33	29,08	12,49	0,46
43	1,19	1,75	3,35	17,53	12,28	0,41
44	1,18	1,76	3,34	23,1	12,4	0,42
45	1,18	1,73	3,36	22,24	12,56	0,42
46	1,22	1,73	3,36	20,58	12,49	0,41
47	1,19	1,74	3,38	29,67	12,65	0,46
48	1,19	1,76	3,37	30,02	12,3	0,44
49	1,20	1,75	3,34	29,2	12,63	0,49
50	1,20	1,76	3,37	31,29	12,65	0,48
Promedio	1,20	1,76	3,35	24,85	12,52	0,45

Pruebas del sistema

Fase de apoyo simple

Se realizaron diferentes pruebas con datos de un marchista oficial con ayuda del aprendizaje profundo automático, para determinar si el sistema está dando como resultado los ángulos que son en cada fase de la marcha atlética, en la tabla 31 se muestra los ángulos en las extremidades inferiores y superiores.

Tabla 31: Ángulos en fase de apoyo simple.

Elaborado por: Fredy Llambo.

Prueba	Apoyo Simple (ángulos)			
	Brazo Derecho	Brazo Izquierdo	Pierna Derecha	Pierna Izquierda
1	90	87	112	179
2	88	88	110	178
3	89	89	112	179
4	92	90	110	179
5	88	88	109	178
6	92	89	108	180
7	89	87	110	179
8	90	89	109	179
9	91	88	111	178
10	88	90	111	179
11	91	89	110	178
12	88	87	108	178
13	90	90	109	180
14	89	89	109	180
15	89	88	110	180
16	92	88	111	180
17	88	88	111	178
18	92	89	108	180
19	88	90	110	178
20	92	87	112	179

21	92	87	111	178
22	92	90	109	180
23	92	90	109	178
24	92	88	109	180
25	91	88	111	179
26	88	87	109	178
27	91	89	111	179
28	88	90	111	178
29	91	90	111	180
30	89	89	112	180
31	92	90	108	178
32	92	90	108	179
33	91	88	112	180
34	91	87	111	179
35	88	90	112	178
36	90	90	108	180
37	92	87	111	180
38	89	89	112	180
39	91	88	111	178
40	92	88	111	180
41	91	89	112	179
42	90	90	109	178
43	92	89	110	179
44	88	87	109	178
45	91	87	112	180
46	91	89	111	178
47	89	87	112	178
48	91	87	110	179
49	92	87	108	179
50	90	90	110	180
Promedio	90	88	110	178

En la fase de apoyo simple como se muestra en el Anexo I los practicantes de la marcha atlética deben tener un ángulo aproximado a 90°, la pierna izquierda un aproximadamente de 180°. [46]

Fase apoyo doble

En la fase de apoyo doble como se muestra en el Anexo I el marchista tiene las dos piernas en contacto con el suelo, la pierna de apoyo (pierna derecha) debe estar completamente extendida es decir con un ángulo de 180°, la mano derecha debe estar perpendicular el codo y hombro es decir tener un ángulo de 180 y la mano izquierda un ángulo de 90°, en la tabla 32 se puede observar que son valores que no tienen un alto porcentaje de error. [46]

Tabla 32: Fase de apoyo doble.

Elabora por: Fredy Llambo.

Prueba	Apoyo Doble (ángulos)			
	Brazo Derecho	Brazo Izquierdo	Pierna Derecha	Pierna Izquierda
1	179	87	179	162
2	179	90	179	164
3	179	89	180	164
4	180	87	180	169
5	179	90	180	165
6	179	89	180	170
7	180	90	179	160
8	179	89	179	163
9	180	89	180	165
10	179	88	179	170
11	180	88	179	165
12	179	90	180	166
13	180	88	179	163
14	180	87	180	162
15	179	90	180	164
16	180	89	180	169
17	179	89	179	160
18	179	89	180	163
19	179	88	179	160
20	179	89	179	162
21	179	89	180	170
22	180	87	179	165
23	179	87	179	160
24	180	87	179	163

25	179	89	179	168
26	180	89	179	166
27	179	89	179	170
28	180	90	180	166
29	180	89	180	170
30	179	88	180	167
31	180	90	180	165
32	180	87	179	160
33	180	90	179	161
34	179	87	180	170
35	180	88	180	168
36	180	87	179	166
37	179	88	180	168
38	179	87	179	160
39	179	87	179	164
40	180	89	179	170
41	180	88	180	161
42	179	87	179	170
43	180	88	180	168
44	179	89	180	170
45	179	87	180	164
46	179	87	179	170
47	180	88	179	160
48	179	89	179	161
49	179	90	180	169
50	179	88	179	160
Promedio	179	88	179	165

Al realizar las pruebas respectivas en nuestro prototipo se puede observar que el sistema de análisis profundo automático tiene los valores especificados en la marcha atlética, tal cual se muestra en la tabla 31 y 32.

3.2.4 Presupuesto

En la tabla 33 se puede observar a detalle el presupuesto del presente proyecto de investigación, en el cual el costo del diseño e implementación de todo el sistema electrónico esta cotizado a un valor aproximado de \$1227,99. Se ha optado por cobrar \$70 por la mano de obra, ya que se requiere de un día para armar de la estructura que es todo portable y la instalación de los elementos electrónicos, además se estableció un monto de \$150 por el diseño del prototipo.

Tabla 33: Presupuesto del sistema.

Elaborado por: Fredy Llambo.

Presupuesto					
Ítem	Descripción	Unidad	Cantidad	Precio Unitario \$	Precio Total \$
1	Arduino Mega	c/u	1	25	25
2	Cámara web	c/u	1	34,99	34,99
3	Motor nema 17	c/u	1	19	19
4	Pantalla LCD	c/u	1	7	7
5	Servo Motor	c/u	2	4,5	9
6	Sensor ultrasónico	c/u	1	4	4
7	Fuente de alimentación	c/u	1	35	35
8	Final de carrera	c/u	2	2	4
9	Botón emergencia	c/u	1	2,5	2,5
10	Banda	c/m	6	4,5	27
11	Polea	c/u	2	0,75	1,5
12	Barrilla lisa	c/u	2	15,5	31
13	Polea rodamiento	c/u	6	25	150
14	Tarjeta de Video	c/u	1	78	78
15	Procesador	c/u	1	480	480
subtotal					907,99
Estructura					100
Diseño del prototipo					150
Mano de obra					70
Total \$					1227,99

CAPITULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

El diseño e implementación del sistema electrónico de corrección del gesto deportivo en base a la biomecánica en deportistas de marcha atlética, se obtuvieron las conclusiones que se describen a continuación:

- Los resultados recabados durante el análisis de factores que son causantes de cometer errores en los marchistas son: la fractura en la columna porque los ejercicios requieren de potenciación y flexibilización de columna, la edad a simple vista es importante por el desgaste en las zonas de trabajo en la marcha y la condición física porque es una rama del deporte.
- El algoritmo desarrollado a la estimación humana ha permitido la detección de puntos claves con ayuda de machine learning, visión artificial y deep learning, llegando a verificar que las mejores bibliotecas de gran ayuda en los campos mencionados es OpenCV y DNN.
- La eficiencia del software se ha determinado mediante el análisis del porcentaje de error en la estimación humana, obteniendo una eficiencia del 98%. La eficiencia de detección dependerá mucho de la tarjeta de video que este instalada en la laptop en uso del software, en caso de no poseer una tarjeta de video de arquitectura mínima para CUDA 5 la eficiencia no será satisfactoria produciendo recalentamiento, daños internos del ordenador y daños en el software.
- La implementación del sistema electrónico requirió de una fuente de 0.5A, de 5, 10 y 15 voltios como mínimo para que el prototipo funcione de manera instantánea y estable.

4.2 Recomendaciones

- Dar a conocer a los marchistas la excesiva flexión o extensión de los brazos, falta de relajación en tronco y caderas, perder en algún ciclo el contacto con el suelo y elevación excesiva de hombros y brazos, que lleva a cometer errores dentro de las prácticas y así no perder los sueños que anhela todo deportista para alcanzar el éxito, por ello deben tener cuidado en las actividades que realizan para evitar fracturas, tener constantes chequeos médicos para que les receten vitaminas para los huesos y ser constantes en el entrenamiento.
- Difundir el algoritmo desarrollado porque permite la visualización de la biomecánica del marchista, para un análisis del procesamiento de imágenes recolectadas.
- El análisis del procesamiento de video tiene un alto consumo en el ordenador, por lo que es preferible optar por tarjetas de video que soporte la arquitectura actual de los controladores para un análisis claro de la recolección de imágenes.
- Los voltajes utilizados son propicios para la implementación del prototipo, a pesar que el desplazamiento va depender del peso del soporte el motor para activarse requiere de la corriente nominal y mientras jala el soporte no hay manera de evitar caídas de tensión.

REFERENCIAS BIBLIOGRÁFICAS

- [1] S. Qiao, Y. Wang, and J. Li, “Real-Time Human Gesture Grading Based on OpenPose,” 2017.
- [2] D. Mehta *et al.*, “Single-Shot Multi-Person 3D Pose Estimation From Monocular RGB,” *2018 Int. Conf. 3D Vis.*, no. 335545, pp. 120–130, 2018, doi: 10.1109/3DV.2018.00024.
- [3] R. Dabral, A. Ai, and A. Ai, “Multi-Person 3D Human Pose Estimation from Monocular Images Arjun Jain,” *2019 Int. Conf. 3D Vis.*, pp. 405–414, 2019, doi: 10.1109/3DV.2019.00052.
- [4] Z. Cao *et al.*, “OpenPose : Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields,” 2019, doi: 10.1109/TPAMI.2019.2929257.
- [5] Asociación Internacional de Federaciones de Atletismo, “Reglamento de Competición 2018-2019,” p. 254, 2018, [Online]. Available: <http://www.iaaf.org>.
- [6] C. M. F. Morales, “El nivel de coordinación y su relación con la técnica de la marcha atlética en los seleccionados de la categoría pre juvenil de la federación deportiva de chimborazo periodo 2015,” no. December, pp. 2–4, 2015.
- [7] J. M. Rubio, “Capacidades condicionantes de la marcha atlética en los estudiantes del ciclo básico del instituto tecnológico superior sucre,” p. 188, 2014, [Online]. Available: <http://www.dspace.uce.edu.ec/bitstream/25000/4442/1/T-UCE-0016-008.pdf>.
- [8] D. R. Seshadri, C. Drummond, J. Craker, J. R. Rowbottom, and J. E. Voos, “Wearable Devices for Sports: New Integrated Technologies Allow Coaches, Physicians, and Trainers to Better Understand the Physical Demands of Athletes in Real time,” *IEEE Pulse*, vol. 8, no. 1, pp. 38–43, 2017, doi: 10.1109/MPUL.2016.2627240.
- [9] C. Lic and K. Y. Fisiatria, “Facultad De Medicina Anatomia Funcional Y

- Biomecánica Programa De La Asignatura 2018,” pp. 1–122, 2018.
- [10] G. C. ELENA, “Lesiones en miembro inferior asociadas a la práctica deportiva de la marcha atlética en la provincia de Alicante y Murcia.,” 2017.
- [11] J. Barreto and S. C. Morales, “Biomecánica de la marcha atlética. Análisis cinemático de su desarrollo y comparación con la marcha normal,” *Rev. Cuba. Investig. Biomédicas*, vol. 36, no. 2, 2017.
- [12] J. Campos Granell and J. E. Gallach Lazcorreta, *Las técnicas de atletismo: manual práctico de enseñanza*. Barcelona, SPAIN: Editorial Paidotribo México, 2017.
- [13] N. R. Council, M. Commission on Physical Sciences and Applications, N. S. Board, and P. on C. S. and A. Intelligence, “Computer Science and Artificial Intelligence,” National Academies Press, Washington, D.C., UNITED STATES, 1997. [Online]. Available: <http://ebookcentral.proquest.com/lib/uta-ebooks/detail.action?docID=3376144>.
- [14] L. Museros, O. Pujol, and N. Agell, *Artificial Intelligence Research and Development: Recent Advances and Applications*. Burke, NETHERLANDS, THE: IOS Press, 2014.
- [15] Ramón Heredia, “10 aplicaciones de Inteligencia artificial para no quedarse atrás en el 2018 | Ebanking News,” *eBankingNews*, Apr. 20, 2018. <https://www.ebankingnews.com/noticias/10-aplicaciones-de-inteligencia-artificial-para-no-quedarse-atras-en-el-2018-0040490> (accessed Jul. 29, 2020).
- [16] E. Tyugu, *Algorithms and Architectures of Artificial Intelligence*. Amsterdam, NETHERLANDS, THE: IOS Press, 2007.
- [17] A. G. Muñiz, “Aplicaciones de Técnicas de Inteligencia Artificial Basadas en Aprendizaje Profundo (deep learning) al Análisis y Mejora de la Eficiencia de Procesos Industriales,” no. February 2018, p. 75, 2018, doi: 10.13140/RG.2.2.15144.72967.
- [18] T. Oladipupo Ayodele, “Types of Machine Learning Algorithms,” United Kingdom, 2010. Accessed: Sep. 18, 2020. [Online]. Available:

www.intechopen.com.

- [19] L. Sneeringer, *Professional Python*. Hoboken, UNITED STATES: John Wiley & Sons, Incorporated, 2015.
- [20] S. Bassi and S. Bassi, *Python for Bioinformatics*. London, UNITED KINGDOM: CRC Press LLC, 2016.
- [21] J. Howse, *OpenCV Computer Vision with Python*. Olton, UNITED KINGDOM: Packt Publishing, Limited, 2013.
- [22] opencv, “OpenCV,” *opencv team*. <https://opencv.org/> (accessed Jul. 29, 2020).
- [23] Google Sites, “Procesamiento De Imágenes en OpenCV.” <https://sites.google.com/site/cg05procesamientodeimagenes/> (accessed Aug. 01, 2020).
- [24] Michael Elkan, “Procesamiento de imágenes en OpenCV.” <http://acodigo.blogspot.com/2013/05/procesamiento-de-imagenes-en-opencv.html> (accessed Aug. 01, 2020).
- [25] P. Alley, “Introductory Microcontroller Programming by,” no. May, 2011.
- [26] A. Baškys, *Microcontrollers*. 2012.
- [27] M. J. Epstein, J. F. Manzoni, and M. P. Bates, *Interfacing PIC Microcontrollers : Embedded Design by Interactive Simulation*. Jordan Hill, UNITED KINGDOM: Elsevier Science & Technology, 2006.
- [28] S. Dawoud and R. Peplow, *Digital System Design : Use of Microcontroller*. Aalborg, DENMARK: River Publishers, 2010.
- [29] J. Morton, *The PIC Microcontroller: Your Personal Introductory Course*. Jordan Hill, UNITED KINGDOM: Elsevier Science & Technology, 2005.
- [30] J. Vetelino and A. Reghu, *Introduction to Sensors*. Bosa Roca, UNITED STATES: Taylor & Francis Group, 2010.
- [31] Miguel Angel Muñoz Navarro, “Brújula.” https://www.manualvuelo.es/2inst/29_brujl.html (accessed Sep. 20, 2020).

- [32] J. K. Fink, *Polymeric Sensors and Actuators*. Somerset, UNITED STATES: Wiley, 2012.
- [33] Academic Resource Center, “SENSORS: Types and Characteristics Academic Resource Center.”
- [34] E. Sadun, *Digital Video Essentials : Shoot, Transfer, Edit, Share*. Alameda, UNITED STATES: Wiley, 2003.
- [35] S. Giancola, M. Valenti, and R. Sala, *A Survey on 3D Cameras: Metrological Comparison of Time-Of-Flight, Structured-Light and Active Stereoscopy Technologies*. Cham, SWITZERLAND: Springer, 2018.
- [36] S. K. Ghosh, *Digital Image Processing*. New Delhi, INDIA: Alpha Science International, 2012.
- [37] A. V Totsky, A. A. Zelensky, and V. F. Kravchenko, *Bispectral Methods of Signal Processing : Applications in Radar, Telecommunications and Digital Image Restoration*. Berlin/Boston, GERMANY: De Gruyter, Inc., 2014.
- [38] T. C. Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Li Wang, Gang Wang, Jianfei Cai, “Recent advances in convolutional neural network acceleration,” *Neurocomputing*, 2017, doi: 10.1016/j.neucom.2018.09.038.
- [39] J. Wu, “Convolutional Neural Networks,” 2020, doi: 10.5220/0006972204340439.
- [40] David Erroz Arroyo, “Visualizando neuronas en Redes Neuronales Convolucionales,” UPNA, Pamplona, 2019.
- [41] Monson H. Hayes, *Digital Signal Processing*. The McGraw-Hill Companies, 1999.
- [42] D. Knudson, *The Fundamentals of Biomechanics*. 2007.
- [43] A. Morecki, *Biomechanics of motion*. 1980.
- [44] Y. C. Fung, *Biomechanics*. 1990.

- [45] V. Polischuk, *Atletismo: iniciación y perfeccionamiento (3a. ed.)*. Ciudad de México, MEXICO: Editorial Paidotribo México, 2018.
- [46] J. Weineck, *Anatomía deportiva (5a. ed.)*. Barcelona, SPAIN: Editorial Paidotribo México, 2013.
- [47] Adesso, “Cámara web USB 480P con micrófono incorporado - Adesso Inc ::: Su especialista en dispositivos de entrada :::” <https://www.adesso.com/product/cybertrack-h2-480p-hd-usb-webcam-with-built-in-microphone/> (accessed Aug. 01, 2020).
- [48] Genius, “FaceCam-1000X | Genius ES Website.” <http://es.geniusnet.com/product/facecam-1000x/> (accessed Aug. 01, 2020).
- [49] Alibaba, “Full Hd 720p Webcam Usb Computer Camera Pc Digital Web Camera For Student Study Video Calling Web Camera Web Cam - Buy Webcam,Webcamera,Webcam Product on Alibaba.com.” https://www.alibaba.com/product-detail/Full-HD-720P-Webcam-USB-Computer_62569466625.html (accessed Aug. 01, 2020).
- [50] MTEC, “Cámaras – MTEC.” <https://www.mtec-ec.com/categoria-producto/camaras/> (accessed Aug. 01, 2020).
- [51] Proveedora Cano, “Sensor de proximidad fotoeléctrico Infrarrojo E18-D80NK.” <http://proveedoracano.com/eshop/sensor-e18> (accessed Jul. 30, 2020).
- [52] Datasheetpdf, “E18-D80NK Sensor/Switch Datasheet pdf - Proximity Sensor/Switch. Equivalent, Catalog,” 2014. <https://datasheetpdf.com/pdf/1311840/tinkbox/E18-D80NK/1> (accessed Jul. 30, 2020).
- [53] AliExpress, “Sensor de interruptor fotoeléctrico réflex de valoración E3F R2N12 DC NPN NO + NC diámetro 18mm Distancia 2 m transductor|distance switch|transducer sensor - AliExpress,” *AliExpress*. <https://es.aliexpress.com/item/32219483079.html> (accessed Jul. 30, 2020).
- [54] Naylamp mechatronics, “Sensor de proximidad fotoeléctrico reflex E3F-R2N12

- Naylamp Mechatronics - Perú.” <https://naylampmechatronics.com/sensores-proximidad/294-sensor-de-proximidad-fotoelectronico-reflex-e3f-r2n12.html> (accessed Jul. 30, 2020).
- [55] Web-Robótica, “Como usar el módulo sensor de Infrarrojos IR FC-51 para evitar obstáculos con Robot Arduino/Genuino - Web-Robótica.com.” <https://www.web-robotica.com/arduino/conceptos-basicos-arduino/como-usar-el-modulo-sensor-de-infrarrojos-ir-fc-51-para-evitar-obstaculos-con-robot-arduinogenuino> (accessed Jul. 30, 2020).
- [56] PiBorg, “PiBorg | Sensor de distancia ultrasónico (HC-SR04).” <https://www.piborg.org/sensors-1136/hc-sr04> (accessed Jul. 30, 2020).
- [57] DatasheetPDF, “28BYJ-48 Motor Datasheet pdf - Stepper Motor. Equivalent, Catalog.” <https://datasheetpdf.com/pdf/1006817/Kiatronics/28BYJ-48/1> (accessed Jul. 31, 2020).
- [58] Rambal, “Stepper motor Nema 34 88 kg-cm (8.6 N.m) 1.8°.” <https://rambal.com/stepper-paso-a-paso/1024-stepper-motor-nema-34-86-kg-cm.html> (accessed Jul. 31, 2020).
- [59] didacticaselectronicas, “Motores Paso a Paso: Motor paso a paso NEMA 34 DualShaft 5A 18kg/cm.” <https://www.didacticaselectronicas.com/index.php/elementos-electromecanicos/motores-y-solenoides-1/motores-paso-a-paso/motor-paso-a-paso-nema-34-dual-shaft-200-pasos-34hs46-5004d-step-step-stepper-detail> (accessed Jul. 31, 2020).
- [60] Web Robotica, “Motor de pasos 28BYJ-48 con driver ULN2003 y Arduino Uno - Web-Robótica.com.” <https://www.web-robotica.com/arduino/motor-de-pasos-28byj-48-con-driver-uln2003-y-arduino-uno> (accessed Jul. 31, 2020).
- [61] StepperOnline, “hoja de datos del motor paso a paso nema 17: Nema 17 Unipolar 1.8deg ...” <https://www.omc-stepperonline.com/nema-17-stepper-motor-datasheet/nema-17-unipolar-1-8deg-26ncm-37oz-in-0-4a-12v-42x42x39mm-6-wires.html> (accessed Jul. 31, 2020).
- [62] BricoGeek, “Motor paso a paso 14 Kg/cm - NEMA 23 Pololu 1475 |

- BricoGeek.com.” <https://tienda.bricogeek.com/motores-paso-a-paso/639-motor-paso-a-paso-14-kg-cm.html> (accessed Jul. 31, 2020).
- [63] Antony Garcia Gonzáles, “Arduino Mega: Características, Capacidades y donde conseguirlo en Panamá | Panama Hitek.” <http://panamahitek.com/arduino-mega-caracteristicas-capacidades-y-donde-conseguirlo-en-panama/> (accessed Jul. 31, 2020).
- [64] ABC Electronica, “Arduino LEONARDO - ABC Proyectos Electrónicos.” <https://www.abcelectronica.net/productos/microcontroladores/ma2/> (accessed Jul. 31, 2020).
- [65] Electronilab, “Arduino Nano V3 - ATmega328 5V + Cable USB Compatible - Electronilab.” <https://electronilab.co/tienda/arduino-nano-v3-atmega328-5v-cable-usb/> (accessed Jul. 31, 2020).
- [66] Patagoniatec, “>> Arduino Leonardo ATmega32u4 USB Nativo | PatagoniaTec.” <https://saber.patagoniatec.com/2014/12/arduino-leonardo-arduino-argentina-ptec/> (accessed Jul. 31, 2020).
- [67] Proyecto Arduino, “▷ Arduino UNO R3 Características, Especificaciones | Proyecto Arduino.” <https://proyectoarduino.com/arduino-uno-r3/> (accessed Jul. 31, 2020).
- [68] Joel Guerrero, “Arduino Uno: Especificaciones y características | PlusElectric,” 2014. <https://pluselectric.wordpress.com/2014/09/21/arduino-uno-especificaciones-y-caracteristicas/> (accessed Jul. 31, 2020).
- [69] Patagoniatec, “>>> LCD 16x2 -16x4 - 16x1- 20x4 Funcionan Igual | PatagoniaTec.” <https://saber.patagoniatec.com/2019/08/lcd-16x2-para-arduino/> (accessed Aug. 01, 2020).
- [70] OMRON, “Finales de carrera | Omron, España.” <https://industrial.omron.es/es/products/mechanical-sensors-limit-switches> (accessed Aug. 04, 2020).
- [71] La Electrónica, “Final de Carrera SPDT, Largo - Guatemala.” <https://laelectronica.com.gt/final-de-carrera-spdt-largo> (accessed Aug. 04,

2020).

- [72] Intel, “Especificaciones de productos Intel®.”
<https://ark.intel.com/content/www/es/es/ark.html#@PanelLabel122139>
(accessed Jul. 31, 2020).
- [73] AMD, “Ryzen™ PRO Desktop Processor | AMD.”
<https://www.amd.com/es/ryzen-pro> (accessed Jul. 31, 2020).
- [74] NVIDIA, “CUDA Zone | NVIDIA Developer.”
<https://developer.nvidia.com/cuda-zone> (accessed Aug. 07, 2020).
- [75] M. Gregoire, N. A. Solter, and S. J. Kleper, *Professional C++*. Hoboken, UNITED STATES: John Wiley & Sons, Incorporated, 2011.
- [76] Arduino, “Arduino - Home.” <https://www.arduino.cc/> (accessed Aug. 04, 2020).
- [77] D. Ronzani, “Introduction to Arduino,” 2017.
- [78] M. Peshev, “XAMPP and LAMP.”
- [79] Sublime Text, “Download - Sublime Text.” <https://www.sublimetext.com/3>
(accessed Aug. 06, 2020).
- [80] Python, “Python Releases for Windows | Python.org.”
<https://www.python.org/downloads/windows/> (accessed Aug. 06, 2020).
- [81] SourceForge, “OpenCV download | SourceForge.net.”
<https://sourceforge.net/projects/opencvlibrary/> (accessed Aug. 06, 2020).
- [82] OpenCV, “OpenCV: Cargar modelos de marco de Caffé.”
https://docs.opencv.org/3.4/d5/de7/tutorial_dnn_googlenet.html (accessed Aug. 07, 2020).
- [83] TensorFlow, “Compatibilidad con GPU | TensorFlow.”
<https://www.tensorflow.org/install/gpu> (accessed Aug. 05, 2020).
- [84] NVIDIA, “Download Drivers | NVIDIA.”
<https://www.nvidia.com/download/index.aspx?lang=en-us> (accessed Aug. 05,

2020).

- [85] OpenCV, “OpenCV: Deep Neural Network module.”
https://docs.opencv.org/master/d6/d0f/group__dnn.html (accessed Aug. 07, 2020).

- [86] NVIDIA, “Installation Guide Windows :: CUDA Toolkit Documentation.”
<https://docs.nvidia.com/cuda/archive/10.0/cuda-installation-guide-microsoft-windows/index.html> (accessed Aug. 07, 2020).

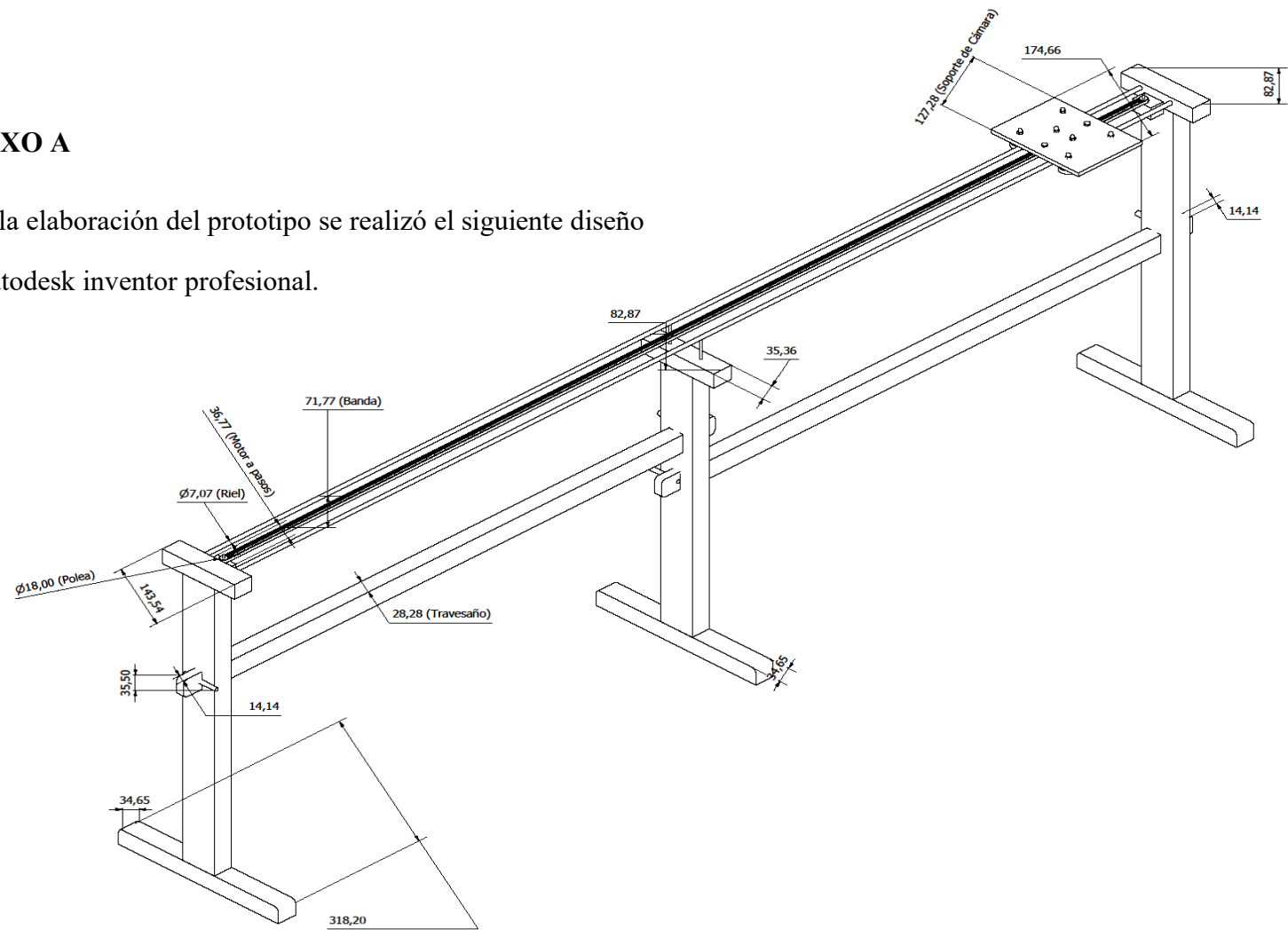
- [87] Dev47Apps, “Windows | Dev47Apps.”
<https://www.dev47apps.com/droidcam/windows/> (accessed Aug. 10, 2020).

- [88] Apache Friends, “XAMPP Installers and Downloads for Apache Friends.”
<https://www.apachefriends.org/es/index.html> (accessed Aug. 11, 2020).

ANEXOS

ANEXO A

Para la elaboración del prototipo se realizó el siguiente diseño en autodesk inventor profesional.



Título:

VISTA ARRIBA: "SISTEMA ELECTRÓNICO DE CORRECCIÓN DEL GESTO DEPORTIVO EN BASE A LA BIOMECÁNICA EN DEPORTISTAS DE MARCHA ATLÉTICA."

AUTOR:

LLAMBO YANSAGUANO FREDY BLADIMIR

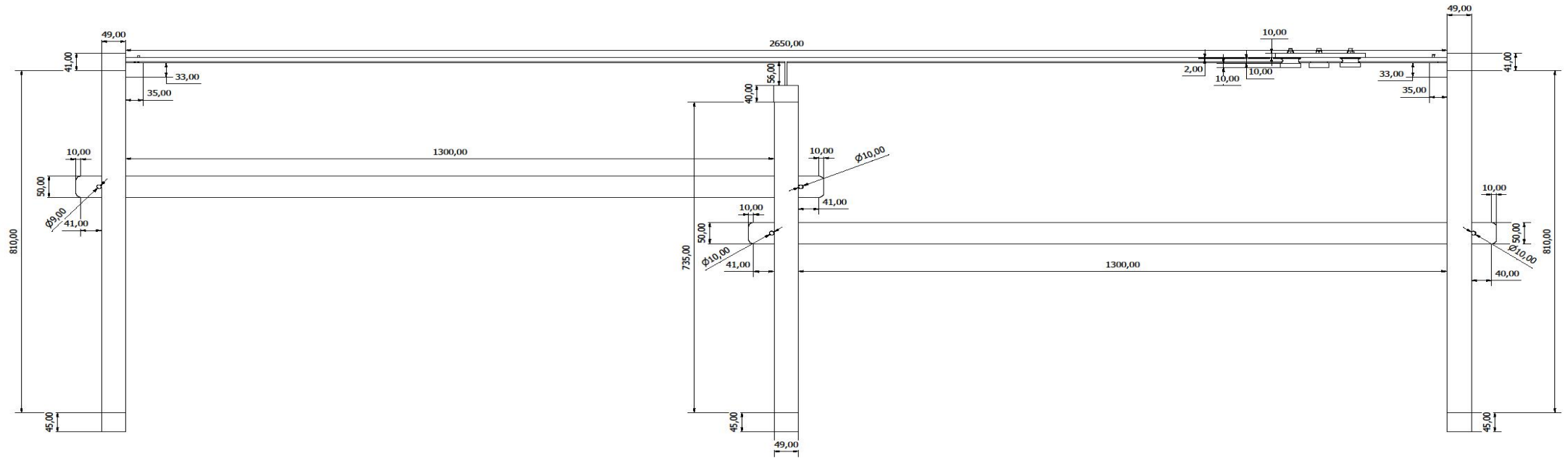
Escala:

1:4

Lamina:

1

Vista frontal del diseño del prototipo.



Título:

VISTA FRONTAL: "SISTEMA ELECTRÓNICO DE CORRECCIÓN DEL GESTO DEPORTIVO EN BASE A LA BIOMECÁNICA EN DEPORTISTAS DE MARCHA ATLÉTICA."

AUTOR:

LLAMBO YANSAGUANO FREDY BLADIMIR

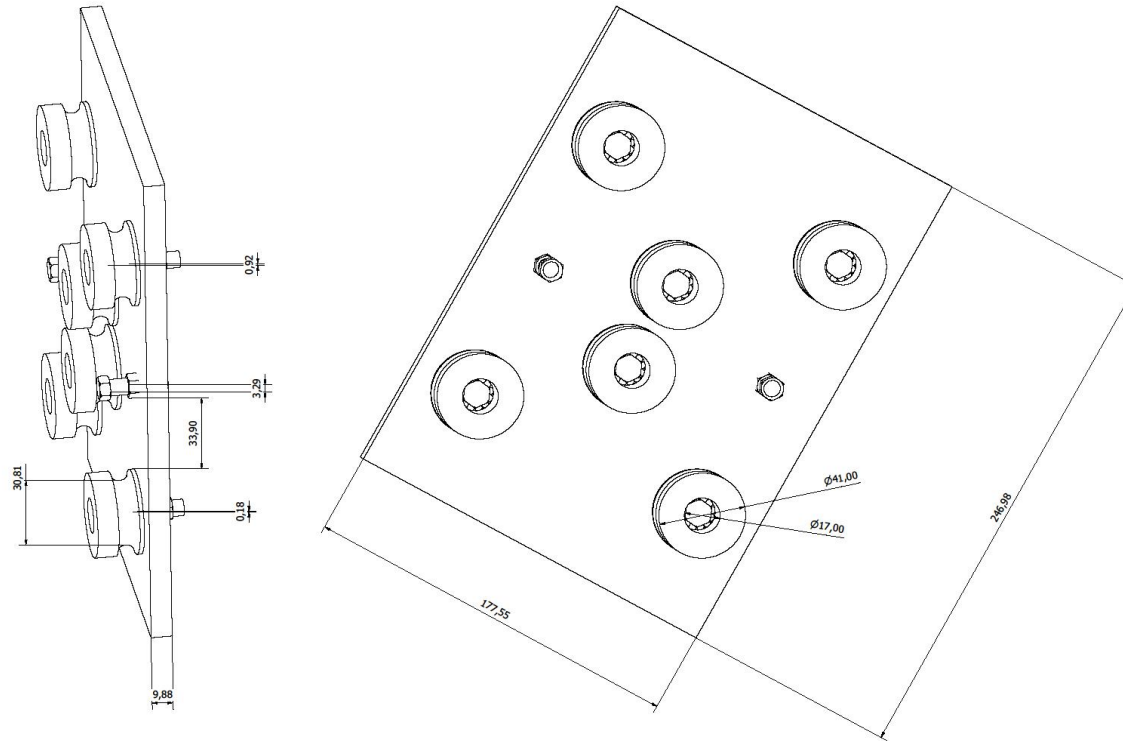
Escala:

1:4

Lamina:

2

Diseño del soporte de la cámara y poleas de acople al riel.



Título:

SOPORTE CON RODAMIENTO DE CÁMARA

AUTOR:

LLAMBO YANSAGUANO FREDY BLADIMIR

Escala:

1:2

Lamina:

3

ANEXO B

El código fuente del archivo cámara.py se muestra a continuación y se utilizó para la comprobación de la librería OpenCV

```
import cv2 #Librería de OpenCV

video=1 #Especificación de la cámara para usar, puede ir de 0 a 99. Ejemplo 0 es para
la cámara propia de la laptop, 1 una cámara extra.

camara = cv2.VideoCapture(video)#captura de video

while(camara.isOpened()):#Ingresa al ciclo while si existe alguna cámara

    ret, frame = camara.read()#Lee fotogramas presentes detectada por la cámara

    if ret == True: #Si existe el fotograma

        cv2.imshow("Capturando Video",frame)#presenta video a color

        if cv2.waitKey(1) & 0xFF == ord('q'):#Presiona la tecla q para salir

            break

    else:

        break

camara.release()#Cierra el archivo de video o el dispositivo de captura.

cv2.destroyAllWindows()#Cierra la ventana de captura.
```

ANEXO C

El código fuente que se indica a continuación llamada Estimación_humana.py se utilizó para el aprendizaje profundo de la recolección de imágenes sin CUDA.

```
import cv2 #importar OpenCV

import time #imodulo para tiempos

import numpy as np #extension para vectores

import imutils#libreria para procesamiento de imágenes

from imutils.video import FPS#libreria para obtener los fotogramas por segundo

import math#libreria para calculos

MODELO="BODY_25"#modelo para estimacion del cuerpo humano

if MODELO is "COCO":

    arquitectura = "pose/coco/pose_deploy_linevec.prototxt"

    pesos = "pose/coco/pose_iter_440000.caffemodel"

    puntos = 18#puntos clave del modelo

    uniones = [
[1,0],[1,2],[1,5],[2,3],[3,4],[5,6],[6,7],[1,8],[8,9],[9,10],[1,11],[11,12],[12,13],[0,14],[
0,15],[14,16],[15,17]]#enlace de puntos clave

elif MODELO is "MPI" :

    arquitectura = "pose/mpi/pose_deploy_linevec_faster_4_stages.prototxt"

    pesos = "pose/mpi/pose_iter_160000.caffemodel"

    puntos = 15

    uniones = [[0,1], [1,2], [2,3], [3,4], [1,5], [5,6], [6,7], [1,14], [14,8], [8,9], [9,10],
[14,11], [11,12], [12,13] ]

elif MODELO is "BODY_25" :

    arquitectura = "pose/body_25/pose_deploy.prototxt"

    pesos = "pose/body_25/pose_iter_584000.caffemodel"
```



```

puntos = 15

uniones = [[0,1], [1,2], [2,3], [3,4], [1,5], [5,6], [6,7], [1,8], [8,9], [8,12], [9,10],
[10,11], [12,13], [13,14] ]

ancho = 368#ancho para blob

alto = 368#alto para blob

limite = 0.1#variable para reducir detecciones falsas

fourcc = cv2.VideoWriter_fourcc(*'XVID')#formato de salida de video

salidavideo = cv2.VideoWriter('salida.avi', fourcc, 5.0, (640,480))#salida de video

#video = "sample_video.mp4"#en caso de observar un video con aprendizaje
automatico

video = 1#variable de la cámara a usar

#video= "salida.avi"#nombre del video de salida

camara = cv2.VideoCapture(video)#Captura de video

#creación de constructor rescala_cuadros

def rescala_cuadro(cuadros, porcentaje):

    anchura = int(cuadros.shape[1] * porcentaje/ 100)#cambiar anchura

    altura = int(cuadros.shape[0] * porcentaje/ 100)#cambiar altura

    dimension = (anchura, altura)#dimension ancho x alto

    return cv2.resize(cuadros, dimension, interpolation =cv2.INTER_AREA)#envía
del proceso una dimensión de cuadros modificado

#creación constructor procesos

def procesos(redneuronal, cuadros, altocuadro, anchocuadro):

    inpBlob = cv2.dnn.blobFromImage(cuadros, 1.0/500, (ancho, alto),(0, 0, 0),
swapRB=False, crop=False)#conversión de formato

    redneuronal.setInput(inpBlob)#configuracion del formato

    return redneuronal.forward()#envia la conversion procesado con puntos clave

redneuronal = cv2.dnn.readNetFromCaffe(arquitectura, pesos)#aplicación de red
neuronal a los modelos

```

```

fps = FPS().start()#fotograma por segundo

#condición de error de la cámara

if camara.isOpened() == False:

    print("Error de Cámara")#impresión del error de video

#condición de la detección de cámara

while camara.isOpened():

    t = time.time()#variable para el tiempo

    SNCuadros, cuadros = camara.read()#lectura de cámara

    cuadros = rescala_cuadro(cuadros, porcentaje=100)#llamada de constructor
rescala_cuadro

    copiacuadros = np.copy(cuadros)#copia de lectura de cámara

    if SNCuadros == True:#condición de fotograma

        anchocuadro = cuadros.shape[1]

        altocuadro = cuadros.shape[0]

        output = procesos(redneuronal, cuadros, altocuadro, anchocuadro)#salida de
aprendizaje automatico

        H = output.shape[2]

        W = output.shape[3]

        points = []#crea vector vacío para los puntos clave

        # condición para rango de puntos

        for i in range(puntos):

            probMap = output[0, i, :, :]

            minVal, prob, minLoc, point = cv2.minMaxLoc(probMap)

            x = (anchocuadro * point[0]) / W

            y = (altocuadro * point[1]) / H

            #condicion para reducir puntos detecciones falsas

```

```

    if prob > limite :

        cv2.circle(copiacuadros, (int(x), int(y)), 8, (0, 255, 255), thickness=-1,
lineType=cv2.FILLED)#bibuja circulo en el punto clave

        points.append((int(x), int(y)))

    else :

        points.append(None)

#condición para las uniones de los puntos

    for pares in uniones:

        partA = pares[0]

        partB = pares[1]

#condicion de la division de puntos

        if points[partA] and points[partB]:

            cv2.circle(cuadros, points[partA], 2, (255, 255, 255), thickness=-1,
lineType=cv2.FILLED)

            cv2.circle(cuadros, points[partB], 2, (0, 0, 255), thickness=-1,
lineType=cv2.FILLED)

            cv2.line(cuadros, points[partA], points[partB], (255, 0, 0), 2,
lineType=cv2.LINE_AA)

        salidavideo.write(cuadros)#escritura de salida video

        cv2.imshow('Biomecanica', cuadros)#Titulo de la pantalla de video

        fps.update()#actualiza fps

        if cv2.waitKey(1) & 0xFF == 27:#salida por Tecla 27-esc" código ascii"

            break

fps.stop()#finaliza fps

print("Tiempo: {:.2f}".format(fps.elapsed()))#imprime el tiempo que recorrio el
video

print("FPS: {:.2f}".format(fps.fps()))#imprime los fps del video

camara.release()#Elimina información de cámara

```

`salidavideo.release()`#Cierra el archivo de video o el dispositivo de captura.

`cv2.destroyAllWindows()`#Cierra la ventana de captura.

Anexo D

El código fuente que se indica a continuación llamada `Estimación_cuda.py` se utilizó para el aprendizaje profundo de la recolección de imágenes sin CUDA.

```
import cv2 #importar OpenCV

import time #imodulo para tiempos

import numpy as np #extension para vectores

import imutils#libreria para procesamiento de imágenes

from imutils.video import FPS#libreria para obtener los fotogramas por segundo

import math#libreria para calculos

MODELO="BODY_25"#modelo para estimacion del cuerpo humano

if MODELO is "COCO":

    arquitectura = "pose/coco/pose_deploy_linevec.prototxt"

    pesos = "pose/coco/pose_iter_440000.caffemodel"

    puntos = 18#puntos clave del modelo

    uniones = [
[1,0],[1,2],[1,5],[2,3],[3,4],[5,6],[6,7],[1,8],[8,9],[9,10],[1,11],[11,12],[12,13],[0,14],[
0,15],[14,16],[15,17]]#enlace de puntos clave

elif MODELO is "MPI" :

    arquitectura = "pose/mpi/pose_deploy_linevec_faster_4_stages.prototxt"

    pesos = "pose/mpi/pose_iter_160000.caffemodel"

    puntos = 15

    uniones = [[0,1], [1,2], [2,3], [3,4], [1,5], [5,6], [6,7], [1,14], [14,8], [8,9], [9,10],
[14,11], [11,12], [12,13] ]

elif MODELO is "BODY_25" :

    arquitectura = "pose/body_25/pose_deploy.prototxt"

    pesos = "pose/body_25/pose_iter_584000.caffemodel"
```

```

puntos = 15

uniones = [[0,1], [1,2], [2,3], [3,4], [1,5], [5,6], [6,7], [1,8], [8,9], [8,12], [9,10],
[10,11], [12,13], [13,14] ]

ancho = 368#ancho para blob

alto = 368#alto para blob

limite = 0.1#variable para reducir detecciones falsas

fourcc = cv2.VideoWriter_fourcc(*'XVID')#formato de salida de video

salidavideo = cv2.VideoWriter('salida.avi', fourcc, 5.0, (640,480))#salida de video

#video = "sample_video.mp4"#en caso de observar un video con aprendizaje
automatico

video = 1#variable de la cámara a usar

#video= "salida.avi"#nombre del video de salida

camara = cv2.VideoCapture(video)#Captura de video

#creación de constructor rescala_cuadros

def rescala_cuadro(cuadros, porcentaje):

    anchura = int(cuadros.shape[1] * porcentaje/ 100)#cambiar anchura

    altura = int(cuadros.shape[0] * porcentaje/ 100)#cambiar altura

    dimension = (anchura, altura)#dimension ancho x alto

    return cv2.resize(cuadros, dimension, interpolation =cv2.INTER_AREA)#envía
del proceso una dimensión de cuadros modificado

#creación constructor procesos

def procesos(redneuronal, cuadros, altocuadro, anchocuadro):

    inpBlob = cv2.dnn.blobFromImage(cuadros, 1.0/500, (ancho, alto),(0, 0, 0),
swapRB=False, crop=False)#conversión de formato

    redneuronal.setInput(inpBlob)#configuracion del formato

    return redneuronal.forward()#envia la conversion procesado con puntos clave

redneuronal = cv2.dnn.readNetFromCaffe(arquitectura, pesos)#aplicación de red
neuronal a los modelos

```

```

redneuronal.setPreferableBackend(cv2.dnn.DNN_BACKEND_CUDA)

redneuronal.setPreferableTarget(cv2.dnn.DNN_TARGET_CUDA)

fps = FPS().start()#fotograma por segundo

#condición de error de la cámara

if camara.isOpened() == False:

    print("Error de Cámara")#impresión del error de video

#condición de la detección de cámara

while camara.isOpened():

    t = time.time()#variable para el tiempo

    SNCuadros, cuadros = camara.read()#lectura de cámara

    cuadros = rescala_cuadro(cuadros, porcentaje=100)#llamada de constructor
rescala_cuadro

    copiacuadros = np.copy(cuadros)#copia de lectura de cámara

    if SNCuadros == True:#condificion de fotograma

        anchocuadro = cuadros.shape[1]

        altocuadro = cuadros.shape[0]

        output = procesos(redneuronal, cuadros, altocuadro, anchocuadro)#salida de
aprendizaje automatico

        H = output.shape[2]

        W = output.shape[3]

        points = []#crea vector vacío para los puntos clave

        # condición para rango de puntos

        for i in range(puntos):

            probMap = output[0, i, :, :]

            minVal, prob, minLoc, point = cv2.minMaxLoc(probMap)

            x = (anchocuadro * point[0]) / W

```

```

y = (altocuadro * point[1]) / H

#condicion para reducir puntos detecciones palsas

if prob > limite :

    cv2.circle(copiacuadros, (int(x), int(y)), 8, (0, 255, 255), thickness=-1,
lineType=cv2.FILLED)#bibuja circulo en el punto clave

    points.append((int(x), int(y)))

else :

    points.append(None)

#condición para las uniones de los puntos

for pares in uniones:

    partA = pares[0]

    partB = pares[1]

#condicion de la division de puntos

if points[partA] and points[partB]:

    cv2.circle(cuadros, points[partA], 2, (255, 255, 255), thickness=-1,
lineType=cv2.FILLED)

    cv2.circle(cuadros, points[partB], 2, (0, 0, 255), thickness=-1,
lineType=cv2.FILLED)

    cv2.line(cuadros, points[partA], points[partB], (255, 0, 0), 2,
lineType=cv2.LINE_AA)

    salidavideo.write(cuadros)#escritura de salida video

    cv2.imshow('Biomecnica', cuadros)#Titulo de la pantalla de video

    fps.update()#actualiza fps

    if cv2.waitKey(1) & 0xFF == 27:#salida por Tecla 27-esc" código ascii"

        break

fps.stop()#finaliza fps

print("Tiempo: {:.2f}".format(fps.elapsed()))#imprime el tiempo que recorrio el
video

```



```
print("FPS: {:.2f}".format(fps.fps()))#imprime los fps del video
```

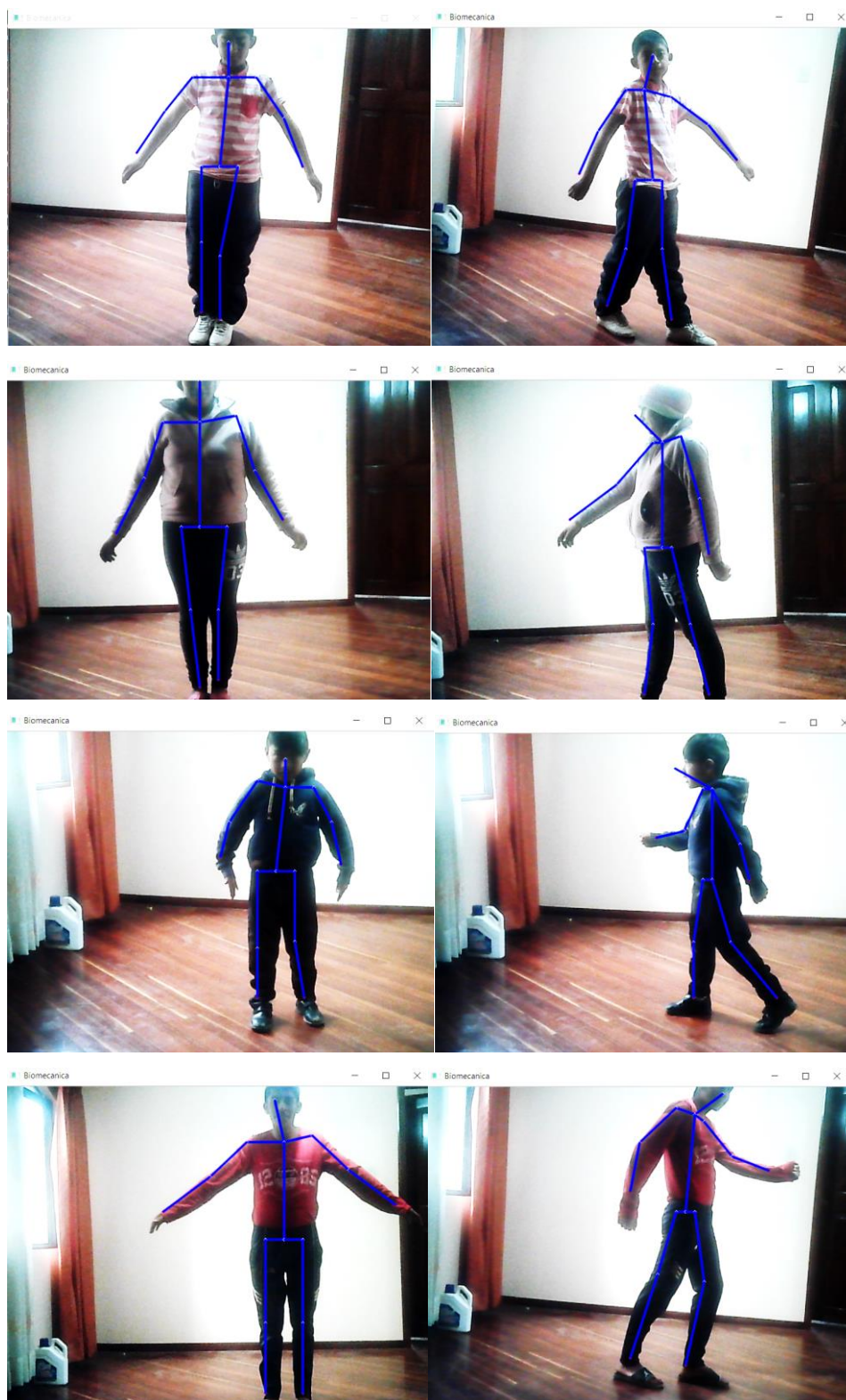
```
camara.release()#Elimina información de cámara
```

```
salidavideo.release()#Cierra el archivo de video o el dispositivo de captura.
```

```
cv2.destroyAllWindows()#Cierra la ventana de captura.
```

ANEXO E

A continuación, se muestra las pruebas de estimación con CUDA y el modelo BODY_25.



Se utilizó redes neuronales artificiales para procesar la imagen y determinar el punto clave, a continuación, se muestra los valores que se obtuvo al realizar la convolución entre la imagen y la matriz Kernel.

Salida RNA	FPS BODY25	Tiempo (s)	CPU (%)	GPU (%)
	3,38	18,36	12,49	0,43
<p>[[[-8.04024749e-05 1.12393405e-04 6.25308603e-05 ... 5.28693199e-05 6.52051531e-05 4.04748134e-05] [1.93053856e-05 2.44546682e-05 1.25668012e-05 ... 1.78511254e-05 4.33996320e-05 5.56283630e-05] [5.87897375e-05 2.58539803e-05 5.32274134e-05 ... 6.92578033e-05 5.51561825e-05 5.17894514e-05] ... [1.57013070e-04 2.38385983e-05 2.54875049e-05 ... 4.34764661e-05 5.59729524e-05 7.67763704e-05] [7.82338902e-05 4.11393121e-05 5.19757159e-05 ... 6.75315969e-05 7.64876604e-05 8.61259177e-05] [3.86647880e-05 9.40635800e-08 7.86273740e-05 ... 9.26218927e-05 8.54623504e-05 -2.92435288e-07]]</p> <p>[[4.86429635e-05 2.44645271e-05 9.04723129e-06 ... 1.13163405e-05 1.45413069e-05 5.94410267e-05] [-2.39353540e-06 -1.89415150e-05 -1.81772484e-05 ... -1.14291033e-05 -4.36538539e-06 7.04214472e-06] [4.05080209e-06 -1.71284482e-05 -7.02639227e-06 ... 6.52740709e-06 -3.29071190e-06 2.84914131e-06] ... [3.14074859e-05 -3.41549458e-06 6.13950397e-06 ... 5.43767237e-06 5.73526631e-06 9.00834129e-06] [2.00825307e-05 1.27167514e-05 7.17148214e-06 ... 4.32172237e-06 1.14474824e-05 1.74459492e-05] [6.50105794e-05 -6.78015931e-06 1.67135076e-05 ... 1.93712185e-06 9.61382466e-06 6.37871271e-05]]</p> <p>[[1.47088198e-04 -1.51434215e-05 -4.33756504e-05 ... -3.48172616e-05</p>				

5.56616578e-06 6.80190278e-05
 [4.63030301e-05 -3.76796816e-05 -2.89112795e-05 ... 5.65650407e-06
 4.27663326e-06 7.02054240e-06]
 [4.67882492e-05 -3.36992089e-05 -2.21664086e-05 ... 1.79061899e-05
 2.08347337e-05 9.46642831e-06]
 ...
 [[[-7.41914846e-05 1.13395974e-04 6.20200299e-05 ... 5.33103012e-05
 6.53062016e-05 3.65087762e-05]
 [2.75732018e-05 2.67201103e-05 1.26021914e-05 ... 1.77263282e-05
 4.21740115e-05 5.42802736e-05]
 [6.57220371e-05 2.59685330e-05 5.16576692e-05 ... 7.09006563e-05
 5.51338308e-05 5.13554551e-05]
 ...
 [1.52024440e-04 2.80519016e-05 2.52872705e-05 ... 4.17772681e-05
 5.50532714e-05 7.81556591e-05]
 [7.69016333e-05 4.21758741e-05 5.00599854e-05 ... 6.76405616e-05
 7.62580894e-05 8.73873942e-05]
 [2.68202275e-05 1.90613791e-05 8.01044516e-05 ... 9.57711600e-05
 8.66460614e-05 -1.85891986e-06]]
 [[4.98173395e-05 2.57528445e-05 9.22822073e-06 ... 1.12154594e-05
 1.41497221e-05 5.90010241e-05]
 [6.86821295e-07 -1.75568421e-05 -1.78124756e-05 ... -1.18182361e-05
 -5.18167508e-06 7.04199192e-06]
 [5.08647645e-06 -1.67418766e-05 -6.47421984e-06 ... 5.73189027e-06
 -4.19685966e-06 2.37058703e-06]
 ...
 [2.06276745e-05 -2.98941450e-06 5.54534927e-06 ... 5.51177800e-06
 5.97241888e-06 9.32021067e-06]
 [1.68758634e-05 6.52117888e-06 6.95142808e-06 ... 5.32995909e-06
 1.24741346e-05 1.79636336e-05]
 [5.82656685e-05 -6.64117397e-06 1.59894262e-05 ... 2.77043728e-06
 1.04490828e-05 6.46149856e-05]]

[[1.49191124e-04 -1.19477045e-05 -4.25156904e-05 ... -3.49034090e-05
4.61097807e-06 6.72636088e-05]

[4.90868697e-05 -3.47240129e-05 -2.80142995e-05 ... 4.35754191e-06
1.71607826e-06 4.19933349e-06]

[4.80160816e-05 -3.38961836e-05 -1.96610345e-05 ... 1.60842901e-05
1.82675431e-05 6.49690628e-06]

...

[8.36607069e-05 -2.45841220e-05 3.05741560e-06 ... 2.37831846e-05
4.81592724e-05 4.14098613e-05]

[1.17240939e-04 8.06455500e-06 3.08935996e-05 ... 4.01287107e-05
6.82801474e-05 6.23153755e-05]

[1.72686996e-04 3.13015189e-05 6.42859377e-05 ... 7.37619121e-05
8.02773284e-05 1.51970657e-04]]

...

[[3.32179363e-04 2.95883510e-04 4.18632932e-04 ... 2.49457429e-04
5.33782004e-04 3.54356482e-04]

[3.61860613e-04 4.74124652e-04 3.39821039e-04 ... 3.34526179e-04
7.27284234e-04 3.72613722e-04]

[3.87484382e-04 3.32557771e-04 -5.46546653e-06 ... -8.91644740e-05
3.89048946e-04 1.39007869e-04]

...

[2.85704562e-04 -6.37285411e-05 -1.95107772e-04 ... 2.60878005e-05
5.67537849e-04 3.25930945e-04]

[3.79355042e-05 1.58737530e-04 -5.99105842e-05 ... 1.27863837e-04
5.32186881e-04 5.56948711e-04]

[9.90204280e-04 2.01670977e-04 9.36883735e-05 ... 2.11895909e-04
3.81838006e-04 1.32693176e-03]]

[[-1.26260193e-03 1.33626178e-04 3.33333912e-04 ... 3.49317765e-04
1.99696617e-04 2.70575401e-04]

[6.70568552e-05 -9.63713683e-05 -1.13515111e-04 ... -1.28267144e-04
-9.81309859e-05 -3.41669889e-04]

[1.01597034e-04 -1.13711067e-04 3.03636662e-05 ... 1.49587315e-04

3.06632719e-04 2.86267859e-05]
...
[5.10049460e-04 1.14928931e-04 4.44647390e-04 ... 5.40340661e-05
1.62035896e-04 1.42270379e-04]
[4.64817043e-04 4.08401102e-04 4.36558534e-04 ... 2.04134340e-04
1.72803586e-04 1.24135229e-04]
[2.06756778e-03 3.82278376e-04 4.00692195e-04 ... 3.91432171e-04
1.01466058e-03 -6.13248092e-04]]

[[9.34295007e-04 3.78994329e-04 2.10291531e-04 ... 1.60654075e-04
2.33529863e-04 2.72592151e-04]
[-3.98536795e-05 1.08664972e-04 -1.00499194e-04 ... 2.28429853e-04
1.29214139e-04 1.27413310e-04]
[-2.50539044e-04 3.26856971e-05 -1.54019333e-04 ... 3.04163841e-04
2.70526594e-04 1.79752300e-04]
...
[8.53943056e-05 1.85971905e-04 -1.74613146e-04 ... 4.02072910e-05
9.55685973e-05 1.74910354e-04]
[-4.63246834e-05 2.49893084e-04 1.18769822e-04 ... 1.54152687e-04
2.56029976e-04 2.83278030e-04]
[-5.20544650e-04 -8.14875821e-05 -2.51324323e-04 ... -1.44657097e-05
-4.17486124e-04 6.15224242e-04]]]]

ANEXO F

Para el envío de los datos se utilizó el siguiente código fuente denominado datos_envio.py.

```
import pymysql #Librería para acceso a la base de datos

# Conectar con base de datos

conexion = pymysql.connect(host="localhost",

                            user="root",

                            passwd="",

                            database="tesis")

cursor = conexion.cursor()

Recuperar registros de la tabla 'datos'

registros = "SELECT * FROM datos;"

# Mostrar registros

cursor.execute(registros)

#Insertar

Lista=[1, 2, 3, 4]

a=Lista[0]

print(a)

"""registro = "INSERT INTO datos ('id', 'BrazoDerecho', 'BrazoIzquierdo',

'PiernaDerecha', 'PiernaIzquierda') VALUES (NULL, '1', '3', '2', '1');"

cursor.execute(registro)"""

registro = "INSERT INTO datos VALUES (NULL,

'{}','{}','{}','{}')".format(Lista[0],Lista[1],Lista[2],Lista[3])

cursor.execute(registro)

# Finalizar

conexion.commit()

conexion.close()
```

ANEXO G

El código fuente que se indica a continuación denominada Sistema_Biomecnica.py es toda la programación utilizada en el sistema desde la seguridad hasta el envío de datos a la base de datos.

```
#-----Importacion-Librerias-----  
  
import cv2  
  
import time  
  
import numpy as np  
  
import imutils  
  
from imutils.video import FPS  
  
import math  
  
#libreria para interfaz grafica  
  
import tkinter as tk  
  
from tkinter import *  
  
from tkinter import messagebox  
  
from tkinter import ttk  
  
from tkinter import filedialog  
  
#---Libreria BDD-----  
  
import pymysql  
  
#-----Libreria Comunicación Serial-----  
  
import serial, time  
  
import threading  
  
import webbrowser  
  
import os
```



```

#-----Control de Comunicación-----

global arduino

try:

    #-----Conexión con arduino-----

    arduino = serial.Serial('COM3', baudrate = 9600, timeout = 1)

    time.sleep(2)

except:

    print("Arduino/Puerto no conectado")

    #-----

#.....Modelos.....

MODELO="BODY_25"

if MODELO is "COCO":

    arquitectura = "pose/coco/pose_deploy_linevec.prototxt"

    pesos = "pose/coco/pose_iter_440000.caffemodel"

    puntos = 18

    uniones = [
[1,0],[1,2],[1,5],[2,3],[3,4],[5,6],[6,7],[1,8],[8,9],[9,10],[1,11],[11,12],[12,13],[0,1
4],[0,15],[14,16],[15,17]]

elif MODELO is "MPI" :

    arquitectura = "pose/mpi/pose_deploy_linevec_faster_4_stages.prototxt"

    pesos = "pose/mpi/pose_iter_160000.caffemodel"

    puntos = 15

    uniones = [[0,1], [1,2], [2,3], [3,4], [1,5], [5,6], [6,7], [1,14], [14,8], [8,9],
[9,10], [14,11], [11,12], [12,13] ]

```

```

elif MODELO is "BODY_25" :

    arquitectura = "pose/body_25/pose_deploy.prototxt"

    pesos = "pose/body_25/pose_iter_584000.caffemodel"

    puntos = 15

    uniones = [[0,1], [1,2], [2,3], [3,4], [1,5], [5,6], [6,7], [1,8], [8,9], [8,12],
[9,10], [10,11], [12,13], [13,14] ]

#---Parámetros de Ventana---

ancho = 368

alto = 368

limite = 0.1

#-----Dirección de almacenamiento de imágenes-----

dirVideo='D:\\Usuario\\Fredy\\Documentos\\Practica-Tesis\\Arduino-
Python\\SalidaVideos'

#-----Elección de cámara

video = 0

#-----Captura de video con OpenCV

camara = cv2.VideoCapture(video)

#----- Método para asignar puntos con ayuda de mouse-----

def mousePoints(event, x, y, flags, params):

    if event == cv2.EVENT_LBUTTONDOWN:

        size = len(puntosLista)

```

```

if size != 0 and size % 3 != 0:

    cv2.line(img, tuple(puntosLista[round((size-1)/3)*3]),(x,y),(0,0,255))

cv2.circle(img, (x,y), 5, (0,0,255), cv2.FILLED)

puntosLista.append([x,y])

#-----Método para cálculo de gradiente -----
def gradient(pt1, pt2):

    return(pt2[1]-pt1[1])/(pt2[0]-pt1[0])

#-----Cálculo del ángulo-----
def getAngle(puntosLista):

    pt1, pt2, pt3 = puntosLista[-3:]

    #print(pt1, pt2, pt3)

    m1 = gradient(pt1, pt2)

    m2 = gradient(pt1, pt3)

    angR = math.atan((m2-m1)/(1+m2*m1))

    angA = round(math.degrees(angR))

    #angA = round(180 - math.degrees(angR))

    cv2.putText(img, str(angA),(pt1[0]-20,pt1[1]-20),
cv2.FONT_HERSHEY_COMPLEX,

                1,(0,0,255), 2)

    lista.append(angA)

    print(lista)

    if len(lista)==4:

        registro = "INSERT INTO datos VALUES (NULL, '{}','{}','{}','{}',
current_timestamp())".format(lista[0],lista[1],lista[2],lista[3])

        cursor.execute(registro)

```

```

#-----Escala de los FPS del video-----

def rescala_cuadro(cuadros, porcentaje):

    anchura = int(cuadros.shape[1] * porcentaje/ 100)

    altura = int(cuadros.shape[0] * porcentaje/ 100)

    dimension = (anchura, altura)

    return cv2.resize(cuadros, dimension, interpolation =cv2.INTER_AREA)

#-----Proceso de aprendizaje automático.....

def procesos(redneuronal, cuadros, altocuadro, anchocuadro):

    inpBlob = cv2.dnn.blobFromImage(cuadros, 1.0/500, (ancho, alto),(0, 0, 0),
    swapRB=False, crop=False)

    redneuronal.setInput(inpBlob)

    return redneuronal.forward()

#-----Configuración a Tarjeta de Video-----

redneuronal = cv2.dnn.readNetFromCaffe(arquitectura, pesos)

redneuronal.setPreferableBackend(cv2.dnn.DNN_BACKEND_CUDA)

redneuronal.setPreferableTarget(cv2.dnn.DNN_TARGET_CUDA)

fps = FPS().start()

#.....Método interfaz (Menu).....

def menu():

    global ventana

    #variable para creación de interfaz

    ventana=Tk()

    ventana.title("Biomecánica del cuerpo humano")#Tema de la Ventana

```

```

ventana.iconbitmap("icono.ico")#Creación de Icono en Ventana

miframe=Frame()#variable para configurar parámetros de fondo

miframe.pack()

miframe.config(width="1000", height="750")#Dimension de frame

miframe.config(bg="black")#Color de frame

#-----Personalización de Frame-----

miImagen=PhotoImage(file="prototipo.png")

Label(miframe, image=miImagen).place(x=0,y=0)

milabel=Label(miframe, text="UNIVERSIDAD TÉCNICA DE
AMBATO")#Ingreso de texto en frame

milabel.place(x=375, y=0)#posición de texto

milabel.config(font=20)#tamaño de letra

milabel.config(bg=None)#fondo del label

milabel=Label(miframe, text="FACULTAD DE INGENIERÍA EN
SISTEMAS ELECTRÓNICA E INDUSTRIAL")#Ingreso de texto en frame

milabel.place(x=200, y=25)#posición de texto

milabel.config(font=20)#tamaño de letra

milabel.config(bg=None)#fondo del label

milabel=Label(miframe, text="CARRERA DE INGENIERÍA EN
ELECTRÓNICA Y COMUNICACIONES")#Ingreso de texto en frame

milabel.place(x=220, y=50)#posición de texto

milabel.config(font=20)#tamaño de letra

milabel.config(bg=None)#fondo del label

milabel=Label(miframe, text="Proyecto de graduación. Modalidad: Proyecto
de Investigación")#Ingreso de texto en frame

milabel.place(x=275, y=150)#posición de texto

```

```

milabel.config(font=20)#tamaño de letra

milabel.config(bg=None)#fondo del label

milabel=Label(miiframe, text="Autor: Fredy Llambo")#Ingreso de texto en
frame

milabel.place(x=450, y=175)#posición de texto

milabel.config(font=20)#tamaño de letra

milabel.config(bg=None)#fondo del label

milabel=Label(miiframe, text="Tutor: Dr. Jaime Guilcapi")#Ingreso de texto en
frame

milabel.place(x=450, y=200)#posición de texto

milabel.config(font=20)#tamaño de letra

milabel.config(bg=None)#fondo del label

milabel=Label(miiframe, text="TEMA: CONTROL PANEL DEL SISTEMA
ELECTRÓNICO DE LA BIOMECÁNICA EN MARCHA
ATLÉTICA")#Ingreso de texto en frame

milabel.place(x=50, y=275)#posición de texto

milabel.config(font=20)#tamaño de letra

milabel.config(bg=None)#fondo del label

#-----Creación de Botones-----

usuariolabel=Label(miiframe, text="Usuario")

usuariolabel.place(x=300,y=440)

usuariolabel.config(font=("bold",10))

usuariolabel=Label(miiframe, text="Contraseña")

usuariolabel.place(x=277,y=470)

usuariolabel.config(font=("bold",10))

```

```

#-----usuario y Contraseña-----

global usuario

global contraseña

usuario=Entry(miframe, text="Usuario")

usuario.place(x=350,y=440)

usuario.config(font=("bold",10))

contraseña=Entry(miframe, text="Contraseña")

contraseña.place(x=350,y=470)

contraseña.config(font=("bold",10), show="*")

#-----Start-----

inicio=Button(miframe, text="Iniciar Software", command=verificar)

inicio.place(x=350,y=540)

inicio.config(cursor="hand2", bg="white", font=("bold",10), border=10,
relief="raised")# Tipo de cursor

#-----Usuario-----Salir-----

salir=Button(miframe, text="Cerrar Programa", command=ventana.destroy)

salir.place(x=350,y=600)

salir.config(cursor="hand2", bg="white", font=("bold",10), border=10,
relief="raised")# Tipo de cursor

#-----Condicionamiento-----

ventana.mainloop()#cierre de la interfaz

def verificar():

    if (usuario.get()==" and contraseña.get()==""):

```

```

principal()

else:

    messagebox.showwarning("Acceso Denegado", "Por favor verifique sus
datos")

def boton_activar():

    valor1=str(1)

    arduino.write((valor1+'\n').encode())

    camaravideo()

def boton_parar():

    arduino.write(b"3")

def boton_reiniciar():

    arduino.write(b"2")

def angulos1(int):

    angulo1=str(servomotor1.get())

    arduino.write((angulo1+'\n').encode())

def angulos2(int):

    angulo2=str(servomotor2.get())

    arduino.write((angulo2+'\n').encode())

def software():

    messagebox.showinfo("Proyecto de Titulación", " Biomecánica marcha V
1.0 \n")

```



```

"    Autor: Fredy LLambo \n"
"    contacto: fllambo577@gmail.com\n"
"    Tutor: Dr. Jaime Guilcapi \n"
"    UTA-FISEI\n"
"    Ambato-Ecuador\n"
"    2020")

```

```
def sugerencia_utilizacion():
```

```
    messagebox.showinfo("Utilización del Software", "Sugerencias\n"
```

```
        "1. Para iniciar con el programa primero asegure
de calibrar la posición de la cámara para la captación correcta del video.\n"
```

```
        "2. Presione iniciar programa e indique al
practicante que ya puede iniciar la práctica.\n"
```

```
        "3. En caso de que falle el Sistema Operativo
presionar el paro de emergencia del sistema y reiniciar todo.\n"
```

```
        "4. En caso de observar fallos en el riel en el
HMI presionar parar sistema o reiniciar.\n"
```

```
        "5. Para el análisis de video debe escoger el
video que va analizar y enviar los datos para evaluación de los resultados.")
```

```
def principal():
```

```
    ventana.destroy()
```

```
    global ventana1
```

```
    ventana1 = Tk()
```

```
    ventana1.title("HMI Biomecánica del cuerpo humano")#Tema de la Ventana
```

```
    ventana1.iconbitmap("icono.ico")#Creación de Icono en Ventana
```

```
    ventana1.geometry("400x300")#Dimensión de frame
```

```
    ventana1.config(bg="blue")
```

```

ventana1.resizable(width= False, height= False)

#Estilo de pestañas

style = ttk.Style()

settings = {"TNotebook.Tab": {"configure": {"padding": [5, 1], "background":
"white"},

                "map": {"background": [("selected", "#B1E7F4"),("active",
"#B1E7F4")]} }}

style.theme_create("mi_estilo", parent="alt", settings=settings)

style.theme_use("mi_estilo")

notebook = ttk.Notebook(ventana1)

notebook.pack(fill="both", expand="yes")

pestaña1=ttk.Frame(notebook)

pestaña2=ttk.Frame(notebook)

pestaña3=Menu(ventana1)

menuAyuda=Menu(pestaña3)

menuAyuda.add_command(label="Sobre Software", command=software)

menuAyuda.add_command(label="Como Funciona")

notebook.add(pestaña1, text="Calibración Cámara")

notebook.add(pestaña2, text="Panel de Control")

pestaña3.add_cascade(label="Ayuda", menu=menuAyuda,
command=sugerencia_utilizacion)

ventana1.config(menu=pestaña3)

#-----Pestaña 1-----

global servomotor1

global servomotor2

servomotor1=Scale(pestaña1, label="Vertical", orient=VERTICAL, from_=4,
to=180, command=angulos1)

```

```

servomotor1.pack()

servomotor11=Scale(from_=1, to=1, command=boton_activar)

servomotor1.pack()

servomotor2=Scale(pestaña1, label="Horizontal", orient=HORIZONTAL,
from_=181, to=360, command=angulos2)

servomotor2.pack()

#-----Pestaña 2-----

activar=Button(pestaña2, text="Iniciar Programa", command=boton_activar)

activar.place(x=40,y=100)

parar=Button(pestaña2, text="Parar Programa", command=boton_parar)

parar.place(x=150,y=100)

reiniciar=Button(pestaña2, text="Reiniciar Programa",
command=boton_reiniciar)

reiniciar.place(x=250,y=100)

basedatos=Button(pestaña2, text="Base de Datos", command=base_datos)

basedatos.place(x=150,y=50)

#####Captura de video#####

def camaravideo():

    fourcc = cv2.VideoWriter_fourcc(*'XVID')

    salidavideo = cv2.VideoWriter(os.path.join(dirVideo, 'salida.avi'), fourcc, 5.0,
(640,480))

    if camara.isOpened() == False:

        print("Error de Camara")

    while camara.isOpened():

        t = time.time()

```

```

SNCuadros, cuadros = camara.read()

cuadros = rescala_cuadro(cuadros, porcentaje=100)

copiacuadros = np.copy(cuadros)

if SNCuadros == True:

    anchocuadro = cuadros.shape[1]

    altocuadro = cuadros.shape[0]

    output = procesos(redneuronal, cuadros, altocuadro, anchocuadro)

    H = output.shape[2]

    W = output.shape[3]

    points = []

    for i in range(puntos):

        probMap = output[0, i, :, :]

        minVal, prob, minLoc, point = cv2.minMaxLoc(probMap)

        # Scale the point to fit on the original image

        x = (anchocuadro * point[0]) / W

        y = (altocuadro * point[1]) / H

        if prob > limite :

            cv2.circle(copiacuadros, (int(x), int(y)), 8, (0, 255, 255), thickness=-
1, lineType=cv2.FILLED)

            # Añade puntos en la imagen

            points.append((int(x), int(y)))

        else :

            points.append(None)

    # Dibuja líneas uniendo puntos

```

```

for pares in uniones:

    partA = pares[0]

    partB = pares[1]

    if points[partA] and points[partB]:

        cv2.circle(cuadros, points[partA], 2, (255, 255, 255), thickness=-1,
lineType=cv2.FILLED)

        cv2.circle(cuadros, points[partB], 2, (0, 0, 255), thickness=-1,
lineType=cv2.FILLED)

        cv2.line(cuadros, points[partA], points[partB], (255, 0, 0), 2,
lineType=cv2.LINE_AA)

    salidavideo.write(cuadros)

    cv2.imshow('Biomecanica', cuadros)

    fps.update()

    if cv2.waitKey(1) & 0xFF == 27:

        break

fps.stop()

print("Tiempo: {:.2f}".format(fps.elapsed()))

print("FPS: {:.2f}".format(fps.fps()))

camara.release()#Elimina información de captura

salidavideo.release()#Cierra el archivo de video o el dispositivo de captura.

cv2.destroyAllWindows()

#-----

def datos_ingreso():

    global puntosLista

```

```

global lista

puntosLista = []

lista=[]

global conexion

global cursor

try:

    #-----Conexión con base de datos-----

    conexion = pymysql.connect(host="localhost",

                                user="root",

                                passwd="",

                                database="tesis")

    cursor = conexion.cursor()

    #-----

except:

    print("Comunicación con MYSQL no establecida")

global img

archivo = filedialog.askopenfilename(title="Abrir", initialdir=dirVideo,
filetypes=(("Archivos PNG",

            "*.png"), ("Archivos JPG", "*.jpg")))

img=cv2.imread(archivo)

while True:

    if len(puntosLista) % 3 == 0 and len(puntosLista) != 0:

        getAngle(puntosLista)

    # Finalizar

    cv2.imshow('Video', img)

```

```

cv2.setMouseCallback('Video', mousePoints)

if cv2.waitKey(0) & 0xFF == 27:
    break

conexion.commit()

conexion.close()

cv2.destroyAllWindows()

def datos_borrar():
    global conexion

    global cursor

    try:
        #-----Conexión con base de datos-----

        conexion = pymysql.connect(host="localhost",
                                    user="root",
                                    passwd="",
                                    database="tesis")

        cursor = conexion.cursor()

        #-----

    except:
        print("Comunicación con MYSQL no establecida")

        cursor.execute("TRUNCATE TABLE datos")

        conexion.close()

def datos_actualizar():
    global img

```

```

    archivo = filedialog.askopenfilename(title="Abrir", initialdir=dirVideo,
filetypes=(("Archivos PNG",
        "*.png"), ("Archivos JPG", "*.jpg")))

img=cv2.imread(archivo)

while True:

    if len(puntosLista) % 3 == 0 and len(puntosLista) != 0:

        getAngle(puntosLista)

        # Finalizar

        cv2.imshow('Video', img)

        cv2.setMouseCallback('Video', mousePoints)

        if cv2.waitKey(0) & 0xFF == 27:

            break

conexion.commit()

conexion.close()

cv2.destroyAllWindows()

def base_datos():

    vid='D:\\Usuario\\Fredy\\Documentos\\Practica-Tesis\\Arduino-
Python\\SalidaVideos\\salida.avi'

    count=0

    videoC = cv2.VideoCapture(vid)

    while videoC.isOpened():

        ret, frame = videoC.read()

        if ret == True:

            cv2.imshow("Capturando Video",frame)

            key = cv2.waitKey(0)

```



```

if key == ord('q'):
    break

#Etapa de colección de imágenes al presionar la tecla a
elif key == ord('a'):
    count=count + 1

    #crea imagines con diferentes nombres la cantidad que desee el
    #usuario

    cv2.imwrite(os.path.join(dirVideo ,
'Imagen_{}.jpg'.format(count)),frame)

    print('Imagen_{}.jpg'.format(count))

    print(count)

else:
    break

videoC.release()

cv2.destroyAllWindows()

ventana1.destroy()

global ventana2

ventana2 = Tk()

ventana2.title("Base de datos de Biomecánica")#Tema de la Ventana

ventana2.iconbitmap("icono.ico")#Creación de Icono en Ventana

ventana2.geometry("400x300")#Dimensión de frame

ventana2.config(bg="white")

ventana2.resizable(width= False, height= False)

ingresodatos=Button(ventana2, text="Ingresar Datos",
command=datos_ingreso)

ingresodatos.place(x=150,y=70)

```

```
eliminar_datos=Button(ventana2, text="Eliminar Datos",  
command=datos_borrar)
```

```
eliminar_datos.place(x=150,y=140)
```

```
actualizar_datos=Button(ventana2, text="Actualizar Datos",  
command=datos_actualizar)
```

```
actualizar_datos.place(x=150,y=210)
```

```
menu()
```

ANEXO H

El código fuente para inicio del sistema llamado Sistema_biomecanica.ino se presenta a continuación.

```
//libreria de servomotor

#include <Servo.h>

//libreria de motor a pasos nema 17

#include <AccelStepper.h>

//declaracion de pines de control del driver del motor a pasos

#define dirPin 4

#define stepPin 5

#define motorInterfaceType 1

const int enable = 6;

//declaración de los pines del sensor ultrasonico

const int PinTrig = 9;

const int PinEcho = 10;

long tiempo;

int distancia;

//declaración de la variable de la comunicación serial

int dato = 0;

// Declaración de los pines de salida de los servos motores

Servo servo_1;

Servo servo_2;
```

```

//Declaración de los finales de Carrera

int finalcarrera = 7;

int finalcarrera = 8;

//Variable del movimiento del motor a pasos

AccelStepper stepper = AccelStepper(motorInterfaceType, stepPin, dirPin);

//acondicionamiento de la comunicación serial

const int reinit=0;

void setup()
{
  Serial.begin(9600);

  stepper.setMaxSpeed(1000);//Configuración máxima de velocidad de motor

  pinMode(PinTrig, OUTPUT); // Pin Trig configuro como salida

  pinMode(PinEcho, INPUT); // Pin echi configuro como entrada

  pinMode(enable, OUTPUT);

  digitalWrite(enable, HIGH);

  pinMode(finalcarrera, INPUT);

  servo_1.attach(11); // Pines de señal para el servo

  servo_2.attach(12);
}

void loop()
{
  //Metodo de la comunicacion serial

  if (Serial.available() > 0)
  {

```

```

int dato = Serial.parseInt();//repcion de dato enviado

//condiciones de movimiento de los servo motores

if (dato >= 4 & dato <= 180) {

    servo_1.write(dato);

}

if (dato >= 181 & dato <= 360) {

    int motor1 = map(dato, 180, 360, 1, 180);

    servo_2.write(motor1);

}

//Inicio de programa

if (dato == 1)

{

```

der:

```

int b = 0;

while (b == 0) {

    digitalWrite(PinTrig, LOW);

    delayMicroseconds(2);

    digitalWrite(PinTrig, HIGH);

    delayMicroseconds(10);

    digitalWrite(PinTrig, LOW);

    tiempo = pulseIn(PinEcho, HIGH);

    distancia = tiempo * 0.034 / 2;

    if (distancia > 3 && distancia < 5) {

        int d = 0;

        digitalWrite(enable, LOW);

```



```

stepper.setSpeed(-600);//Configuracion rpm
stepper.runSpeed();//Configuracion de velocidad constante

int dato = Serial.read();

if (dato == '1' || dato== 1) {

    goto der;

}

if (dato == '3') {

    goto paro;

}

}

}

if (dato == '3')

{

paro:

    digitalWrite(enable, HIGH);

    Serial.println(enable);

}

}

}

```

ANEXO I

