



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y REDES DE COMUNICACIÓN

**“SISTEMA DE DETECCIÓN DEL ÁNGULO ARTICULAR EN LOS
MOVIMIENTOS DE MIEMBRO SUPERIOR PARA EVALUACION EN
FISIOTERAPIA MEDIANTE VISIÓN ARTIFICIAL”**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERA EN ELECTRÓNICA Y REDES DE COMUNICACIÓN**

AUTOR: JOSÉ LUIS JARRÍN CHACÓN

DIRECTOR: MSC. JAIME ROBERTO MICHILENA CALDERÓN

ASESOR: MSC. EDGAR ALBERTO MAYA OLALLA

ASESOR: MSC DANIELA ALEXANDRA ZURITA PINTO

IBARRA-ECUADOR

2020



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD
TÉCNICA DEL NORTE

IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DEL CONTACTO	
Cédula de identidad	1718898362
Apellidos y nombres	Jarrín Chacón José Luis
Dirección	Cayambe, Parroquia. Ayora, Calle Unopac N3-18
E-mail	jljarrin@utn.edu.ec
Teléfono móvil	0939022835
DATOS DE LA OBRA	
Título	SISTEMA DE DETECCIÓN DEL ÁNGULO ARTICULAR EN LOS MOVIMIENTOS DE MIEMBRO SUPERIOR PARA EVALUACIÓN EN FISIOTERAPIA MEDIANTE VISIÓN ARTIFICIAL.
Autor	José Luis Jarrín Chacón
Fecha	27 de Febrero del 2020
Programa	Pregrado
Título	Ingeniero en Electrónica y Redes de Comunicación
Director	Ing. Jaime Michilena Calderón, MSc

CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 27 días del mes de febrero de 2020

EL AUTOR:



José Luis Jarrín Chacón
CI:171889836-2



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

MSc. JAIME ROBERTO MICHILENA CALDERÓN, DIRECTOR DEL PRESENTE
TRABAJO DE TITULACIÓN CERTIFICA:

Que, el presente trabajo de titulación "SISTEMA DE DETECCIÓN DEL ÁNGULO
ARTICULAR EN LOS MOVIMIENTOS DE MIEMBRO SUPERIOR PARA
EVALUACIÓN EN FISIOTERAPIA MEDIANTE VISIÓN ARTIFICIAL.", fue realizado en
su totalidad por el Sr. José Luis Jarrin Chacón, bajo mi supervisión.

Es todo en cuanto puedo certificar en honor a la verdad.



MSc. Jaime Michilena
DIRECTOR

DEDICATORIA

Dedicado a mis abuelitos: Julio, José Ignacio y María; que han sido fuente de inspiración y guía en cada etapa de mi vida, su recuerdo permanece intacto en mi corazón por siempre.

A mis padres y hermanos que siempre estuvieron para apoyarme incondicionalmente y darme su amor.

AGRADECIMIENTOS

Agradezco a mis padres Vicente Jarrín y Lucía Chacón por todo su apoyo y sacrificio, por inculcarme valores y demostrarme todo el amor del mundo, siempre estaré agradecido por darme su mano en los momentos más difíciles de mi vida y ser parte de cada logro, por su paciencia y brindarme las posibilidades que han estado a su alcance para siempre hacerme feliz. De igual manera a mis hermanos David, Paola, Darío, Silvia, Rosario, a mis tíos Germán, Elizabeth y Verónica quienes me dieron su apoyo incondicional y sobretodo su cariño tanto en los buenos como en los malos momentos de cada una de mis etapas.

A mis siempre recordados abuelitos que están en el cielo Julio, José Ignacio y María, a mi querida abuelita Zoila que me sigue acompañando y ha estado pendiente de mí con su amor de madre siempre estaré agradecido por todo.

A mi director de tesis Ingeniero Jaime Michilena por no solo ser una guía académica sino un amigo que gracias a sus consejos, orientación e incondicional disponibilidad me permitió crecer como persona y cumplir con el desarrollo de mi proyecto de titulación.

A mis amigos y compañeros que estuvieron a mi lado estos años compartiendo experiencias, tristezas y alegrías, por su amistad y consejos para conseguir la meta que nos propusimos desde el primer día en nuestra querida Universidad.

ÍNDICE

IDENTIFICACIÓN DE LA OBRA.....	I
CONSTANCIAS.....	II
CERTIFICACIÓN.....	III
DEDICATORIA.....	IV
AGRADECIMIENTOS.....	V
ÍNDICE.....	VI
ÍNDICE DE FIGURAS.....	IX
ÍNDICE DE ECUACIONES.....	XII
Resumen.....	XIII
Abstract.....	XIV
Capítulo 1. Antecedentes.....	1
1.1. Tema.....	1
1.2. Problema.....	1
1.3. Objetivos.....	3
1.3.1. Objetivo general.....	3
1.3.2. Objetivos específicos.....	3
1.4. Alcance.....	3
1.5. Justificación.....	5
Capítulo 2. Fundamentación Teórica.....	7
2.1. Fisioterapia.....	7
2.1.1. Biomecánica.....	7
2.1.1.1. Biomecánica del codo.....	8
2.1.1.2. Biomecánica del hombro.....	9
2.1.2. Goniometría.....	12
2.1.3. Objetivos de la Goniometría en terapia física.....	12
2.1.4. Instrumentos de Medición.....	13
2.1.4.1. Goniómetro.....	13
2.1.4.2. Inclímetro.....	14
2.1.4.3. Electrogoniómetro.....	15
2.1.5. Goniometría de hombro y codo.....	16
2.1.5.1. Abducción y aducción de hombro.....	16
2.1.5.2. Flexo-extensión de codo.....	18
2.1.6. Métodos de medición.....	19
2.1.7. Registro de resultados.....	20
2.2. Inteligencia Artificial.....	20

2.2.1. Visión artificial	21
2.3. Aprendizaje Automático	23
2.3.1. Métodos de aprendizaje automático.....	23
2.3.1.1. Aprendizaje supervisado	23
2.3.1.2. Aprendizaje no supervisado	26
2.3.1.3. Aprendizaje reforzado	28
2.3.1.4. Aprendizaje profundo.....	28
2.4. Redes neuronales.....	29
2.4.1. Redes neuronales profundas	29
2.4.1.1. Redes neuronales recurrentes (RNN).....	31
2.4.1.2. Redes neuronales convolucionales (CNN's).....	32
2.5. Sistemas de estimación de posición de articulaciones	35
2.5.1. OpenPose	36
2.5.2. Vnect.....	36
2.5.3. AlphaPose	37
2.6. Software Libre.....	37
2.6.1. Phyton.....	37
2.6.2. OpenCv.....	38
2.6.3. Tensorflow.....	38
2.7. Estándar IEEE 29148.....	39
Capítulo 3. Diseño del Sistema.....	44
3.1. Descripción general del sistema.....	44
3.1.1. Alcance del sistema.....	44
3.1.2. Limitaciones del sistema.....	44
3.2. Situación Actual	44
3.3. Metodología	47
3.4. Desarrollo	48
3.4.1. Propósito del sistema	48
3.4.2. Ámbito del sistema	49
3.5. Requerimientos del sistema.....	49
3.6. Requerimientos iniciales del sistema	50
3.7. Requerimientos de arquitectura.....	53
3.8. Requerimientos de stakeholders.....	55
3.9. Selección de hardware y software.....	57
3.9.1. Hardware.....	57
3.9.1.1. CPU.....	57
3.9.1.2. GPU.....	58

3.9.1.3. Cámara	60
3.9.2. Software	60
3.9.2.1. Software de programación	61
3.10. Diseño del Sistema	61
3.10.1. Arquitectura del Sistema.....	62
3.10.2. Diagrama de Bloques	64
3.10.2.1. Bloque1. Recolección de datos	65
3.10.2.2. Bloque 2. Procesamiento.....	66
3.10.2.3. Bloque 3. Datos de Salida	86
3.11. Pruebas de integración	92
3.11.1. Pruebas de registro extracción de datos	92
3.12. Diagramas de Flujo	94
3.12.1. Obtención de imágenes	95
3.12.2. Procesamiento	96
3.12.3. Salida de datos	97
3.13. Esquema del sistema	98
Capítulo 4. Pruebas de Funcionamiento y Resultados.....	100
4.1. Pruebas de funcionamiento del sistema	100
4.1.1. Pruebas por movimiento	101
4.2. Evaluación de la flexo-extensión de codo.....	102
4.2.1. Resultados obtenidos de la flexo-extensión.....	105
4.3. Evaluación de la abducción y aducción de hombro	106
4.3.1. Resultados obtenidos de la abducción y aducción.....	109
4.4. Validación del sistema	110
4.4.1. Prueba de hipótesis	110
4.4.2. Índice de Fallas	113
Conclusiones	116
Recomendaciones	117
Referencias.....	119
Glosario de Términos y Acrónimos.....	124
ANEXOS	126
Anexo A. Script de programación para el inicio del sistema.....	126
Anexo B. Programación Calculo de Ángulos Articulares.....	127
Anexo D. Elaboración del formulario para la base de datos del sistema	132
Anexo E. Pacientes del Centro de Rehabilitación UTN.....	138
Anexo F. Datos validados por el experto	150
Anexo G. Certificación de pruebas de funcionamiento	152

ÍNDICE DE FIGURAS

<i>Figura 1.</i> Movimientos de Supinación y Pronación del codo	9
<i>Figura 2.</i> Flexión y extensión de codo	9
<i>Figura 3.</i> Movimientos del plano sagital del hombro.....	10
<i>Figura 4.</i> Movimientos del Plano Frontal del Hombro	11
<i>Figura 5.</i> Movimientos del plano horizontal del hombro.....	11
<i>Figura 6.</i> Uso del Goniómetro en la articulación del codo.....	14
<i>Figura 7.</i> Uso del inclinómetro.....	15
<i>Figura 8.</i> Manejo del inclinómetro	15
<i>Figura 9.</i> Abducción de hombro en la plantilla frontal	17
<i>Figura 10.</i> Angulo máximo del movimiento de abducción.....	17
<i>Figura 11.</i> Aducción de hombro.....	18
<i>Figura 12.</i> Medición de la Flexo-extensión de codo en el plano anatómico medial ...	19
<i>Figura 13.</i> Relación entre la inteligencia artificial y la visión artificial	21
<i>Figura 14.</i> Etapas de fundamentos de un sistema de visión artificial	23
<i>Figura 15.</i> Proceso de la capa convolucional	33
<i>Figura 16.</i> Red neuronal convolucional con múltiples capas agrupadas.....	35
<i>Figura 17.</i> Estadística total de usuarios atendidos en el C.R.UTN	45
<i>Figura 18.</i> Total de usuarios atendidos.....	46
<i>Figura 19.</i> Modelo en V	47
<i>Figura 20.</i> Arquitectura del funcionamiento del sistema	63
<i>Figura 21.</i> Diagrama de bloques del sistema.....	64
<i>Figura 22.</i> Metodología del algoritmo de estimación de pose humana.....	66
<i>Figura 23.</i> Diagrama de bloques de Openpose.....	67
<i>Figura 24.</i> Pares y partes para el conjunto de datos del algoritmo.....	68

<i>Figura 25.</i> Arquitectura del modelo de estimación de pose	69
<i>Figura 26.</i> Etapa 1 modelo VGG-19	70
<i>Figura 27.</i> Etapas y ramas del modelo Openpose	70
<i>Figura 28.</i> Mapas de calor	72
<i>Figura 29.</i> Campo de afinidad de partes.....	73
<i>Figura 30.</i> Asociación entre partes para el reconocimiento del brazo	74
<i>Figura 31.</i> Script donde se encuentran las tuplas del conjunto de datos COCO.....	75
<i>Figura 32.</i> Muestra del esqueleto en 2D de un ser humano	76
<i>Figura 33.</i> Ingreso al ambiente virtual openpose	77
<i>Figura 34.</i> Ejecución del ambiente virtual	77
<i>Figura 35.</i> Carpeta que aloja el script de ejecución del sistema.....	78
<i>Figura 36.</i> Extracción de coordenadas de los puntos que forman el brazo izquierdo .	79
<i>Figura 37.</i> Producto escalar de dos vectores	80
<i>Figura 38.</i> Angulo formado por los vectores del brazo y antebrazo	81
<i>Figura 39.</i> Angulo a evaluar en la flexión y extensión de codo	82
<i>Figura 40.</i> Puntos para calcular los ángulos articulares del hombro.....	83
<i>Figura 41.</i> Medición del ángulo formado por dos vectores para el hombro	84
<i>Figura 42.</i> Lógica medición Angulo resultante de la Abducción.....	85
<i>Figura 43.</i> Lógica de medición del Angulo resultante de la Aducción de hombro.....	86
<i>Figura 44.</i> Ubicación del script para el desarrollo de la Interfaz grafica	87
<i>Figura 45.</i> Interfaz Gráfica del Sistema de medición de Ángulos Articulares.....	87
<i>Figura 46.</i> Ejecución de la opción Prueba.....	88
<i>Figura 47.</i> Ingreso de datos del paciente y evaluación.....	89
<i>Figura 48.</i> Evaluación del Paciente	90
<i>Figura 49.</i> Datos completos del Paciente	91

Figura 50. Plantilla para la muestra de la información completa del paciente.....92

Figura 51. Diagrama de flujo de la obtención de datos.....95

Figura 52. Diagrama de flujo del procesamiento del sistema96

Figura 53. Diagrama de flujo de la salida de datos97

Figura 54. Diagrama de sistema.....98

Figura 55. Plantilla para la toma de muestras de la flexo-extensión..... 102

Figura 56. Toma de datos manualmente con el goniómetro 105

Figura 57. Muestra de datos capturada con el sistema..... 105

Figura 58. Plantilla para la toma de datos de la abducción y aducción de hombro .. 106

Figura 59. Evaluación de la abducción manualmente con el goniómetro..... 109

Figura 60. Evaluación de la aducción a través del sistema 109

Figura 61. Certificado de aprobación de los datos obtenidos con el sistema..... 115

INDICE DE TABLAS

Tabla 1. Algoritmos de Aprendizaje Supervisado.....25

Tabla 2. Algoritmos de Aprendizaje no supervisado.....27

Tabla 3. Funciones de la Capa Pooling34

Tabla 4. Definición de acrónimos.....50

Tabla 5. Prioridad de los Requerimientos del sistema.....50

Tabla 6. Requerimientos Iniciales del Sistema.....52

Tabla 7. Requerimientos de Arquitectura.....54

Tabla 8. Lista de Stakeholders del sistema.....56

Tabla 9. Requerimientos de Stakeholder.....56

Tabla 10. Elección de CPU.....57

Tabla 11. Características del CPU58

<i>Tabla 12.</i> Elección de la GPU.....	59
<i>Tabla 13.</i> Características de la GPU	59
<i>Tabla 14.</i> Elección de la Cámara	60
<i>Tabla 15.</i> Selección del Software de Programación	61
<i>Tabla 16.</i> Puntos de las extremidades a trabajar.....	79
<i>Tabla 17.</i> Planos de evaluación del paciente	93
<i>Tabla 18.</i> Comparación de ángulos	93
<i>Tabla 19.</i> Evaluación de hombro	101
<i>Tabla 20.</i> Evaluación de codo.....	102
<i>Tabla 21.</i> Evaluación de pacientes en reposo	103
<i>Tabla 22.</i> Evaluación de la flexo-extensión de codo	104
<i>Tabla 23.</i> Evaluación en reposo del paciente.....	107
<i>Tabla 24.</i> Datos Obtenidos de la evaluación en el hombro	108

ÍNDICE DE ECUACIONES

<i>Ecuación 1.</i> Función de pérdida de la red neuronal.....	71
<i>Ecuación 2.</i> Representación matemática para la generación de mapas de confianza..	72
<i>Ecuación 3.</i> Representación matemática de los PAF.....	74
<i>Ecuación 4.</i> Producto escalar	80
<i>Ecuación 5.</i> Ecuación para el cálculo del ángulo articular del sistema	81
<i>Ecuación 6.</i> Tasa de error de lectura.....	94
<i>Ecuación 7.</i> Ecuación hipótesis nula	111
<i>Ecuación 8.</i> Ecuación hipótesis alternativa	111
<i>Ecuación 9.</i> Ecuación para el cálculo de estadístico de contraste	111
<i>Ecuación 10.</i> Ecuación de índice de fallas.....	113

Resumen

El presente trabajo está enfocado en el desarrollo de un sistema de medición de ángulos articulares en miembros superiores para evaluación en fisioterapia mediante visión artificial con el uso de un algoritmo de estimación de pose humana el cual ubica los puntos articulares formando extremidades en un esquema en 2D del paciente mostrado en video y en tiempo real.

El sistema detecta los puntos articulares de hombro y codo que permite al usuario evaluar los diferentes arcos de movimiento de los miembros superiores en los planos corporales (frontal y sagital), obteniendo así una lectura clara de los rangos de movimiento del usuario en tiempo real, así como una base de datos que respalda la evaluación del paciente.

Las pruebas de funcionamiento del sistema se realizaron bajo la supervisión de un experto en el área de fisioterapia, lograron determinar validez y confiabilidad del mismo, las pruebas fueron realizadas dentro del Centro de Rehabilitación de la Universidad Técnica del Norte, contando con la participación de 9 pacientes entre estudiantes y administrativos de diferente edad, estatura y patología.

Los resultados obtenidos en las pruebas de funcionamiento tienen coherencia objetiva demostrando que el sistema es válido y confiable con un porcentaje del 92,60%, luego de comparación entre las mediciones realizadas manualmente con un goniómetro y las realizadas por el sistema de medición de ángulos articulares planteado que permitió obtener datos en tiempo real facilitando la evaluación de los pacientes.

Abstract

The present work is focused on the development a system measurement system of articular angles in upper limbs for evaluation in physiotherapy through artificial vision with the use of an algorithm human pose estimation which locates the articular points forming limbs in a 2D scheme of the patient shown on video and in real time.

The system detects the joint points of the shoulder and elbow that allows the user to evaluate the different arcs of movement of the upper limbs in the body planes (frontal and sagittal), thus obtaining a clear reading of the user's ranges of movement in real time, as well as a database that supports the evaluation of the patient.

The system performance tests were carried out under the supervision of an expert in the area of physiotherapy, thus determining its validity and reliability, the tests were carried out within the Rehabilitation Center at Técnica del Norte University of the North, with the participation of 9 patients between students and administrators of different age, height and pathology.

The results obtained in the functional tests have objective coherence demonstrating that the system is valid and reliable with a percentage of 92.60%, after the comparison between the measurements made manually with a goniometer and those made by the joint angle measurement system raised that allowed to obtain data in real time, facilitating the evaluation of the patients.



Capítulo 1. Antecedentes

Este capítulo dará a conocer detalles de la problemática, objetivos y la posible solución planteada.

1.1. Tema

SISTEMA DE DETECCIÓN DEL ÁNGULO ARTICULAR EN LOS MOVIMIENTOS DE MIEMBRO SUPERIOR PARA EVALUACIÓN EN FISIOTERAPIA MEDIANTE VISIÓN ARTIFICIAL.

1.2. Problema

La rehabilitación a través de fisioterapia es realizada por técnicas que permiten la recuperación de un paciente que ha sufrido una lesión tanto a nivel muscular, cirugías, daños cerebrales o fracturas, ocasionando la falta de movilidad en articulaciones expuestas demasiado tiempo a inmovilización, la recuperación de cada paciente depende entre varios aspectos a la constancia en los métodos que utiliza el profesional encargado de la rehabilitación, y el rol del paciente en asistir continuamente a las sesiones de fisioterapia, la medición de ángulos articulares en movimientos de extremidades es evaluado manualmente, con herramientas como lo son graduadores o goniómetros que proporcionan cierto error de medición por estimación visual propio del ser humano, es así que toma importancia la inclusión de la tecnología para determinar el avance en la rehabilitación de movilidad de las extremidades de una persona de manera en que el profesional encargado tenga herramientas de evaluación mucho más efectivas e integrando un valor agregado en la experiencia del paciente generando una motivación extra para su recuperación.

La medición y el registro de los ángulos de movilidad o goniometría, término utilizado que significa medición de ángulos, depende de herramientas y de la precisión de evaluación ser humano, el profesional encargado de la rehabilitación física del paciente realiza mediciones

continuas en cada sesión para determinar el punto de inicio del tratamiento dependiendo de la patología, la evaluación y progreso del paciente así como también determinar las secuelas y el punto final del tratamiento, la información es llevada en fichas médicas e informes hechos manualmente por el fisioterapeuta (Taboadela, 2007). La Universidad Técnica del Norte brinda a través de su Centro de Rehabilitación mejor calidad de vida tanto a estudiantes, deportistas, docentes y personal administrativo propios de la institución que han sufrido una determinada lesión física que imposibilita su desempeño diario, siendo el 40% del total de patologías tratadas destinadas a pacientes con lesiones en miembros superiores (J. Zambrano, comunicación personal, 10 de diciembre de 2018). Es por eso la necesidad de buscar herramientas con ayuda de la tecnología que permitan la evaluación de este tipo de patologías con el fin de ayudar al profesional a realizar un trabajo más óptimo y permitir al paciente conocer de manera más interactiva su evolución.

La posible solución planteada a esta problemática está basada en la aplicación de un modelo de algoritmo pre-entrenado de visión artificial que permite la determinación de posición estimada de extremidades en 2D y por ende calcular el ángulo de movilidad articular de las extremidades en pacientes en recuperación, así como una base de datos periódica de desarrollo del paciente que permita determinar al profesional encargado optimizar recursos en herramientas de medición y tiempo en toma de datos así como una planificación más llevadera de las sesiones de terapia.

La recuperación de pacientes con falta de movilidad en extremidades es determinada por diversos factores entre los que se cita: la constancia y los métodos aplicados por el fisioterapeuta, la integración de la tecnología en este tipo de actividades permite la optimización de recursos y una mejor planificación como es la solución basada en el algoritmo pre-entrenado que permite determinar la posición estimada de extremidades en 2D en tiempo

real a través de visión artificial y por ende calcular el nivel ángulos de movilidad funcional articular de las extremidades superiores para pacientes en recuperación.

1.3. Objetivos

1.3.1. Objetivo general

Determinar los ángulos de movilidad funcional articular en miembro superior en la flexo-extensión de codo y abducción y aducción de hombro para evaluación de terapia física en pacientes del Centro de Rehabilitación de la Universidad Técnica del Norte mediante un algoritmo de estimación de posición de articulaciones por visión artificial en 2D

1.3.2. Objetivos específicos

- Realizar una revisión de literatura respecto a la metodología utilizada por los profesionales en fisioterapia para la evaluación de la movilidad de extremidades superiores en pacientes de rehabilitación física.
- Usar la metodología en V para el diseño del sistema y la implementación del estándar IEEE 29148 para la elección del software y hardware apegado a las necesidades del sistema.
- Implementar el sistema de medición de ángulos articulares usando un algoritmo de estimación de posición de articulaciones por visión artificial y la interfaz para la visualización de resultados.
- Realizar pruebas de funcionamiento y efectuar la corrección de errores del sistema para garantizar la implementación.

1.4. Alcance

El proyecto se basa en la utilización de un algoritmo pre-entrenado que permite la estimación de posición de extremidades del cuerpo humano a través de visión artificial en 2D,

y una posterior aplicación en el desarrollo de un sistema que permita la medición de los ángulos articulares en los movimientos de flexo-extensión de codo y abducción y aducción de hombro en personas con problemas de movilidad en sus extremidades superiores en el Centro de Rehabilitación de la Universidad Técnica del Norte, convirtiéndose en una herramienta de apoyo para el profesional y el paciente.

Se realizará un análisis literario para comprender la metodología utilizada para la rehabilitación física de extremidades superiores con el fin de canalizar la información necesaria en la implementación del sistema de medición en base a visión artificial.

La selección de hardware y software se realizará en base a la norma IEEE 29148 misma que sigue un proceso de selección en base a requerimientos previos con la intervención de los stakeholders y a través de un modelo en V se escogerá los elementos que se apeguen a las necesidades de la implementación y ubicación estratégica del sistema para su funcionamiento.

El desarrollo del sistema se realizara a través de la implementación de un algoritmo que será previamente elegido y que se apegue a las necesidades de funcionamiento del proyecto que permitirá la extracción de datos sobre la postura de extremidades en tiempo real mediante el uso de una cámara de alta definición conectada a un ordenador que procesara el video capturado del paciente, mediante una interfaz se indicara un esquema en 2D de las extremidades superiores del plano frontal del paciente para una posterior evaluación de los grados articulares en los movimientos de miembro superior en la flexo extensión de codo, abducción y aducción de hombro, el ángulo dado se indicara en la interfaz de cada extremidad evaluada y se tendrá una base de datos del nombre del paciente número de cedula y una breve descripción de su patología para el registro continuo de las sesiones de fisioterapia.

Se realizarán pruebas de funcionamiento en dos escenarios, en una primera fase se realizará pruebas de integración con un cierto número de personas sin problemas físicos en sus

extremidades para tener una referencia de la medición que realiza el sistema y la correcta funcionalidad en la toma y análisis de datos. Una segunda fase se realizará con pruebas de funcionamiento para efectuar la corrección de errores del sistema con el fin de garantizar su implementación y la evaluación de datos obtenidos con los pacientes del Centro de Rehabilitación Física de la Universidad Técnica del Norte.

1.5. Justificación

El objetivo principal de la rehabilitación física es proveer de mejor calidad de vida al paciente que ha sufrido una determinada lesión que le impide realizar actividades de forma autónoma a causa de su discapacidad, se entiende por discapacidad a las deficiencias o limitaciones de una persona en relación al entorno donde vive. Según datos de la Organización Mundial de la Salud: el 15% de la población mundial sufre de algún tipo de discapacidad que le impide tener acceso con facilidad a servicios de sanidad (OMS, 2018).

Las estadísticas proporcionadas por la Organización Mundial de la Salud y el Grupo Mundial han permitido desarrollar un informe mundial sobre la discapacidad con datos relevantes que permiten implementar políticas y proyectos innovadores que mejoren la calidad de vida de las personas con una discapacidad en específico, cumpliendo con el artículo 1 de la Convención de Naciones Unidas sobre los Derechos de las Personas con Discapacidad: El propósito de la presente Convención es promover, proteger y asegurar el goce pleno y en condiciones de igualdad de todos los derechos humanos y libertades fundamentales por todas las personas con discapacidad, y promover el respeto de su dignidad inherente (Naciones Unidas, 2006).

En el Ecuador, el Gobierno viene implementando políticas que contribuyen a construir una "Patria para todos y todas", con planes que abordan temáticas que se enfocan en proveer a sus habitantes de una sociedad que promueva la inclusión social de todos los sectores, en

especial aquellos expuestos a una vulnerabilidad inminente y que requieren de una atención especial como lo son las personas con discapacidad. El Plan Nacional del Buen Vivir manifiesta en sus incisos objetivos que incluyen programas en beneficio de los grupos vulnerables y son de suma importancia ser tomados en cuenta al momento de diseñar e implementar un proyecto.

Para cumplir con los objetivos de una adecuada rehabilitación física y mejorar la condición de salud de un paciente con un determinado problema físico este debe acudir constantemente a un centro especializado que cuente con los equipos necesarios hasta completar el tratamiento, hay mucha gente que no completa su terapia por temas económicos y falta de motivación al observar un lento progreso en su recuperación. En las sesiones de fisioterapia terapeuta es el encargado de controlar los ejercicios y técnicas necesarias dependiendo de la patología del paciente para lograr en el menor tiempo posible su recuperación. Actualmente la tecnología es una alternativa para desarrollar nuevas formas de evaluación de un paciente de manera que arroje datos más precisos, mejores planificaciones, una aceleración en el proceso de recuperación y la integración del paciente con sesiones de fisioterapia interactivas y que generen mayor motivación al interactuar visualmente a través de la tecnología con los avances en el movimiento de una determinada extremidad.

Capítulo 2. Fundamentación Teórica

El presente capítulo abordara tópicos necesarios para el desarrollo del presente proyecto, se detallará aspectos importantes para comprender el proceso de medición de ángulos articulares de miembros superiores, partiendo de la goniometría, su uso, técnicas medición y los detalles a tomar en cuenta para los miembros superiores, así como también una breve introducción a la visión artificial, sus aplicaciones y entrar en aspectos fundamentales tanto del Aprendizaje Automático como el Aprendizaje Profundo, haciendo énfasis en el uso de redes neuronales para el procesamiento de imágenes así como los algoritmos de estimación de posición humana que serán explicados en breves rasgos permitiendo el desarrollo de modelos y sistemas que permiten la aplicación del presente proyecto .

2.1. Fisioterapia

El término o concepto fisioterapia engloba un sin número de técnicas y métodos necesarios utilizados por un profesional en el campo que permiten la recuperación de una persona que sufre de una determinada patología o problema físico (Gallego Izquierdo, 2007).

El desarrollo de la recuperación de un paciente con una patología que impida su desenvolvimiento en actividades diarias de cualquier índole, dependerá de la interacción entre el paciente y el fisioterapeuta, es decir la asistencia continua a sesiones de fisioterapia, permiten a largo o corto plazo que el tratamiento utilizado por el profesional encargado determinar el éxito en la recuperación parcial de una persona en relación a pacientes que tienen sesiones discontinuas o que no completan el proceso de rehabilitación (Bigorda-Sague & Abadia-Rosa, 2016).

2.1.1. Biomecánica

Puede decirse que la biomecánica analiza el movimiento humano mediante la observación de los fenómenos mecánicos y cinemáticos que denotan vida en los seres

humanos. Tiene como objetivo compilar información cuantitativa sobre la mecánica del sistema musculoesquelético durante la ejecución de cualquier movimiento (Lu & Chang, 2012).

2.1.1.1. Biomecánica del codo

El codo es una articulación compleja que incide en el enlace entre el hombro y la mano, funciona como un punto de apoyo del antebrazo, es importante para el posicionamiento correcto de la mano en el espacio, permitiendo realizar actividades de la vida cotidiana. El codo es lo suficientemente robusto como para permitir que funcione como una articulación de soporte de peso. La biomecánica es necesaria para mantener su estabilidad y función (Felstead & Ricketts, 2017).

El arco de flexión en el codo es de 0 a alrededor de 140 grados de flexión. Normalmente hay 85 grados de supinación y 75 grados de pronación. El rango de movimiento funcional es de 30 a 130 grados de flexión, y 50 grados de pronación y supinación (Frangiamore et al., 2018).

Los movimientos que la articulación del codo permite son:

- **Supinación y pronación.** - El movimiento primario del antebrazo es supinación y pronación, con el eje de rotación que pasa de la cabeza radial proximal a la superficie articular convexa del cúbito, permite una supinación promedio de 75 ° y pronación media de 70 °. Para la mayoría de las actividades de la vida cotidiana Figura 1.

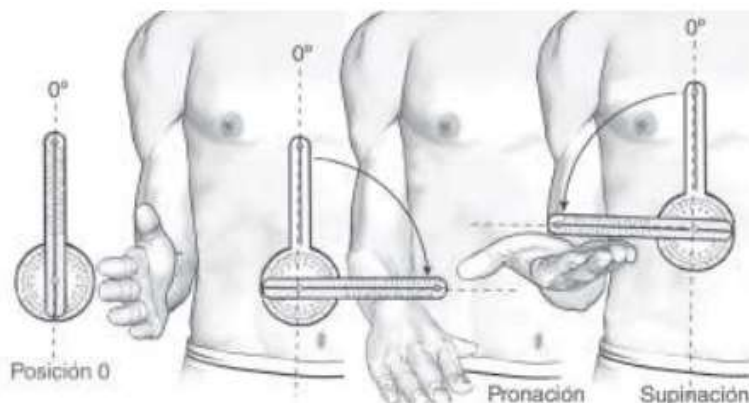


Figura 1. Movimientos de Supinación y Pronación del codo

Fuente: Adaptado de (Malagelada, Dalmau-Pastor, Vega, & Golanó, 2014)

- **Flexión y extensión.** - El rango de movimiento del codo consiste en flexión y extensión de 0° (extensión completa) a 140° de flexión, la mayoría de las actividades de la vida diaria pueden ser realizadas en el rango de 30 grados a 130 grados. El codo es a menudo catalogado erróneamente como una articulación de bisagra simple debido a la articulación ulnohumeral congruente estable Figura 2.

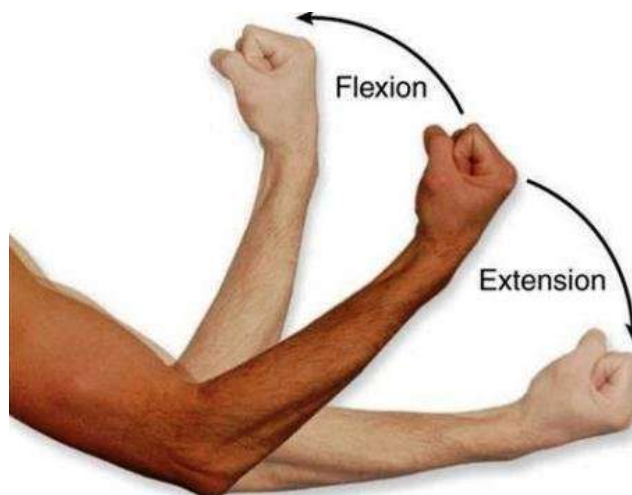


Figura 2. Flexión y extensión de codo

Fuente: Adaptado de (Fornalski, Gupta, & Lee, 2003)

2.1.1.2. Biomecánica del hombro

El hombro es considerado la articulación que mayor grado de movilidad en el cuerpo humano, pero también una de las más inestables y por ende propensa a lesiones. El funcionamiento normal del hombro depende de factores como: el rango de movimiento,

estabilidad y la fuerza, permitiendo que el brazo se pueda colocar en el espacio para el desarrollo de diversas actividades diarias (Suárez & Osorio, 2013).

Los movimientos que la articulación del hombro permite son:

- Dentro del plano sagital se puede realizar los movimientos de flexoextensión, como se observa en la Figura 3.

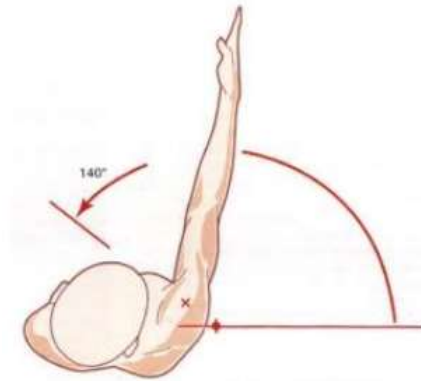


Figura 3. Movimientos del plano sagital del hombro

Fuente: Adaptado de (Kapandji, 2006)

- En el plano frontal los movimientos de abducción en el cual el miembro se aleja del cuerpo y aducción donde el miembro superior se aproxima al plano de simetría Figura 4.



Figura 4. Movimientos del Plano Frontal del Hombro

Fuente: Adaptado de (Yamamoto & Itoi, 2015)

- En el plano horizontal los movimientos de flexión y extensión se realizan con el brazo en aducción de 90 grados Figura 5.

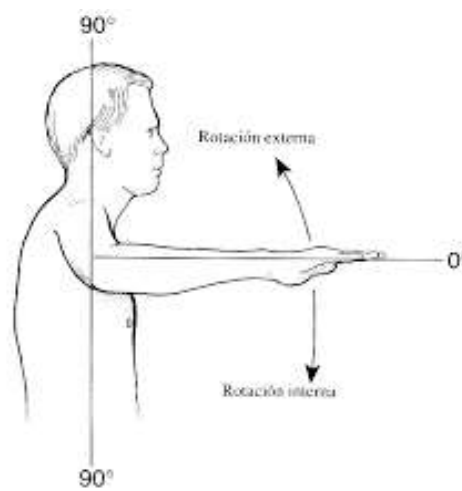


Figura 5. Movimientos del plano horizontal del hombro

Fuente: Adaptado de (Wolff & Rosenzweig, 2017)

2.1.2. Goniometría

Las herramientas de medición utilizadas en terapia física médica son relevantes para la evaluación de pacientes, y más aún la importancia del uso correcto de estas por parte del evaluador, la medición de ángulos articulares permiten establecer un punto de partida y final en la rehabilitación de un paciente con falta de movilidad en sus extremidades, métodos de medición como el uso de rayos x expone al paciente y al examinador a radiaciones perjudiciales a la salud, métodos prácticos y eficientes se han estandarizado por parte de la AAOS (Academia Americana de Cirujanos Ortopédicos) con el uso de herramientas de medición como el goniómetro (Taboadela, 2007).

La goniometría es el método más utilizado para evaluar el rango de movimiento en pacientes con secuelas de una determinada lesión tanto muscular o esquelética (Parry et al., 2019). Por ende, la goniometría no es más que la medición de los ángulos formados por los huesos distales y proximales de una articulación para determinar la cantidad de movimiento que esta puede estimar (Norkin & White, 2016).

2.1.3. Objetivos de la Goniometría en terapia física

La goniometría ha sido utilizada en múltiples ramas de estudio como arquitectura, ingeniería, y sobre todo la medicina que es el tema de aplicación del presente proyecto. Existen dos objetivos en los cuales la goniometría médica es utilizada:

- Cuantificar, evaluar y documentar la falta de movilidad de una articulación y sus limitaciones.
- Cuantificar y evaluar la movilidad de una articulación. (Taboadela, 2007)

2.1.4. Instrumentos de Medición

La valoración de la movilidad en una articulación puede ser realizada a partir de estimación visual del fisioterapeuta con el uso de herramientas que permita la medición de su flexibilidad o presencia de movimiento. (Araújo, 2005)

2.1.4.1. Goniómetro

El goniómetro es la herramienta más utilizada para medir rangos de movimiento en articulaciones, es un instrumento de 360° bastante eficiente para la medición de grados de movilidad, consta un punto central que es ubicado sobre la articulación y dos brazos móviles que permiten determinar el ángulo de movilidad (Clarkson, Hurabielle, Doy, & Marlowe, 2003), como se muestra en la Figura 6.

Tiene algunas limitaciones en cuanto el médico debe utilizar ambas manos para la toma de datos, lo que causa inestabilidad y conduce al error, puesto que se debe poner atención en el goniómetro y la posición correcta de la articulación. (Behnoush et al., 2016)



Figura 6. Uso del Goniómetro en la articulación del codo

Fuente: Adaptado de (Chapleau, Canet, Petit, Laflamme, & Rouleau, 2011)

2.1.4.2. Inclinómetro

El inclinómetro es una alternativa frente al goniómetro, hace uso de la gravedad constante como punto de referencia, contiene en su interior fluidos, estos instrumentos son ligeros y económicos, requieren un entrenamiento similar al uso del goniómetro, es colocado sobre la articulación a medir, como se puede observar en la Figura 7, son propensos a errores en su colocación debido a que no son ubicados en el punto de referencia a medir, que son llamados errores técnicos mismos que arrojan lecturas inexactas, es uno de los métodos con mayor probabilidad de error.(Kolber, Pizzini, Robinson, Yanez, & Hanney, 2013)



Figura 7. Uso del inclinómetro

Fuente: Adaptado de (Taboadela, 2007)

2.1.4.3. Electrogoniómetro

El electrogoniómetro es una herramienta similar al goniómetro, utiliza dos sensores y registra su medición a través de software, su costo es relativamente alto y su poco uso como lo muestra la Figura 8. (Ruiz-Olaya et al., 2017)



Figura 8. Manejo del inclinómetro

Fuente: Adaptado de (Taboadela, 2007)

2.1.5. Goniometría de hombro y codo

La medición de ángulos articulares en miembros superiores para caso del presente proyecto como hombro y codo permiten la evaluación y cuantificación de la recuperación a través de terapia física con el objetivo de alcanzar la funcionalidad del miembro afectado.

2.1.5.1. Abducción y aducción de hombro

El hombro es la articulación de mayor movilidad de todas las que componen el cuerpo humano, es una articulación proximal de miembro superior que consta de tres ejes: Eje transversal, vertical y anteroposterior el cual permite los movimientos de abducción y aducción (Kapandji, 2006).

La abducción y aducción de hombro son patrones de movimientos cuyo objetivo pretende aislar los abductores del hombro, así como también los aductores, este procedimiento se lo puede trabajar con el paciente tanto de pie como sentado en el plano frontal (Lowe, Chaitow, Lowe, & Chaitow, 2009).

La goniometría del hombro a través de la abducción y aducción se la realiza con el goniómetro ubicado en el punto 0 en la articulación del hombro, con la mano colocada sobre el hombro apegado contra la pared para fijar el movimiento, el brazo fijo del goniómetro alineado paralelo al esternón en línea al punto 0 mientras que el brazo móvil sigue el movimiento del brazo (Taboadela, 2007). La Figura 9 indica el movimiento de abducción.

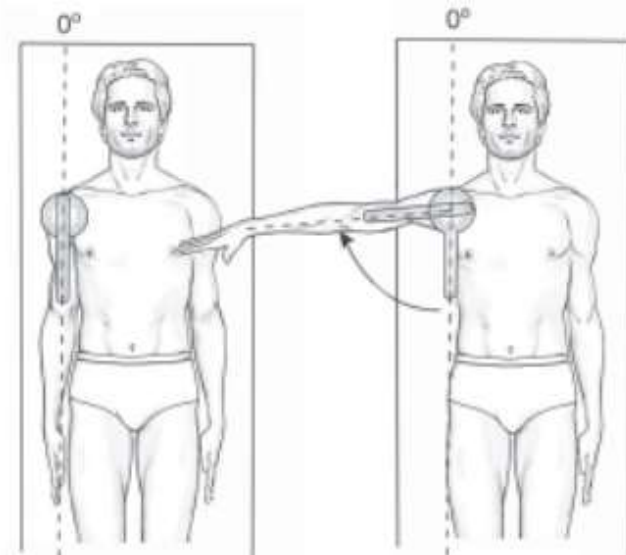


Figura 9. Abducción de hombro en la plantilla frontal

Fuente: Adaptado de (Taboadela, 2007)

En el caso de la abducción el hombro permite que el movimiento del brazo se aleje del tronco formando un ángulo de 180° en la amplitud máxima en relación al eje imaginario 0 como se muestra en la Figura 10.



Figura 10. Ángulo máximo del movimiento de abducción

Fuente: Adaptado de (Kapandji, 2006)

El movimiento de aducción acerca al miembro superior en referencia al tronco, a diferencia del movimiento de abducción la amplitud máxima de este movimiento es mucho menor es decir el rango va de 0° a 30° , el proceso de evaluación de este movimiento a través del goniómetro es similar al movimiento de abducción, (Sabari, Maltzev, Lubarsky, Liskay, & Homel, 1998) . El proceso de medición del ángulo articular del hombro en el movimiento de aducción se muestra en la Figura 11.

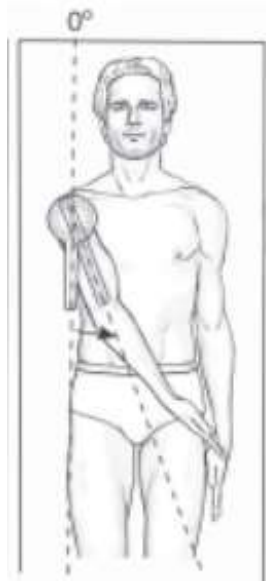


Figura 11. Aducción de hombro

Fuente: Adaptado de (Taboadela, 2007)

2.1.5.2. Flexo-extensión de codo

El codo es la articulación compleja que une tanto el brazo y el antebrazo, por ende, encargada de algunas de las funciones de los miembros superiores, en específico la colocación de la mano en el espacio y su desplazamiento. (Magee et al., 2016)

El movimiento de flexión permite que el antebrazo se proyecte dentro del brazo, se caracteriza por tener un ángulo máximo promedio de 150° , mientras que el movimiento de extensión el antebrazo se aleja del brazo, con ausencia de movimiento del codo, puede llegar a un ángulo promedio de 10° . (Kapandji, 2006)

La goniometría del codo a través de la flexo-extensión se la realiza ubicando el punto 0 en el codo, el brazo fijo del goniómetro se lo alinea con la línea vertical 0° mientras que el brazo móvil acompaña el movimiento del brazo del paciente en el plano anatómico sagital.

Figura 12.

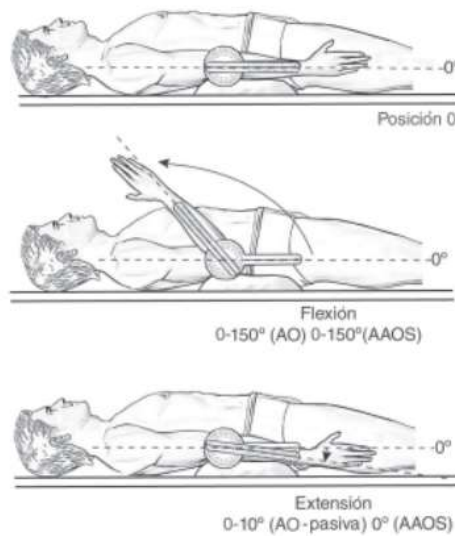


Figura 12. Medición de la Flexo-extensión de codo en el plano anatómico medial

Fuente: Adaptado de (Taboadela, 2007)

2.1.6. Métodos de medición

El primer método de medición utilizado fue el de $180^\circ - 0^\circ$ donde el sujeto se encuentra en posición anatómica y las articulaciones a ser evaluadas establecen una posición inicial de 180° , en el método de medición del cero neutro la posición de medición comienza a partir de la posición 0° o neutro, se considera el método estándar actualmente implementado por la AO (Asociación para el Estudio de Osteosíntesis) y con el que se trabajara en el presente proyecto de estudio.

2.1.7. Registro de resultados

Se lo realiza a través de tablas numéricas donde la información se encuentra tabulada en forma numérica y hace referencia a los grados de movimiento para cada una de las articulaciones, este registro consta de nombre, edad, sexo y el número de grados máximo de movilidad del examinado. (Taboadela, 2007)

2.2. Inteligencia Artificial

La inteligencia artificial o IA puede definirse como el estudio y desarrollo de sistemas inteligentes, haciendo énfasis en la simulación de la inteligencia humana aplicada en las máquinas abarcando procesos como el aprendizaje y el razonamiento; autores como Deyi Li mencionan que la inteligencia artificial es: *“Un área de estudio en el campo de la informática en el cual la inteligencia artificial está preocupada por el desarrollo de computadoras capaces de involucrarse en procesos propios de los seres humanos de pensamiento como el aprendizaje, el razonamiento y la auto corrección”* (Li, D., & Du, Y, 2017). La inteligencia artificial se incorpora a una variedad de diferentes tipos de tecnología tales como:

- Reconocimiento de voz
- Visión Artificial
- Aprendizaje Automático
- Robótica
- Minería de datos
- Automatización

La investigación de esta rama de la informática plantea nuevos desafíos y la aparición de aplicaciones en áreas de interés económico, a nivel de salud, educación y gubernamental.

2.2.1. Visión artificial

Dentro del desarrollo de la Inteligencia Artificial una de sus ramas en auge y con mayor desarrollo es la visión artificial o también llamada visión por computador es un campo multidisciplinario definido como un subcampo de la inteligencia artificial y el aprendizaje automático como muestra la Figura 13, que puede hacer uso de métodos y algoritmos de aprendizaje, encargada de tomar parámetros que sirven de base para la detección de objetos y el procesamiento de imágenes.

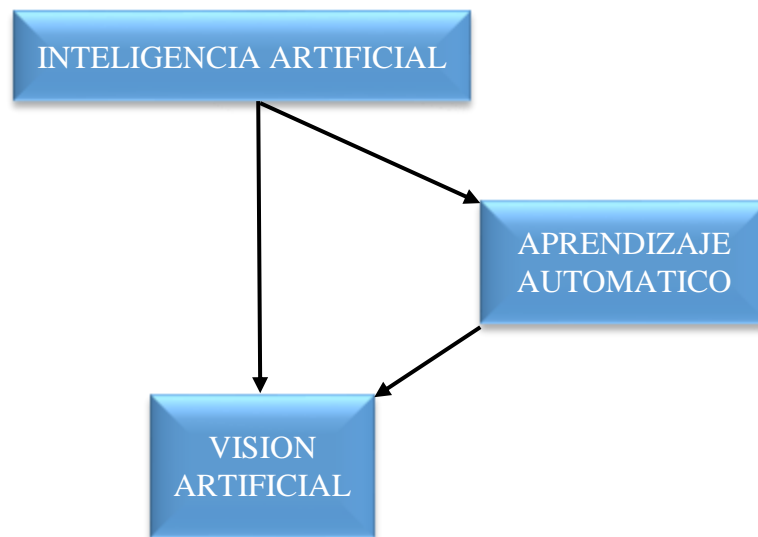


Figura 13. Relación entre la inteligencia artificial y la visión artificial

Fuente: Autoría

El objetivo principal de la visión artificial es el de emular la capacidad de visión humana desarrollando métodos que permitan la comprensión del contenido de imágenes digitales.

Para que un sistema de visión artificial cumpla con su objetivo y funcione de acuerdo a las necesidades este depende de 6 etapas importantes, el campo de estudio en el presente trabajo de investigación depende del procesamiento de imágenes, que consiste en optimizar las imágenes en estudio para su posterior manipulación, análisis, reconocimiento e interpretación como muestra la Figura 14.

- **Captación:** es el proceso base, a través de una cámara o sensor se encarga de obtener una imagen visual en formato digital para luego cumplir con el objetivo de escalar la imagen y realizar la conversión de colores RGB (Paleta de colores red, green, blue) a gris o viceversa
- **Pre-procesamiento:** se encarga de la manipulación de la imagen para mejorar los resultados en una aplicación en específico, hace uso de técnicas tales como la reducción de ruido y realce de detalles que resaltan ciertas características de interés de la imagen, En este paso, habría una transición de procesos que generan imágenes, a procesos que generan atributos de imagen.
- **Segmentación:** Es el proceso que divide a una imagen en objetos o partes constituyentes, se puede decir que cuanto más preciso sea la segmentación, el reconocimiento tiene mayor probabilidad de tener éxito.
- **Representación y Descripción:** en la representación se realiza la toma de decisión en base a los datos de cada salida de una etapa de segmentación a través del uso de algoritmos, se los considera como un límite o región, la representación de límites obtiene características convenientes para diferenciar un objeto de otro en base a esquinas e inflexiones, mientras que la representación de regiones e enfoca en las propiedades internas como texturas.

A su vez la descripción es el proceso de selección de características a fin de extraer atributos que dan como resultado información específica de interés para un posterior procesamiento informático en el que conlleva el reconocimiento.

- **Reconocimiento e Interpretación:** Es el proceso por el cual se asigna una etiqueta a un objeto basado en la información obtenida de las etapas posteriores, tareas que son realizadas por un algoritmo que realiza tareas en específico del procesamiento de la imagen.

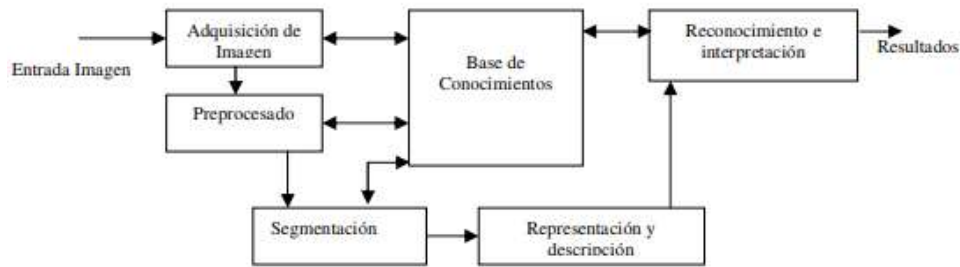


Figura 14. Etapas de fundamentos de un sistema de visión artificial

Fuente: Adaptado de (Chávez, 2001)

2.3. Aprendizaje Automático

El aprendizaje automático es el estudio computacional de algoritmos que mejora el rendimiento y la capacidad de aprender a las computadoras sin ser programadas explícitamente, es un subcampo de la inteligencia artificial (IA). El aprendizaje automático se centra en el desarrollo de sistemas y programas ligados a la informática que pueden cambiar de acuerdo a los datos y condiciones que se les presenten. (Zulkarnain et al., 2017)

2.3.1. Métodos de aprendizaje automático

En este apartado se dará a conocer los tipos de algoritmos de aprendizaje automático que con su aplicación permite el reconocimiento de objetos mediante modelos matemáticos necesarios para el que un sistema pueda trabajar de forma automática, estos son el aprendizaje supervisado, no supervisado, reforzado y profundo.

2.3.1.1. Aprendizaje supervisado

El aprendizaje supervisado es una herramienta poderosa para clasificar y procesar datos utilizando lenguaje de máquina, la idea principal es que el usuario intervenga en el entrenamiento de un algoritmo cuyos datos de entrenamiento de entrada y salida son proporcionados y a la vez etiquetados. Con el aprendizaje supervisado, el conjunto de datos se utiliza como base para predecir la clasificación de otros elementos no etiquetados datos a través del uso de algoritmos de aprendizaje automático. (Shobha & Rangaswamy, 2018)

Un modelo se desarrolla mediante un proceso de capacitación en el cual es necesario hacer predicciones mismas que deben ser corregidas cuando son incorrectas, las técnicas que abordan el aprendizaje supervisado son de clasificación y regresión lineal.

- **Clasificación:** son técnicas que se enfocan en predecir una respuesta cualitativa analizando datos y reconociendo patrones, es recomendado el uso de este tipo de algoritmos cuando los datos se pueden etiquetar o separar en grupos específicos, las aplicaciones que se desarrollan en base a esta técnica son: calificación de crédito bancario, reconocimiento de voz, verificación de correos spam. Los problemas de clasificación se pueden resolver con un sinnúmero de algoritmos entre los que se citan:
 - Máquinas de Vector Soporte (SVM)
 - Árboles de decisión
 - K vecinos más cercanos (k-NN)
 - Redes Neuronales
 - Regresión Logística
- **Regresión Lineal:** esta técnica de aprendizaje supervisado generalmente se usa para predecir, pronosticar y encontrar relaciones entre datos cuantitativos. Es una de las primeras técnicas de aprendizaje, que todavía se usa ampliamente. Esta técnica se puede utilizar en aplicaciones tales como examinar si hubo una relación entre el presupuesto publicitario de una empresa y sus ventas. También podría usarlo para determinar si existe una relación lineal entre una radioterapia particular y tamaños tumorales.

En la Tabla 1 se muestra los algoritmos de aprendizaje supervisado que permiten el entrenamiento de modelos y desarrollo de aplicaciones que se presentan en la literatura relacionada al ámbito de la inteligencia artificial.

Tabla 1. Algoritmos de Aprendizaje Supervisado

Algoritmo	Descripción	Aplicaciones	Ventajas	Desventajas
Árbol de Decisión	Busca clasificar los elementos mediante el uso de una serie de variables y posibilidades. En términos simples, un árbol de decisión le presta la ayuda para tomar una decisión sobre el elemento de datos	Economía, toma de decisiones para entrega de créditos (Timarán, Calderón, & Hidalgo, 2017). - Industrial, detección de niveles de agua (Rivas, 2008).	Llega a una conclusión lógica presentando las posibles consecuencias y eventos fortuitos, construyendo un modelo.	En cada elección considera únicamente una hipótesis por lo que lo hace un proceso lento.
K vecinos más cercanos (k- nn)	Utiliza el conjunto de datos completo como conjunto de entrenamiento, en lugar de dividir el conjunto de datos en un conjunto de entrenamiento y un conjunto de prueba	Industrial, clasificador del producto para predicción de precios (Troncoso, 2007).	Es un algoritmo sencillo debido a que no es necesario construir un modelo explícito, ni hacer suposiciones sobre los datos	Lentitud en la clasificación cuando existen muchos datos, dependencia de la función distancia, gran influencia del ruido en los datos.
Naïve bayes	Es capaz de clasificar un nuevo elemento, supone que una característica particular en una clase no está directamente relacionada directamente con ninguna otra característica.	Industria automotriz, como coches autónomos. - Informática, localización de correos y clasificación de artículos.	Toma decisiones óptimas razonando con probabilidades, útil para un conjunto de datos grande	Elevado coste computacional en el proceso de actualización de probabilidades, requiere conocer gran número de elementos.
Regresión ordinaria por mínimos cuadrados	Utiliza el método matemático llamado regresión, es más adecuada para la clasificación binaria: conjuntos de datos donde $y = 0$ o 1 , donde 1 denota la clase predeterminada.	Climatología, análisis de los patrones del comportamiento del clima	Permite realizar predicciones a partir de datos parciales, lo que le hace un algoritmo de funcionamiento rápido.	No es apto para el manejo de variables aleatorias continuas.
Regresión logística	Manera estadística de representar un resultado binomial con variables explicativas, estima la probabilidad de ocurrencia de un suceso.	Medicina, identificación de variables endocrinológicas (Domínguez & Padilla, 2001). - Académico, reconocimiento de patrones de rendimiento estudiantil (Reyes, Escobar, Duarte, & Ramírez, 2007)	Se utiliza en escenarios de grandes volúmenes de datos por su eficiencia, además el resultado del modelo se puede interpretar como una probabilidad.	Resultados satisfactorios en caso de que los atributos y clases sean linealmente separables
Support vector machines (svm)	Es de clasificación binaria, con dos grupos de distintos tipos, el SVM buscará la manera de separarlos mediante líneas rectas estando lo más lejos posible entre ellos.	Reconocimiento de sitios de empalme humano, detección de genero basado en imágenes y clasificación de imágenes a gran escala.	Predice el grupo al que pertenece una nueva entrada, basándose en una sola variable. Permite minimizar el error en la separación,	Es un clasificador binario y no está diseñado para identificar los atributos importantes que construyen la regla discriminante.

			utilizando pocos parámetros a estimar.	
Redes neuronales	Simulan el comportamiento de un cerebro humano, conectando varios procesadores entre sí.	Telecomunicaciones, reconocimiento de patrones, filtrado de señales, reconocimiento de caracteres (Galán & Martínez, 2015). - Robótica, identificación de movimientos (Moya, Herrero, & Guerrero, 1998). -Psicología, identificación de características para diagnóstico (Montaño, 2002).	Son capaces de aprender de la experiencia, crea su propia organización y tiene un bajo porcentaje de fallos o errores.	Complejidad de aprendizaje para grandes tareas, por su gran cantidad de datos, requieren de una gran capacidad de hardware

Fuente: Adaptado de

2.3.1.2. Aprendizaje no supervisado

El aprendizaje no supervisado permite que una máquina explore un conjunto de datos e identifique patrones ocultos que vinculan diferentes variables. Este método se puede utilizar para agrupar los datos en grupos basándose solo en sus propiedades estadísticas. Una buena aplicación del aprendizaje no supervisado es la vinculación de registros probabilísticos, una técnica que extrae conexiones entre elementos de datos y se basa en ellos para identificar individuos y organizaciones y sus conexiones en el mundo físico o virtual. (Celebi & Aydin, 2012.)

El aprendizaje no supervisado es lo opuesto al aprendizaje supervisado, es importante conocer que los datos no están etiquetados porque no existe un conjunto de entrenamiento, ninguno de los datos se puede clasificar o pre clasificar de antemano, por lo que es el algoritmo de aprendizaje automático es más complejo y el procesamiento requiere de mucho tiempo. Con el aprendizaje no supervisado, el algoritmo de aprendizaje automático clasifica un conjunto de datos al descubrir una estructura a través de elementos comunes en los datos, las técnicas que hacen uso de este modelo de aprendizaje es el clustering y el análisis de componentes principales. (Hoffmann, 2015)

El Clustering o análisis de agrupamiento es un tipo de técnica de aprendizaje no supervisado utilizada para encontrar elementos comunes entre los elementos de datos que de otro modo no estarían etiquetados y sin categorizar. El objetivo de la agrupación es encontrar grupos distintos o agrupaciones dentro de un conjunto de datos. Usando un algoritmo de lenguaje de máquina, la herramienta crea grupos donde los elementos en un grupo similar tendrán en general características similares el uno del otro.

El Análisis de componentes principales es una técnica de aprendizaje no supervisado que resume un gran conjunto de variables y lo reduce a un representante más pequeño variables, llamados componentes principales. El objetivo de este tipo de análisis es identificar patrones en los datos y expresar sus similitudes y diferencias a través de sus correlaciones.

Varios de los algoritmos que forman parte de la solución de problemas en el aprendizaje no supervisado se muestran en la Tabla 2.

Tabla 2. Algoritmos de Aprendizaje no supervisado

Algoritmo	Descripción	Aplicaciones	Ventajas	Desventajas
k-means	es un algoritmo iterativo que agrupa datos similares en grupos. Calcula los centroides de k grupos y asigna un punto de datos a ese grupo que tiene la menor distancia entre su centroide y el punto de datos.	-Segmentación de imágenes (Pinto, 2015). -Geoestadísticas (Honarkhah & Caers, 2010).	Es un algoritmo simple y trabaja muy eficientemente gracias a que tomo decisiones rápidas con la característica concreta.	Es muy sensible al ruido y está limitado a encontrar clusters globulares.
K-medoids	Es una partición clásica de agrupación a un conjunto de datos de n objetos en k grupos conocidos, selecciona los puntos centroides y mediante un matriz arbitraria las distancias entre ellos.	-Agricultura, Agrupar genes y proteínas (Gravano & Mislej, 2015). - Telecomunicaciones, detección para la distribución de una red híbrida (Amoroso & Avila, 2015)	Aporta una mayor precisión diferencial entre grupos, además es un algoritmo más robusto frente al ruido.	Requiere indicar el número de clusters que se van a crear, además las agrupaciones resultantes varían con la asignación inicial de los centroides

Análisis de Componentes Principales (PCA)	Algoritmo de reducción de dimensionalidad, es invariante para escalar los datos.	de -Teledetección, de detección de objetos terrestres para 2013).	Presenta mayor robustez cuando las variables tienen unidades distintas, además son fáciles de interpretar sus resultados.	Utiliza los vectores propios de la matriz de covarianza y considera que los datos se distribuyen de forma gaussiana, no considera otra forma.
---	--	---	---	---

Fuente: Adaptado de

2.3.1.3. Aprendizaje reforzado

Con el aprendizaje por refuerzo, permite que la máquina interactúe con su entorno (por ejemplo, moviendo los productos dañados de un transportador a un contenedor) y proporcione una recompensa cuando hace lo correcto. Al automatizar el cálculo de la recompensa, puedes dejar que la máquina aprenda a su debido tiempo. (Sugiyama, 2016)

Muchos robots utilizan el aprendizaje por refuerzo, así como el aprendizaje profundo para determinar cuánta presión deben usar al agarrar objetos y la mejor manera de agarrar estos artículos en el inventario.

2.3.1.4. Aprendizaje profundo

El aprendizaje profundo realiza tipos de aprendizaje tales como el aprendizaje sin supervisión o de refuerzo. En términos generales, el aprendizaje profundo imita algunos aspectos de cómo las personas aprenden, principalmente mediante el uso de redes neuronales para identificar las características del conjunto de datos con más detalle. (Martinez, Hossain, Romero, & Little, 2016.)

El aprendizaje profundo, en la forma de redes neuronales profundas, se ha utilizado para acelerar la detección de alto contenido para el descubrimiento de fármacos. Implica la aplicación de técnicas de aceleración de aprendizaje para procesar múltiples imágenes en un tiempo significativamente menor, al tiempo que extrae una mayor comprensión de las características de imagen que el modelo aprende en última instancia. (Qiao, Wang, & Li, 2017)

2.4. Redes neuronales

Las redes neuronales artificiales o también conocidas como sistemas conexionistas son un modelo computacional vagamente inspirado en el comportamiento observado en su homólogo biológico. Consiste en un conjunto de unidades, llamadas neuronas artificiales, conectadas entre sí para transmitirse señales. (Liu, Wu, Wu, & Guo, 2012)

Las redes neuronales artificiales tratan de simular el procesamiento del cerebro, se estructuran como numerosos procesadores conectados entre sí. Matick (2001) define a una red neuronal artificial como un sistema para el tratamiento de la información, cuya unidad básica de procesamiento está fundamentada en la célula principal del sistema nervioso humano, es decir, la neurona. Una neurona artificial es un objeto lógico (se trata de software no de hardware) que recibe diversas entradas, hace una suma ponderada de las mismas y produce una salida a partir de la aplicación de una función umbral a la media ponderada.

Este tipo de algoritmos son una herramienta robusta para resolver problemas altamente complejos dentro de las ciencias físicas y recientemente por los estudiosos en investigación organizacional como herramientas digitales que permiten procesos más rápidos de recopilación y procesamiento de datos. (Shahid, Rappon, & Berta, 2019)

2.4.1. Redes neuronales profundas

A medida que se estudiaba la manera de lograr resolver problemas usando reglas, redes semánticas y modelos probabilísticos, aparecieron problemas más complejos como el gran número de variables, aprendizaje autónomo o auto organización de información, los cuales no se podían procesar usando un tipo de razonamiento lógico. Es por ello que se buscó la manera de realizar una solución como lo hace la inteligencia humana, con neuronas.

Las redes neuronales artificiales son procesadores interconectados con organización jerárquica, la cual, consta de una unidad procesadora de 4 elementos funcionales: receptor,

sumador, activador y salida. La interconexión de estas neuronas forma la estructura básica de una red neuronal artificial las cuales se dividen en tres capas generales llamadas: capa de entrada, cuyas neuronas se conectan a las neuronas de la capa oculta y estas conectadas a la capa de salida. La capa de entrada es la que recibe directamente las señales de fuentes externas, mientras que la capa de salida es la que transfiere la información obtenida en la red neuronal hacia el exterior para ser analizada. La capa oculta puede contener varios niveles internos y estas no tiene contacto con el exterior, esta capa contiene neuronas procesadoras que pueden estar conectadas de distintas maneras, de acuerdo a la topología que se plantee de red neuronal. Esta estructura básica es demasiado simple para obtener una red optima, es por ello que incrementar el número de capas a una red neuronal permite obtener mayor información de los datos de entrada. Las denominadas Redes Neuronales Profundas son aquellas que tienen una o más capas intermedias entre la entrada y la salida, estas redes son la base del aprendizaje profundo (López M., 2016). Las redes neuronales profundas utilizan una jerarquía de características en cada capa, lo que le permite aprender y clasificar, no únicamente el objeto deseado sino también las similitudes de todos los objetos presentes en la imagen. Estas redes utilizan una combinación de aprendizaje supervisado y no supervisado en sus diferentes capas ocultas internas, de esta manera reconstruyen la imagen de entrada aprendiendo características propias en cada una de ellas. Las redes neuronales profundas a diferencia de las redes neuronales artificiales, son capaces de extraer un gran número de características de la imagen completa de entrada, es decir, extrae características de todos los objetos presentes en la imagen y esto permite agrupar, de cada objeto, las características principales. Esto es posible debido a que una red neuronal profunda es capaz de tener en su estructura una gran profundidad en su capa oculta haciendo que los datos pasen por varios procesos de reconocimiento de patrones, en cambio las redes neuronales artificiales cuentan con una estructura sencilla, en términos de número de capas. El aprendizaje profundo se realiza mediante el uso de un gran número de

datos etiquetados y la arquitectura de la red neuronal que, sin necesidad de realizar una extracción manual de características, aprende directamente de los datos. Gracias a la cantidad de capas ocultas que esta red puede tener, es posible obtener mayor información del objeto de entrada y tomar un resultado final más preciso, de esta manera, a medida que se avanza en la red, cada capa oculta obtiene características más complejas de la imagen de entrada (Jones, 2018).²³ Para su respectivo entrenamiento los parámetros de la red comienzan con una suposición y, al final de pasar los datos por toda la red, se obtiene el índice de error y la influencia de cada parámetro; con esta información el entrenamiento continúa actualizando sus parámetros. Una vez entrenada la red, es capaz de realizar su tarea con datos no etiquetados, los cuales la red no había visto antes

Diversas arquitecturas se presentan en el ámbito redes neuronales profundas que han permitido ampliar la cantidad de soluciones a problemas que una red neuronal puede proveer, como la extracción de características, precisión, actualización de parámetros, capacidad limitada, entre otros; de esta manera, mientras más datos se usen para entrenar a una red, mejor nivel de precisión se obtendrá mediante diferentes estructuras profundas de redes entre ellas se encuentran:

2.4.1.1. Redes neuronales recurrentes (RNN)

Una Red neuronal recurrente (RNN) es una de las arquitecturas fundamentales debido a que, a partir de esta, se construye otras arquitecturas de aprendizaje profundo. Una red multicapa se diferencia de una red recurrente en sus conexiones, una RNN tiene conexiones de retroalimentación entre capas o a la misma capa, lo cual permite una memoria de entradas pasadas, es por esto que requieren de mayor tiempo de procesamiento (Oropeza, 2007). Las RNN constan de un amplio conjunto de arquitecturas, todas con su característica principal de retroalimentación y tienen un comportamiento variante en el tiempo, lo cual permite resolver

problemas con no – linealidades temporales significativas, es decir, son ideales para la identificación y clasificación de patrones secuenciales que tengan diferentes probabilidades de suceder en el tiempo

2.4.1.2. Redes neuronales convolucionales (CNN's)

Las redes neuronales convolucionales son un tipo de redes neuronales utilizadas para datos altamente dimensionados tal es el caso de imágenes y video, debido a que son capaces de detectar características simples como por ejemplo detención de bordes, líneas, etc y componer en características más complejas hasta detectar lo que se busca (Wang, Zhang, Wang, & Hu, 2017).

Las redes neuronales convolucionales vienen a solucionar el problema en el cual las redes neuronales ordinarias no escalan bien para imágenes de mucha definición (Mou, Li, Zhang, Wang, & Jin, 2016). Trabajan modelando de forma consecutiva pequeñas piezas de información, y luego combinando esta información en las capas más profundas de la red. Una manera de entenderlas es que la primera capa intentará detectar los bordes y establecer patrones de detección de bordes. Luego, las capas posteriores trataran de combinarlos en formas más simples y, finalmente, en patrones de las diferentes posiciones de los objetos, iluminación, escalas, etc. De esta forma las redes neuronales convolucionales son capaces de modelar complejas variaciones y comportamientos dando predicciones bastantes precisas (Sakhavi, Guan, & Yan, 2015).

2.4.1.2.1. Arquitectura

En general, las redes neuronales convolucionales van a estar construidas con una estructura que contendrá 3 tipos distintos de capas, así también como se puede apreciar en la Figura 16:

- Una capa convolucional, que es la que le da el nombre a la red, La capa convolucional está compuesta por un conjunto de núcleos convolucionales. Estos núcleos están asociados con un área pequeña de la imagen conocida como campo receptivo. Funciona dividiendo la imagen en pequeños bloques y convirtiéndolos en un conjunto específico de pesos con los elementos correspondientes del campo receptivo. (Khan, Sohail, Zahoor, & Qureshi, n.d.), compuesto de un conjunto de filtros entrenables que realizan producto punto con los valores de la capa precedente, es decir, se aplica un mismo filtro a todos los conjuntos que se formen en la imagen 30 y se obtiene una imagen convolucionada, esto permite reducir el número de conexiones y el número de parámetros a entrenar en comparación con una red multicapa full-connected, tal y como se muestra en la Figura 15.

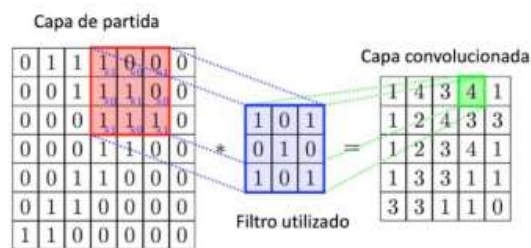


Figura 15. Proceso de la capa convolucional

Fuente: Adaptado de (Calvo, 2017)

- Una capa de reducción o de pooling, la cual va a reducir la cantidad de parámetros al quedarse con las características más comunes, los motivos de las características, que resultan como una salida de la operación de convolución, pueden ocurrir en diferentes ubicaciones en la imagen. Una vez que se extraen las características, su ubicación exacta se vuelve importante siempre que se trate de una posición aproximada con respecto a las demás. (Lee, Gallagher, & Tu, 2016)

La salida de esta capa depende de la elección del usuario, debido a que se puede decidir qué operación realizar. Estas operaciones son: Promediar, maxpooling o elegir el máximo valor y subsampling que consiste en utilizar una función de tangente hiperbólica. La Tabla 3 presenta las funciones aplicables en esta capa:

Tabla 3. Funciones de la Capa Pooling

Tipo de pooling	Implementación	Descripción
Max-pooling	$\max_{rxr}(Xi)$	Máximo de una región cuadrada de tamaño rxr.
Promedio (average) o downsampling	$\text{mean}_{rxr}(Xi)$	Promedio de una región cuadrada de tamaño rxr.
Subsampling	$\tanh\left(\beta \sum_{rxr} \frac{Xi}{r^2} + b\right)$	Promedio de la entrada de una región cuadrada de tamaño rxr y multiplicado por un escalar entrenable (β) y sumado un bias entrenable (b), calculando el resultado a través de una función no lineal.

Fuente: Autoría

La matriz resultante es de dimensiones considerablemente menores que la matriz de características obtenida en la capa de convolución. El objetivo de esta capa es disminuir aún más la carga computacional del sistema y ayuda a la carnetización de la imagen obteniendo rasgos predominantes en ella.

- Una capa llamada función ReLU es la recomendada para redes neuronales convolucionales, esta función de activación se calcula como $f(x) = \max(0, x)$ Se ha demostrado que los algoritmos de entrenamiento convergen más rápido que con las funciones de tangente hiperbólica y sigmoideal, debido a que ReLU es lineal. Sin embargo, algunas neuronas nunca se activan, es decir, solo producen 0, reduciendo la capacidad de representación de la red
- Una capa clasificadora totalmente conectada, la cual nos va dar el resultado final de la red, las redes utilizan generalmente capas completamente conectados en la que cada

pixel se considera como una neurona separada al igual que en una red neuronal regular. Esta última capa clasificadora tendrá tantas neuronas como el número de clases que se debe predecir. (Murphy, n.d.)

Su objetivo es transformar todas las activaciones de red en la capa de salida final a una serie de valores que puede interpretarse como valores de vector de probabilidad entre 0 y 1. Softmax es la más utilizada para la clasificación en las redes neuronales convolucionales (Guachi, y otros, 2018). Esta aplica una distribución de probabilidad categórica basada en una función exponencial, proporcionando un valor cercano a uno para la entrada máxima y un valor cercano a cero para el resto de las entradas.

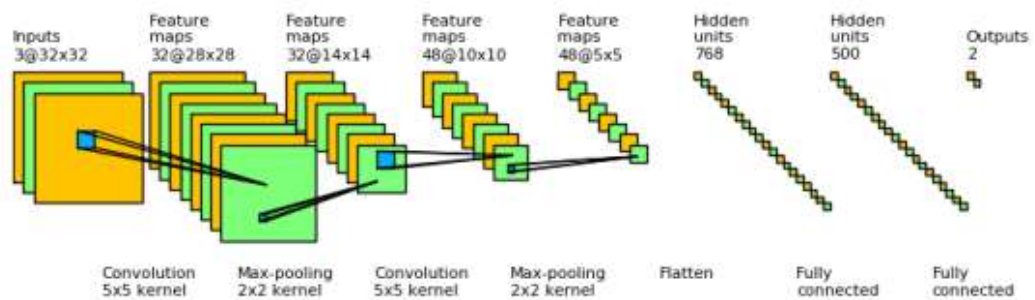


Figura 16. Red neuronal convolucional con múltiples capas agrupadas

Fuente: Adaptado de (QiaoC et al., 2018)

2.5. Sistemas de estimación de posición de articulaciones

La inteligencia artificial estrecha las capacidades de razonamiento, cálculo y capacidad de manejar datos entre las máquinas y los seres vivos. La Visión Artificial tiene como objetivo último la generación de descripciones explícitas, y con significado, de objetos físicos a partir de datos visuales (imágenes).

La estimación de pose humana es una de las principales áreas de investigación en visión artificial. La razón de su importancia es la abundancia de aplicaciones que pueden beneficiarse

de dicha tecnología. Aquí hay una introducción a las diferentes técnicas utilizadas en la estimación de pose humana basada en el aprendizaje profundo.

2.5.1. OpenPose

OpenPose representa el primer sistema de varias personas en tiempo real para detectar conjuntamente los puntos clave del cuerpo humano, la mano, la cara y el pie (en total 135 puntos clave) en imágenes individuales (Qiao et al., 2018). Detecta puntos clave de varias personas en tiempo real 2D:

- Estimación de 15 o 18 o 25 puntos clave cuerpo / pie punto clave.
- Estimación de puntos clave de mano 2x21-punto clave. Actualmente, el tiempo de ejecución depende del número de personas detectadas.
- Estimación de puntos clave de cara de 70 puntos clave. Actualmente, el tiempo de ejecución depende del número de personas detectadas.

2.5.2. Vnect.

VNect fue presentado por un equipo del Instituto Max Planck de Alemania; la Universidad de Saarland de Alemania y la Universidad Rey Juan Carlos de España en el 2017. Captura la pose esquelética 3D global completa de un ser humano de una manera estable y temporariamente consistente con una sola cámara RGB (Mehta et al., 2017). El método combina un nuevo regresor de pose basado en una red neuronal convolucional (CNN) con ajuste de esqueleto cinemático. La novedosa formulación de postura completamente convolucional hace retroceder las posiciones de las articulaciones 2D y 3D conjuntamente en tiempo real y no requiere marcos de entrada muy recortados. (Zulkarnain et al., 2017)

2.5.3. AlphaPose

Es un método popular de arriba hacia abajo de estimación de pose. Los autores afirman que los métodos de arriba hacia abajo generalmente dependen de la precisión del detector de personas, ya que la estimación de pose se realiza en la región donde se encuentra la persona. Por lo tanto, los errores en la localización y las predicciones de cuadro de límite duplicado pueden hacer que el algoritmo de extracción de pose funcione de manera óptima.

Los autores propusieron el uso de la Red de transformadores espaciales simétricos (SSTN) para extraer una región de una sola persona de alta calidad de un cuadro delimitador inexacto. Se utiliza un estimador de pose de una sola persona en esta región extraída para estimar el esqueleto de pose humana para esa persona. Se utiliza una red espacial de transformadores para reasignar la postura humana estimada al sistema de coordenadas de la imagen original. Finalmente, se usa una técnica de supresión paramétrica de pose no máxima (NMS) para manejar el problema de las deducciones de pose redundantes. La característica más destacada de esta solución es que esta técnica se puede extender a cualquier combinación de un algoritmo de detección de personas y un estimador de pose humana de una sola persona.

2.6. Software Libre.

Dentro de las plataformas de software libre que permiten el desarrollo de proyectos que involucran a la visión artificial se citaran a continuación las más importantes.

2.6.1. Python.

Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Python es un lenguaje de programación indispensable para cualquier informático o aprendiz de desarrollo web. Constituye una base sólida para quienes deseen formarse en el

área, porque se trata de un lenguaje dinámico que se implementa en una variedad de plataformas, por lo cual permite crear no solo sitios sino aplicaciones en una amplia variedad de sistemas operativos como iOS, Android, Windows o Mac

2.6.2. OpenCv.

(Open Source Computer Vision) es una librería software open-source de visión artificial y machine learning. OpenCV provee una infraestructura para aplicaciones de visión artificial. OpenCV tiene una licencia BSD, lo que permite utilizar y modificar el código, tiene una comunidad de más de 47000 personas y más de 7 millones de descargas. Es una librería muy usada a nivel comercial, desde Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota, Applied Minds, VideoSurf, Zeite. La librería tiene más de 2500 algoritmos, que incluye algoritmos de machine learning y de visión artificial para usar.

Estos algoritmos permiten identificar objetos, caras, clasificar acciones humanas en vídeo, hacer tracking de movimientos de objetos, extraer modelos 3D, encontrar imágenes similares, eliminar ojos rojos, seguir el movimiento de los ojos, reconocer escenarios

2.6.3. Tensorflow.

Es una biblioteca de código abierto dirigida al aprendizaje automático a través de una serie de tareas. Ha sido desarrollado por Google para satisfacer las necesidades de sistemas capaces de construir y entrenar redes neuronales para detectar y descifrar patrones y correlaciones, análogos al aprendizaje y razonamiento usados por los humanos. Actualmente es utilizado tanto para la investigación como para la producción de productos de Google, reemplazando el rol de su predecesor de código cerrado.

2.7. Estándar IEEE 29148.

La norma IEEE 29148 señala los procesos para la identificación de requerimientos en sistemas y productos de software, especifica el contenido requerido de los elementos de información necesarios y da pautas para el formato de los elementos de información requeridos y relacionados. Por tal razón se aplicará esta norma en el desarrollo del presente proyecto. La norma IEEE 29148 es el resultado de la unión de los siguientes estándares:

- ISO/IEC 12207:2008 (IEEE Std 12207-2008), Systems and software engineering — Software life cycle processes.
- ISO/IEC 15288:2008 (IEEE Std 15288-2008), Systems and software engineering — System life cycle processes.
- ISO/IEC/IEEE 15289:2011, Systems and software engineering — Content of lifecycle information products (documentation).
- ISO/IEC TR 19759, Software Engineering — Guide to the Software Engineering Body of Knowledge (SWEBOK).
- IEEE Std 830, IEEE Recommended Practice for Software Requirements Specifications.

Proporciona un tratamiento unificado de los procesos y productos relacionados con los requerimientos de ingeniería en todo el ciclo de vida de los sistemas. Mediante este estándar se puede realizar el proceso de levantamiento y gestión de actividades relacionadas con los requerimientos que se van a implementar para el desarrollo de sistemas y software, apoyados en lineamientos que dan pautas para aplicar los requisitos y requisitos relacionados con procesos.

El estándar propone elaborar tres documentos: `

- Stakeholder requirements specification document (StRS). Documento de especificación de requisitos de las partes interesadas (ERI): representan los usuarios, clientes y otras partes interesadas en relación con el problema o proyecto a desarrollar, como un conjunto de requisitos para una solución que puede proporcionar los servicios que necesitan las partes interesadas en un entorno definido.
- System requirements specification document (SyRS). Documento de especificación de requisitos del sistema (ERSis): es un documento que describe las características y el comportamiento de un sistema. Incluye una variedad de elementos que intenta definir la funcionalidad deseada requerida por el cliente para satisfacer a sus diferentes usuarios.
- Software requirements specification document (SRS), Documento de especificación de requisitos de software (ERS): Establece el marco que seguirán todos los equipos involucrados en el desarrollo, se utiliza para proporcionar información crítica a múltiples equipos: desarrollo, garantía de calidad, operaciones y mantenimiento, el uso del SRS ayuda a garantizar que se cumplan los requisitos. Y también puede ayudarlo a tomar decisiones sobre el ciclo de vida de su producto, por ejemplo, cuándo retirar una función.

A continuación, se detalla brevemente el contenido del esquema de especificación de requerimientos del estándar IEEE 29148.

Propósito. - Delinear el propósito del software a ser desarrollado.

Alcance. - Describir el alcance del software bajo las siguientes consideraciones:

- a) Identificar los productos de software que se van a producir por nombre por ejemplo Generador de Reportes.
- b) Explicar qué funcionalidad tiene el producto.
- c) Describir la aplicación del software incluyendo los beneficios relevantes, objetivos y metas.

Descripción del producto. - Definir la relación del sistema con otros productos relacionados.

- Interfaces del sistema. - Se enumera cada interfaz del sistema e identifica la funcionalidad del software para lograr que el requisito y la descripción de la interfaz coincidan con el sistema.
- Interfaces de usuario. - Se debe especificar lo siguiente:
 - a) Las características lógicas de cada interfaz entre el producto de software y sus usuarios. Esto incluye aquellas características de configuración, por ejemplo, formatos de pantalla requeridos, diseños de páginas o ventanas, contenido de cualquier informe o menú, disponibilidad de teclas de función programables.
 - b) Todos los aspectos de la optimización de la interfaz con la persona que utiliza o proporciona otro tipo de apoyo al sistema. Esto simplemente puede comprender una lista de cosas que hacer y qué no hacer en cómo el sistema aparecerá al usuario. Un ejemplo puede ser un requisito para la opción de mensajes de error largos o cortos.
- Interfaces de hardware. -Especificar las características lógicas de cada interfaz entre el producto de software y los elementos de hardware del sistema. Esto

incluye características de configuración (número de puertos, conjuntos de instrucciones, dispositivos a utilizar).

- Interfaces de software. - Especificar el uso de otros productos de software requeridos (por ejemplo, un sistema de gestión de datos, un sistema operativo, o un paquete matemático), o que interactúa con otros sistemas de aplicación (por ejemplo, el enlace entre un sistema de cuentas por cobrar y un sistema de contabilidad general).

Interfaces de comunicaciones. - Especificar las distintas interfaces a las comunicaciones, como los protocolos de red local.

Funciones del producto. - Se debe proporcionar un resumen de las principales funciones que el software realizará. Por ejemplo, una ERS para un programa de contabilidad puede usar esta parte para tratar el mantenimiento de cuentas de clientes, la declaración de clientes y la preparación de facturas sin mencionar la gran cantidad de detalle que cada una de esas funciones requiere.

Limitaciones. - Proporcionar una descripción general de cualquier otro elemento que limite las opciones del proveedor.

Distribución de los requisitos. - Asignar los requisitos de software a los elementos de software. Para requisitos que requieran la implementación sobre múltiples elementos de software, o cuando la asignación a un elemento de software es inicialmente indefinida, esto debe ser dicho. Se utilizará una tabla de referencia cruzada por función y elemento de software para resumir los prorrateos.

Requisitos específicos. - Especificar todos los requisitos de software a un nivel de detalle suficiente.

Requerimientos de usabilidad. - Definir los requisitos de usabilidad (calidad en el uso). Los requisitos y objetivos de usabilidad para el sistema incluyen la efectividad medible, la eficiencia y los criterios de satisfacción en contextos específicos de uso.

Requisitos de rendimiento. - Especifique los requisitos numéricos estáticos y dinámicos colocados en el software o en la interacción humana con el software como un todo.

Capítulo 3. Diseño del Sistema

En el presente capítulo se presenta los requerimientos, metodología y el desarrollo del proyecto, detalles que permiten cumplir con los objetivos trazados y las etapas para correcto funcionamiento del sistema.

3.1. Descripción general del sistema

En esta sección se presentará aspectos importantes para el desarrollo del sistema como es el alcance y limitación con el fin de cumplir con las expectativas del presente proyecto.

3.1.1. Alcance del sistema

El sistema medirá los grados articulares en los movimientos de miembro superior en la flexo-extensión de codo, abducción y aducción de hombro mediante la captura de video con una cámara web conectada a un computador, el ángulo dado se indicará en la interfaz de cada extremidad evaluada y se tendrá una base de datos con los resultados y datos del paciente.

3.1.2. Limitaciones del sistema

El algoritmo del sistema con el que se trabaja reconoce todas las articulaciones del cuerpo humano por lo que se dificulta graficar únicamente las de codo y hombro para el caso de estudio del presente proyecto, por ende, se mostrara el esquema en 2D de todo el cuerpo del paciente a evaluar, este sistema no garantiza la recuperación total del paciente, solo es una herramienta de evaluación del rango de movimiento de la articulación lesionada.

3.2. Situación Actual

El Centro de Rehabilitación de la Universidad Técnica del Norte es un centro el cual está destinado a la formación de estudiantes que están cursando los últimos niveles de la carrera de Terapia Física Médica, con el fin de formar profesionales capaces de promover y mejorar la

calidad de vida de los pacientes, y el mejoramiento de la calidad en la atención al usuario a través de la rehabilitación brindada con el uso de equipos y la intervención fisioterapéutica.

Por tal motivo es que el Centro de Rehabilitación de la Universidad Técnica del Norte, facilita la práctica Pre Profesional al estudiante para que puedan aportar los conocimientos adquiridos durante todo el periodo académico, ya que el centro recibe un sin número de pacientes con distintas patologías a tratar.

De acuerdo a datos obtenidos del Centro de Rehabilitación de la Universidad Técnica del Norte en el periodo académico octubre 2018- febrero 2019, en el CENTRO DE REHABILITACIÓN DE LA UNIVERSIDAD TÉCNICA DEL NORTE se han atendido un total de 20 pacientes, correspondientes a un 25% que son mujeres y un 75% hombres. Así como también los usuarios que han sido atendidos en el centro de rehabilitación UTN corresponden a un 15% perteneciente al personal administrativo, un 25% a docentes de la universidad técnica del norte y un 60% estudiantes de distintas carreras que se muestra en la Figura 17 (Entrevista, Lic Jorge Zambrano).



Figura 17. Estadística total de usuarios atendidos en el C.R.UTN

Fuente: Autoría

Por otro lado pese a que un factor importante para la recuperación de un paciente es la constancia con la que este asista a las sesiones de fisioterapia el cuadro estadístico de la Figura 18 nos indica que de los pacientes que acuden a su tratamiento al Centro de Rehabilitación UTN, solo un 35 % de los pacientes han culminado su tratamiento de rehabilitación el cual se les ha dado de alta después de las sesiones establecidas y su respectiva evaluación, un 50% siguen en tratamiento latente, y un 15 % indica que los pacientes que acudieron no terminaron todo el tratamiento de rehabilitación.

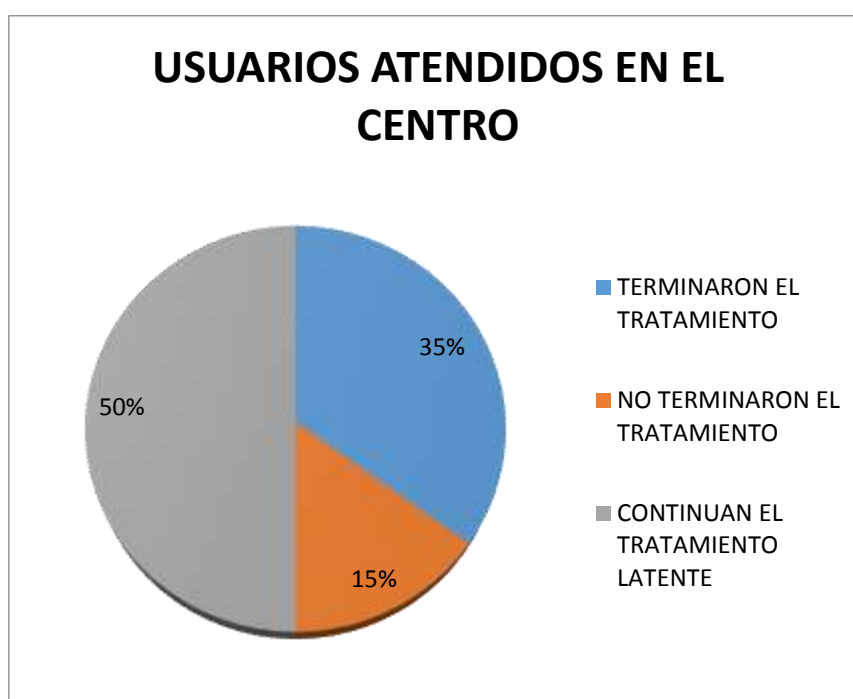


Figura 18. Total de usuarios atendidos

Fuente: Autoría

Es por esto que el análisis realizado en esta sección del documento sustenta la realización del presente proyecto tanto como una herramienta que permita la evaluación de pacientes con el uso de la tecnología, tener una base de datos personalizada de cada paciente, así como también ser un sistema que interactúe con el paciente y su recuperación con evaluaciones que arrojen datos valerosos y en menor tiempo que permitan mantener al paciente motivado a terminar su recuperación.

3.3. Metodología

Para el desarrollo de este estudio se seguirá una metodología de desarrollo denominado “Modelo en V”, el cual cumple con objetivos para garantizar y mejorar la calidad de proyectos y reducir recursos optimizándolos en cada uno de sus 4 niveles, de forma que se pueda validar el desarrollo, implementación, aplicación y funcionamiento del software en base a la aplicación y verificación de requerimientos, procesos que arranca con la parte izquierda de la V iniciando el proyecto, posteriormente con las especificaciones, diseño preliminar y codificación del software, continuando con la parte derecha de la V donde se realizan las pruebas de verificación y validación del sistema así como su mantenimiento y finalización del proyecto. Con esto se facilita en gran manera la gestión del proyecto por la estrecha relación entre cada nivel de la metodología y los diversos tipos de pruebas que permiten focalizar un determinado fallo a lo largo del desarrollo desde su inicio hasta la culminación involucrando al usuario en cada una de las pruebas. En la Figura 19 se muestra el modelo en V y se especifica a detalle cada una de sus fases.

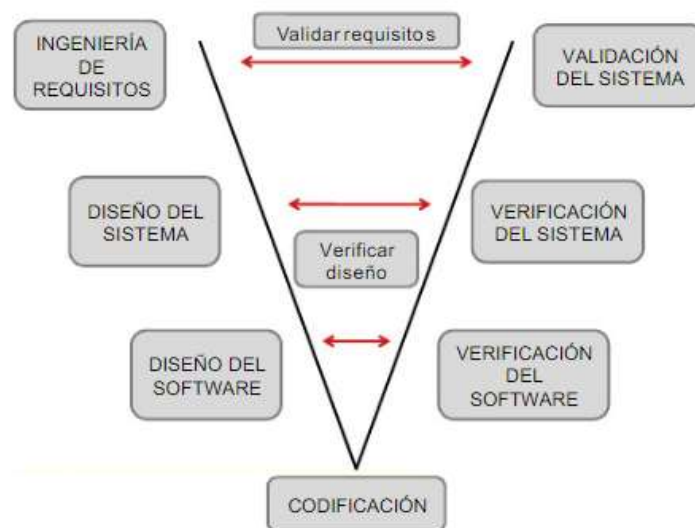


Figura 19. Modelo en V

Fuente: Adaptado de (Herrera, 2010)

El primer nivel de este modelo define parámetros iniciales y requisitos para el desarrollo del proyecto, se compone de un análisis general de especificaciones necesarias para que el ciclo de inicio y fin del proyecto e lleve a cabo correctamente.

El segundo nivel detalla las características funcionales del proyecto, es una de las fases más importantes que se lleva a cabo puesto que de esta depende el éxito y funcionalidad del sistema en desarrollo, de igual manera el análisis de funcionalidad del sistema es realizado a través de pruebas que determinan el correcto trabajo de cada una de las etapas que conforman el proyecto.

El tercer nivel define los componentes de software y hardware que provean de una perspectiva sobre el diseño del proyecto, así como la arquitectura del sistema, además se realiza las pruebas unitarias que consienten en revisar cada una de las etapas del proyecto para certificar con las expectativas generadas al inicio del proyecto.

El cuarto nivel es la fase de implementación desarrollo y programación del software, obteniendo el sistema en cuestión funcionando de acuerdo a lo previsto.

3.4. Desarrollo

En este apartado se detalla los involucrados, requerimientos y desarrollo del sistema, ítems que fueron realizados en base a artículos científicos, bibliografía y revistas referentes al uso de algoritmos que permiten la estimación de posición humana mediante visión artificial y su integración tanto con el hardware y software necesario para el funcionamiento del sistema.

3.4.1. Propósito del sistema

El propósito de este sistema es la detección de articulaciones para la evaluación y medición de ángulos de miembros superiores como codo y hombro a través de visión artificial, mediante el uso de un algoritmo de estimación de posición humana. El objetivo es desarrollar

una herramienta que permita medir de manera más eficiente el proceso de goniometría realizado por los profesionales en fisioterapia, así como también llevar un registro continuo de las sesiones realizadas al paciente en rehabilitación.

3.4.2. Ámbito del sistema

Se ha considerado que el sistema de detección de ángulos articulares en miembros superiores para evaluación en fisioterapia sea implementado para sus pruebas en el Centro de Rehabilitación de la Universidad Técnica del Norte, y sus resultados estén acordes con las mediciones realizadas a los pacientes con el sistema tradicional de los profesionales en fisioterapia.

3.5. Requerimientos del sistema

Para el análisis de los requerimientos del sistema se tomó como referencia el estándar ISO/ IEC / IEEE 29148: 2011 el mismo que contiene directrices para el proceso relacionado a la ingeniería de requisitos, específicamente ha sido desarrollado para ser implementado en los sistemas y productos de software y servicios a lo largo del ciclo de vida, ya que permite definir cada función que se requiere en el proyecto, las restricciones necesarias, y especificar los requisitos y funciones del sistema.

Las tablas que se muestran a continuación se han diseñado teniendo en cuenta las consideraciones del estándar, éstas contienen los requerimientos iniciales más relevantes del sistema, requerimientos de arquitectura y los requerimientos de stakeholders, el objetivo es presentar de una manera concisa dicha información la cual permitirá realizar la selección de software, hardware y algunos aspectos específicos para el diseño del sistema. La Tabla 4 muestra los acrónimos empleados para referirse de forma abreviada a cada requerimiento.

Tabla 4. Definición de acrónimos

ACRÓNIMO	DESCRIPCIÓN
SYSR	Requerimientos iniciales del Sistema
SRSR	Requerimientos de Arquitectura
STSR	Requerimientos de Stakeholder

Fuente: Autoría

El diseño propuesto para cada tabla incluye una columna donde se identifica el número de requerimiento, una columna destinada a la descripción detallada del requerimiento, la siguiente columna está destinada a indicar la prioridad del requerimiento y a su vez se subdivide en Alta, Media y Baja, esta valoración se puede visualizar en la Tabla 5 y es de suma importancia para la selección del software y hardware, se incluye también una columna de relación que se utilizará en el caso de que un requerimiento sea totalmente dependiente de otro.

Tabla 5. Prioridad de los Requerimientos del sistema.

PRIORIDAD	DESCRIPCIÓN
ALTA	Es un requerimiento crítico que debe incluirse durante el desarrollo del sistema. Si no se implementa puede afectar la funcionalidad.
MEDIA	El no incluir este tipo de requerimiento puede afectar la decisión final del sistema, sin embargo, se puede omitir este requerimiento en condiciones de fuerza mayor.
BAJA	Si no se incluye este requerimiento no se espera un impacto significativo en la decisión final del sistema.

Fuente: Adaptado de (Pérez García, 2013)

3.6. Requerimientos iniciales del sistema

En los requerimientos iniciales del sistema (SySR) se definen los límites funcionales del sistema en términos de comportamiento y propiedades del proyecto, que constituyen la

descripción de los requerimientos de interfaces, de performance, de modos y estados, y requerimientos físicos. A continuación, se describen los requerimientos iniciales según su categoría:

- *Requerimientos de Interfaces* Dentro de los requerimientos comunes de las interfaces del sistema se contemplan ciertos parámetros que definen cómo se requiere que el sistema interactúe con las interfaces de usuario, hardware y software para contribuir al buen funcionamiento del sistema. Las interfaces de usuario a considerar en el desarrollo del proyecto son la detección de posición de articulaciones de miembros superiores visualizada a través de un software instalado en un equipo o servidor, en las interfaces de hardware se contemplan la comunicación cableada vía USB de la cámara de video y el servidor donde se ejecuta el software de medición de ángulos articulares.
- *Requerimientos de performance* Dentro de los requerimientos de performance se consideran parámetros que definen el rendimiento del sistema y en qué condiciones una función debe realizarse. Los requerimientos de performance a analizar en el proyecto son: la detección y reconocimiento de articulaciones de miembro superior tales como codo y hombro, obtención y procesamiento de los cuadros de video en tiempo real y registro de información de los ángulos de movilidad tomados del sistema.
- *Requerimientos físicos* Dentro de los requerimientos físicos se contemplan restricciones con respecto a la ubicación de la cámara dependiendo de la plantilla a usar para la medición de ángulos articulares del paciente.

En la Tabla 6 se muestran los requerimientos iniciales del sistema (SySR).

Tabla 6. Requerimientos Iniciales del Sistema.

SYSR					
REQUERIMIENTOS INICIALES DEL SISTEMA					
#	REQUERIMIENTOS	PRIORIDAD			RELACION
		Alta	Media	Baja	
REQUERIMIENTO DE INTERFAZ					
SYSR1	El sistema deberá mostrar la detección de articulaciones (hombro y codo) a través de un software.	X			
SYSR2	El registro de la medición de movilidad articular debe mostrarse en tiempo real.		X		
SYSR3	El sistema debe interactuar con una cámara a través de una conexión cableada USB.	X			
SYSR4	El sistema debe interactuar con una GPU(Unidad de Procesamiento Gráfico).	X			
SYSR5	El sistema debe interactuar con una CPU(Unidad de Central de Procesos).		X		
SYSR6	El sistema requiere conexión a la red eléctrica.	X			
REQUERIMIENTO DE PERFORMANCE					
SYSR7	Reconocimiento codo para el cálculo de la flexo extensión	X			
SYSR8	Reconocimiento de hombro para el cálculo de la abducción y aducción.	X			
SYSR9	Detección y reconocimiento de articulaciones en el Centro de Rehabilitación Física UTN	X			
SYSR10	Detección de miembros inferiores.			X	
SYSR11	Obtención de video en tiempo real de la cámara.		X		
REQUERIMIENTO FISICOS					
SYSR12	El sistema debe estar situado correctamente en un lugar donde no interfiera con la actividad de los pacientes del centro de rehabilitación UTN.	X			
SYSR13	La cámara del sistema debe estar colocada en un lugar estratégico donde se enfoque directamente al paciente y donde aspectos como la luminosidad y ruido en sean mínimos.	X			
SYSR14	La cámara del sistema debe estar situada a una altura apropiada dependiendo de la estatura del paciente.	X			

Fuente: Autoría

3.7. Requerimientos de arquitectura

En los requerimientos de arquitectura (SRS_H) se definen los requerimientos de hardware, software y el sistema eléctrico. A continuación, se describen los requerimientos de arquitectura considerados para el diseño del sistema:

- *Requerimientos de Diseño* Dentro de los requerimientos de diseño se considera parámetros de la ubicación de la cámara y del servidor para la recolección y procesamiento de la información.
- *Requerimientos de Hardware* Dentro de los requerimientos de hardware se contemplan las especificaciones técnicas de los elementos que se van a emplear como son: una CPU o GPU para el tratamiento de imágenes, una conexión cableada vía USB entre el servidor y una cámara web, un disco duro para el almacenamiento de los registros, una cámara web de alta resolución, y la compatibilidad del hardware del servidor con cualquier sistema operativo.
- *Requerimientos de Software* Dentro de los requerimientos de software se considera la compatibilidad del sistema con las bibliotecas de visión artificial de OpenCV, bibliotecas de aprendizaje automático, software de base de datos y de diseño de interfaces de usuario.

Las directrices descritas en los requerimientos de Arquitectura se muestran en la Tabla 7 y son necesarias para la selección del hardware y software a emplear en el proyecto.

Tabla 7. Requerimientos de Arquitectura.

SRSH					
REQUERIMIENTOS DE ARQUITECTURA					
#	REQUERIMIENTOS	PRIORIDAD			RELACION
		Alta	Media	Baja	
REQUERIMIENTO DE DISEÑO					
SRSH1	La cámara debe estar empotrada en una base que permita la medición de ángulos articulares de hombro y codo.	X			SRSH2
SRSH2	El cable de conexión tipo USB debe ser extendido de manera correcta y que no interfiera al lugar donde se instalara el sistema.	X			SRSH1
SRSH3	El servidor que aloja el sistema debe estar situado en un lugar espacioso y con ventilación.		X		
REQUERIMIENTO DE HARDWARE					
SRSH4	El sistema requiere una unidad central de procesamiento (CPU) que permita el tratamiento de imágenes en tiempo real.		X		
SRSH5	El sistema requiere una unidad de procesamiento grafico (GPU) que permita el tratamiento de imágenes en tiempo real.	X			
SRSH6	Se requiere una placa madre con conexión vía puerto USB para la recepción del flujo de video proveniente de la cámara web.	X			
SRSH7	El sistema requiere una GPU que permita el uso de modelos de aprendizaje automático y profundo.	X			
SRSH8	El sistema requiere una GPU con tecnología de procesamiento de computo paralelo CUDA.	X			
SRSH9	Se requiere de una cámara de alta resolución para la efectividad de las tareas de detección y reconocimiento de articulaciones.		X		
SRSH10	Las imágenes de la cámara deben ser procesadas con rapidez en la GPU.	X			
SRSH11	Se requiere disco duro de almacenamiento de gran capacidad para el almacenamiento de los registros.		X		
SRSH12	El procesador del servidor debe ser compatible con la arquitectura de cualquier sistema operativo.	X			
SRSH13	El procesador debe ser de última generación.		X		

SRSH14	El procesador debe tener una tarjeta de gráficos integrados sobre el mismo chip	X
SRSH15	Se requiere una cámara con una conexión vía puerto USB.	X
REQUERIMIENTO DE SOFTWARE		
SRSH16	Se requiere de un sistema operativo y lenguaje de programación de código abierto.	X
SRSH17	Se requiere compatibilidad con la librería OpenCV y la cámara.	X
SRSH18	Se requiere que el software permita ejecutar el código de visión artificial en tiempo real en el servidor.	X
SRSH19	Se requiere de un sistema operativo que ejecute con rapidez los hilos de procesamiento del sistema.	X
SRSH20	Se requiere software que permita usar los recursos de la GPU en el uso de modelos de visión artificial.	X
SRSH21	Se requiere de un software de base de datos no relacional que permita obtener un bajo impacto o costo en el rendimiento del sistema.	X
SRSH22	Se requiere de un software de diseño de interfaces de usuario (GUI) para la visualización de resultados.	X
SRSH23	Se requiere compatibilidad de software con bibliotecas de aprendizaje automático y profundo.	X
SRSH24	Se requiere compatibilidad de software con tecnología de computo paralelo CUDA.	X

Fuente: Autoría

3.8. Requerimientos de stakeholders

Los requerimientos de stakeholders comprenden a un grupo o individuo que tiene un interés directo en el resultado obtenido por el desarrollo del proyecto. La definición de los requerimientos del stakeholder (StSR) tiene como finalidad identificar los requisitos de los interesados por el sistema. Específicamente se analizan un conjunto de requerimientos operacionales y de usuario que tienen que ver con la interacción directa de los usuarios

involucrados con el sistema. La Tabla 9 muestra los requisitos empleados en los requerimientos de stakeholder.

En la Tabla 8 se listan a los implicados o stakeholder que se toman en cuenta para el desarrollo de este proyecto:

Tabla 8. Lista de Stakeholders del sistema

LISTA DE STAKEHOLDERS
1. Estudiantes y docentes de la FICA
2. Personal administrativo de la FICA
3. Msc. Jaime Michilena (director del trabajo de titulación)
4. Msc. Edgar Maya (codirector del trabajo de titulación)
5. Msc. Daniela Zurita (codirectora del trabajo de titulación)
6. José Luis Jarrín (desarrollador del proyecto)

Fuente: Autoría

Tabla 9. Requerimientos de Stakeholder.

STSR					
REQUERIMIENTOS DE STAKEHOLDERS					
#	REQUERIMIENTOS DE USO	PRIORIDAD			RELACION
		Alta	Media	Baja	
REQUERIMIENTO OPERACIONALES					
STSR1	El sistema debe implementarse en las instalaciones del Centro de Rehabilitación de la UTN	X			
STSR2	Adquisición de datos para la aplicación del algoritmo	X			
REQUERIMIENTO DE USUARIOS					
STSR4	Los usuarios directos del sistema pueden manipular las opciones del sistema		X		
STSR5	Los usuarios indirectos del sistema deben mirar hacia la cámara al menos una vez	X			
STSR6	Para la obtención de muestras de los usuarios, se debe capturar video de frente y de lado.	X			

Fuente: Autoría

3.9. Selección de hardware y software

Para la selección de los componentes de hardware y software se realiza una tabla comparativa de especificaciones según los atributos de los requerimientos de Stakeholders, Sistemas y de Arquitectura, se evalúa un componente y mediante dicha tabla se obtiene una valoración de los atributos correspondientes (StRS, SySR, SRSH) y al final se elige al componente de mayor puntuación. Definiendo que un valor de puntuación de “1” cumple con el requerimiento y de una puntuación de “0” cuando no cumple con el requerimiento.

3.9.1. Hardware

La selección del Hardware se realiza de acuerdo a los requerimientos de hardware establecidos en la Tabla 10 sobre Requerimientos de Arquitectura. Específicamente se seleccionará la unidad central de procesamiento (CPU), la unidad de procesamiento gráfico (GPU), y la cámara a emplearse para el análisis de las imágenes por visión artificial,

3.9.1.1. CPU.

Para la elección del procesador se eligieron 4 opciones que se adaptan a las necesidades del proyecto. La Tabla 10 muestra la valoración de cada requerimiento para la elección del procesador (CPU).

Tabla 10. Elección de CPU

HARDWARE	REQUERIMIENTOS					VALORACION TOTAL
	SRSH4	SRSH13	SRSH14	SRSH15	SRSH16	
AMD/RYZEN 7 1800X	1	1	1	1	0	4
AMD/ RYZEN 7 1700X	1	1	1	1	0	4
INTEL/CORE I7-7700K	1	1	1	1	1	5
INTEL/CORE I7-8700	1	1	1	1	1	5

1 Cumple

0 No cumple

Elección: en la selección del procesador según la tabla de requerimientos de arquitectura, se concluye que es óptimo el uso de procesadores intel core i7 7700k/8700, ya que cumple con todos los requerimientos de una arquitectura robusta para el despliegue de aplicaciones de visión artificial, esto quiere decir que cualquiera de los procesadores puede realizar un tratamiento a las imágenes, pero de manera ineficiente comparado con una GPU dedicada.

Fuente: Autoría

A continuación, en la Tabla 11 se muestran las características técnicas principales del procesador INTEL/Core i7-8700, las especificaciones completas se pueden obtener en el siguiente enlace: <https://ark.intel.com/es/products/126686/Intel-Core-i7-8700-Processor-12M-Cache-up-to-4-60-GHz->

Tabla 11. Características del CPU

ESPECIFICACIONES	PROPIEDADES
Cantidad de núcleos	6
Cantidad de subprocesos	12
Frecuencia básica del procesador	3.20 Ghz
Frecuencia turbo máxima	4.60 Ghz
Caché	12 MB smartcache
Gráficos incorporados	Gráficos HD Intel® 630

Fuente: (Intel Corporation)

3.9.1.2. GPU.

Para la elección del procesador se eligieron 2 opciones que se adaptan a las necesidades del proyecto. La Tabla 12 muestra la valoración de cada requerimiento para la elección de la tarjeta de video (GPU).

Tabla 12. Elección de la GPU

HARDWARE					VALORACION TOTAL
	SRSH5	SRSH8	SRSH9	SRSH11	
NVIDIA GEFORCE GTX 1080	1	1	1	1	4
AMD RADEON RX 480	1	0	0	1	2

1 Cumple
0 No cumple

Elección: en la selección de la tarjeta de video según la tabla de requerimientos de arquitectura, se concluye que es óptimo el uso de un módulo nvidia geforce gtx 1080, ya que cumple con todos los requerimientos de arquitectura enfocados al procesamiento de imágenes mediante tecnología cuda (arquitectura unificada de dispositivos de cómputo), la cual se encarga de realizar cálculos paralelos entre cpu y gpu de manera repartida y eficientemente rápida, proporcionando un incremento extraordinario del rendimiento del sistema.

Fuente: Autoría

A continuación, en la Tabla 13 se muestran las características técnicas principales de la tarjeta de video, las especificaciones completas se pueden obtener en el siguiente enlace:
<https://www.nvidia.com/en-us/geforce/products/10series/geforce-gtx-1080/>

Tabla 13. Características de la GPU

ESPECIFICACIONES	PROPIEDADES
Cantidad de núcleos CUDA	2560
Reloj base	1708 mhz
Velocidad de memoria	10 gbps
Memoria de video dedicada	8 gb gddr5x
Ancho de interfaz de memoria	256 bits
Ancho de banda de memoria	320 /s

Fuente: (Nvidia Corporation)

3.9.1.3. Cámara

Para la elección de la cámara se seleccionó 3 opciones, las cuales se adaptaban a las necesidades del proyecto. La Tabla 14 muestra la elección de la cámara y el modo de valoración se realizó de igual forma que para el sistema embebido.

Tabla 14. Elección de la Cámara

HARDWARE	VALORACION TOTAL		
	SRSH5	SRSH11	
HIKVISION CAMERA IP DS-2CD2142FWD-I(W)(S)	1	0	1
DAHUA IPC HDW2100	1	0	1
WEBCAM CAMERA LOGITECH C270	1	1	2
1 CUMPLE			
0 NO CUMPLE			

Elección: en la selección de la cámara web según la tabla de requerimientos de arquitectura, se concluye que es óptimo el uso de una cámara web marca: Logitech, ya que ofrecen transmisión de datos a gran velocidad y video con resolución 1080p y 720p.

Fuente: Autoría

3.9.2. Software

Una vez seleccionado el hardware se procede a la selección del software, la cual se la realiza en base a los requerimientos de software establecidos en la Tabla 4 sobre Requerimientos de Arquitectura. En esta sección se consideran algunas alternativas en cuanto a la codificación del sistema sobre una plataforma de programación adecuada y bien documentada. Debido a que la naturaleza de este proyecto se centra en capacitación e implementación de modelos de aprendizaje automático y profundo casi en su totalidad, es crucial seleccionar con el mayor cuidado posible, ya que, al ser una investigación en constante avance dentro del campo de la inteligencia artificial algunas plataformas conllevan mayor dificultad de implementación que en otras y son compatibles

con tecnología de procesamiento paralelo CUDA. Además, debe tener compatibilidad con la biblioteca de visión artificial OpenCV.

3.9.2.1. Software de programación

Para la selección del software de programación se seleccionaron 4 opciones, las cuales se adaptan parcialmente a los requerimientos del proyecto para su codificación. La Tabla 15 muestra la valoración de cada requerimiento para la elección del software de programación.

Tabla 15. Selección del Software de Programación

HARDWARE	VALORACION						TOTAL
	SRSH 19	SRSH 20	SRSH 21	SRSH 23	SRSH 26	SRSH 27	
PYTHON	1	1	1	1	1	1	6
MATLAB	0	1	1	1	1	1	5
VISUAL STUDIO	0	1	1	1	1	1	5
JAVA	1	1	1	1	0	1	4
1 Cumple							
0 No cumple							

Elección: en la selección del software idóneo según los requerimientos del software especificados en la tabla de requerimientos de arquitectura la mayor valoración fue obtenida por python. Este software de programación es altamente ideal para el diseño de aplicaciones de inteligencia artificial, ya que tiene compatibilidad con las más relevantes bibliotecas de aprendizaje automático y profundo (tensorflow) utilizadas en este proyecto.

Fuente: Autoría

3.10. Diseño del Sistema

En este apartado se empieza con el desarrollo del sistema, por lo cual se toma en cuenta detalles descritos posteriormente tales como la situación actual, alcance, limitación y

requerimientos que son parte del análisis inicial para empezar con el sistema de medición de ángulos articulares mediante visión artificial.

Se incluye diagramas de bloques, arquitectura del sistema, así como también la funcionalidad e implementación del algoritmo de estimación de posición humana y scripts pertinentes para la implementación y pruebas unitarias del sistema para demostrar su funcionalidad.

3.10.1. Arquitectura del Sistema

En la Figura 20 se muestra el sistema de medición de ángulos articulares, en el cual se tiene la conexión de la cámara web que se encarga de la toma de datos a partir de la detección de video al paciente en cuestión a ser evaluado, esta cámara se encuentra conectada vía USB al CPU que constará de una tarjeta de video NVIDIA GTX1080 que será el GPU encargado de procesar la imagen recibida a través del sistema de detección de estimación de posición humana Openpose, que mostrara un esquema en 2D del paciente y permitirá la evaluación del ángulo de flexo extensión de codo así como también la abducción y aducción de hombro.



Figura 20. Arquitectura del funcionamiento del sistema

Fuente: Autoría

3.10.2. Diagrama de Bloques

El diagrama de bloques se muestra la integración entre el hardware y software en cada una de las etapas del sistema previamente detallado de acuerdo al funcionamiento deseado en este proyecto, cada bloque muestra características independientes que deben realizar y determinaran el éxito del proyecto.

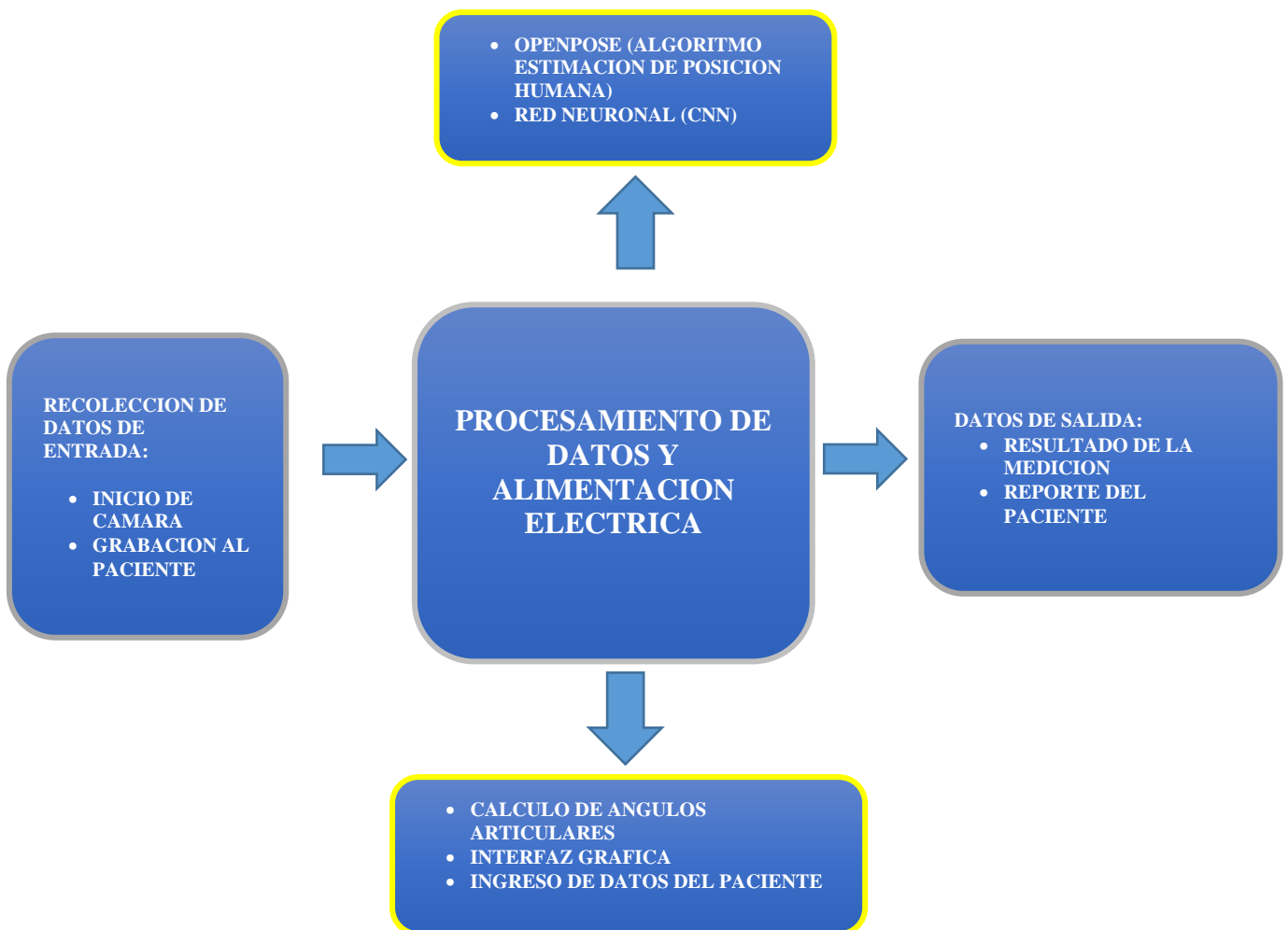


Figura 21. Diagrama de bloques del sistema

Fuente: Autoría

La Figura 21 muestra el diagrama del sistema que está compuesto por 3 bloques, los cuales se encuentran interconectados el uno al otro, se detalla al bloque 2 que representa al procesamiento de datos que está compuesto por dos sub bloques.

3.10.2.1. Bloque1. Recolección de datos

La primera fase del sistema se encuentra representada en este bloque, para la recolección de datos se trabaja con una cámara web Logitech C720 de 720p que trabaja a 30 Fps (Frames por segundo) misma que está conectada vía puerto USB. Puesto que el sistema a desarrollar se lo realiza en el sistema operativo Ubuntu 16.04, la cámara debe ser habilitada a través de la instalación de un paquete que permite el reconocimiento del hardware, puesto que la marca usada no tiene soporte para plataformas Linux es suficiente con la habilitación del paquete “apt-get install cameras -y”.

Una vez instalado el paquete hacemos uso de la aplicación por defecto de Ubuntu 16.04 para comprobar que la cámara funciona y captura tanto imágenes como video, ya habilitada la cámara se puede hacer uso de esta para la implementación del sistema de visión artificial.

- **Obtención de la imagen**

Para la obtención de información se procura que las imágenes a capturar tengan una buena luminosidad, no cuente con obstrucciones o a su vez imperfecciones y ningún tipo de cambio en su escala. La imagen que proporciona la cámara web será una captura RGB. Para que el sistema pueda ejecutar estas características de calidad se debe tener presente que la cámara a utilizar tiene como principal objetivo transformar las señales luminosas en señales analógicas. Una cámara cuenta con un sensor digital que está conformado de celdas fotosensibles, las cuales analizan la luz que arriba a las celdas y es ahí donde se capta una pequeña escena de la imagen que se requiere obtener en forma de señales luminosas para más luego poder transformarlas en señales analógicas, además poseemos la parte óptica que realiza la función de preparar elementos a una distancia focal prudencial. La cámara Logitech C270 que fue seleccionada anteriormente para este proyecto tiene la capacidad de conectarse a cualquier ordenador vía puerto USB con el fin de producir fotografías y también videos HD.

Este elemento posee un enfoque de foco fijo compatibles con 720p. La conexión de la Logitech C270 se la realiza por medio de un cable USB de 15 cm que va dirigido directamente al puerto USB del ordenador que será el encargado del procesamiento del video.

3.10.2.2. *Bloque 2. Procesamiento*

La estimación de la pose humana en dos dimensiones tiene como objetivo localizar los puntos anatómicos de una persona, se ha abordado como un método que toma, como entrada, una imagen a color de tamaño xyz y produce, como salida, las ubicaciones en dos dimensiones de los puntos anatómicos de cada persona en la imagen como muestra la Figura 22.

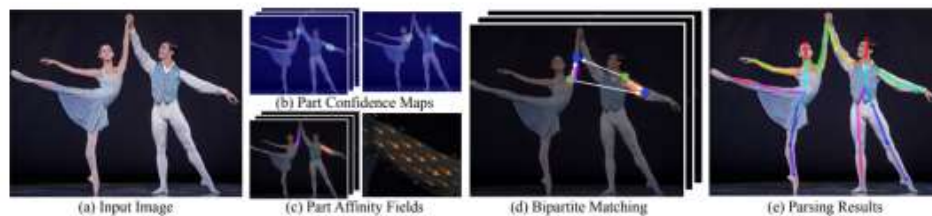


Figura 22. Metodología del algoritmo de estimación de pose humana

Fuente: Adaptado de (Martinez et al., n.d.)

- **Algoritmo de Estimación de Pose Humana**

En el desarrollo del presente proyecto se hace uso sistema Openpose, la estimación de la pose se refiere a técnicas de visión por computadora que detectan figuras humanas en imágenes y videos, de modo que uno podría determinar, por ejemplo, dónde aparece el codo de alguien en una imagen. Para ser claros, esta tecnología no reconoce quién está en una imagen. El algoritmo es simplemente estimar dónde están las articulaciones clave del cuerpo.

El diagrama de bloques del funcionamiento de Openpose se resume en la Figura 23:

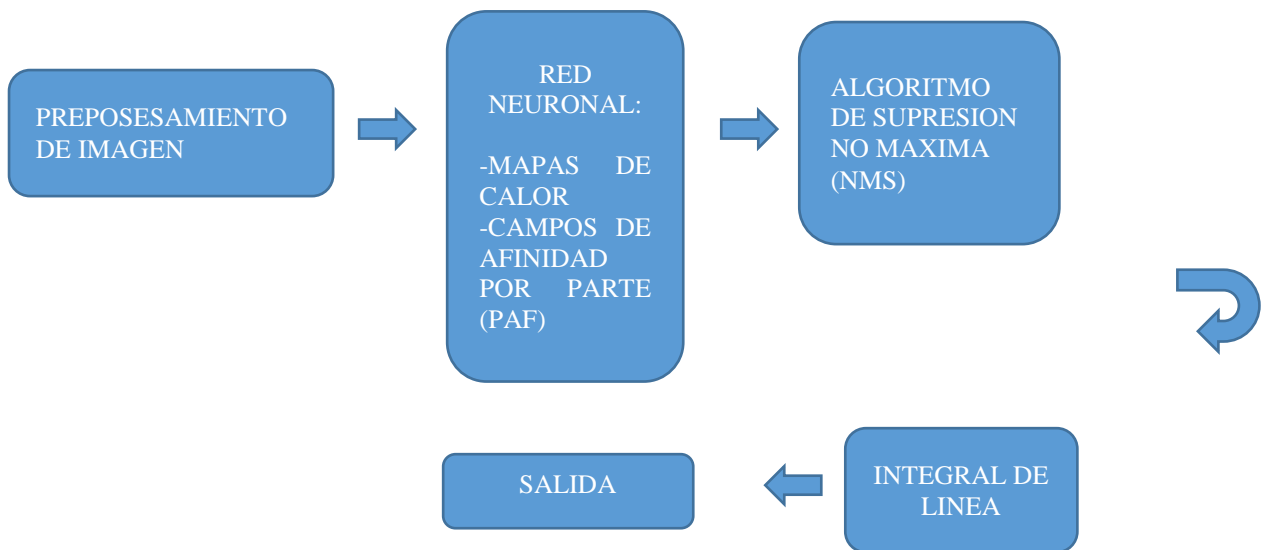


Figura 23. Diagrama de bloques de Openpose

Fuente: Autoría

Openpose reúne tres conjuntos de modelos entrenados: uno para la estimación de la postura del cuerpo, otro para las manos y el último para las caras.

Cada conjunto tiene varios modelos dependiendo del conjunto de datos que se encuentren en (COCO Common Objects in Context) o MPII (Conjunto de datos de estimación articulada), debido a que el poder del aprendizaje profundo es proporcional a la calidad, la cantidad y la disponibilidad de los conjuntos de datos utilizados.

El algoritmo está basado en el reconocimiento de partes y pares para formar el esqueleto en 2D de la persona identificada en 18 puntos mismas que difieren una de la otra como muestra la Figura 22, una parte del cuerpo es un elemento del cuerpo, como el cuello, el hombro izquierdo o la cadera derecha, un par es un par de partes, es decir una conexión entre partes, se podría decir extremidad, pero la conexión entre la nariz y el ojo izquierdo definitivamente no es una extremidad. Se podrá tener conexiones entre orejas y hombros que no existen en la vida real.

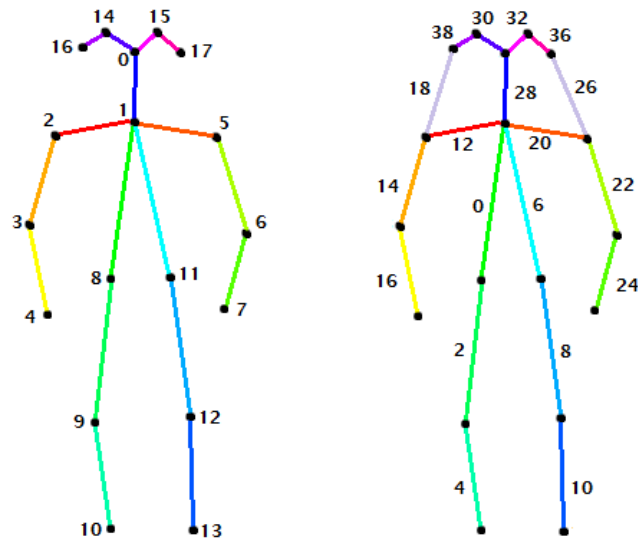


Figura 24. Pares y partes para el conjunto de datos del algoritmo

Fuente: Adaptado de (Pinheiro, Collobert, & Dollár, 2017)

- **Arquitectura de la Red**

Primero la imagen se pasa a través de una red de línea de base para extraer mapas de características, son utilizadas las primeras 10 capas del modelo VGG-19 que es una red neuronal convolucional (CNN) pre entrenada que ha sido entrenada en aproximadamente 1.2 millones de imágenes (Simonyan & Zisserman, 2015).

La red está dividida en dos CNN de múltiples etapas, ilustradas en la Figura 23, el modelo toma como entrada una imagen en color de tamaño $h \times w$ y produce, como salida, una matriz de matrices que consiste en los mapas de confianza de los puntos clave y los mapas de calor de afinidad parcial para cada par de puntos clave.

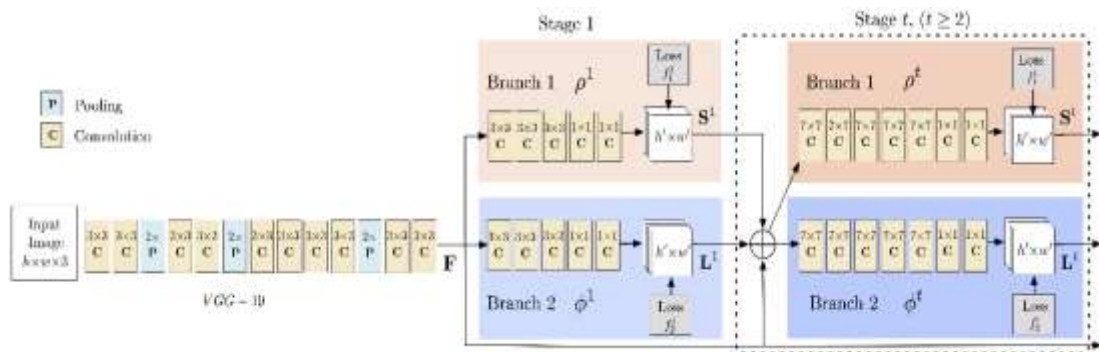


Figura 25. Arquitectura del modelo de estimación de pose

Fuente: Adaptado de (Qiao et al., 2018)

La red está dividida en dos ramas: la rama superior, en color beige, predice los mapas de confianza, mientras que la rama inferior, en azul, es la que predice los campos vectoriales de afinidad entre las partes.

La arquitectura de red anterior consta de dos etapas como se explica a continuación:

- **Etapas 0:** las primeras 10 capas de VGGNet se utilizan para crear mapas de características para la imagen de entrada.

VGG-19 es una red neuronal entrenada convolucional, desde Visual Geometry Group, Departamento de Ingeniería de la Universidad de Oxford. El número 19 representa el número de capas con pesos entrenables. 16 capas convolucionales y 3 capas completamente conectadas. Las redes VGG, , siguen el esquema arquetípico de las redes convolucionales clásicas: una serie de convoluciones, max pooling, capas de activación y finalmente algunas capas de clasificación completamente conectadas.

Esta etapa recoge la imagen y es analizada por la red neuronal convolucional diseñada de acuerdo al modelo VGG-19 que genera un mapa de características de interés para el entrenamiento de red F que es la entrada de cada una de las etapas de cada rama del modelo como se muestra en la Figura 26.

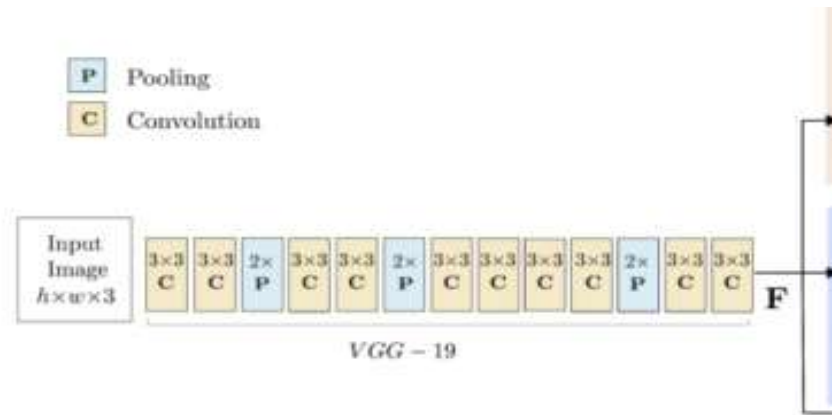


Figura 26. Etapa 1 modelo VGG-19

Fuente: Adaptado de (Ortega, 2015)

En la primera etapa, la produce un conjunto de mapas de confianza $S^1 = \rho^1(\mathbf{F})$ y un conjunto de campos de afinidad de partes $L^1 = \phi^1(\mathbf{F})$ donde:

ϕ^1 y ρ^1 Representan las redes neuronales convolucionales de cada etapa como se muestra en la Figura 27.

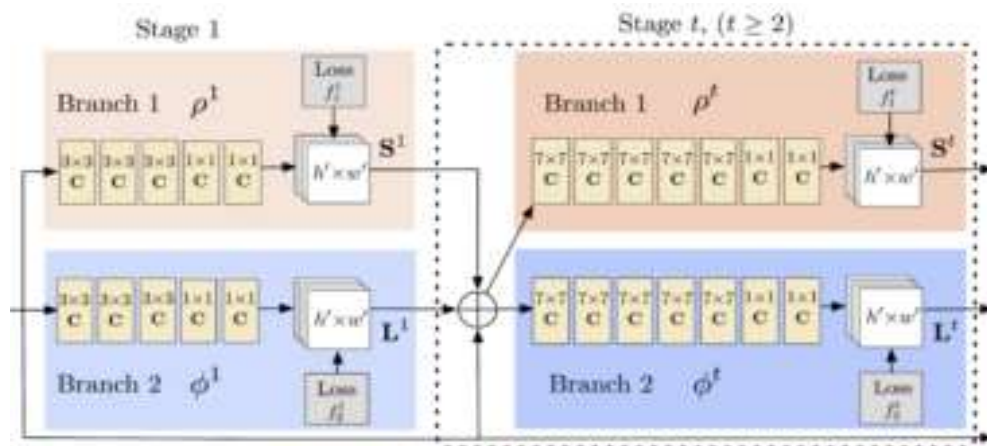


Figura 27. Etapas y ramas del modelo Openpose

Fuente: Adaptado de (Ortega, 2015)

Para que la red neuronal aprenda a generar los mejores conjuntos de \mathbf{S} y \mathbf{L} , la función de pérdida está definida por la Ecuación 1, se aplica una función de pérdida al final de cada etapa mencionada posteriormente.

$$f = \sum_{t=1}^T (f_s^t + f_L^t)$$

Ecuación 1. Función de pérdida de la red neuronal

Fuente: Adaptado de (Meng, Shengbing, Yi, & Meng, 2014)

Estas funciones permiten el refinamiento de las predicciones del modelo entre las predicciones estimadas en conjunto con los datos dados al modelo entre los mapas de confianza y los campos de afinidad por parte en un conjunto de predicciones de múltiples etapas t , dentro del aprendizaje automático se define una función de pérdida a la función que cuantifica el error cuadrático cometido durante la clasificación de un objeto.

- **Etapas 1:** se utiliza una CNN de múltiples etapas de 2 ramas donde:

La primera rama predice un conjunto de mapas de confianza o mapas de calor S a partir de los datos etiquetados a la entrada de la imagen en la red neuronal, es una matriz que almacena la confianza que tiene la red de que cierto píxel contiene cierta parte. Hay 18 (+1) mapas de calor asociados con cada una de las partes e indexados como se mostró en los esquemas del esqueleto formado por los pares y partes, extrayendo la ubicación de las partes del cuerpo de estas 18 matrices, los mapas de calor no precisamente miden temperatura, por ejemplo, el mapa de calor para la cadera como se muestra en la Figura 28 tiene valores altos en todas las ubicaciones donde hay una cadera.

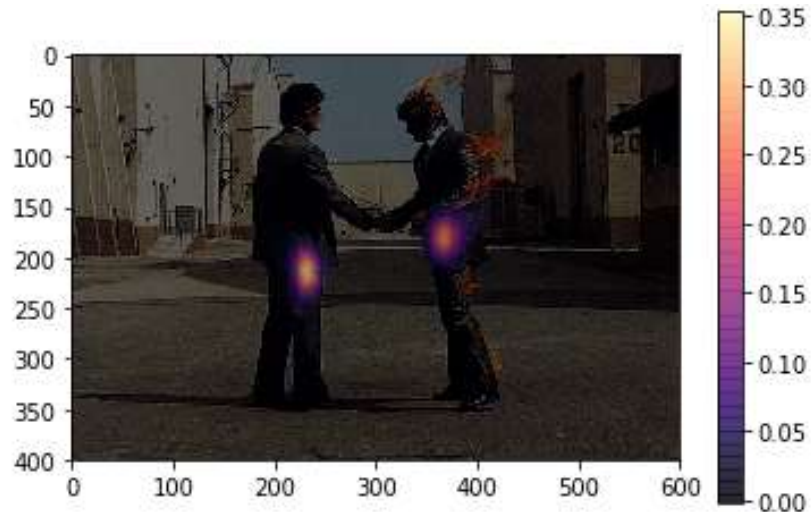


Figura 28. Mapas de calor

Fuente: Adaptado de (Kim & Reiter, 2017)

Cada mapa de confianza realiza una representación en dos dimensiones de la certeza de que una determinada parte del cuerpo j se encuentre ubicada en un determinado pixel de la imagen de una persona k .

El mapa de confianza generado en una determinada ubicación p se lo representa en la Ecuación 2:

$$S_{j,k}(p) = \exp\left(-\frac{p - x_{j,k}}{\sigma^2}\right)$$

Ecuación 2. Representación matemática para la generación de mapas de confianza

Fuente: Adaptado de (Viola & Jones, 2004)

Donde:

σ^2 controla la extensión de los picos generados si existiesen más de una persona.

- La segunda rama predice un conjunto de campos vectoriales 2D de los campos de afinidad por parte (PAF), que codifican el grado de asociación entre partes o puntos clave.

Las matrices 20 a 57 son las matrices PAF. En la Figura 29, se muestra la afinidad parcial entre el cuello y el hombro izquierdo existiendo una gran afinidad entre las partes que pertenecen a la misma persona.



Figura 29. Campo de afinidad de partes

Fuente: Adaptado de (Ortega, 2015)

Como se mostró en la figura anterior cada pixel se junta para representar en un vector una extremidad que tiene un campo de afinidad que une sus dos partes del cuerpo asociadas, en la Figura 30 se muestra una extremidad donde se puede observar las posiciones de las partes de cuerpo representadas por los puntos $x_{j1,k}$ y $x_{j2,k}$ de las partes del cuerpo $j1$ y $j2$ de una extremidad c de una persona k que forma un vector con un punto p que está posicionado en la extremidad de la persona, en caso que el resto de p no pertenezcan a la extremidad su valor inmediato es igual a cero.

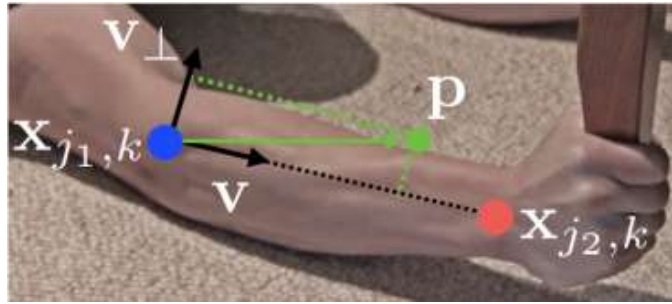


Figura 30. Asociación entre partes para el reconocimiento del brazo

Fuente: Autoría

El campo de afinidad por partes es calculado entre partes de datos previamente etiquetados promediando los campos de las personas en una imagen representada por la Ecuación 3:

$$L_c(p) = \frac{1}{n_c(p)} \sum_k L_{c,k}(p)$$

Ecuación 3. Representación matemática de los PAF

Fuente: Adaptado de (Meng, Shengbing, Yi, & Meng, 2014)

Donde:

$n_c(p)$ es el número de vectores distintos de cero en el punto p para las k personas, es decir el promedio de pixeles donde se superponen extremidades de diferentes personas.

C es el número total de extremidades, para el conjunto de datos es un número de 19 que contiene la base de datos COCO como indica la Figura 31.

```

CocoPairs = [
    (1, 2), (1, 5), (2, 3), (3, 4), (5, 6), (6, 7), (1, 8), (8, 9), (9, 10), (1, 11),
    (11, 12), (12, 13), (1, 0), (0, 14), (14, 16), (0, 15), (15, 17), (2, 16), (5, 17)
] # = 19

```

Figura 31. Script donde se encuentran las tuplas del conjunto de datos COCO

Fuente: Autoría

- Ya ubicados los mapas de calor se debe transformar la confianza en certeza extrayendo ubicaciones de piezas de un mapa de calor, aplicando el algoritmo de supresión no máxima (NMS). Para una sola persona, es muy fácil encontrar la ubicación de cada punto clave con solo encontrar el máximo del mapa de confianza, como se muestra a continuación los pasos que realiza el algoritmo:
 - Comienza en el primer píxel del mapa de calor.
 - Rodea el píxel con una ventana y encuentra el valor máximo en esa área.
 - Sustituye el valor del píxel central por ese máximo
 - Desliza la ventana un píxel y repite estos pasos una vez que se haya cubierto todo el mapa de calor compare el resultado con el mapa de calor original. Esos píxeles que permanecen con el mismo valor son los picos que se busca. Suprime los otros píxeles configurándolos con un valor de 0.

Una vez que se ha encontrado los candidatos para cada una de las partes del cuerpo, se necesita conectarlos para formar pares.

Aquí es donde los PAF toman importancia. El modelo calcular la integral de línea a lo largo del segmento que conecta cada par de candidatos parciales, sobre los PAF correspondientes (x e y) para ese par. Una integral de línea mide el efecto de un campo dado en este caso, los campos de afinidad de parte a lo largo de una curva dada como son las posibles conexiones entre candidatos de parte.

Finalmente, lo que se obtiene es una colección de conjuntos humanos, donde cada humano es un conjunto de partes, donde cada parte contiene su índice, sus coordenadas relativas y su puntaje como indica la Figura 32.



Figura 32. Muestra del esqueleto en 2D de un ser humano

Fuente: Autoría

- **Cálculo de ángulos articulares**

En este apartado se realiza un análisis del cálculo de ángulos articulares haciendo uso del algoritmo de estimación de pose humana, el propósito de este bloque es hacer uso del código libre que Openpose brinda en su licencia para desarrollos en investigaciones académicas.

Haciendo uso del hardware previamente elegido se procede a instalar las herramientas que permitirán el procesamiento e implementación del algoritmo, es importante destacar el uso de los drivers actualizados de la tarjeta de video para la instalación de los drivers CUDA, necesarios para la activación de los procesos de GPU que permitirán el uso tanto de Tensorflow, OpenCv y Python de forma óptima, puesto que no se hace uso del CPU en gran medida para el procesamiento y uso de Openpose.

Python permite trabajar con ambientes virtuales debido a la integración en las versiones de bibliotecas y herramientas para el desarrollo de varios proyectos, para el caso de este sistema se ha empleado el ambiente virtual Openpose como muestra la Figura 33 que indica el comando para trabajar en el mencionado ambiente y la Figura 34 que muestra que su ejecución, en el cual están instalados todos los paquetes para que el ejecute el sistema.



```
jrt@JRT977: ~/Escritorio/tesis/tf-pose-estimation
jrt@JRT977:~/Escritorio/tesis/tf-pose-estimation$ workon openpose
```

Figura 33. Ingreso al ambiente virtual openpose

Fuente: Autoría



```
(openpose) jrt@JRT977:~/Escritorio/tesis/tf-pose-estimation$
```

Figura 34. Ejecución del ambiente virtual

Fuente: Autoría

De acuerdo al previo análisis en el cual se especificó la manera en el que Openpose trabaja se ha podido determinar cómo trabajar con el código libre para el cálculo de ángulos articulares, puesto que para poner en marcha el sistema se debe ejecutar el script *run_webcam.py* que se encuentra dentro de la carpeta */Escritorio/tesis/tf-pose-estimation/* como muestra la Figura 35.

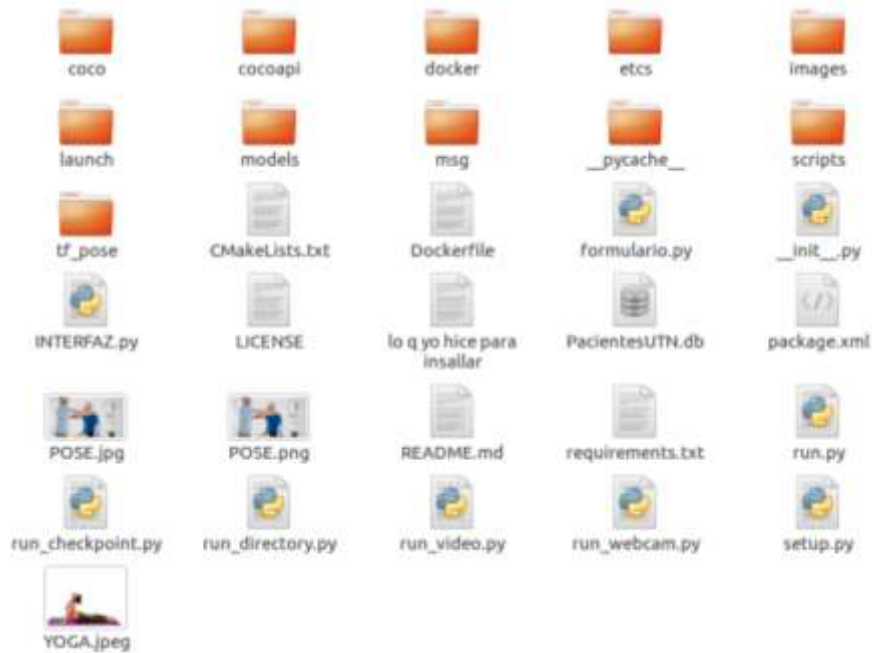


Figura 35. Carpeta que aloja el script de ejecución del sistema

Fuente: Autoría

Luego de abrir el script de código abierto se evidencia los métodos que realiza el sistema para su ejecución, es de vital importancia puesto que integra todas las clases y métodos necesarios, tomando relevancia el método `tf_pose.estimator import TfPoseEstimator` que llama al script en el cual se realiza la asignación de cada una de las partes y coordenadas del cuerpo humano que la red neuronal discrimina, es por eso que se decide trabajar dentro del script `estimator.py`, puesto que se puede manipular el código y a su vez identificar los puntos necesarios para el cálculo de los ángulos de codo y hombro.

El análisis de los puntos con los que el sistema trabaja mostro que son 18, cada una de las partes del cuerpo humano están especificadas con un número fijo que las identifica, puesto que necesitamos medir tanto los ángulos del codo y hombro se estimó trabajar con los puntos que muestra la Tabla 16.

Tabla 16. Puntos de las extremidades a trabajar

EXTREMIDAD	PUNTOS
BRAZO IZQUIERDO	5,6,7
BRAZO DERECHO	2,3,4
HOMBRO IZQUIERDO	1,5,6
HOMBRO DERECHO	1,2,3

Fuente: Autoría

El proceso que se realiza para trabajar con los puntos de cada una de las articulaciones propuestas en este proyecto es identificar el rango de puntos que son del 1 al 7 y extrayendo las coordenadas en x,y de cada punto como muestra la Figura 36 del script *estimator.py*

```

-----
body_part = human.body_parts[i]
center = (int(body_part.x * image_w + 0.5), int(body_part.y * image_h + 0.5))
#print(len(body_part))
centers[i] = center
if(i>=1 and i<=7):
    #print(center)
    cv2.circle(npimg, center, 3, common.CocoColors[i], thickness=7, lineType=10, shift=0)
    asd+=1
#print(asd)
#####COODOS#####
try:
    if(centers[5] != "" and centers[6] != "" and centers[7] != "" and asd >= 6):
        print("PUNTO D")
        d = centers[5]
        print(d)
        print("PUNTO E")
        e = centers[6]
        print(e)
        print("PUNTO F")
        f = centers[7]
        print(f)
        ang1 = math.degrees(math.atan2(f[1]-e[1], f[0]-e[0]) - math.atan2(d[1]-e[1], d[0]-e[0]))

```

Figura 36. Extracción de coordenadas de los puntos que forman el brazo izquierdo

Fuente: Autoría

- **Cálculo del ángulo de flexión y extensión de codo**

Una vez funcional el modelo de visión artificial y obtenidos las coordenadas de los 3 puntos que forman las articulaciones a medir es necesario encaminar el sistema a la medición de ángulos de movilidad articular, a través del estudio del producto escalar se puede calcular el ángulo formado entre dos vectores formados por los 3 puntos antes mencionados.

El producto escalar, Ecuación 4, se define como el producto de los módulos de los vectores considerados multiplicados por el coseno del ángulo que forman, como muestra la Figura 37.

$$\vec{BA} \cdot \vec{BC} = |\vec{BA}| \cdot |\vec{BC}| \cdot \cos\theta$$

Ecuación 4. Producto escalar

Fuente: Adaptado de (Nguyen & Bai, 2010)

Dónde:

- A, B, C son los 3 puntos formados cada uno por las coordenadas en (x, y) .
- $\vec{BA} = A - B$
- $\vec{BC} = C - B$
- $|\vec{BA}| = \sqrt{A^2 - B^2}$
- $|\vec{BC}| = \sqrt{C^2 - B^2}$

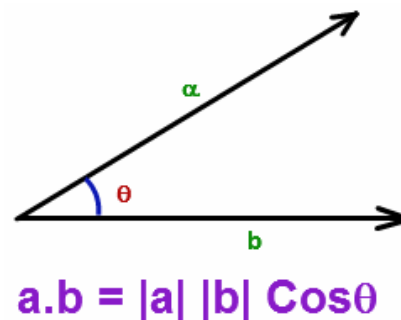


Figura 37. Producto escalar de dos vectores

Fuente: Adaptado de (Burbano de Ercilla, Burbano García, & Gracia Muñoz, 2014)

De la Ecuación 4 del producto escalar despejando θ se obtiene el ángulo formado por dos vectores, como se muestra en la Ecuación 5.

$$\theta = \cos^{-1} \frac{\vec{BA} \cdot \vec{BC}}{|\vec{BA}| \cdot |\vec{BC}|}$$

Ecuación 5. Ecuación para el cálculo del ángulo articular del sistema

Fuente: Adaptado de (Nguyen & Bai, 2010)

Puesto que el codo se forma del brazo y antebrazo cada uno de ellos pueden ser representados como un vector como se aprecia en la Figura 38 además se tiene los 3 puntos de las articulaciones necesarias con sus respectivas coordenadas gracias a aplicación del algoritmo de pose humana, cuyos puntos son para el brazo izquierdo son (2,3,4) y para el brazo derecho los puntos son (5,6,7) para formar los mencionados vectores y calcular el ángulo interior de flexión y extensión del codo.

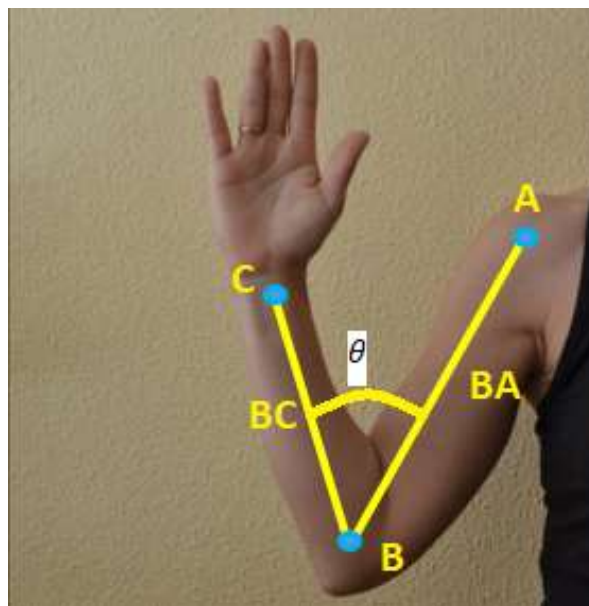


Figura 38. Angulo formado por los vectores del brazo y antebrazo

Fuente: Autoría

En relación al análisis bibliográfico de la medición goniométrica, el ángulo a calcular es aquel formado por el recorrido de la articulación desde el punto cero (en el cual el brazo se encuentra totalmente en reposo y extensión total), para el caso ideal de flexión y que permite a su vez en el funcionamiento del sistema medir la extensión, es decir 180° hacia el punto en el

cual el paciente necesite ser evaluado como muestra la Figura 39, el ángulo resultante para la evaluación es $Ang = 180 - \theta$.



Figura 39. Angulo a evaluar en la flexión y extensión de codo

Fuente: Autoría

- **Calculo de la Abducción y Aducción de Hombro**

La lógica tomada para el cálculo de los ángulos de abducción y aducción de hombro es similar a la de la flexo-extensión de codo puesto que se necesitan 3 puntos referenciales para formar dos vectores que permitan calcular θ , tanto para el hombro derecho e izquierdo se tiene un punto en común que es el punto 1 que indica el punto medio de unión entre el cuello y el centro del tronco de la persona, por ende como se analizó anticipadamente los puntos a trabajar son para hombro izquierdo (1,5,6) y para hombro derecho (1,2,3) como indica la Figura 40.

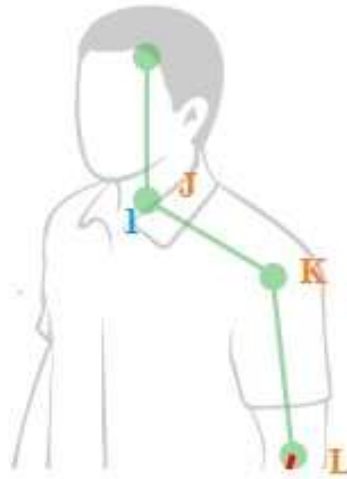


Figura 40. Puntos para calcular los ángulos articulares del hombro

Fuente: Autoría

Tanto la abducción como la aducción fueron desarrolladas cada una con un caso en particular:

- **Angulo de la Abducción**

Debido que la posición anatómica en reposo el paciente debe estar en 0° de acuerdo al sistema de medición del cero neutro, aplicando el cálculo del producto escalar del vector que forma el cuello con el hombro y el vector que forma el hombro con el antebrazo se puede obtener θ a medida que el paciente aleja el brazo del torso como muestra la Figura 41.

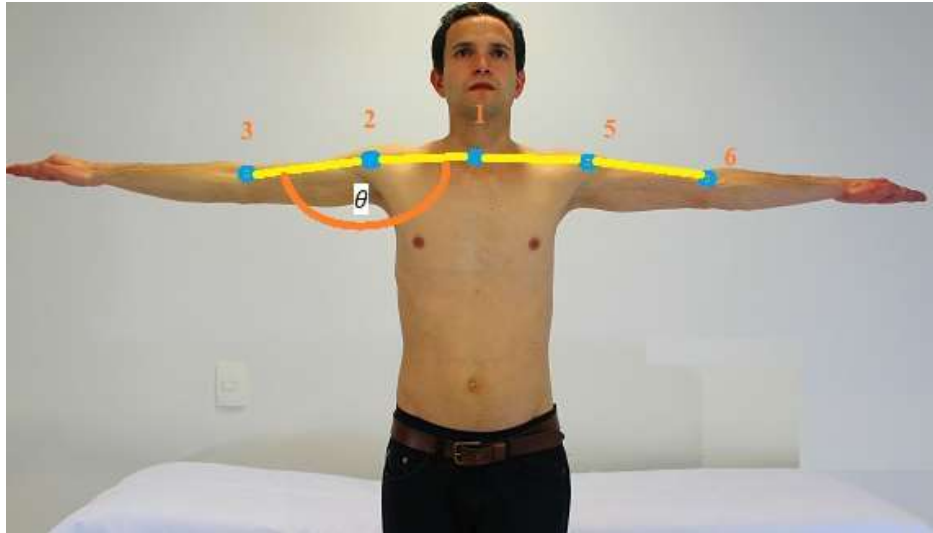


Figura 41. Medición del ángulo formado por dos vectores para el hombro

Fuente: Autoría

Pese a que se obtiene un ángulo llamado θ , este no es el deseado para el caso de este estudio, los puntos 2 y 5 que representan a las articulaciones del hombro son vitales para el cálculo del ángulo de abducción, de estos se traza una línea imaginaria en el eje de coordenadas y que representa al punto de inicio o cero que es desde donde se empieza con la medición y que forma 90° con el vector formado entre el punto referencial del hombro y cuello tal como indica la Figura 42, es por eso que el ángulo θ obtenido en una primera instancia es útil para el cálculo del ángulo resultante $Ang1 = \theta - 90$.

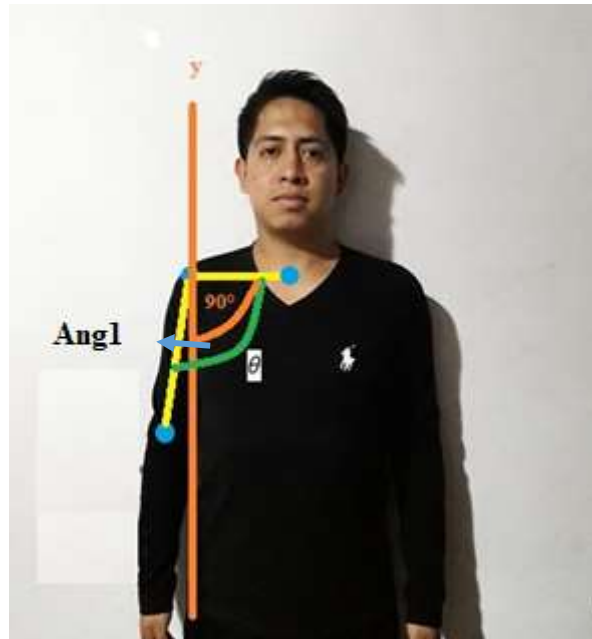


Figura 42. Lógica medición Angulo resultante de la Abducción

Fuente: Autoría

- **Angulo de Aducción**

El cálculo de este ángulo contiene la misma lógica previamente explicada para la abducción, la diferencia es que debido a que en este movimiento el brazo se acerca o se adentra al torso del paciente el ángulo resultante $Ang2 = 90 - \theta$, tal como muestra la Figura 43.

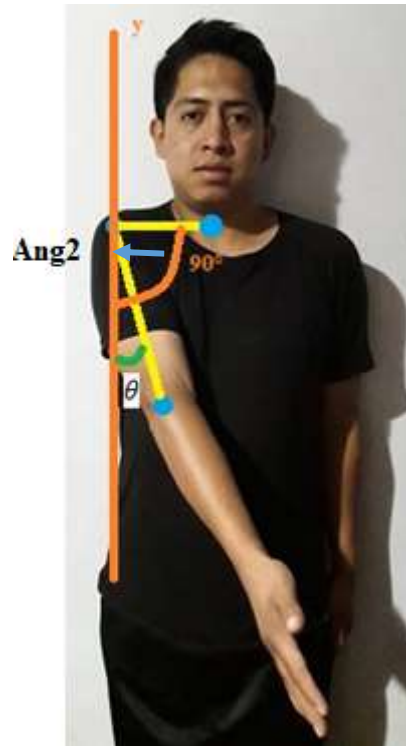


Figura 43. Lógica de medición del Angulo resultante de la Aducción de hombro

Fuente: Autoría

3.10.2.3. Bloque 3. Datos de Salida

Al terminar el proceso de cálculo de los ángulos articulares para hombro y codo se procede a almacenarlos en variables para hacer uso posteriormente de ellos en un reporte tanto en la interfaz de usuario a desarrollar, así como también en una base de datos en la que se almacenara los datos del paciente y su evaluación. Para el desarrollo de la interfaz de usuario del sistema se utilizó la biblioteca Tkinter, que contiene un kit completo de herramientas estándar para el desarrollo de GUI's en el intérprete de Python. A continuación, se mostrará de manera unificada a través de una interfaz de usuario, cada función implementada en el sistema de medición de ángulos articulares.

El script “INTERFAZ.py” muestra la codificación empleada para el diseño de la ventana principal del sistema como muestra la Figura 44.

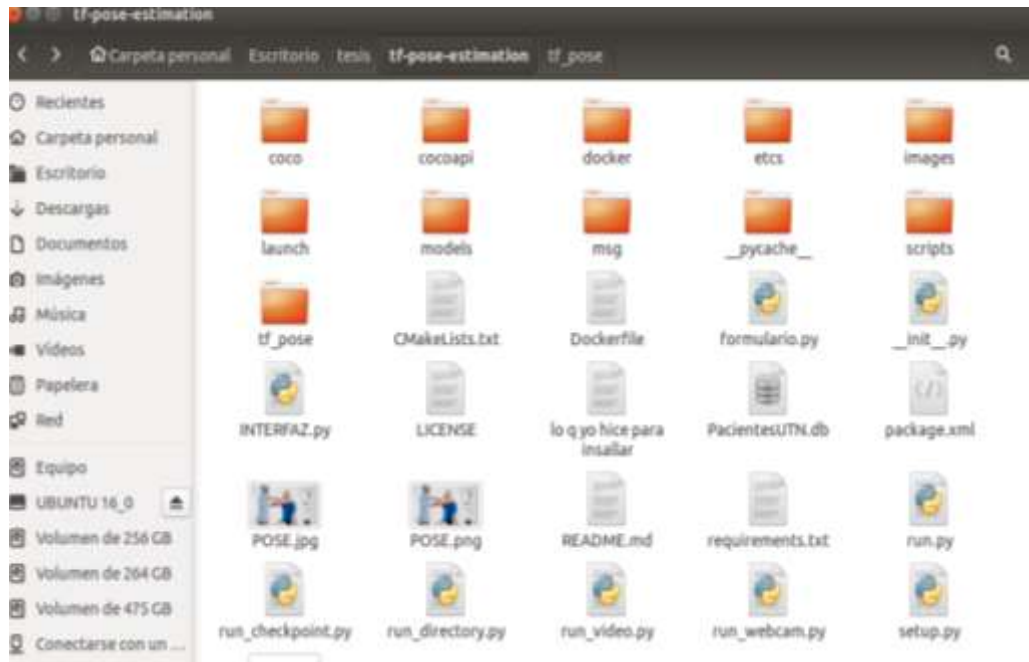


Figura 44. Ubicación del script para el desarrollo de la Interfaz grafica

Fuente: Autoría

En la Figura 45 se puede visualizar el conjunto de funciones disponibles en el sistema y son las siguientes: Prueba e Ingresar Paciente. En la siguiente sección se hará mención de las funciones que conllevan cada botón del sistema.

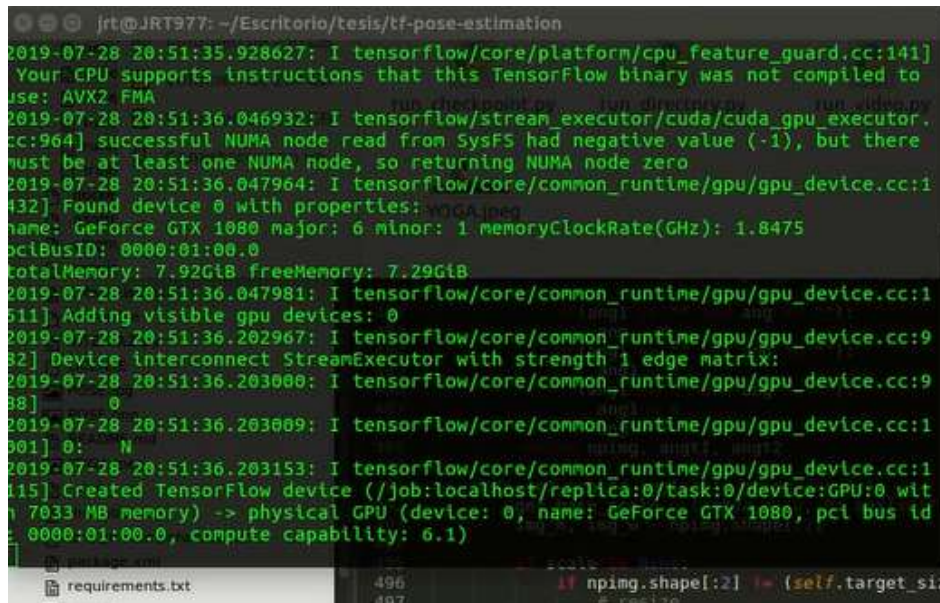


Figura 45. Interfaz Gráfica del Sistema de medición de Ángulos Articulares

Fuente: Autoría

- **Prueba de la interfaz gráfica del sistema de ángulos articulares**

La opción prueba permite al usuario comprobar el correcto funcionamiento del sistema, en cuanto refiere a la conexión de la cámara, ejecución del algoritmo e identificación de las articulaciones, así como también observar el rendimiento del sistema y la fluidez con la que este trabaja, tal como muestra la Figura 46.



```

jrt@JRT977: ~/Escritorio/tesis/tf-pose-estimation
2019-07-28 20:51:35.928627: I tensorflow/core/platform/cpu_feature_guard.cc:141]
Your CPU supports instructions that this TensorFlow binary was not compiled to
use: AVX2 FMA
2019-07-28 20:51:36.046932: I tensorflow/stream_executor/cuda/cuda_gpu_executor.
cc:964] successful NUMA node read from SysFS had negative value (-1), but there
must be at least one NUMA node, so returning NUMA node zero
2019-07-28 20:51:36.047964: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1
432] Found device 0 with properties:
name: GeForce GTX 1080 major: 6 minor: 1 memoryClockRate(GHz): 1.8475
pciBusID: 0000:01:00:0
totalMemory: 7.92GiB freeMemory: 7.29GiB
2019-07-28 20:51:36.047981: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1
511] Adding visible gpu devices: 0
2019-07-28 20:51:36.202967: I tensorflow/core/common_runtime/gpu/gpu_device.cc:9
82] Device interconnect StreamExecutor with strength 1 edge matrix:
2019-07-28 20:51:36.203000: I tensorflow/core/common_runtime/gpu/gpu_device.cc:9
88] 0
2019-07-28 20:51:36.203009: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1
001] 0:  N
2019-07-28 20:51:36.203153: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1
115] Created TensorFlow device (/job:localhost/replica:0/task:0/device:GPU:0 wit
h 7033 MB memory) -> physical GPU (device: 0, name: GeForce GTX 1080, pci bus id
: 0000:01:00:0, compute capability: 6.1)

```

Figura 46. Ejecución de la opción Prueba

Fuente: Autoría

- **Ingresar Paciente**

Esta opción despliega una Ficha de ingreso, donde se introducirán los datos de los pacientes del sistema, Figura 47, de forma manual. Los campos de información personal que se han determinado fueron abordados como se explica en el apartado 2.1.7 y son: Cédula, Apellidos, Nombres, Patología, Género, Edad, y Ocupación.

The screenshot shows a web application window titled "FICHA DE INGRESO". It is divided into two main sections: "Formulario de registro" (Registration Form) and "Toma de datos" (Data Collection).

Formulario de registro:

- Cédula(ID):
- Apellidos:
- Nombres:
- Patología:
- Género: Masculino Femenino
- Edad:
- Ocupación:

Toma de datos:

- Movimiento:
- FLEXION-EXTENSION CODO:
 - Codo izquierdo:
 - Codo derecho:
 -
- ABDUCCION HOMBRO:
 - Hombro izq:
 - Hombro der:
 -
- ADUCCION HOMBRO:
 - Hombro izq:
 - Hombro der:
 -

At the bottom of the form, there are two buttons: and .

Figura 47. Ingreso de datos del paciente y evaluación

Fuente: Autoría

Una vez ingresada la información se procede a realizar la medición tanto de la flexo-extensión de codo o de la abducción y aducción del paciente cualquiera fuese el caso a través del botón “Iniciar” como se indica Figura 48.

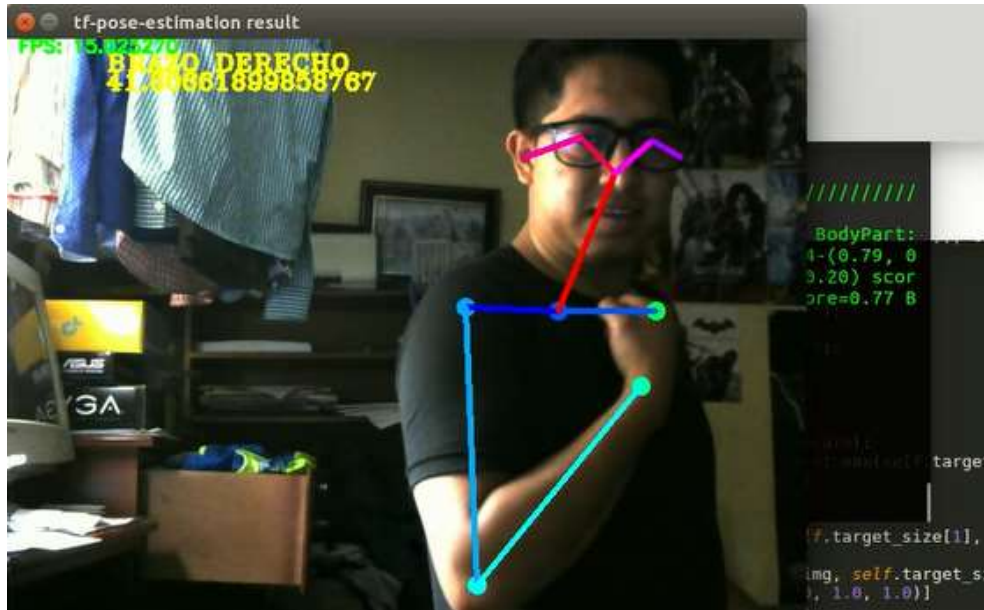


Figura 48. Evaluación del Paciente

Fuente: Autoría

Una vez terminada la evaluación de cualquiera de los movimientos se procede a presionar ESC para capturar los ángulos articulares medidos que se almacenaran en un TextBox de la interfaz dependiendo del movimiento medido y se procede a presionar en “Registrar” para enviar todos los datos del paciente a una base de datos como muestra la Figura 49.

FICHA DE INGRESO

Formulario de registro

Cédula (ID): 1003745187

Apellidos: Ipiales Ruano

Nombres: Christian Eduard

Patología: Luxación

Género:
 Masculino
 Femenino

Edad: 21

Ocupación: Estudiante

Toma de datos

Movimiento: Seleccionar

FLEXION-EXTENSION CODO:

Codo izquierdo: 0 0

Codo derecho: 10.3 128.7

Final Final

ABDUCCION HOMBRO:

Hombro izq:

Hombro der:

Final Final

ADUCCION HOMBRO:

Hombro izq:

Hombro der:

Final Final

Registrar

Guardar

Figura 49. Datos completos del Paciente

Fuente: Autoría

- **Registro.**

Finalmente, la opción “Registro” trasladará la información a una base de datos y a su vez a un archivo con extensión “.txt”, una vez terminado el proceso la opción guardar permite enviar la información a un archivo en formato “.xlsx” que será la ficha completa del paciente. Además, se almacenará las evaluaciones a las cuales fue sometido con el sistema, la plantilla del archivo se muestra en la Figura 50.

CENTRO DE DOCENCIA Y ATENCION FISIOTERAPEUTICA UTN							
FECHA	CEDULA	NOMBRE	APELLIDO	PATOLOGIA	GENERO	EDAD	OCCUPACION
Codigo Izquierdo_Rango_Inicial	Codigo Izquierdo_Rango_Final	Codigo Derecho_Rango_Inicial	Codigo Derecho_Rango_Final				
Hombro_ABD_Izquierdo_Rango_Inicial	Hombro_ABD_Izquierdo_Rango_Final	Hombro_ABD_Derecho_Rango_Inicial	Hombro_ABD_Derecho_Rango_Final				
Hombro_AD_Izquierdo_Rango_Inicial	Hombro_AD_Izquierdo_Rango_Final	Hombro_AD_Derecho_Rango_Inicial	Hombro_AD_Derecho_Rango_Final				

RANGO DE MOVIMIENTO DEL HOMBRO
 ABDUCCION 0 A 180 (AAOS)
 ADUCCION 0 A 30 (AAOS)

RANGO DE MOVIMIENTO DEL CODO
 EXTENSION 0 A 10 (AAOS)
 FLEXION 0 A 150 (AAOS)

OBSERVACIONES:

Figura 50. Plantilla para la muestra de la información completa del paciente

Fuente: Autoría

3.11. Pruebas de integración

Para el desarrollo de las pruebas de integración se realiza las pruebas de comunicación entre la cámara y el ordenador es decir se verifica que el sistema arranque y permita la captura de la posición humana. A partir de eso se verifica la comunicación entre la cámara y el software de desarrollo (aplicación).

3.11.1. Pruebas de registro extracción de datos

Para este tipo de pruebas se consideró verificar que la toma de datos sea realizada con las plantillas que se analizó previamente para la evaluación goniometría la Tabla 17 indica el plano en el cual cada movimiento será evaluado.

Tabla 17. Planos de evaluación del paciente

Movimiento	Plano de evaluación	Rango de movimiento articular
Flexo-extensión de codo	Lateral	0° a 150°
Abducción de hombro	Frontal	0° a 180°
Aducción de hombro	Frontal	0° a 30°

Fuente: Autoría

La prueba de ángulos detectados por el sistema se realizó con el objetivo de medir el error que existe entre los ángulos medidos por el dispositivo y los ángulos reales medidos con el goniómetro. Para determinar la variación en la medición de un ángulo se consideró realizar una comparación de ángulos, que en la Tabla 18 se detalla la información recolectada.

Tabla 18. Comparación de ángulos

N° de persona	Movimiento	Angulo medido con el sistema	Angulo medido manualmente	Diferencia
1	Flexión	0° - 143°	0° - 138°	5°
2	Abducción	0° - 169°	0° - 163°	6°
3	Aducción	0° - 38°	0° - 42°	9°
4	Abducción	0° - 172°	0° - 170°	2°

Fuente: Autoría

Se determinó el valor promedio de la diferencia de los ángulos medidos manualmente con el goniómetro y los ángulos medidos por el sistema a través de visión artificial, para obtener un porcentaje que expresa la tasa de error de lectura se utiliza la Ecuación 6.

$$P = \frac{S}{N}$$

Ecuación 6. Tasa de error de lectura

Fuente: Adaptado de (Pérez López, 2005)

Donde:

P = Es el promedio de las mediciones realizadas

S = Suma de las diferencias obtenidas en la medición

N = Numero de muestras

$$P = \frac{22}{4} = 5,5$$

Como resultados se obtiene que el error entre la medición a través del sistema comparado con la medición con el goniómetro manualmente tiene un estimado de 5,5° de diferencia a favor del sistema propuesto.

La solución planteada para el desarrollo del sistema cumple con las expectativas puesto que el algoritmo permite reconocer los puntos de las articulaciones necesarias para la posterior medición de los ángulos de estudio, además el sistema aplica la lógica planteada para la solución y entrega de los datos necesarios para la comparación entre la toma de datos manualmente y la que realiza el sistema desarrollado, para ello pese a que no se puede tener una base de pacientes con la misma altura se ha determinado una distancia de 2.50m entre la cámara y los pacientes a evaluar para que el sistema pueda detectar solamente las extremidades superiores.

3.12. Diagramas de Flujo

En este apartado se esquematiza en base a diagramas de flujo cada uno de los procesos y las decisiones tomadas en el desarrollo de cada uno de los bloques del sistema, dando una

pauta de que parámetros se tomaron en cuenta al momento de realizar la programación e ingreso de instrucciones para la ejecución del sistema de medición de ángulos articulares.

3.12.1. Obtención de imágenes

Para el uso de la cámara Logitech C270 es primordial la instalación del paquete de instalación de cámaras al sistema operativo Ubuntu, luego de esto el dispositivo puede ser ocupado para la extracción de imágenes y video a través del sistema de procesamiento de imágenes, como muestra la Figura 51.

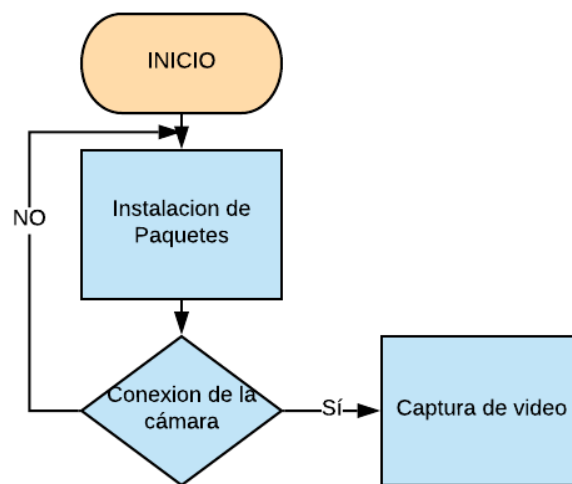


Figura 51. Diagrama de flujo de la obtención de datos

Fuente: Autoría

3.12.2. Procesamiento

En la Figura 52 del diagrama de bloques se calibra el video obtenido por la cámara web, donde se realiza una auto resolución de la imagen, además de hacer uso del sistema de estimación de pose humana y el grafico de los puntos articulares.

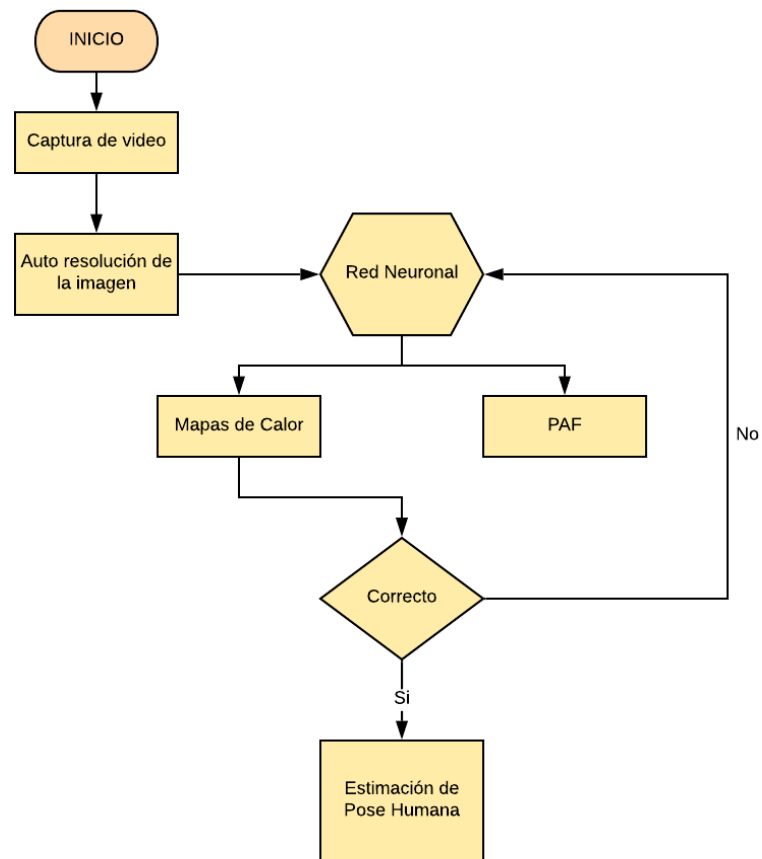


Figura 52. Diagrama de flujo del procesamiento del sistema

Fuente: Autoría

3.12.3. Salida de datos

Para la salida de datos se toma en consideración la conexión de la cámara, los puntos articulares obtenidos a partir del algoritmo y la red neuronal aplicada, así como también el cálculo de ángulos articulares y el registro en una base de datos de la información obtenida como se aprecia en la Figura 53.

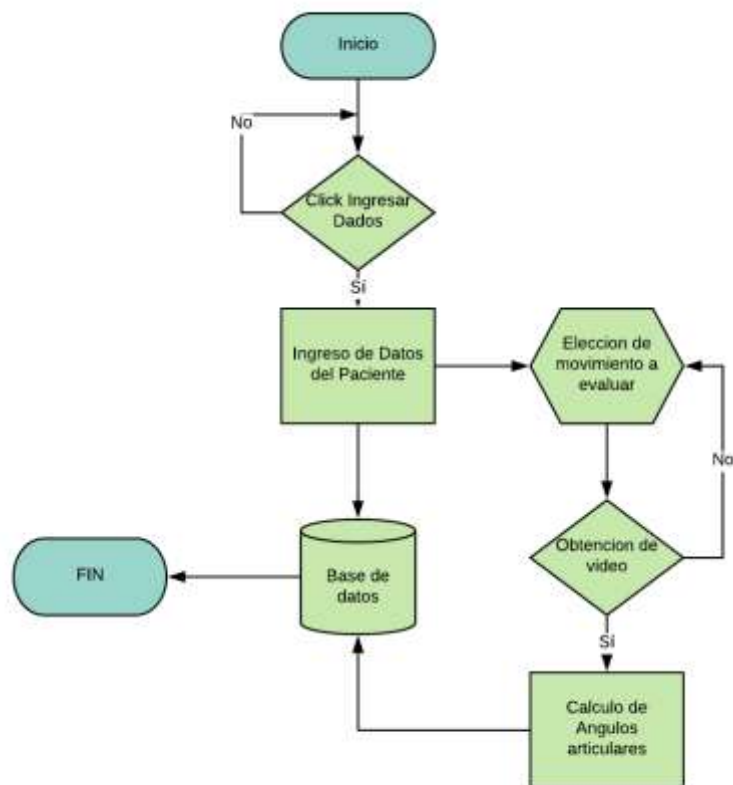


Figura 53. Diagrama de flujo de la salida de datos

Fuente: Autoría

3.13. Esquema del sistema

En la Figura 54 se muestra el esquema del sistema, de forma general como interactúan los 3 elementos principales que para el funcionamiento de este proyecto se consideró el uso de un procesador core i7, una cámara (Logitech c270) con una fuente de alimentación de conexión USB hacia la placa del ordenador, y una tarjeta de video NVIDIA 1080GTX como procesador GPU.

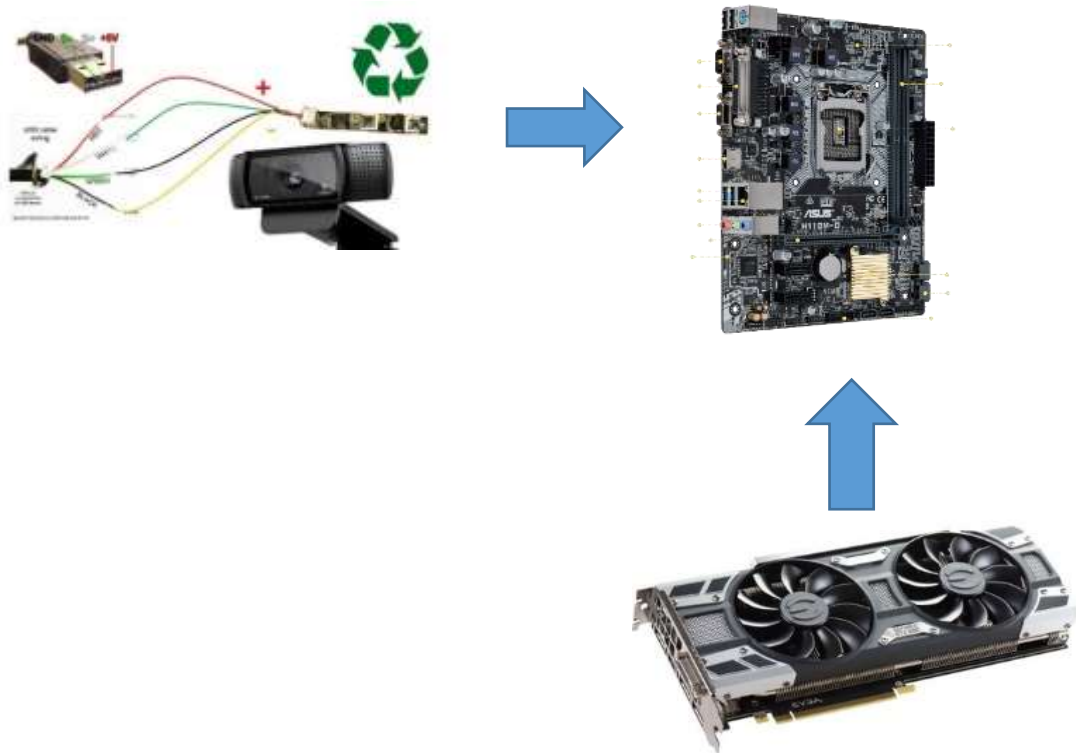


Figura 54. Diagrama de sistema

Fuente: Autoría

Finalmente el diseño del sistema en marcha permitió constatar que tanto el algoritmo de estimación de pose humana, como la interfaz gráfica y la base de datos convergen para ejecutar la evaluación de ángulos articulares en miembros superiores tanto para flexo-extensión de codo como para la aducción abducción de hombro, las pruebas de campo fueron realizadas para determinar el trabajo de la interfaz y el manejo del sistema por parte del usuario comprobando así que los datos tomados sean coherentes, en tiempo real, mismos que arrojaron un error promedio de $5,5^{\circ}$ en relación a la medición manual.

Capítulo 4. Pruebas de Funcionamiento y Resultados

En este capítulo se describe los resultados obtenidos al realizar las pruebas de funcionamiento del sistema en el Centro de Rehabilitación de la “Universidad Técnica del Norte”.

Además, en este capítulo con las pruebas ya realizadas, se llega a concluir sobre los resultados logrados por el sistema de monitoreo al finalizar el presente proyecto. Esto define si el sistema cumple con los objetivos propuestos en un inicio y también como se fueron superando las restricciones mientras se realizaba el proyecto.

4.1. Pruebas de funcionamiento del sistema

Una vez terminada la etapa del diseño del sistema se procede a realizar las pruebas que validen todo lo propuesto en el presente proyecto, la finalidad de estas pruebas es verificar que se tenga datos con coherencia objetiva para comparar la medición del sistema versus la medición manual.

Para la prueba de funcionamiento del sistema se cuenta con el criterio de un experto es decir el responsable del Centro de Rehabilitación de la “UTN”, el Lic. Jorge Zambrano; en el Anexo F se evidencia la evaluación por parte del experto. La evaluación del experto permite realizar una comparación entre la evaluación del sistema y la de la medición manual, basado en:

- Una muestra con la medición manual del profesional hacia 9 pacientes los cuales fueron evaluados en los movimientos de flexo-extensión de codo y abducción y aducción de hombro, de igual manera se tomaron dos muestras con los datos arrojados por el sistema para obtener un promedio en la medición realizada y poder comparar los instrumentos de medición utilizados.

- Una referencia tanto del ángulo inicial como del ángulo límite de movilidad de la articulación evaluada del paciente.
- Un plano anatómico en el que permita la medición del paciente tanto para la medición manual como para la medición del sistema.
- Se consideró la evaluación del experto en base a dos criterios que son: Acepta (El sistema arroja coherencia objetiva con un máximo de error en la medición de 5 grados en comparación con la medición tradicional usando el goniómetro), No Acepta (Los valores exceden los 5 grados y por ende no pueden ser considerados aptos para el registro del paciente).

4.1.1. Pruebas por movimiento

Los resultados obtenidos de la implementación del sistema de medición de ángulos articulares, permiten evaluar el funcionamiento del sistema en su totalidad, como también determinar si los ángulos evaluados con los planos predeterminadas se encuentran en los rangos establecidos por la Asociación para el Estudio de Osteosíntesis (AO) de Suiza, los rangos de movimiento articular para la abducción y aducción se muestra en la Tabla 19, mientras que para los movimientos de flexo-extensión se detallan en la Tabla 20.

Tabla 19. Evaluación de hombro

Posición	Rango de ángulos (AO)
Posición Cero	0°
Abducción	0° hasta 160° o 180°
Aducción	0° a 30°

Fuente: Autoría

Tabla 20. Evaluación de codo

Posición	Rango de ángulos (AO)
Posición Cero	0°
Flexión – Extensión	0° hasta 130° o 150°

Fuente: Autoría

4.2. Evaluación de la flexo-extensión de codo

Para la obtención de las muestras de esta fase, se evalúa 9 pacientes a los cuales se toman dos muestras del ángulo que se genera entre el antebrazo y el brazo con el sistema y una muestra de forma manual con el goniómetro, tanto en reposo, así como para los movimientos de flexo-extensión de codo izquierdo o derecho cual fuese el caso, todas las posiciones deben tomarse en el plano sagital como muestra la Figura 55 y con las normas para goniometría que se abordaron en el apartado 2.1.5.2.



Figura 55. Plantilla para la toma de muestras de la flexo-extensión

Fuente: Autoría

En la Tabla 21 se muestra los datos resultantes de las muestras tomadas al paciente en reposo y el criterio generado por el experto.

Tabla 21. Evaluación de pacientes en reposo

Paciente	Muestra	Angulo Goniómetro	Angulo Sistema	Angulo Promedio	Error	Criterio del experto
1	1	10°	5,6°	7,65°	2,35°	Acepta
	2		9,7°			
2	1	11°	12,3°	11,65°	0,65°	Acepta
	2		11°			
3	1	12,5°	11,9	12°	0,5°	Acepta
	2		12,1°			
4	1	15°	12,3°	12,85°	2,15°	Acepta
	2		13,4°			
5	1	7°	14,9°	12,45°	5,45°	No acepta
	2		10°			
6	1	9°	8°	8,75°	0,25°	Acepta
	2		9,5°			
7	1	12°	13,6°	11,4°	0,2°	Acepta
	2		10,8°			
8	1	10,2°	10,3°	11,15°	0,95°	Acepta
	2		12°			
9	1	9,8°	13,6°	11,8°	2°	Acepta
	2		10°			
Promedio		10,7°		11,07°	1.6°	

Fuente: Autoría

La Tabla 22 muestra los datos obtenidos con el movimiento de flexión tanto con la medición manual, la comparación con la medición del sistema y el criterio del experto.

Tabla 22. Evaluación de la flexo-extensión de codo

Paciente	Muestra	Codo	Angulo Goniómetro	Angulo Sistema	Angulo Promedio	Error	Criterio del experto
1	1	Derecho	127°	140,6°	138,8°	11,8°	No
	2			137°			Acepta
2	1	Izquierdo	132°	131,3°	131,9°	0,1°	Acepta
	2			133°			
3	1	Izquierdo	145°	148,1°	147,55°	2,55°	Acepta
	2			147°			
4	1	Izquierdo	135°	134,6°	133,8°	1,2°	Acepta
	2			133°			
5	1	Derecho	120°	124,7°	124,95°	4,9°	Acepta
	2			125,2°			
6	1	Derecho	123°	128,7°	126,9°	3,9°	Acepta
	2			125,1°			
7	1	Izquierdo	137°	136,3°	137,15°	0,15°	Acepta
	2			138°			
8	1	Derecho	141°	138,7°	139,6°	1,4°	Acepta
	2			140,5°			
9	1	Derecho	133°	128,6°	133°	3,65°	Acepta
	2			130,1°			
Promedio			131,7°		134,8°	3,29°	

Fuente: Autoría

Las Figuras 56 y 57 muestran un ejemplo del plano en el cual el paciente fue evaluado con los dos métodos mencionados anteriormente.



Figura 56. Toma de datos manualmente con el goniómetro

Fuente: Autoría



Figura 57. Muestra de datos capturada con el sistema

Fuente: Autoría

4.2.1. Resultados obtenidos de la flexo-extensión

Considerando la evaluación del experto al momento de ejecución del sistema en el Centro de Rehabilitación de la Universidad Técnica del Norte la tasa de error promedio se ve afectada en $1,6^\circ$ en mediciones para reposo y $3,6^\circ$ en mediciones con movimiento, que son

valores considerables puesto que en cada una de las pruebas estas arrojan datos con coherencia objetiva en la parte clínica, debido a que la manipulación del goniómetro manualmente está sujeto a movimientos y a una estimación visual propia del ser humano así como también las mediciones tomadas por parte del sistema que se ven afectadas por falta de precisión en el eje z de profundidad en la posición del paciente al momento de tomar los datos.

4.3. Evaluación de la abducción y aducción de hombro

Para finalizar con la obtención de las muestras de esta fase, se evalúa 9 pacientes a los cuales se les toman dos muestras del ángulo que se genera con los movimientos de abducción y aducción de hombro izquierdo o derecho cual fuese el caso con el sistema y una muestra con la medición manual a través del goniómetro, todas las posiciones deben tomarse en el plano frontal, como se muestra en la Figura 58 y tomando las prácticas de goniometría para abducción y aducción de hombro que se abordaron en el apartado 2.1.5.1.



Figura 58. Plantilla para la toma de datos de la abducción y aducción de hombro

Fuente: Autoría

En la Tabla 23 se muestra los datos resultantes de las muestras tomadas al paciente en reposo y el criterio generado por el experto.

Tabla 23. Evaluación en reposo del paciente

Paciente	Muestra	Angulo Goniómetro	Angulo Sistema	Angulo Promedio	Error	Criterio del experto
1	1	7°	9,7°	7°	2,4°	Acepta
	2		9,1°			
2	1	18°	12,3°	14,8°	3,2°	Acepta
	2		17,3°			
3	1	10°	7,9°	9,05°	0,95°	Acepta
	2		10,2°			
4	1	16°	10,7°	13°	3°	Acepta
	2		15,3°			
5	1	24°	18,7°	19,45°	4,5°	Acepta
	2		20,2°			
6	1	9°	14,5°	13,05°	4,05°	Acepta
	2		11,6°			
7	1	7°	7,8°	7,55°	0,55°	Acepta
	2		7,3°			
8	1	10°	8,2°	9,85°	0,15°	Acepta
	2		11,5°			
9	1	20°	17,4°	19,3	0,7°	Acepta
	2		21,2°			
Promedio		13,4°		12,5°	2,1°	

Fuente: Autoría

La Tabla 24 muestra los datos obtenidos con el movimiento de abducción y aducción tanto con la medición manual, la comparación con la medición del sistema y el criterio del experto.

Tabla 24. Datos Obtenidos de la evaluación en el hombro

Paciente	Muestra	Movimiento	Hombro	Angulo Goniómetro	Angulo Sistema	Angulo Promedio	Error	Criterio del experto
1	1	Abducción	Derecho	145°	150,5°	149,15°	4,15	Acepta
	2				147,8°			
2	1	Abducción	Izquierdo	100°	97,5°	96,25°	3,75	Acepta
	2				95°			
3	1	Abducción	Izquierdo	164°	166,5°	167,75°	3,75	Acepta
	2				169°			
4	1	Aducción	Izquierdo	16°	14,5°	16,35°	0,35	Acepta
	2				18,2°			
5	1	Aducción	Derecho	18°	12,3°	14,1°	3,9	Acepta
	2				15,9°			
6	1	Abducción	Derecho	120°	118,7°	120,5°	0,5	Acepta
	2				122,3			
7	1	Abducción	Derecho	115°	115,5°	114,8°	0,2	Acepta
	2				114,1°			
8	1	Abducción	Izquierdo	104°	103,6°	101,9°	2,05	Acepta
	2				100,3°			
9	1	Aducción	Izquierdo	21°	22,8°	20,5°	0,5	Acepta
	2				18,2°			
Promedio				89,2°		89,03°	2,1°	

Fuente: Autoría

A continuación, en las Figuras 59 y 60 se muestra el plano en el cual fueron realizadas las mediciones tanto con el goniómetro, así como también con el sistema.



Figura 59. Evaluación de la abducción manualmente con el goniómetro

Fuente: Autoría



Figura 60. Evaluación de la aducción a través del sistema

Fuente: Autoría

4.3.1. Resultados obtenidos de la abducción y aducción

De acuerdo al criterio del experto encargado de supervisar la toma de datos, cada una de las muestras están de acuerdo con los resultados esperados, existe un error promedio de $2,1^{\circ}$ tanto en las muestras en reposo como en las muestras en movimiento que son valores a tomar

en cuenta puesto que los valores son sumamente bajos y están dentro del límite propuesto por el experto lo cual en una primera instancia da fe que el sistema trabaja correctamente, así como resultando menos complejo el sistema de medición de ángulos articulares a través de visión artificial.

Mediante los resultados obtenidos en la implementación del sistema, como también la información generada por las muestras de las pruebas realizadas, se puede afirmar que el sistema de propuesto permite evaluar el movimiento corporal humano en tiempo real.

4.4. Validación del sistema

El sistema de medición de ángulos articulares mediante visión artificial será evaluado en base a las mediciones obtenidas en cada uno de los movimientos de miembro superior en comparación al goniómetro determinando la confiabilidad del sistema a partir de dos métodos que son la prueba de hipótesis y el índice de fallas.

4.4.1. Prueba de hipótesis

La prueba de hipótesis es un procedimiento estadístico basado en datos de muestra, una prueba de hipótesis evalúa dos afirmaciones sobre una población para determinar que afirmación es mejor respaldar por los datos de la muestra.

El objetivo es plantear dos hipótesis, una hipótesis nula H_0 que se muestra en la Ecuación 7 y una hipótesis alternativa H_1 , Ecuación 8, los análisis estadísticos de los datos servirán para determinar si se puede aceptar o no la hipótesis nula (H_0). Cuando se rechaza a la hipótesis nula significa que automáticamente se acepta la hipótesis alternativa.

Los pasos para la prueba de hipótesis son:

1. Planteamiento de la hipótesis nula o alternativa para el caso de este proyecto se utilizará una hipótesis alternativa en la que media muestral

sea diferente a la poblacional por lo cual la hipótesis nula planteara que las mediciones tendrán concordancia o relativamente iguales.

Donde:

$$H_0 = \bar{X} = u$$

Ecuación 7. Ecuación hipótesis nula

Fuente: Adaptado de (Kingma & Ba, 2014)

$$H_1 = \bar{X} \neq u$$

Ecuación 8. Ecuación hipótesis alternativa

Fuente: Adaptado de (Kingma & Ba, 2014)

2. Elegir el nivel de significancia δ que normalmente para casos de investigación es de 0,05
3. Determinar la zona de rechazo o aceptación de la hipótesis nula, donde el estadístico de contraste se muestra en la Ecuación 9:

$$T = \frac{\bar{X} - u_o}{\sqrt{S/\sqrt{n}}}$$

Ecuación 9. Ecuación para el cálculo de estadístico de contraste

Fuente: Adaptado de (Kingma & Ba, 2014)

Donde:

S= número de muestras

\bar{X} = media muestral

u_o = media poblacional

n = numero de muestras

4. Determinación pivotal, t si es para una muestra menor de 30
5. Utilizar el valor del cálculo pivotal para determinar la aceptación o rechazo de la hipótesis nula.

- **Flexo-Extensión de codo**

Datos:

$$H_0 = u_{gon} = u_{sist}$$

$$H_1 = u_{gon} \neq u_{sist}$$

Goniómetro	Sistema
120	138,8
132	131,9
145	147,55
135	133,8
120	124,95
123	126,9
137	137,15
141	139,6
133	133

Muestras n_1 y $n_2 = 9$

Media $n_1 = 131,77$ $n_2 = 134,85$

Nivel de significancia= 0,05

Calculo región crítica:

$$t_c = \frac{131,77 - 134,85}{\sqrt{9,03} / \sqrt{9}}$$

$$t_c = 0,004946035$$

$$\text{Entonces si : } t_t = n_1 + n_2 - 2 = 9+9-2$$

$$t_t = 16 = 2,120$$

Dado que $t_c < t_t$ se acepta H_0

En conclusión gracias a este método se puede determinar que los valores medidos con el goniómetro, así como los del sistema tienen concordancia y, por eso se puede asegurar que los resultados son fiables en las pruebas realizadas.

4.4.2. Índice de Fallas

El índice de fallas es un indicador que permite conocer la confiabilidad del sistema, este se lo representa con la Ecuación 10:

$$TF\% = \frac{\text{Numero de fallas}}{\text{Numero de examinados}}$$

Ecuación 10. Ecuación de índice de fallas

Fuente: Adaptado de (Koldo, 2018)

Donde:

El número de fallas está relacionado con el criterio del experto de no aceptar aquellas mediciones realizadas que sobrepasen o igualen los 5 grados de diferencia entre la medición manual y la del sistema que previamente están mostradas en las Tablas 21, 22, 23 y 24.

El número de examinados representa el número total de muestras examinadas en cada uno de los casos.

Por lo tanto:

$$TF\% = \frac{2}{27} = 7,40\%$$

De acuerdo al índice de fallas se puede determinar que el sistema tiene una confiabilidad del 92,60%. Las fallas representan a mediciones realizadas por el sistema que

sobrepasan en 5 grados a la medición realizada por el goniómetro, varios factores al momento de evaluar al paciente se presentan con el sistema tales como:

- Postura del paciente en el eje z que altera la medición en profundidad.
- Masa muscular diferente en cada uno de los pacientes, por ende, fallos propios del algoritmo al momento de detectar los puntos de las articulaciones.
- Ambientes no controlados exceso de luz.

Terminadas las pruebas de funcionamiento se logró obtener un sistema de medición de ángulos articulares para miembros superiores a través de visión artificial que muestra con coherencia objetiva datos aprobados por el criterio de un experto en el área de fisioterapia, el Lic. Jorge Zambrano como muestra en la Figura 61 y el Anexo G, permitiendo evaluar a los pacientes y mantener una base de datos digital de ellos, el sistema es confiable y valedero puesto que las pruebas a las que fue sometido las supero con un pequeño margen de error en comparación con los instrumentos de medición como el goniómetro que son usados tradicionalmente por los especialistas.

CERTIFICADO

Ibarra, 31 de julio de 2019

A quien corresponda:

YO, Jorge Luis Zambrano Vásquez, con cédula de identidad 0401696216 en mi calidad de Fisioterapeuta responsable del Centro de Rehabilitación de la Universidad Técnica del Norte, certifico que el señor José Luis Jarrín Chacón con cédula de identidad 1718898362 realizó pruebas de funcionamiento de su proyecto de investigación titulado: SISTEMA DE DETECCIÓN DEL ÁNGULO ARTICULAR EN LOS MOVIMIENTOS DE MIEMBRO SUPERIOR PARA EVALUACIÓN EN FISIOTERAPIA MEDIANTE VISIÓN ARTIFICIAL, cuyos resultados arrojaron coherencia objetiva en relación a las pruebas clínicas realizadas manualmente con el goniómetro.

Es todo cuanto puedo certificar en honor a la verdad, el señor José Luis Jarrín puede hacer uso del presente certificado como estime conveniente.

Atentamente:

UTN Centro de
Rehabilitación
FISIO UTN

Lic. Jorge Zambrano FT.

LIC. JORGE L. ZAMBRANO
FISIOTERAPEUTA
REG. SENECEYLR015-2017-1855986

Figura 61. Certificado de aprobación de los datos obtenidos con el sistema

Fuente: Autoría

Conclusiones

Se desarrolló un sistema de medición de ángulos articulares a través de visión artificial con el uso de cualquier cámara para la evaluación miembros superiores mismo que tiene un nivel de confiabilidad del 92,60%, dato obtenido luego de la evaluación del sistema con los datos de las pruebas de funcionamiento.

La aplicación de la tecnología en ramas como la medicina provee de herramientas que simplifican tanto el trabajo de los profesionales, así como optimizar el tiempo y mejorar los procesos de evaluación de miembro superior en codo y hombro y diagnóstico de un paciente.

El sistema de medición de ángulos articulares para extremidades superiores es un instrumento funcional para la utilización en el área de fisioterapia, ya que, en diferentes pruebas realizadas con movimientos de evaluación en goniometría, se observa una respuesta en tiempo real de los arcos de movimiento en los distintos planos evaluados y no supera los 5° de error en comparación con la medición manual a través del goniómetro.

El desarrollo del presente proyecto prescinde del uso de tecnologías como Kinect para la detección de puntos articulares y extremidades, logrando el uso de cualquier cámara a través de la aplicación de un algoritmo de estimación de pose humana.

El sistema de medición de ángulos articulares a través de visión desarrollado no es invasivo para la evaluación al paciente por ende permite una fluidez en el registro de datos por parte del profesional encargado, además de convertirse en una herramienta innovadora con ciertas limitaciones en su aplicación puesto que el estudio de movimientos del cuerpo humano es extenso y solamente se ha utilizado dos articulaciones en este caso.

La ayuda de un profesional en el área de fisioterapia en el diseño del sistema permite tener una mayor claridad en los datos que se desea tomar, así como también la manera de sistematizar una base de datos, con datos adecuados y valederos

Recomendaciones

Analizar a detalle la arquitectura y funcionamiento de una red neuronal con la finalidad de comprender el funcionamiento general de cualquier variación de las mismas.

CUDA es una tecnología de procesamiento paralelo tremendamente importante en el área de la computación de altas prestaciones, ya que aprovecha la potencia de cálculo de los núcleos físicos de una GPU Nvidia para el tratamiento de cantidades masivas de datos enfocados al desarrollo de investigaciones y/o aplicaciones en diferentes campos científicos, además de ser útil en este estudio.

Es recomendable emplear una arquitectura de redes convolucionales que no demande de muchos recursos computacionales o a su vez optimizar la existente, ya que el simple uso de una CPU convencional no cumple con la demanda para la aplicación de modelos robustos de visión artificial.

En cuanto a la precisión de un modelo de aprendizaje automático, sería muy conveniente extender el conjunto de datos con los que se trabaja, con el fin de generar más pesos que generen mayor precisión en los procesos de reconocimiento de pose humana que puede ser un caso de estudio a futuro mejorando el sistema.

Existe una gama de marcos o frameworks que permiten simplificar la excesiva programación y la complejidad en el campo de la inteligencia artificial, por lo que se recomienda probar con diferentes marcos tales como: Caffe2, Tensorflow 2.0, Pytorch, Keras, entre otros.

El uso de cámaras de resolución promedio de 360p, 480p y 720p para sistemas de visión artificial comúnmente por su coste y adquisición son una buena opción en entornos controlados, por el contrario, una cámara de resolución alta de 720p, 1080p, 2k o inclusive 4k

puede resultar mucho mejor para entornos no controlados que se presenta en la mayoría de casos.

Es necesario que el usuario a ser evaluado por el sistema desarrollado no use ropa holgada o chompa, ya que obstruye la visualización correcta de las extremidades superiores generando errores de detección esquelética.

Se recomienda realizar las mediciones de ángulos articulares a una medida superior a los 2,5m en la cual se puede tener evaluaciones del paciente sin importar cuál sea su altura y en entornos con un fondo de preferencia blanco y sin objetos ni personas que intervengan a excepción del evaluador.

Para evitar los errores por movimiento del paciente en el plano a evaluarse recomienda el uso de una plantilla o marcadores que permita la ubicación correcta del paciente en el plano Z.

Se debe considerar mejorar la experiencia del usuario implementando a futuro un modelo optimizado que pueda ser portable usando menos recursos de hardware y permita la evaluación del paciente con las mismas características tanto en tiempo de respuesta para la detección de extremidades como la de toma de datos.

Referencias

- Behnoush, B., Tavakoli, N., Bazmi, E., Nateghi Fard, F., Pourgharib Shahi, M. H., Okazi, A., & Mokhtari, T. (2016). Smartphone and Universal Goniometer for Measurement of Elbow Joint Motions: A Comparative Study. *Asian Journal of Sports Medicine*, 7(2), e30668. <https://doi.org/10.5812/asjasm.30668>
- Bigorda-Sague, A., & Abadia-Rosa, S. (2016). Revisión sistemática del tratamiento de fisioterapia en el reimplante de extremidad superior. *Fisioterapia*, 38(6), 303–312. <https://doi.org/10.1016/j.ft.2015.12.001>
- Burbano de Ercilla, S., Burbano García, E., & Gracia Muñoz, C. (2014). *Física general*.
- Chapleau, J., Canet, F., Petit, Y., Laflamme, G.-Y., & Rouleau, D. M. (2011). Validity of Goniometric Elbow Measurements: Comparative Study with a Radiographic Method. *Clinical Orthopaedics and Related Research*®, 469(11), 3134–3140. <https://doi.org/10.1007/s11999-011-1986-8>
- Felstead, A. J., & Ricketts, D. (2017). Biomechanics of the shoulder and elbow. *Orthopaedics and Trauma*, 31(5), 300–305. <https://doi.org/10.1016/j.mporth.2017.07.004>
- Fornalski, S., Gupta, R., & Lee, T. Q. (2003). *Anatomy and Biomechanics of the Elbow Joint*. Retrieved from [http://courses.washington.edu/bioen520/notes/Elbow_Anatomy_&_Biomechanics_\(Fornalski,\).pdf](http://courses.washington.edu/bioen520/notes/Elbow_Anatomy_&_Biomechanics_(Fornalski,).pdf)
- Frangiamore, S. J., Bigart, K., Nagle, T., Colbrunn, R., Millis, A., & Schickendantz, M. S. (2018). Biomechanical analysis of elbow medial ulnar collateral ligament tear location and its effect on rotational stability. *Journal of Shoulder and Elbow Surgery*, 27(11), 2068–2076. <https://doi.org/10.1016/J.JSE.2018.05.020>
- Gallego Izquierdo, T. (2007). *Bases teóricas y fundamentos de la fisioterapia*. Médica Panamericana.

- Herrera, M. (2010). *INGENIERÍA DEL SOFTWARE: METODOLOGÍAS Y CICLOS DE VIDA*. Laboratorio Nacional de Calidad del Software. Retrieved from https://www.academia.edu/9795641/INGENIERÍA_DEL_SOFTWARE_METODOLOGÍAS_Y_CICLOS_DE_VIDA_Laboratorio_Nacional_de_Calidad_del_Software
- Hoffmann, A. G. (2015). Artificial and Natural Computation. In *International Encyclopedia of the Social & Behavioral Sciences: Second Edition* (pp. 27–31). <https://doi.org/10.1016/B978-0-08-097086-8.43004-7>
- Kapandji, A. (2006). *FISIOLOGIA ARICULAR*. Madrid: Editorial medica panamericana.
- Khan, A., Sohail, A., Zahoor, U., & Qureshi, A. S. (n.d.). *A Survey of the Recent Architectures of Deep Convolutional Neural Networks*. Retrieved from <https://arxiv.org/ftp/arxiv/papers/1901/1901.06032.pdf>
- Kim, T. S., & Reiter, A. (2017). *Interpretable 3D Human Action Analysis with Temporal Convolutional Networks*. Retrieved from <http://arxiv.org/abs/1704.04516>
- Kolber, M. J., Pizzini, M., Robinson, A., Yanez, D., & Hanney, W. J. (2013). The reliability and concurrent validity of measurements used to quantify lumbar spine mobility: an analysis of an iPhone® application and gravity based inclinometry. *International Journal of Sports Physical Therapy*, 8(2), 129–137. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/23593551>
- Lee, C.-Y., Gallagher, P. W., & Tu, Z. (2016, May 2). *Generalizing Pooling Functions in Convolutional Neural Networks: Mixed, Gated, and Tree* (pp. 464–472). pp. 464–472. Retrieved from <http://proceedings.mlr.press/v51/lee16a.html>
- Liu, L., Wu, X., Wu, L., & Guo, T. (2012). Static Human Gesture grading based on Kinect. *2012 5th International Congress on Image and Signal Processing, (CISP)*, 1390–1393. <https://doi.org/10.1109/CISP.2012.6469917>
- Lowe, W., Chaitow, L., Lowe, W., & Chaitow, L. (2009). Orthopedic Massage. *Orthopedic*

- Massage*, 75–76. <https://doi.org/10.1016/B978-0-443-06812-6.00024-6>
- Lu, T.-W., & Chang, C.-F. (2012). Biomechanics of human movement and its clinical applications. *The Kaohsiung Journal of Medical Sciences*, 28(2), S13–S25. <https://doi.org/10.1016/j.kjms.2011.08.004>
- Magee, D. J., Zachazewski, J. E., Quillen, W. S., Manske, R. C., Wilk, K. E., Arrigo, C. A., ... Aviles, S. A. (2016). Pathology and Intervention in Musculoskeletal Rehabilitation. *Elbow*, 290–343. <https://doi.org/10.1016/B978-0-323-31072-7.00009-9>
- Malagelada, F., Dalmau-Pastor, M., Vega, J., & Golanó, P. (2014). Elbow Anatomy. In *Sports Injuries* (pp. 1–30). https://doi.org/10.1007/978-3-642-36801-1_38-1
- Martinez, J., Hossain, R., Romero, J., & Little, J. J. (n.d.). *A simple yet effective baseline for 3d human pose estimation*. Retrieved from <https://github.com/una-dinosauria/>
- Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.-P., ... Theobalt, C. (2017). *VNect: Real-time 3D Human Pose Estimation with a Single RGB Camera*. Retrieved from <http://gvv.mpi-inf.mpg.de/projects/VNect/>
- Mou, L., Li, G., Zhang, L., Wang, T., & Jin, Z. (2016). Convolutional Neural Networks over Tree Structures for Programming Language Processing. *Thirtieth AAAI Conference on Artificial Intelligence*. Retrieved from <https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/viewPaper/11775>
- Murphy, J. (n.d.). *An Overview of Convolutional Neural Network Architectures for Deep Learning*. Retrieved from <https://pdfs.semanticscholar.org/64db/333bb1b830f937b47d786921af4a6c2b3233.pdf>
- Norkin, C. C., & White, D. J. (2016). *Measurement of joint motion : a guide to goniometry* (5th ed.). Philadelphia: F.A DAVIS COMPANY.
- Parry, I., Richard, R., Aden, J. K., Yelvington, M., Ware, L., Dewey, W., ... Sen, S. (2019). Goniometric Measurement of Burn Scar Contracture: A Paradigm Shift Challenging the

- Standard. *Journal of Burn Care & Research*. <https://doi.org/10.1093/jbcr/irz038>
- Pinheiro, P. O., Collobert, R., & Dollár, P. (2017). *Learning to Segment Object Candidates*. Retrieved from <https://papers.nips.cc/paper/5852-learning-to-segment-object-candidates.pdf>
- Qiao, S., Wang, Y., & Li, J. (2017). Real-time human gesture grading based on OpenPose. *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 1–6. <https://doi.org/10.1109/CISP-BMEI.2017.8301910>
- Qiao, S., Wang, Y., & Li, J. (2018). Real-time human gesture grading based on OpenPose. *Proceedings - 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2017, 2018-Janua*, 1–6. <https://doi.org/10.1109/CISP-BMEI.2017.8301910>
- Ruiz-Olaya, A. F., Callejas-Cuervo, M., Lara-Herrera, C. N., Ruiz-Olaya, A. F., Callejas-Cuervo, M., & Lara-Herrera, C. N. (2017). Wearable low-cost inertial sensor-based electrogoniometer for measuring joint range of motion. *DYNA*, *84*(201), 180. <https://doi.org/10.15446/dyna.v84n201.59054>
- Sabari, J. S., Maltzev, I., Lubarsky, D., Liskay, E., & Homel, P. (1998). *Goniometric Assessment of Shoulder Range of Motion: Comparison of Testing in Supine and Sitting Positions*. Retrieved from [https://www.archives-pmr.org/article/S0003-9993\(98\)90038-7/pdf](https://www.archives-pmr.org/article/S0003-9993(98)90038-7/pdf)
- Sakhavi, S., Guan, C., & Yan, S. (2015). Parallel convolutional-linear neural network for motor imagery classification. *2015 23rd European Signal Processing Conference (EUSIPCO)*, 2736–2740. <https://doi.org/10.1109/EUSIPCO.2015.7362882>
- Shahid, N., Rappon, T., & Berta, W. (2019). Applications of artificial neural networks in health care organizational decision-making: A scoping review. *PLOS ONE*, *14*(2), e0212356.

<https://doi.org/10.1371/journal.pone.0212356>

Shobha, G., & Rangaswamy, S. (2018). Machine Learning. In *Handbook of Statistics* (Vol. 38, pp. 197–228). <https://doi.org/10.1016/bs.host.2018.07.004>

Simonyan, K., & Zisserman, A. (2015). *VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION*. Retrieved from <http://www.robots.ox.ac.uk/>

Suárez, N., & Osorio, A. (2013). *Biomecánica del hombro y bases fisiológicas de los ejercicios de Codman*. Retrieved from <http://www.scielo.org.co/pdf/cesm/v27n2/v27n2a08.pdf>

Sugiyama, M. (2016). Semisupervised Learning. In *Introduction to Statistical Machine Learning* (pp. 375–390). <https://doi.org/10.1016/b978-0-12-802121-7.00044-3>

Taboadela, C. (2007). *Goniometria: Una herramienta para la evaluación de las incapacidades laborales*. Buenos Aires: ASOCIART SA ART.

Wang, H., Zhang, Q., Wang, Y., & Hu, H. (2017). *Structured Probabilistic Pruning for Convolutional Neural Network Acceleration*. Retrieved from <http://arxiv.org/abs/1709.06994>

Wolff, A. L., & Rosenzweig, L. (2017). Anatomical and biomechanical framework for shoulder arthroplasty rehabilitation. *Journal of Hand Therapy*, 30(2), 167–174. <https://doi.org/10.1016/j.jht.2017.05.009>

Yamamoto, N., & Itoi, E. (2015). A review of biomechanics of the shoulder and biomechanical concepts of rotator cuff repair. *Asia-Pacific Journal of Sports Medicine, Arthroscopy, Rehabilitation and Technology*, 2(1), 27–30. <https://doi.org/10.1016/J.ASMART.2014.11.004>

Zulkarnain, R. F., Kim, G.-Y., Adikrishna, A., Hong, H. P., Kim, Y. J., & Jeon, I.-H. (2017). Digital data acquisition of shoulder range of motion and arm motion smoothness using Kinect v2. *Journal of Shoulder and Elbow Surgery*, 26(5), 895–901. <https://doi.org/10.1016/j.jse.2016.10.026>

Glosario de Términos y Acrónimos.

Articulación: estructura anatómica que permite la unión entre dos huesos o entre un hueso y un cartílago.

BSD: Es una licencia de software libre permisiva.

C#: significa lenguaje de programación, el cual es utilizado con mayor frecuencia.

Codificación: proceso de conversión en símbolos de una determinada información con el fin de ser comunicada.

CUDA: plataforma de computación en paralelo.

Fps: Los cuadros por segundo es la medida de fotogramas, cuadros o imágenes distintas que un video muestra en un segundo.

GPU: coprocesador dedicado al procesamiento de gráficos u operaciones, para aligerar la carga de trabajo del procesador central.

Inferencia: Proceso de aprendizaje de las ponderaciones de los pesos de una CNN.

NMS.- Algoritmo de supresión no máxima

Machine Learning: Aprendizaje automático es una rama de la inteligencia artificial cuyo objetivo es desarrollar técnicas que permitan a las computadoras aprender.

OpenCv: Es una librería de código abierto para el desarrollo de funciones de programación dirigidas principalmente a visión por computador y con un fuerte enfoque en aplicaciones que necesitan ser interpretadas en tiempo real.

PAF.- Campos de afinidad por parte

Python: Es un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional.

RGB: es un modelo de color utilizado normalmente para presentar color en los sistemas de video, cámaras, y monitores de ordenadores. Representa todos los colores como combinaciones de rojo, verde y azul.

SySR: Requerimientos iniciales del sistema se definen los límites funcionales del sistema en términos de comportamiento y propiedades del proyecto

SRSR: Requerimiento de arquitectura se definen los requerimientos de hardware, software y el sistema eléctrico.

Stakeholder: Es todo grupo o individuo que tiene un interés directo en el resultado obtenido por el desarrollo del proyecto.

StSR: Requerimientos de stakeholders identificar los requisitos de los interesados para el sistema.

Vector: segmento de línea que con dirección y sentido.

ANEXOS

Anexo A. Script de programación para el inicio del sistema

```

import cv2
import numpy as np

from tf_pose.estimator import TfPoseEstimator
from tf_pose.networks import get_graph_path, model_wh

def runpose():
    print("asd!!!")
    logger = logging.getLogger('TfPoseEstimator-WebCam')
    logger.setLevel(logging.DEBUG)
    ch = logging.StreamHandler()
    ch.setLevel(logging.DEBUG)
    formatter = logging.Formatter('%(asctime)s] %(name)s] %(levelname)s] %(message)s')
    ch.setFormatter(formatter)
    logger.addHandler(ch)
    fps_time = 0
    resize = '432x368'
    resize_out_ratio = 4.0
    model = 'cmu'
    logger.debug('initialization %s : %s' % (model, get_graph_path(model)))
    w, h = model_wh(resize)
    if w > 0 and h > 0:
        e = TfPoseEstimator(get_graph_path(model), target_size=(w, h))
    else:
        e = TfPoseEstimator(get_graph_path(model), target_size=(432, 368))
        #logger.debug('cam read+')
    cam = cv2.VideoCapture(0)
    ret_val, image = cam.read()
    logger.info('cam image=%dx%d' % (image.shape[1], image.shape[0]))

    while True:
        ret_val, image = cam.read()

        #logger.debug('image process+')
        humans = e.inference(image, resize_to_default=(w > 0 and h > 0),
upsample_size=resize_out_ratio)
        print("////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////")
        print(humans)

        #logger.debug('postprocess+')
        image,ci,cd,hi,hd = TfPoseEstimator.draw_humans(image, humans, imgcopy=False)
        # print("//////////////////////////////////////////////////////////////////ANGULOS OBTENIDOS////////////////////////////////////////////////////////////////")
        # print(ci)
        # print(cd)
        # print("////////////////////////////////////////////////////////////////")
        #logger.debug('show+')
        cv2.putText(image,
                    "FPS: %f" % (1.0 / (time.time() - fps_time)),
                    (10, 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
                    (0, 255, 0), 2)
        cv2.imshow('tf-pose-estimation result', image)
        fps_time = time.time()

```

```

if cv2.waitKey(1) == 27:
    break
    #logger.debug('finished+')

cv2.destroyAllWindows()
return ci, cd, hi, hd

```

Anexo B. Programación Calculo de Ángulos Articulares

A continuación, se detalla el código de programación realizado en el script estimator.py del sistema donde fue implementado el cálculo de ángulos articulares para codo y hombro en el lenguaje de programación Python.

```

#print(len(humans))
global ang
global ang1
global ang2
global ang3
global angt1
global angt2
global angt3
global angt4
asd = 0
ang = 0
ang1 = 0
ang2 = 0
ang3 = 0
angt1 = 0
angt2 = 0
angt3 = 0
angt4 = 0
for human in humans:
    # draw point
    for i in range(common.CocoPart.Background.value):
        if i not in human.body_parts.keys():
            continue

        body_part = human.body_parts[i]
        center = (int(body_part.x * image_w + 0.5), int(body_part.y * image_h + 0.5))
        #print(len(body_part))
        centers[i] = center
        if(i>=1 and i<=7):
            #print(center)
            cv2.circle(npimg, center, 3, common.CocoColors[i], thickness=7, lineType=10,
shift=0)

            asd+=1
            #print(asd)

#####CODOS#####
try:
    if(centers[5] != "" and centers[6] != "" and centers[7] != "" and asd >= 6):

```

```

print("PUNTO D")
d = centers[5]
print(d)
print("PUNTO E")
e = centers[6]
print(e)
print("PUNTO F")
f = centers[7]
print(f)
ang1 = math.degrees(math.atan2(f[1]-e[1], f[0]-e[0]) - math.atan2(d[1]-e[1],
d[0]-e[0]))

print("ANGULO IZQUIERDO CODO")
if (ang1 < 0):
    ang1 = ang1*(-1)
    "{0:.1f}".format(ang1)
    print(ang1)
else:
    ang1 = (ang1-360)*(-1)
    print(ang1)
angt1 = "{0:.1f}".format(180 - ang1)

cv2.putText(npimg, "CODO IZQUIERDO", (320,25),
cv2.FONT_HERSHEY_COMPLEX_SMALL,1, (0, 255, 255), thickness=2, lineType=4)
cv2.putText(npimg, "{0:.1f}".format(ang1), (320,40),
cv2.FONT_HERSHEY_COMPLEX_SMALL,1, (0, 255, 255), thickness=2, lineType=4)

except KeyError:
    pass

try:
if(centers[2] != "" and centers[3] != "" and centers[4] != "" and asd >= 6):
print("PUNTO A")
a = centers[2]
print(a)
print("PUNTO B")
b = centers[3]
print(b)
print("PUNTO C")
c = centers[4]
print(c)
ang = math.degrees(math.atan2(c[1]-b[1], c[0]-b[0]) - math.atan2(a[1]-b[1],
a[0]-b[0]))

print("ANGULO DERECHO CODO")
if (ang < 0):
    ang = ang + 360
    "{0:.1f}".format(ang)
    print(ang)
else:
    "{0:.1f}".format(ang)
    print(ang)
angt2 = "{0:.1f}".format(180 - ang)

cv2.putText(npimg, "CODO DERECHO", (80,25),
cv2.FONT_HERSHEY_COMPLEX_SMALL,1, (0, 255, 255), thickness=2, lineType=4)
cv2.putText(npimg, "{0:.1f}".format(ang), (80,40),
cv2.FONT_HERSHEY_COMPLEX_SMALL,1, (0, 255, 255), thickness=2, lineType=4)

```

```

except KeyError:
    pass

#####CODOS#####

#####HOMBROS#####
////

try:
    if(centers[1] != "" and centers[5] != "" and centers[6] != "" and asd >= 6):
        print("PUNTO J")
        j = centers[1]
        print(j)
        print("PUNTO K")
        k = centers[5]
        print(k)
        print("PUNTO L")
        l = centers[6]
        print(l)
        ang2 = math.degrees(math.atan2(l[1]-k[1], l[0]-k[0]) - math.atan2(j[1]-k[1],
j[0]-k[0]))

        print("ANGULO IZQUIERDO HOMBRO")
        if (ang2 < 0):
            ang2 = ang2*(-1)
            "{0:.1f}".format(ang2)
            print(ang2)
        else:
            ang2 = (ang2-360)*(-1)
            print(ang2)
            ang3 = "{0:.1f}".format(ang2 - 90)

            cv2.putText(npimg, "HOMBRO IZQUIERDO", (320,95),
cv2.FONT_HERSHEY_COMPLEX_SMALL,1, (14, 41, 105), thickness=2, lineType=4)
            cv2.putText(npimg, "{0:.1f}".format(ang2-90), (320,75),
cv2.FONT_HERSHEY_COMPLEX_SMALL,1, (14, 41, 105), thickness=2, lineType=4)

except KeyError:
    pass

try:
    if(centers[1] != "" and centers[2] != "" and centers[3] != "" and asd >= 6):
        print("PUNTO G")
        g = centers[1]
        print(g)
        print("PUNTO H")
        h = centers[2]
        print(h)
        print("PUNTO I")
        i = centers[3]
        print(i)
        ang3 = math.degrees(math.atan2(i[1]-h[1], i[0]-h[0]) - math.atan2(g[1]-h[1],
g[0]-h[0]))

```



```

print("ANGULO DERECHO HOMBRO")
if (ang3 < 0):
    ang3 = ang3 + 360
    "{0:.1f}".format(ang3)
    print(ang3)
else:
    "{0:.1f}".format(ang3)
    print(ang3)
angt4 = "{0:.1f}".format(ang3 -90)

    cv2.putText(npimg, "HOMBRO DERECHO", (80,95),
cv2.FONT_HERSHEY_COMPLEX_SMALL,1, (14, 14, 105), thickness=2, lineType=4)
    cv2.putText(npimg, "{0:.1f}".format(ang3-90), (80,75),
cv2.FONT_HERSHEY_COMPLEX_SMALL,1, (14, 41, 105), thickness=2, lineType=4)

except KeyError:
    pass

#####HOMBROS#####
////

    for pair_order, pair in enumerate(common.CocoPairsRender):
        if pair[0] not in human.body_parts.keys() or pair[1] not in
human.body_parts.keys():
            continue
            #if(i>=1 and i<=7):
            # npimg = cv2.line(npimg, centers[pair[0]], centers[pair[1]],
common.CocoColors[pair_order], 3)
            cv2.line(npimg, centers[pair[0]], centers[pair[1]],
common.CocoColors[pair_order], 3)

            # if(ang1 != "" and ang == ""):
            # ang = 0
            # if(ang1 == "" and ang != ""):
            # ang1 = 0
            # if(ang1 == "" and ang == ""):
            # ang1 = 0
            # ang = 0

            #####

            # if(ang2 != "" and ang3 == ""):
            # ang3 = 0
            # angt4 = 0
            # if(ang2 == "" and ang3 != ""):
            # ang2 = 0
            # angt3 = 0
            # if(ang2 == "" and ang3 == ""):
            # ang2 = 0
            # angt3 = 0
            # ang3 = 0
            # angt4 = 0
            print("VALORES NETOS")
            print(angt1)
            print(angt2)
            print(angt3)

```

```

    print(angt4)
    return npimg, angt1, angt2, angt3, angt4

```

Anexo C. Creación de la Interfaz gráfica

A continuación, se detalla el código usado para la elaboración de la interfaz gráfica para el manejo del sistema

```

import numpy
from pygame import mixer
import time
import cv2
from tkinter import *
import tkinter.messagebox
#from detect_face_landmarks_ip_camera_GUI import facelandmarks
from run_webcam import runpose
from formulario import registerinfo

root=Tk()
root.geometry('1000x400')
frame = Frame(root, relief=RIDGE, borderwidth=2)
frame.pack(fill=BOTH,expand=1)
root.title('UNIVERSIDAD TECNICA DEL NORTE')
frame.config(background='light blue')
label = Label(frame, text="Sistema De Medicion de Angulos Articulares",bg='light
blue',font=("Times 35 bold"))
label.pack(side=TOP)
filename = PhotoImage(file="/POSE.png")
background_label = Label(frame,image=filename)
background_label.pack(side=TOP)

def hel():
    help(cv2)

def Contri():
    tkinter.messagebox.showinfo("Desarrolladores","\n1. Jose Luis Jarrin\n2. MsC. Jaime
Michilena \n3. MsC.Edgar Maya \n4. MsC.Daniela Zurita")

def anotherWin():
    tkinter.messagebox.showinfo("Acerca de",'Sistema de Estimacion de pose UTN v1.0\n
Herramientas empleadas\n-Tensorflow\n-OpenCV\n-SQLite\n-Tkinter\n-Plataforma Python
3.5.2')

menu = Menu(root)
root.config(menu=menu)

subm1 = Menu(menu)

```

```

menu.add_cascade(label="Herramientas",menu=subm1)
subm1.add_command(label="Open CV Docs",command=hel)

subm2 = Menu(menu)
menu.add_cascade(label="Acerca de",menu=subm2)
subm2.add_command(label="Estimacion de pose SDK",command=anotherWin)
subm2.add_command(label="Colaboradores",command=Contri)

def CAMARA():
    runpose()
def REGISTERDB():
    registerinfo()
def exitt():
    exit()

# def GENERATEREPORT():
#     makeReport()

but1=Button(frame,padx=5,pady=5,width=20,bg='white',fg='black',relief=GROOVE,command=CAMARA,text='Prueba',font=('helvetica 15 bold'))
but1.place(x=250,y=100)

////////////////////////////////////

but2=Button(frame,padx=5,pady=5,width=20,bg='white',fg='black',relief=GROOVE,command=REGISTERDB,text='Ingresar paciente',font=('helvetica 15 bold'))
but2.place(x=250,y=200)

but3=Button(frame,padx=5,pady=5,width=5,bg='white',fg='black',relief=GROOVE,text='SALIR',command=exitt,font=('helvetica 15 bold'))
but3.place(x=325,y=300)

root.mainloop()

```

Anexo D. Elaboración del formulario para la base de datos del sistema

El siguiente script contiene las líneas de código utilizadas para la elaboración del formulario que alojara la información obtenida por el sistema para la base de datos implementada.

```

//////////Liberias importadas//////////
from tkinter import *
import sqlite3
import cv2
import sqlite3
import tensorflow as tf
import os
import numpy as np
import tkinter.messagebox

```

```

from run_webcam import runpose

def registerinfo():
    global ci,cd,hi,hd
    root2 = Tk()
    root2.geometry('800x600')
    root2.title("FICHA DE INGRESO")

    cedula=StringVar(root2)
    apellidos=StringVar(root2)
    nombres=StringVar(root2)
    patologia=StringVar(root2)
    var = IntVar(root2)
    edad=StringVar(root2)
    ocupacion=StringVar(root2)
    movimiento=StringVar(root2)

    FECO = StringVar(root2)
    FECO1 = StringVar(root2)
    AAHO = StringVar(root2)
    AAHO1 = StringVar(root2)
    AAHO11 = StringVar(root2)
    AAHO111 = StringVar(root2)

    def getinfo():
        ID=cedula.get()
        ap=apellidos.get()
        no=nombres.get()
        pa=patologia.get()
        ge=var.get()
        ed=edad.get()
        oc=ocupacion.get()
        mo=movimiento.get()
        feco = FECO.get()
        feco1 = FECO1.get()
        aaho = AAHO.get()
        aaho1 = AAHO1.get()
        aaho11 = AAHO11.get()
        aaho111 = AAHO111.get()

        if ge == 1:
            ge = "Masculino"
        else:
            ge = "Femenino"

        conn = sqlite3.connect('PacientesUTN.db')
        with conn:
            cursor=conn.cursor()
            cursor.execute('CREATE TABLE IF NOT EXISTS PacientesUTNINFO (Cedula
TEXT,Apellidos TEXT,Nombres TEXT,Patologia TEXT,Genero TEXT,Edad TEXT,Ocupación
TEXT, Movimiento TEXT, Codo_Izquierdo TEXT, Codo_Derecho TEXT,Hombro_ABD_Izquierdo
TEXT, Hombro_ABD_Derecho TEXT,Hombro_AD_Izquierdo TEXT, Hombro_AD_Derecho TEXT)')
            cmd = "SELECT * FROM PacientesUTNINFO WHERE Cedula="+str(ID)
            cursor = conn.execute(cmd)
            RegistroExistente = 0
            for row in cursor:
                RegistroExistente = 1
            if RegistroExistente == 1:

```

```

        cursor.execute("""UPDATE PacientesUTNINFO SET Apellidos = ? ,Nombres =
?,Patologia = ? WHERE Cedula= ? """,(ap,no,pa,ID))
tkinter.messagebox.showinfo("Mensaje","\nRegistro existente, siga las
instrucciones:\n1. Modifique el ultimo registro creado en la BD\n2. Borre el registro existente en
la BD\n3. Ingrese nuevamente los datos en el formulario")
    else:
        cursor.execute('INSERT INTO PacientesUTNINFO
(Cedula,Apellidos,Nombres,Patologia,Genero,Edad,Ocupación,Movimiento,
Codo_Izquierdo,Codo_Derecho,Hombro_ABD_Izquierdo,Hombro_ABD_Derecho,Hombro_AD_Izq
uierdo,Hombro_AD_Derecho)
VALUES(?,?,?,?,?,?,?,?,?,?,?,?,?)',(ID,ap,no,pa,ge,ed,oc,mo,feco,feco1,aaho,aaho1,aaho11,aaho111,
))
        conn.commit()
tkinter.messagebox.showinfo("Mensaje","\nDatos ingresados correctamente en la
BD\n")
        conn.commit()

def FECOT():
    ci, cd, hi, hd = runpose()
    FECO.set(ci)
    FECO1.set(cd)
    print("////////////////////////////////ANGULOS OBTENIDOS CODO////////////////////////////////")
    print(ci)
    print(cd)
    print("////////////////////////////////")

def FEHOTA():
    ci, cd, hi, hd = runpose()
    AAHO.set(hi)
    AAHO1.set(hd)
    print("////////////////////////////////ANGULOS OBTENIDOS HOMBRO////////////////////////////////")
    print(hi)
    print(hd)
    print("////////////////////////////////")

def FEHOTAB():
    ci, cd, hi, hd = runpose()
    hi = (float(hi) -6) * (-1)
    hd = (float(hd) -6) * (-1)
    AAHO11.set(hi)
    AAHO111.set(hd)
    print("////////////////////////////////ANGULOS OBTENIDOS HOMBRO////////////////////////////////")
    print(hi)
    print(hd)
    print("////////////////////////////////")

label_0 = Label(root2, text="Formulario de registro",width=20,font=("bold", 20))
label_0.place(x=90,y=53)

label_1 = Label(root2, text="Toma de datos",width=20,font=("bold", 20))
label_1.place(x=450,y=53)

label_2 = Label(root2, text="Cédula(ID):",width=20,font=("bold", 10))
label_2.place(x=80,y=130)

entry_1 = Entry(root2,textvar=cedula)
entry_1.place(x=240,y=130)

```

```

#####
label_3 = Label(root2, text="FLEXION-EXTENSION CODO:",width=30,font=("bold", 10))
label_3.place(x=500,y=130)

label_31 = Label(root2, text="Codo izquierdo:",width=20,font=("bold", 10))
label_31.place(x=400,y=160)

entry_2 = Entry(root2,textvar=FECO)
entry_2.place(x=535,y=160)

label_32 = Label(root2, text="Codo derecho:",width=20,font=("bold", 10))
label_32.place(x=400,y=180)

entry_21 = Entry(root2,textvar=FECO1)
entry_21.place(x=535,y=180)

#####

label_4 = Label(root2, text="ABDUCCION HOMBRO:",width=30,font=("bold", 10))
label_4.place(x=500,y=260)

label_311 = Label(root2, text="Hombro izq:",width=20,font=("bold", 10))
label_311.place(x=400,y=290)

entry_3 = Entry(root2,textvar=AAHO)
entry_3.place(x=535,y=290)

label_312 = Label(root2, text="Hombro der:",width=20,font=("bold", 10))
label_312.place(x=400,y=310)

entry_31 = Entry(root2,textvar=AAHO1)
entry_31.place(x=535,y=310)

label_41 = Label(root2, text="ADUCCION HOMBRO:",width=30,font=("bold", 10))
label_41.place(x=500,y=380)

label_312 = Label(root2, text="Hombro izq:",width=20,font=("bold", 10))
label_312.place(x=400,y=410)

entry_32 = Entry(root2,textvar=AAHO11)
entry_32.place(x=535,y=410)

label_313 = Label(root2, text="Hombro der:",width=20,font=("bold", 10))
label_313.place(x=400,y=430)

entry_33 = Entry(root2,textvar=AAHO111)
entry_33.place(x=535,y=430)

#####

label_5 = Label(root2, text="Apellidos:",width=20,font=("bold", 10))
label_5.place(x=80,y=180)

entry_4 = Entry(root2,textvar=apellidos)
entry_4.place(x=240,y=180)

```

```

label_6 = Label(root2, text="Nombres:",width=20,font=("bold", 10))
label_6.place(x=80,y=230)

entry_5 = Entry(root2,textvar=nombres)
entry_5.place(x=240,y=230)

label_7 = Label(root2, text="Patologia:",width=20,font=("bold", 10))
label_7.place(x=80,y=280)

entry_6 = Entry(root2,textvar=patologia)
entry_6.place(x=240,y=280)

label_8 = Label(root2, text="Género:",width=20,font=("bold", 10))
label_8.place(x=80,y=330)

Radiobutton(root2, text="Masculino",padx = 5, variable=var,
value=1).place(x=235,y=325)
Radiobutton(root2, text="Femenino",padx = 5, variable=var, value=2).place(x=235,y=340)

label_9 = Label(root2, text="Edad:",width=20,font=("bold", 10))
label_9.place(x=80,y=380)

entry_7 = Entry(root2,textvar=edad)
entry_7.place(x=240,y=380)

label_10 = Label(root2, text="Ocupación:",width=20,font=("bold", 10))
label_10.place(x=80,y=430)

list3 = ['Estudiante','Docente','Personal Administrativo','Otro'];

droplist=OptionMenu(root2,ocupacion, *list3)
droplist.config(width=17)
ocupacion.set('Seleccionar ocupación')
droplist.place(x=232,y=430)

label_11 = Label(root2, text="Movimiento:",width=20,font=("bold", 10))
label_11.place(x=400,y=100)

list4 = ['Flexion','Extension','N/A'];

droplist1=OptionMenu(root2,movimiento, *list4)
droplist1.config(width=17)
movimiento.set('Seleccionar')
droplist1.place(x=525,y=95)

Button(root2,
text='Registrar',width=20,bg='black',fg='green',command=getinfo).place(x=230,y=500)

#####CODO#####

Button(root2,
text='Iniciar1',width=10,bg='black',fg='green',command=FECOT).place(x=560,y=210)

#Button(root2,
text='Capturar',width=10,bg='black',fg='green',command=FECOANGLES).place(x=620,y=210)

#####HOMBRO#####

```

```
    Button(root2,  
text='Iniciar2',width=10,bg='black',fg='green',command=FEHOTA).place(x=560,y=340)  
  
    Button(root2,  
text='Iniciar3',width=10,bg='black',fg='green',command=FEHOTAB).place(x=560,y=460)  
  
    root2.mainloop()
```


Anexo E. Pacientes del Centro de Rehabilitación UTN

Figura 1. Goniometría de codo



Figura 2. Goniometría Flexión de codo



Figura 3. Abducción de hombro



Figura 4. Goniometría de hombro en reposo



Figura 5. Flexión de codo



Figura 6. Goniometría de codo en reposo



Figura 7. Abducción de hombro



Figura 8. Goniometría de hombro en reposo



Figura 9. Abducción de hombro



Figura 10. Goniometría de hombro en reposo



Figura 11. Goniometría de codo e reposo



Figura 12. Goniometría de flexión de codo

Anexo F. Datos validados por el experto

Universidad Técnica del Norte
Centro de Rehabilitación UTN

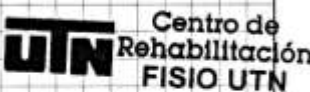
Registro de mediciones para las pruebas de funcionamiento del sistema de medición de ángulos articulares para miembros superiores en la flexión/extension de codo y la aducción y abducción de hombro

Flexo - Extensión de Codo


N°	Angulo Inicial	Angulo Final
1	10°	137°
2	11°	132°
3	12,5°	115°
4	15°	135°
5	7°	120°
6	9°	123°
7	12°	137°
8	10,8°	141°
9	7,8°	136°

Abducción y Aducción de Hombro

N°	Angulo Inicial	Angulo Final
1	7°	115°
2	18°	100°
3	10°	169°
4	16°	16°



 Centro de Rehabilitación
 FISIO UTN



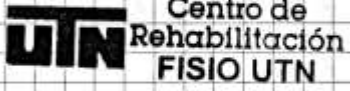
 DR. JORGE L. ZAMBRANO
 FISIOTERAPÉUTA
 RESERVA 1013-01

Educho

5	24°	18°
6	7°	120°
7	7°	115°
8	10°	109°
9	20°	21°



 LIC. JORGE L. ZAMBRANO
 FISIOTERAPEUTA
 REG. SEC. CTTO. 1970/1985/86



UTN Centro de
 Rehabilitación
 FISIS UTN

Las datos obtenidos son válidos para la comparación con la muestra
 de datos que se obtiene con el sistema de medición de ángulos articular
 mediante visión artificial



 Centro de
 Rehabilitación
 FISIS UTN

Anexo G. Certificación de pruebas de funcionamiento**CERTIFICADO**

Ibarra, 31 de julio de 2019

A quien corresponda:

YO, Jorge Luis Zambrano Vásquez, con cédula de identidad 0401696216 en mi calidad de Fisioterapeuta responsable del Centro de Rehabilitación de la Universidad Técnica del Norte, certifico que el señor José Luis Jarrín Chacón con cédula de identidad 1718898362 realizó pruebas de funcionamiento de su proyecto de investigación titulado: SISTEMA DE DETECCIÓN DEL ÁNGULO ARTICULAR EN LOS MOVIMIENTOS DE MIEMBRO SUPERIOR PARA EVALUACIÓN EN FISIOTERAPIA MEDIANTE VISIÓN ARTIFICIAL, cuyos resultados arrojaron coherencia objetiva en relación a las pruebas clínicas realizadas manualmente con el goniómetro.

Es todo cuanto puedo certificar en honor a la verdad, el señor José Luis Jarrín puede hacer uso del presente certificado como estime conveniente.

Atentamente:

UTN Centro de
Rehabilitación
FISIO UTN

Lic. Jorge Zambrano FT.

LIC JORGE L. ZAMBRANO
FISIOTERAPEUTA
REG. SANE. C. V. JULIO 2017-1855986