



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN

Tema:

**SISTEMA INTELIGENTE USANDO VIDEO CÁMARAS Y MOTION
CAPTURE PARA MONITOREO DE LA REHABILITACIÓN DE LA MANO
DERECHA**

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la obtención del Título de Ingeniero en Tecnologías de la Información.

ÁREA: Software

LÍNEA DE INVESTIGACIÓN: Desarrollo de software

AUTOR: Miguel Enrique Bonilla Quishpe

TUTOR: Ing. Rubén Eduardo Nogales Portero Mg.

Ambato – Ecuador

agosto – 2023

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: SISTEMA INTELIGENTE USANDO VIDEO CÁMARAS Y MOTION CAPTURE PARA MONITOREO DE LA REHABILITACIÓN DE LA MANO DERECHA, desarrollado bajo la modalidad Proyecto de Investigación por el señor Miguel Enrique Bonilla Quishpe, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, agosto 2023

Ing. Rubén Eduardo Nogales Portero, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación titulado: SISTEMA INTELIGENTE USANDO VIDEO CÁMARAS Y MOTION CAPTURE PARA MONITOREO DE LA REHABILITACIÓN DE LA MANO DERECHA, es absolutamente original, autentico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, agosto 2023



Miguel Enrique Bonilla Quishpe

C. C. 1805326343

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta de Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, agosto 2023



Miguel Enrique Bonilla Quishpe

C. C. 1805326343

AUTOR

APROBACIÓN TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor Miguel Enrique Bonilla Quishpe, estudiante de la Carrera de Tecnologías de la Información, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado SISTEMA INTELIGENTE USANDO VIDEO CÁMARAS Y MOTION CAPTURE PARA MONITOREO DE LA REHABILITACIÓN DE LA MANO DERECHA, nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, conjuntamente con la señora Presidente del Tribunal.

Ambato, agosto 2023

Ing. Elsa Pilar Urrutia Urrutia, Mg.

PRESIDENTE DEL TRIBUNAL

Ing. Jaime Bolívar Ruiz Banda, Mg.

PROFESOR CALIFICADOR

PhD. José Vicente Morales Lozada

PROFESOR CALIFICADOR

DEDICATORIA

Principalmente, dedico este trabajo a Dios, como fuente de sabiduría y amor incondicional. A través de cada paso, su guía y apoyo han sido invaluable, iluminando mi mente en momentos de duda y brindándome fuerza para seguir adelante. Agradezco profundamente su generosidad al permitirme alcanzar este logro, reconociendo que todo talento proviene de Él.

Con infinito cariño y gratitud dedico este proyecto a mis padres, especialmente a mi amada madre Patricia quien con su amor incondicional ha sido mi guía y sostén durante toda mi vida. Sus palabras de aliento y su ejemplo de perseverancia han sido mi inspiración para seguir adelante incluso en los momentos más difíciles. Gracias por creer en mí y por apoyarme en cada decisión que he tomado. Tu sabiduría y dedicación han sido fundamentales en mi formación como persona y como estudiante. Te dedico este logro con el corazón lleno de gratitud y amor.

A toda mi familia, cada uno de ustedes ha dejado una huella imborrable en mi vida. Su presencia en cada celebración, en cada momento de alegría y en cada desafío ha sido un pilar fundamental en mi desarrollo como individuo. Los valores y la unidad que nos caracterizan como familia son el cimiento sobre el cual construí este logro. Cada uno de ustedes ha sido mi fuente de inspiración y fortaleza.

En especial, quiero dedicar este proyecto a la memoria de mi querido abuelito José Quishpe, quien más que un abuelo, fue un padre ejemplar, cuya sabiduría y ejemplo de lucha incansable han sido un faro que me ha guiado hacia la senda de la fe y el amor en cada desafío. Las enseñanzas que compartió conmigo durante su vida siguen resonando en mi corazón, y su legado perdurará en mis acciones y decisiones.

Miguel Enrique Bonilla Quishpe

AGRADECIMIENTO

En primer lugar, quiero dar gracias a Dios, quien me ha brindado la fuerza, perseverancia y sabiduría necesaria para superar los desafíos que se presentaron a lo largo de este camino.

A mis padres, les debo todo mi amor y gratitud. Gracias por su incondicional apoyo, comprensión y sacrificio.

A cada miembro de mi familia, quienes han sido el pilar fundamental que ha sostenido cada paso de este camino. Su amor incondicional y su constante respaldo han sido mi mayor motivación en los momentos de desafío.

Al Ing. Rubén Nogales, mi tutor, le agradezco su guía experta y apoyo constante, fundamentales para el éxito del trabajo. Ha sido un privilegio contar con su mentoría.

A mis compañeros, gracias a su apoyo, colaboración y amistad que han sido fundamentales en este camino. Gracias por hacer que esta experiencia sea significativa y enriquecedora.

ÍNDICE DE CONTENIDOS

PORTADA	i
APROBACIÓN DEL TUTOR.....	ii
AUTORÍA.....	iii
DERECHOS DE AUTOR.....	iv
APROBACIÓN TRIBUNAL DE GRADO	v
DEDICATORIA.....	vi
AGRADECIMIENTO.....	vii
ÍNDICE DE CONTENIDOS	viii
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE TABLAS.....	xi
RESUMEN EJECUTIVO	xii
ABSTRACT.....	xiii
CAPÍTULO I.- MARCO TEÓRICO	1
1.1. Tema de Investigación.....	1
1.1.1. Planteamiento del Problema.....	1
1.2. Antecedentes Investigativos	4
1.3. Fundamentación Teórica.....	13
1.3.1. Tecnologías de la Información	13
1.3.2. Sistemas Informáticos	14
1.3.3. Inteligencia Artificial	14
1.3.4. Sistema Inteligente.....	15
1.3.5. Rehabilitación de Manos.....	16
1.3.6. Terapia Ocupacional de Manos.....	16
1.3.7. Ingeniería Biomédica Aplicada a la Rehabilitación.....	17
1.3.8. Biomedicina	18
1.4. Objetivos.....	19
1.4.1. Objetivo General.....	19
1.4.2. Objetivos Específicos.....	19
CAPÍTULO II – METODOLOGÍA.....	20
2.1. Materiales	20
2.2. Métodos	21

2.2.1.	Modalidad de la investigación	21
2.2.2.	Población y Muestra.....	21
2.2.3.	Recolección de la Información.....	23
CAPÍTULO III – RESULTADOS Y DISCUSIÓN.....		38
3.1.	Análisis y discusión de los resultados	38
3.1.1.	Proceso de la rehabilitación de la mano derecha	38
3.1.2.	Ejercicios de rehabilitación para la mano derecha.....	43
3.1.3.	Patologías que necesitan rehabilitación	46
3.2.	Desarrollo de la propuesta.....	48
3.2.1.	Adquisición de datos mediante video cámaras.....	48
3.2.2.	Lenguaje para el desarrollo de algoritmos de Machine Learning	51
3.2.3.	Metodología de desarrollo.....	53
3.2.4.	Aplicación de la metodología de desarrollo	56
CAPÍTULO IV. – CONCLUSIONES Y RECOMENDACIONES		80
4.1.	Conclusiones.....	80
4.2.	Recomendaciones.....	81
REFERENCIAS BIBLIOGRÁFICAS.....		82
ANEXOS.....		86

ÍNDICE DE FIGURAS

Figura 1. Uso de sistemas de rehabilitación virtual en la mano derecha	23
Figura 2. Familiaridad con la rehabilitación virtual.....	24
Figura 3. Uso de sistemas informáticos	25
Figura 4. Beneficio de un sistema de rehabilitación virtual para la mano derecha...	26
Figura 5. Confianza en el sistema de monitoreo de rehabilitación	27
Figura 6. Comodidad al usar el sistema de rehabilitación	28
Figura 7. Beneficios de un sistema de monitoreo de la rehabilitación	29
Figura 8. Desventajas de usar un sistema de rehabilitación virtual	30
Figura 9. Características y funcionalidades del sistema	31
Figura 10. Desafíos potenciales en el uso del sistema	33
Figura 11. Costo de terapias en consultorios privados.....	35
Figura 12. Análisis Exploratorio	36
Figura 13. Códigos de codificación	37
Figura 14. Codificación de la entrevista	37
Figura 15. Flexión y Extensión de la Mano	44
Figura 16. Apertura y cierre de la mano.....	45
Figura 17. Pinza de la mano.....	46
Figura 18. Esquema de la metodología.....	57
Figura 19. Gráfico t-SNE del dataset sin extraer características.....	61
Figura 20. Gráfico t-SNE dataset final.....	62
Figura 21. Pantalla de inicio	69
Figura 22. Video de ejemplo del gesto.....	70
Figura 23. Pantalla de captura de datos.....	70
Figura 24. Avance de la rehabilitación.....	71
Figura 25. Relación entre distancia y ángulo del gesto palma abierta.....	72
Figura 26. Relación distancia - muestras del puño	74
Figura 27. Ángulos de flexión de la mano	75
Figura 28. Ángulos de extensión por muestra	77
Figura 29. Relación distancia con ángulos del gesto de la pinza	78

ÍNDICE DE TABLAS

Tabla 1. Cadenas de Búsqueda.....	4
Tabla 2. Criterios de inclusión y exclusión	6
Tabla 3. Resultados Algoritmo SVM	9
Tabla 4. Resultados de los modelos aplicados en el sistema de reconocimiento de gestos utilizando LMC	10
Tabla 5. Resultados de los casos de estudios.....	11
Tabla 6. Resumen de la investigación de la literatura científica	12
Tabla 7. Población de Estudio	22
Tabla 8. Etapas de la rehabilitación.....	38
Tabla 9. Características utilizadas en la revisión patológica	40
Tabla 10. Tabla comparativa de video cámaras.....	48
Tabla 11. Tabla comparativa de lenguajes de desarrollo para Machine Learning	52
Tabla 12. Tabla comparativa de metodologías	55
Tabla 13. Características del dataset.....	59
Tabla 14. Tabla comparativa de algoritmos de Machine Learning.....	63
Tabla 15. Contenido incluido en el SDK de Unity.....	65

RESUMEN EJECUTIVO

El monitoreo de la rehabilitación se ha convertido en un avance destacado en el ámbito tecnológico, gracias a la convergencia de diferentes tecnologías. La integración de estas tecnologías permite un monitoreo más preciso y detallado de los gestos realizados durante la rehabilitación de la mano. Mediante la captura y análisis de datos se proporciona información en tiempo real sobre el progreso de la rehabilitación. Esto permite a los profesionales de la salud y a los usuarios evaluar su desempeño durante el proceso. Además, el sistema no solo brinda un seguimiento visual del avance diario. También motiva a los usuarios a continuar su tratamiento, lo que contribuye al éxito de su rehabilitación.

En este proyecto de investigación se implementan diversas tecnologías para facilitar el monitoreo y análisis de los datos de rehabilitación de la mano. En primer lugar, se utiliza el Leap Motion Controller para capturar los datos de posición y dirección de la mano durante los gestos realizados. Estos datos son procesados mediante un sistema de adquisición desarrollado en Matlab, permitiendo su análisis y generación de un dataset. El dataset es preprocesado y se extraen las características necesarias para la clasificación de cada gesto. Para ello se implementaron diferentes modelos de machine learning, como Redes Neuronales Artificiales (RNA) y K-Nearest Neighbors (KNN), para lograr una alta precisión en la clasificación. Tras el entrenamiento, se seleccionó una RNA con una exactitud del 93%. Una vez entrenado el modelo, se integra en Unity, donde se crea una interfaz interactiva que proporciona retroalimentación visual al usuario, permitiéndole realizar los gestos de forma efectiva y monitorear su progreso durante la rehabilitación.

Palabras clave: Reconocimiento de gestos, Leap Motion Controller, rehabilitación, Machine Learning, sistema inteligente.

ABSTRACT

Rehabilitation monitoring has become a prominent advancement in the technological field, thanks to the convergence of different technologies. The integration of these technologies allows for more accurate and detailed monitoring of hand rehabilitation gestures. Through data capture and analysis real-time information about the rehabilitation progress is provided, enabling healthcare professionals and users to assess their performance during the process. Moreover, the system not only offers visual tracking of daily progress but also motivates users to continue their treatment, contributing to the success of their rehabilitation.

In this research project, various technologies are implemented to facilitate the monitoring and analysis of hand rehabilitation data. Firstly, the Leap Motion Controller is used to capture hand position and direction data during the performed gestures. These data are processed using a custom acquisition system developed in Matlab, enabling their analysis, and generating a dataset. The dataset is preprocessed and relevant features are extracted for the classification of each gesture. For this, different models are implemented, such as Artificial Neural Networks (ANN) and K-Nearest Neighbors (KNN) to obtain a high accuracy in the classification. After training, ANN with 93% accuracy was selected. Once the model is trained, it is integrated into Unity, where an interactive interface is created to provide visual feedback to the user, allowing them to perform the gestures effectively and monitor their progress during rehabilitation.

Keywords: Hand gesture, Leap Motion Controller, rehabilitation, Machine Learning, intelligent system.

CAPÍTULO I.- MARCO TEÓRICO

1.1. Tema de Investigación

SISTEMA INTELIGENTE USANDO VIDEO CÁMARAS Y MOTION CAPTURE PARA MONITOREO DE LA REHABILITACIÓN DE LA MANO DERECHA.

1.1.1. Planteamiento del Problema

En las últimas décadas la tecnología avanza aceleradamente en el campo de la rehabilitación. La innovación tecnológica permite una mayor precisión en el diagnóstico y tratamiento de lesiones y discapacidades. La tecnología ha permitido el desarrollo de nuevos dispositivos y terapias, como la terapia por medio de la realidad virtual, la neuroestimulación y los exoesqueletos. Estas terapias ayudan a mejorar la movilidad y la función en pacientes con diferentes lesiones. Sin embargo, para seguir avanzando en el campo de la rehabilitación virtual, es necesario obtener una gran cantidad de datos. La combinación del mundo físico y virtual es esencial para recopilar y analizar estos datos de manera efectiva. La tecnología de realidad virtual y aumentada, por ejemplo, permite a los pacientes practicar habilidades motoras en un entorno controlado, mientras que el uso de dispositivos conectados en tiempo real permite recopilar datos de pacientes en cualquier lugar y en cualquier momento [1].

Un modelo para tomar en cuenta es el desarrollo de un sistema de rehabilitación para personas con deficiencia del control motor de miembro superior en un entorno de realidad aumentada. Este sistema permite que los pacientes realicen ejercicios de forma lúdica y motivadora, lo que mejora su participación en el tratamiento y aumenta la eficacia de la rehabilitación. Además, al estar en un entorno de realidad aumentada, los pacientes pueden sentir que están en un ambiente distinto al de una clínica u hospital, lo que contribuye a su bienestar emocional. La realidad aumentada también permite una mayor precisión en el seguimiento y evaluación del progreso del

paciente[1].

De igual manera, una de las tendencias tecnológicas más relevantes actualmente es el uso de Señales Electromiográficas (EMG) para generar sistemas de control y monitorización. Uno de los sistemas generados por EMG es el sistema de estimación de posición de la muñeca. Este método utiliza electrodos para medir las señales electromiográficas generadas por los músculos de la muñeca, lo que permite una medición precisa y no invasiva de la posición de la muñeca. Además, esta técnica se está utilizando cada vez más en aplicaciones de realidad virtual y robótica, ya que permite controlar dispositivos y robots con mayor precisión [2].

Algo semejante ocurre con el estudio realizado por Shull [3] que consiste en la adquisición de señales de gestos manuales mediante un arreglo de sensores barométricos. Este sistema utiliza sensores de presión barométrica colocados alrededor de la muñeca, los cuales son utilizados para captar los gestos manuales mediante el cambio de presión atmosférica generado por el movimiento. Esta técnica tiene la ventaja de poseer una alta precisión del 94% en la clasificación de 10 gestos distintos, lo que la hace atractiva para su uso en aplicaciones como el control de dispositivos, la robótica y el juego [4].

En la región de las Américas. Los dispositivos médicos, como los sensores de monitoreo de salud, las aplicaciones móviles de salud y las tecnologías de diagnóstico, han desempeñado un papel importante en mejorar la detección, prevención, diagnóstico, tratamiento y rehabilitación de enfermedades [5]. Estas tecnologías son fundamentales en el diseño de planes y estrategias tanto en el campo clínico como en la ingeniería biomédica con el objetivo de mejorar la salud y bienestar de la población [6].

Con respecto al avance tecnológico se puede observar que en Perú se realiza el seguimiento del paciente en rehabilitación, garantizando que el paciente esté

recibiendo el tratamiento adecuado y que esté progresando en su rehabilitación. La tecnología utilizada para mejorar el seguimiento del paciente es a través del uso de dispositivos como el Kinect 2.0. El Kinect 2.0 es un dispositivo de seguimiento de movimiento que utiliza tecnología de profundidad para rastrear los movimientos del cuerpo humano. Se ha utilizado en la rehabilitación para medir el rango de movimiento, la fuerza y la coordinación, así como para supervisar la ejecución de ejercicios y detectar problemas de movimiento [7].

Por otro lado, en Ecuador se han realizado estudios que permiten el reconocimiento de gestos de la mano mediante técnicas como Leap Motion Controller, Infrared Information y Deep Learning Framework. Logrando así clasificar los sensores no invasivos en dos categorías: sensores portátiles y sensores sin contacto. Los sensores portátiles, como Myo Armband y Smart Gloves, utilizan sensores inerciales como: acelerómetros, magnetómetros y giroscopios, mejorando la interacción, pero presentan limitaciones en términos de sensibilidad, ruido de señales, calibración y comodidad. Los sensores sin contacto, como Microsoft Kinect, Cámaras Intel RealSense y Leap Motion Controller (LMC), ofrecen mayor seguridad y comodidad para el usuario, pero también presentan problemas de sensibilidad a las condiciones de iluminación y oclusión. Estos sensores proporcionan datos de posición espacial e imágenes [8].

Se puede tomar como ejemplo la implementación de la cámara Intel RealSense SR 300 y el software Project Gesture para identificar gestos de la mano. La cámara Intel RealSense SR 300 es un dispositivo de captura de movimiento que utiliza tecnología de profundidad para detectar y rastrear los movimientos de las manos y los dedos. Project Gesture es un software desarrollado por Intel que utiliza esta tecnología para reconocer y analizar los gestos de las manos. Juntos, la cámara Intel RealSense SR 300 y Project Gesture pueden ser utilizados para desarrollar aplicaciones de control de gestos, como la navegación en dispositivos móviles, el control de juegos y la presentación de diapositivas. Además, esta combinación de hardware y software también puede ser utilizada en campos como la robótica, la automatización industrial y la asistencia para personas con discapacidades [9].

En este contexto, el no contar con un adecuado sistema que dé seguimiento a la rehabilitación que realiza el paciente puede traer varias consecuencias, una de ellas puede ser que el paciente no realice correctamente el ejercicio propuesto. Sin un sistema de monitoreo que pueda verificar la precisión y la calidad del gesto, es difícil para el paciente saber si se está realizando el ejercicio correctamente. Sin datos precisos y actualizados, es difícil para el paciente evaluar el progreso que obtiene.

1.2. Antecedentes Investigativos

En el presente trabajo se llevó a cabo una investigación de literatura de artículos publicados en diversas bases de datos científicas: Scopus, IEEE Explore, ACM Digital Librery, Willey, Science Direct y Springer. Se determinaron cadenas de búsqueda utilizando las siguientes palabras claves: Hand Gesture Recognition, Machine Learning, Leap Motion, Motion Capture, Rehabilitation, System y Virtual Reality. Se limitó a artículos publicados entre 2019 y 2023. En la **Tabla 1**, se exhiben las cadenas de búsqueda utilizada y el número de artículos encontrados.

Tabla 1. Cadenas de Búsqueda

	Revista	Cadena	Nº Artículos
Cadena de Búsqueda	Scopus	TITLE-ABS-KEY ("hand gesture recognition" AND "machine learning" AND ("leap motion" OR "motion capture") AND "rehabilitation" AND "system" AND "virtual reality") AND (LIMIT-TO (PUBYEAR , 2019) OR LIMIT-TO (PUBYEAR , 2023))	2
	IEEE Explore	("All Metadata":"hand gesture recognition") AND ("All Metadata":"machine learning") AND ("All Metadata":"leap motion") OR ("All	26

		Metadata:"motion capture") AND ("All Metadata":rehabilitation) AND ("All Metadata":system") AND ("All Metadata":virtual reality") Fliters Applied: 2019 - 2023	
	ACM Digital Library	[All: "hand gesture recognition"] AND [All: "machine learning"] AND [[All: "leap motion"] OR [All: "motion capture"]] AND [All: "rehabilitation"] AND [All: "system"] AND [All: "virtual reality"] AND [E-Publication Date: (01/01/2019 TO 12/31/2023)]	24
	Wiley	"hand gesture recognition" AND "machine learning" AND ("leap motion" OR "motion capture") AND "rehabilitation" AND "system" AND "virtual reality" Applied Filters: 2019-2023	3
	Science Direct	"hand gesture recognition" and "machine learning" and ("leap motion" or "motion capture") and "rehabilitation" and "system" and "virtual reality" Refine by 2019 – 2023	3
	Springer	"hand gesture recognition" and "machine learning" and ("leap motion" or "motion capture") and "rehabilitation" and "system" and "virtual reality" within 2019 - 2023	31
TOTAL			89

Elaborado por: Miguel Bonilla

La finalidad de la revisión de literatura es explorar el uso de sistemas inteligentes que integren tecnología de Motion Capture y sensores infrarrojos. En este estudio, se analizaron las diversas técnicas y herramientas empleadas en sistemas creados anteriormente. Dichos sistemas son utilizados para seguimiento de la recuperación de la mano derecha. Los resultados obtenidos identifican las prácticas más efectivas en el diseño y desarrollo de sistemas.

En este contexto, la revisión de literatura obtiene un conjunto de 89 artículos científicos, los cuales corresponden con las palabras claves utilizadas en la cadena de búsqueda. En la **Tabla 2.** se plantean los criterios de inclusión y exclusión empleados para seleccionar los artículos relevantes para la investigación.

Tabla 2. Criterios de inclusión y exclusión

Inclusión	<ul style="list-style-type: none"> • Artículos que presentan sistemas inteligentes enfocados a la rehabilitación. • Artículos que usan modelos de machine learning para el reconocimiento de gestos de la mano. • Artículos vinculados a tecnologías de rehabilitación de las manos. • Artículos que describen el proceso de captura de datos para reconocimiento de gestos de la mano.
Exclusión	<ul style="list-style-type: none"> • Se tomarán en cuenta solo artículos científicos, no libros. • Artículos que no proporcionan ejemplos de sistemas informáticos. • Artículos que no especifiquen el procedimiento utilizado. • Artículos que eviten el uso de algoritmos de machine learning.

Elaborado por: Miguel Bonilla

Se descartaron un total de 54 artículos, que no cumplen totalmente con los criterios de inclusión. Finalmente se realizó la lectura exhaustiva de 36 artículos que cumplen con

varios de los criterios propuestos. De los 36 artículos, se referencio 6 artículos que cumplen con todos los parámetros de inclusión. Mediante esta selección se logró obtener los artículos más relevantes para la investigación.

Según Ameer Safa [10], se describe un sistema de captura de gestos de la mano mediante la manipulación de imágenes DICOM (Digital Imaging and Communications in Medicine). El sistema emplea el dispositivo LMC para la adquisición de datos sin contacto y contiene dos partes principales: un software orientado a identificar los gestos de la mano y un módulo gráfico para interactuar con los datos de manera visual. El sistema utiliza el clasificador Support Machine Vector (SVM), el cual entrena y evalúa el dataset público LeapGestureDB. El conjunto de datos consta de 11 gestos que se repitió 5 veces por 120 sujetos diferentes, el total de muestras obtenidas es de 6600. Las DICOM contienen metadatos como la información del paciente, datos del estudio e información sobre la adquisición de la imagen. Entrenado el algoritmo SVM con el dataset se alcanzó una precisión de 91.73%. No obstante, se observó una tasa de error del 20% en la identificación de los gestos flexión y extensión en la mano derecha y flexión y extensión en la mano izquierda .

De acuerdo con Heike Brock [11], describe un sistema de reconocimiento multi-gestual mediante la interacción hombre-máquina. Se emplea el dispositivo LMC para el seguimiento de los movimientos de la mano en tiempo real. Para esto se toman características como la posición, orientación y velocidad al realizar los gestos de la mano. El proceso comienza identificando los gestos activos y seguidamente realiza una clasificación secuencial temprana de los movimientos de la mano. Con respecto a la adquisición de datos se tomaron en cuenta a 15 sujetos para que realicen 29 gestos, cada uno repetido en 25 ocasiones. Obteniendo un total de 10875 muestras. Para la clasificación de los gestos se utiliza un algoritmo Random Forest con bootstrapping. Se producen 200 árboles con una profundidad máxima de 100, con un mínimo de 2 muestras para una división y un mínimo de 2 muestras por hoja. El uso de Random Forest proporciona buenos resultados gracias a su robustez y capacidad de adaptación. Además, este algoritmo obtuvo una precisión del 92.2 %, confirmando la eficacia que

posee en tareas de aprendizaje similares. El alto porcentaje que presenta el sistema permite su uso en la rehabilitación de la mano o en terapias de movilidad de la mano.

Basado en la investigación realizada por Hussein Walugembe [12], se detalla un sistema de rehabilitación de mano o dedos, utilizando un sensor LMC para la captura de datos y para la clasificación propone el uso de dos algoritmos conocidos, como lo son Linear Discriminant Analysis (LDA) y SVM. El estudio detalla la captura de datos mediante LMC, el cual registra los movimientos realizados por la mano y los dedos en fotogramas, para posteriormente extraer vectores de dirección de los huesos de los dedos y calcular los ángulos de las articulaciones. Se utilizó una mano de artista para simular pacientes que sufren lesiones cerebrovasculares y recopilar la información. Se obtuvo un total de 600 muestras de cuatro gestos, 150 muestras por gesto. Los parámetros que se consideraron para la obtención de datos fueron: elevación, movimientos laterales, posicionamiento hacia adelante y hacia atrás y rotación. A partir del dataset generado, se utilizó el 62.5 % de las muestras para entrenamiento y el 37.5 % para pruebas. Además, el modelo LDA que se empleó es un multiclase, ya que hay más de dos gestos en el dataset. En este sentido, LDA empieza localizando las dispersiones intraclase e interclase de k gestos independientes en el dataset. Para representar dichos parámetros se utiliza la matriz de dispersión S , la cual sirve para calcular los coeficientes de discriminación lineal, que se utiliza para reflejar los datos de entrada en un espacio de características de menor dimensión. Mediante esto el modelo LDA obtuvo una precisión de 88.44 % en la clasificación de gestos ejecutados por el paciente. Por otro lado, el modelo SVM se utiliza comúnmente para resolver el problema de separación no lineal que se puede dar entre las categorías o clases de un dataset. Por lo tanto, SVM se emplea para reconocer y clasificar observaciones que constan de solo dos clases. Para lograr un reconocimiento multiclase se deben de combinar muchos clasificadores. Con el fin de obtener el mejor rendimiento posible se decidió utilizar cuatro tipos de modelos SVM: lineal, polinómico, radial y sigmoidal. Cada modelo cuenta con diferentes parámetros y características que pueden afectar en la clasificación, por lo cual se testeó cada modelo para definir el más óptimo al momento de reconocer los gestos. Los resultados obtenidos se visualizan en la **Tabla 3**.

Tabla 3. Resultados Algoritmo SVM

Algoritmo	Precisión
SVM Lineal	89.17 %
SVM Polinómico	88.33 %
SVM Radial	87.58 %
SVM Sigmoidal	87.58 %

Realizado por: Miguel Bonilla

Conforme a Diego G. Alonso [13], se describe un enfoque para el reconocimiento de gestos de la mano utilizando LMC mediante una combinación de factores como: la forma de la mano, la dirección del movimiento, la distancia recorrida y la curvatura, estas características mejoran la precisión del reconocimiento de gestos. El uso del sistema de reconocimiento de gestos de la mano basado en LMC se utiliza ampliamente en muchas aplicaciones como los juegos, realidad virtual, la rehabilitación física, la comunión y la inclusión. Para aplicar el sistema en dichos campos se debe de realizar una clasificación de los gestos. Para esto se utiliza una arquitectura basada en la red neuronal convolucional (CNN) y de memoria a largo plazo (LSTM), SVM y Dynamic Time Warping (DTW). Para DTW se toman en cuenta varios parámetros para mejorar el rendimiento. Los parámetros mencionados son: ventana de restricción, función de distancia, peso de los puntos y normalización, lo cual permite mejorar la precisión y reducir el tiempo de cálculo. Todo esto mediante un esquema de validación cruzada de 10 k folds. El dataset obtenido contenía un total de 600 muestras, se tuvo 20 gestos y se realizó una repetición de 30 muestras por cada gesto. Cabe aclarar que 10 fueron gestos estáticos y 10 dinámicos. Ambos tipos de gestos consistían en una secuencia de fotogramas para emular un escenario de reconocimiento de gestos en tiempo real. Para poder simular dicho ambiente el dispositivo LMC lee los datos del sensor y los transmite al software de seguimiento. Este sistema combina los datos con un modelo de una mano humana y los ensambla en forma de fotogramas, para posteriormente realizar clustering y coincidencia de cadenas. Finalizado el proceso se aplican los algoritmos de clasificación, que presentan el resultado obtenido de cada uno de los modelos validados. La precisión de los modelos se encuentra en la **Tabla 4.**

Tabla 4. Resultados de los modelos aplicados en el sistema de reconocimiento de gestos utilizando LMC

Modelos	Precisión
CNN + LSMT	98.83 %
DTW	87.67 %
SVM	83.33 %

Elaborado por: Miguel Bonilla

Según Miri Weiss Cohen [14], se expone un sistema basado en la captura de movimiento mediante el dispositivo LMC. Se utiliza K-Nearest Neighbor (K-NN) para la correspondiente clasificación y regresión, mediante la cual se construye una caracterización del ejercicio para facilitar la retroalimentación sobre el desempeño del paciente en tiempo real. El sistema utiliza diversos juegos serios para fomentar al paciente a continuar realizando más repeticiones del ejercicio. El proceso inicia en el domicilio del paciente que debe de realizar la rehabilitación y un guía que proporciona el movimiento correcto a realizar. Se realiza la adquisición de los movimientos con el controlador LMC, el guía debe repetir los movimientos alrededor de 20 a 30 veces. El proceso de captura de datos tarda entre 5 y 10 minutos. Seguido el algoritmo K-NN calcula la distancia y asigna una etiqueta al punto de prueba, basado en las etiquetas de los puntos de entrenamiento. El sistema presenta tres casos de estudio propuestos, el primer caso se realiza con una mano sana para verificar la calidad de los resultados y representación de características. El segundo caso se ejecuta con una mano que posee limitación en el dedo pulgar, mostrando que el sistema detecta la limitación que posee el paciente. El último caso se realiza con una mano que no puede agarrar ni cerrar el puño, demostrando la retroalimentación en tiempo real proporcionada por el sistema. Los resultados obtenidos de los casos de estudio se reflejan en la **Tabla 5**.

Tabla 5. Resultados de los casos de estudios

Caso	Dedo Pulgar	Dedo Índice	Dedo Mediano	Dedo Anular	Dedo Meñique	Total
Caso 1	89 %	94 %	96 %	92 %	88 %	92 %
Caso 2	46 %	95 %	96 %	89 %	85 %	83 %
Caso 3	48 %	41 %	46 %	46 %	40 %	44 %

Elaborado por: Miguel Bonilla

Según Rubén Nogales [15], se realizó un sistema con LMC que obtiene posiciones y direcciones espaciales de la mano. El LMC es el encargado de rastrear y recuperar estos datos según el eje de coordenadas, cuyo origen viene a ser el centro del dispositivo. A continuación, se extraen las características de los gestos de la mano, los cuales se obtienen mediante el uso de funciones estadísticas. Las funciones utilizadas para la extracción son: tasa de porcentaje de pulso (MYOP), detector logarítmico (LD), longitud de onda (WL), longitud de onda mejorada (EWL), valor de desviación estándar absoluta de diferencia (DASDV) y desviación estándar (SD). Se utilizó el método de selección de características para la extracción de las mismas. Para la extracción de características y entrenamiento del algoritmo se estableció 5 gestos: mano abierta, puño, movimiento de mano hacia adentro, movimiento de mano hacia afuera y pellizco. Para la realización de los gestos se obtuvo la ayuda de 56 voluntarios, entre los que incluyen hombres y mujeres entre 18 a 46 años. Cada voluntario repitió 30 veces cada gesto, por lo cual se obtuvo un total de 8400 observaciones. Para evaluar el dataset se utilizaron tres clasificadores: red neuronal artificial, SVM y una combinación de BiLSTM y ANN (BiLSTM-ANN). La red neuronal obtuvo una precisión de 92.93 % mientras que SVM logró una exactitud de 91.37 %. En cambio, con BiLSTM-ANN se obtuvo una precisión de 99.9912 %, haciendo que este sea el algoritmo elegido por su alta precisión. El sistema de reconocimiento de gestos de mano en tiempo real puede tener varios usos, como en la interacción humano-computadora, la robótica, la realidad virtual y aumentada, la rehabilitación física, entre otros.

Para facilitar el acceso a la información primordial en la **Tabla 6.** se realiza un resumen de la revisión de literatura científica.

Tabla 6. Resumen de la investigación de la literatura científica

Ref	Año	Captura	Descripción	Nº Gestos	Dataset	Algoritmo	Precisión
[10]	2020	Leap Motion Controller	Manipulación de imágenes DICOM.	11	Contiene 6.600 muestras.	Support Vector Machine	91.73%
[11]	2020	Leap Motion Controller	Se captura la posición, orientación y velocidad al realizar los gestos	29	Se integra de 600 muestras.	Random Forest	92.2%
[12]	2019	Leap Motion Controller	Se obtiene características de los ángulos de las articulaciones de los dedos.	4	Se compone de 600 muestras de 4 gestos.	Support Vector Machine	88.44 %
[13]	2020	Leap Motion Controller	Se recopilan datos de los movimientos de la mano y se guardan como secuencias de fotogramas.	20	Tiene 600 muestras.	Approximate String Matching, Support Vector Machine y Dynamic Time Warping	Especificado en la Tabla 4.

[14]	2020	Leap Motion Controller	Se capturan los movimientos de la mano y se obtienen vectores de ángulos entre cada punta de los dedos y vector de dirección de la palma.	2	Se realizan 20 a 30 repeticiones.	K-Nearest Neighbor en 3 casos de estudio.	Detallado en la Tabla 5 .
[15]	2023	Leap Motion Controller	Se capturan las posiciones espaciales, direcciones y velocidades de las manos y dedos.	5	30 repeticiones	BiLSTM-ANN	99.9912 %

Elaborado por: Miguel Bonilla

1.3. Fundamentación Teórica

1.3.1. Tecnologías de la Información

Las TIC incluyen sistemas computacionales que procesan, almacenan, sintetizan, comunican y muestran información en una diversa variedad de formas. Es una colección de herramientas, soportes y canales para tratar y conectar mediante canales nuevos, los que permiten crear, registrar, almacenar y distribuir contenidos de información. Varios ejemplos de estas tecnologías incluyen pizarras digitales, computadoras personales, proyectores multimedia, blogs, podcast y por supuesto, el internet [16].

1.3.2. Sistemas Informáticos

Los sistemas informáticos son conjuntos de hardware y software diseñados para recopilar, almacenar, procesar y transmitir información de manera eficiente y segura. Estos sistemas pueden ser utilizados en una variedad de entornos, incluyendo empresas, gobiernos, educación y hogares. El software es un componente esencial en cualquier sistema informático y es un conjunto de instrucciones que se ejecutan en el hardware. Estas instrucciones permiten al hardware realizar tareas específicas y controlar su funcionamiento. Además, el software también se encarga de manejar y coordinar el funcionamiento del sistema informático en su conjunto, proporcionando una interfaz para el usuario y realizando tareas de gestión en segundo plano. En si el software es la pieza clave que convierte en un sistema informático útil y funcional [17].

El software de un sistema realiza operaciones que utilizan los datos disponibles a los que tienen acceso los programas. Dichos programas deben de proporcionar instrucciones para datos inexactos y los pasos necesarios para poder obtener información de cada tipo de dato utilizado. Ejecutando el programa con un tipo de datos específico, se llevan a cabo las instrucciones correspondientes a cada uno de los datos en cuestión. Dando así paso al uso de una base de datos, operaciones elementales y estructuras de control [18].

1.3.3. Inteligencia Artificial

El término Inteligencia Artificial (IA) es aquella que trabaja en la capacidad de imitar las funciones inteligentes que posee el cerebro humano. La aplicación que posee la IA en el mundo de la tecnología es muy variada, principalmente es usado en el área de la informática y la robótica, así mismo se extiende a otras ramas, cómo las ciencias sociales, los negocios, medicina, realidad virtual y la economía. De igual manera, la creación de redes neuronales artificiales y algoritmos genéticos se está convirtiendo

en tecnologías cada vez más comunes de hallar, especialmente en investigación y pronósticos de diversa índole [19].

Se le puede definir a la IA como una rama de la informática que proporciona diferentes tipos de técnicas y herramientas para crear modelos de entornos y resolver problemas, imitando el comportamiento de los seres conscientes [20]. La IA también se le puede considerar cómo, la ciencia y tecnología que permite diseñar y programar ordenadores de forma que realicen tareas que requieran inteligencia. Otro concepto muy popular de la IA es la ciencia e ingeniería que posibilita imitar la inteligencia humana mediante máquinas [21].

1.3.4. Sistema Inteligente

Los sistemas expertos, también conocidos como Sistemas Inteligentes (Intelligent System - IS), esencialmente son software que realiza las tareas que generalmente realizaría un experto humano. Esto se logra mediante el uso de algoritmos y técnicas de inteligencia artificial, como el aprendizaje automático y el procesamiento del lenguaje natural. Los sistemas inteligentes pueden ser utilizados en una variedad de aplicaciones, como la robótica, la visión por computadora, el diagnóstico médico y el procesamiento del lenguaje natural. Los avances en esta área están permitiendo que estos sistemas se vuelven cada vez más sofisticados y útiles, y se espera que en el futuro se utilicen en una amplia gama de aplicaciones [22].

Cabe destacar que los sistemas inteligentes proporcionan un enfoque metodológico estandarizado para la resolución de problemas mediante el uso de algoritmos y técnicas de inteligencia artificial para ayudar en la toma de decisiones en la producción y la gestión de operaciones. Estos sistemas pueden ayudar a mejorar la eficiencia, reducir los costos y mejorar la calidad en una variedad de industrias [23].

Se puede señalar que la definición concreta de sistemas inteligentes se encuentra sujeta a un gran debate, ya que desde el punto de vista técnico se lo puede caracterizar por su flexibilidad, adaptabilidad, memoria, aprendizaje, dinámica temporal, razonamiento y la capacidad que posee para la gestión de información incierta e imprecisa [24]. Sin tomar en cuenta las diversas definiciones y enfoques que tienen los IS, se puede tener por hecho que la IA es una base esencial para la realización de estos [25].

1.3.5. Rehabilitación de Manos

La rehabilitación de las manos se refiere a un conjunto de técnicas y terapias utilizadas para mejorar o recuperar la función de las manos después de una lesión o enfermedad. Añadiendo una innovación en la rehabilitación de las manos como es el monitoreo y evaluación de los avances en la terapia se ofrece varias ventajas en comparación con los métodos tradicionales. El monitoreo a través de dispositivos tecnológicos, como sensores y sistemas informáticos, permite una medición objetiva y precisa de los movimientos de la mano y los dedos, así como también un seguimiento continuo del progreso [26].

Se puede seguir el caso de un paciente con fusión triquetra-lunar que realizó terapia ocupacional. Tras recibir un tratamiento específico, se llevó a cabo una supervisión durante un periodo de 12 meses, durante el cual, el paciente dejó de experimentar dolor y golpes. Finalizando la terapia se pudo obtener datos, dando una fuerza de agarre de 65 libras en la mano derecha; en la mano izquierda 75 libras, pellizco bilateral 19 libras, flexión de la muñeca derecha 40° y extensión en 75° . Dando muy buenos resultados el tratamiento realizado [27].

1.3.6. Terapia Ocupacional de Manos

La terapia ocupacional de la mano se refiere a la evaluación y tratamiento de problemas relacionados con la función de la mano y el brazo. Puede ser utilizada de varias formas

e incluir ejercicios específicos, terapia con aparatos y dispositivos, y la enseñanza de técnicas alternativas para realizar tareas. Como es el caso de un juego para realizar rehabilitación en niños. La cual, mediante la inclusión y adaptación de herramientas y aparatos para hacer las tareas más fáciles de realizar, permite a los niños mantenerse entretenidos al momento de realizar la terapia. La terapia ocupacional de la mano también puede incluir la evaluación y adaptación del entorno del paciente para asegurar que sea seguro y accesible. Para adaptar el entorno los médicos usan dispositivos como: la mesa de Kanavel, el dispositivo Amadeo de Tyromotion, el dispositivo Diego de Tyromotion, Hand Mentor, Gloreha Glove, consola Nintendo Wii, dispositivos de rehabilitación Kinect de Microsoft [28].

La terapia ocupacional en la mano se enfoca en ayudar a las personas a recuperar la capacidad para realizar actividades funcionales de la vida diaria, como escribir, cocinar, peinar, vestirse y realizar tareas de trabajo o hogar. Los terapeutas ocupacionales trabajan con el paciente para identificar y priorizar las actividades significativas, y luego utilizan esas actividades para guiar el tratamiento y la planificación de metas. La terapia ocupacional se basa en la premisa de que al mejorar la capacidad del paciente para realizar las actividades que son importantes para él o ella, se mejorará su calidad de vida en general [29].

1.3.7. Ingeniería Biomédica Aplicada a la Rehabilitación

La ingeniería biomédica es un campo interdisciplinario que combina la tecnología, la biología y la medicina para crear soluciones innovadoras para problemas médicos. La complejidad de movimientos que posee las articulaciones y la falta de datos precisos y detallados puede dificultar la toma de decisiones y limitar la capacidad en la biomedicina para desarrollar soluciones efectivas. Con los recientes avances tecnológicos se puede realizar simulación de técnicas para mejorar la comprensión de los movimientos y así identificar las articulaciones afectadas para evaluar y tratar la disfunción musculoesquelética [30].

El tratamiento que es más común al momento de intervenir se debe de clasificar en base a variables clínicas que se encuentran al momento de diagnosticar la enfermedad, el tratamiento más común y recomendado es el de la rehabilitación física, claro que todo esto depende de los resultados obtenidos [31]. Lo que se busca es que la ingeniería de rehabilitación utilice métodos de evaluación de programas por defecto, basándose en modelos de control [32].

1.3.8. Biomedicina

La denominación normal de biomedicina se remonta a cuando se destacaba el hecho de que la medicina es totalmente biológica, con la diferencia que está provenía de occidente. El término biomedicina fue otorgada por Gaines y Hahn en 1985, denominando así a la medicina científica. Se puede ver que, en la biomedicina, el aprendizaje automático ha tenido un impacto significativo, especialmente en áreas como la detección de enfermedades, la personalización de tratamientos y la optimización de procesos clínicos. Por ejemplo, se pueden utilizar algoritmos de aprendizaje automático para analizar grandes conjuntos de datos médicos y detectar patrones y tendencias que puedan ser útiles para el diagnóstico y tratamiento de enfermedades. Además, el aprendizaje automático también se utiliza para desarrollar sistemas expertos en biomedicina que pueden ayudar a los médicos a tomar decisiones clínicas informadas y a personalizar tratamientos para cada paciente [33].

El enfoque actual que se le da a la biomedicina es el de cómo se puede lograr avances decisivos en campos como: el diagnóstico clínico, tratamiento de precisión y la gestión de la salud y seguimiento de esta. Lo cual permite describir nuevas oportunidades de aprendizaje y así preparar el enfoque actual de la vanguardia. Esto lleva a la creación de aplicaciones de prestigio que permitan identificar desafíos que se puedan resolver con pleno aprendizaje automático. [34]

1.4. Objetivos

1.4.1. Objetivo General

Diseñar un sistema inteligente que permita la visualización del avance en la rehabilitación de la mano derecha empleando video cámaras y Motion Capture para pacientes de centros de rehabilitación autorizados.

1.4.2. Objetivos Específicos

- Identificar procesos que permitan la rehabilitación de la mano derecha.
- Desarrollar una aplicación de reconocimiento de gestos de la mano orientados a la rehabilitación de la misma empleando video cámaras y Motion Capture.
- Generar un modelo de Machine Learning para la verificación del avance de la rehabilitación en los pacientes.
- Diseñar un sistema inteligente con una interfaz interactiva que facilite la ejecución de las actividades y monitoreo del gesto realizado por el usuario.

CAPÍTULO II – METODOLOGÍA

2.1. Materiales

Para el presente proyecto se realizó una entrevista a fisioterapeutas, identificando procesos y fases necesarias para el monitoreo de la rehabilitación en la mano derecha. En conjunto se aplicó una encuesta a los pacientes usuarios del sistema.

Materiales utilizados para el desarrollo del proyecto

- Laptop.
- Internet.
- Hojas.
- Sistema de captura de datos.
- Sistema de gestión de base de datos.
- Video Cámaras.

Entrevista realizada a fisioterapeutas

La entrevista tiene como objetivo el recolectar información acerca del proceso a seguir en la rehabilitación de la mano derecha y entender como tener un acercamiento al paciente para conocer el proceso que debe de seguir para un correcto uso del sistema de monitoreo. La encuesta fue validada con el software Atlas Ti mediante un análisis exploratorio y codificación de la misma. Para conocer más sobre las preguntas realizadas revisar el **Anexo A**.

Encuesta realizada a pacientes con problemas de movilidad en la mano derecha

La finalidad de la encuesta es obtener información sobre el grado de conocimiento que poseen los pacientes acerca del uso de sistemas informáticos de monitoreo para la rehabilitación de la mano derecha y el tratamiento a seguir para el control que se va a brindar a los pacientes. La encuesta realizada se refleja en el **Anexo B**.

La encuesta fue aplicada a usuarios y posibles usuarios del sistema de monitoreo de rehabilitación de la mano derecha. Los usuarios pueden ser trabajadores en el área de salud, personal que maneja mucho tiempo el computador y aquellos que poseen una patología clínica comprobada. La confiabilidad de la encuesta fue validada mediante el coeficiente Alpha de Cronbach y el coeficiente Kuder Richardson (**Anexo C**).

2.2. Métodos

2.2.1. Modalidad de la investigación

Las modalidades consideradas en la presente investigación son las siguientes:

Modalidad Bibliográfica – se obtiene información de libros, artículos, páginas web y tesis de grado para sustentar el proyecto con literatura científica.

Modalidad de Campo – al estudiar el problema en el lugar dónde se realiza la rehabilitación se puede entender el problema a tratar, obteniendo criterios para cumplir con los objetivos propuestos

2.2.2. Población y Muestra

Se posee una muestra finita, pero con población desconocida por lo que se calculará el número de muestras mínimas aplicando la fórmula:

$$n = \frac{Z^2 * p * q}{e^2}$$

Dónde:

n = número mínimo de muestras

Z = nivel de confianza

p = probabilidad de éxito o proporción esperada

q = probabilidad de fracaso

e = error de estimación máximo aceptado

Realizando el cálculo se tendría:

$$n = \frac{(1.65)^2 * 0.5 * (1-0.5)}{(0.15)^2} = \frac{2.7225 * 0.5 * 0.5}{0.0225} = \frac{0.680625}{0.0225} = 30.25 \approx 30$$

El valor de $Z=1.65$ es un valor crítico en la distribución normal estándar, y se utiliza para calcular el intervalo de confianza, con un nivel de confianza del 90%. La probabilidad $p=0.5$ se utiliza porque se desconoce la probabilidad de eficacia en este tipo de software. Asimismo, el valor de la probabilidad q se obtiene como $q=1-p$. Finalmente, se utiliza un error aceptable de $e=0.15$ según la estimación de una breve revisión de literatura, en la que se estima un promedio de precisión de los sistemas de reconocimiento de gestos de la mano entre 71.5% hasta 80.8%. Lo cual nos permite calcular una muestra proporcional de 30 repeticiones por cada gesto a realizar. En la Tabla 8, se refleja la población con la que se va a trabajar, indicando el número total de gestos y repeticiones que serán considerados para el estudio. Este enfoque de muestreo garantiza una evaluación exhaustiva de los sistemas de reconocimiento de gestos de la mano y permite obtener resultados estadísticamente significativos.

Tabla 7. Población de Estudio

Población	Número	Porcentaje
Fisioterapeutas	2	6.25 %

Repeticiones	30	93.75 %
Total	32	100 %

Elaborado por: Miguel Bonilla

La población en la cual se enfoca el proyecto investigativo no sobrepasa de los cien elementos, por lo cual no será necesario realizar una muestra representativa y se procede a trabajar con la totalidad de la población.

2.2.3. Recolección de la Información

Tras haberse aplicado la encuesta se obtuvo los siguientes resultados:

Resultado de la encuesta aplicada a pacientes y posibles pacientes del sistema

Ítem 1: ¿Ha participado alguna vez en un programa de rehabilitación que utilice movimientos virtuales para la recuperación de su mano derecha?

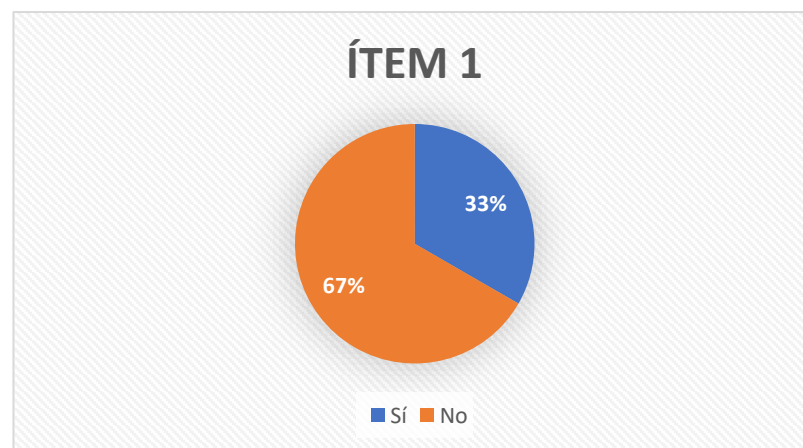


Figura 1. Uso de sistemas de rehabilitación virtual en la mano derecha

Elaborado por: Miguel Bonilla

Análisis e interpretación de resultados

En la **Figura 1**, se puede apreciar que:

- Un 67% no ha participado en programas de rehabilitación virtual de la mano derecha. Esto conlleva a que la mayoría no ha experimentado los beneficios que posee la rehabilitación virtual.
- Un 33% contó con la posibilidad de participar en programas virtuales de rehabilitación de la mano derecha. Estos sujetos han logrado probar las ventajas de esta innovadora tecnología.

Ítem 2: ¿Cuál es su nivel de familiaridad con el concepto de rehabilitación virtual en el contexto del movimiento de la mano derecha?

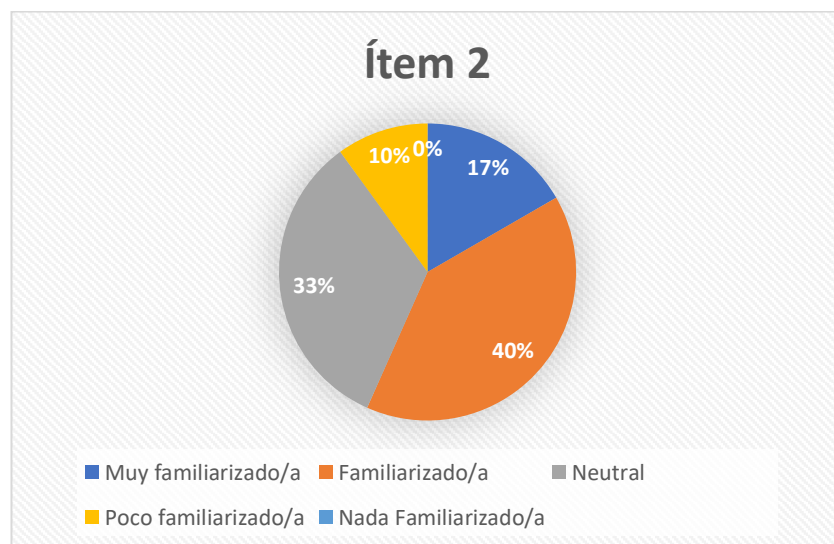


Figura 2. Familiaridad con la rehabilitación virtual

Elaborado por: Miguel Bonilla

Análisis e interpretación de los resultados

Según los resultados de la **Figura 2**, se logra interpretar que:

- El 17% de los encuestados cuentan con un nivel alto de conocimiento del concepto de rehabilitación virtual de la mano derecha. El 40% tiene una comprensión aceptable sobre el tema.
- El 33% no cuenta con la experiencia o el entendimiento necesario sobre la rehabilitación virtual de la mano derecha. Por último, un 10% desconoce las cualidades que posee esta tecnología.

Ítem 3: ¿Cuál es su nivel de conocimiento sobre el uso de sistemas informáticos?

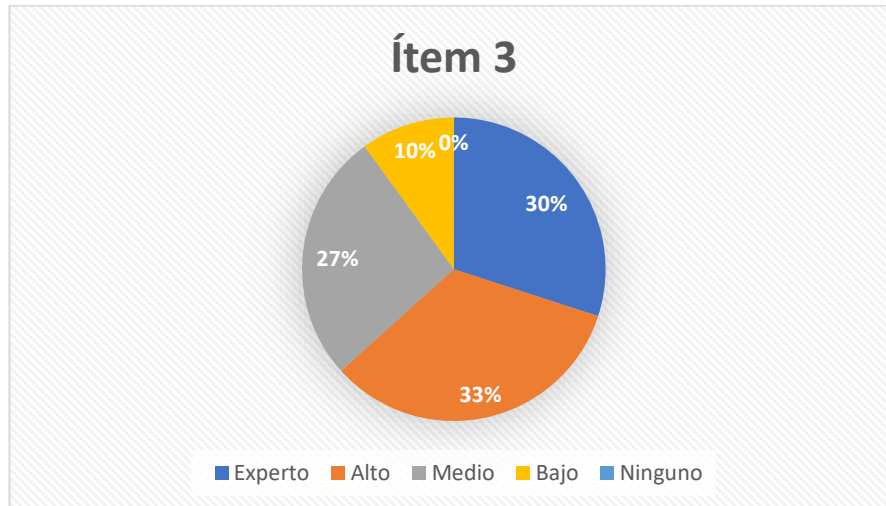


Figura 3. Uso de sistemas informáticos

Elaborado por: Miguel Bonilla

Análisis e interpretación de resultados

Basados en los datos de la **Figura 3**, se puede concluir que:

- Se identifica al 30% como usuarios expertos en el manejo de sistemas informáticos. Estos poseen amplio conocimiento y dominios en el manejo de sistemas. El 33% poseen un nivel alto en habilidades al manejar sistemas informáticos.
- El 27% posee un grado intermedio de conocimientos respecto al uso de sistemas informáticos. Finalmente, un 10% posee habilidades bajas en el uso de la tecnología.

Ítem 4: ¿Considera que un sistema de rehabilitación virtual que monitoree el movimiento realizado sería beneficioso para su proceso de rehabilitación de la mano derecha?

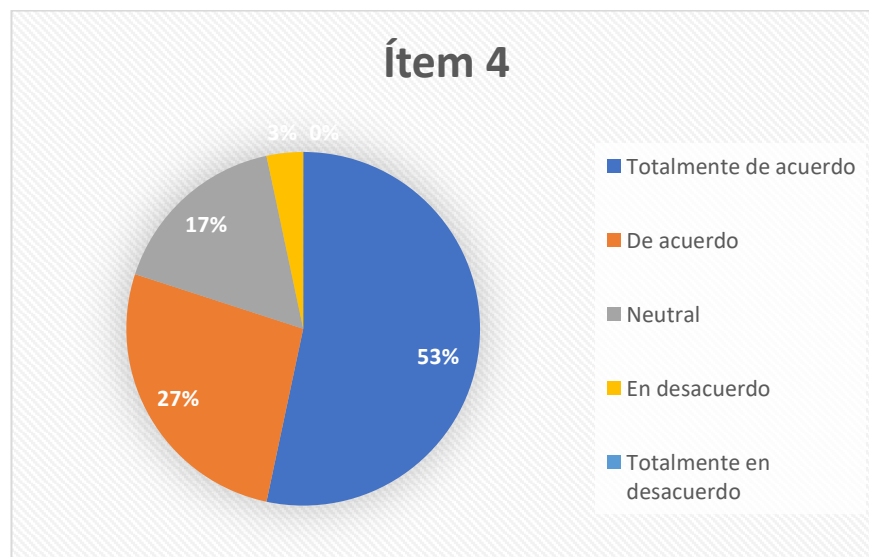


Figura 4. Beneficio de un sistema de rehabilitación virtual para la mano derecha

Elaborado por: Miguel Bonilla

Análisis e interpretación de resultados

A partir de los resultados obtenidos en la **Figura 4.** se obtiene que:

- Un notable índice del 53% está totalmente de acuerdo y un 27% está de acuerdo con el ítem 4. El uso de nuevas tecnologías en la rehabilitación es positivo para poder lograr un correcto monitoreo.
- Un 17% considera una posición neutral, lo que indica que no están ni a favor ni en contra de implementar el sistema. Un limitado 3% de los encuestados mostró desacuerdo al uso de tecnologías en el proceso de rehabilitación.

Ítem 5: ¿Qué nivel de confianza tendría en un sistema inteligente para monitorear su rehabilitación de la mano derecha?

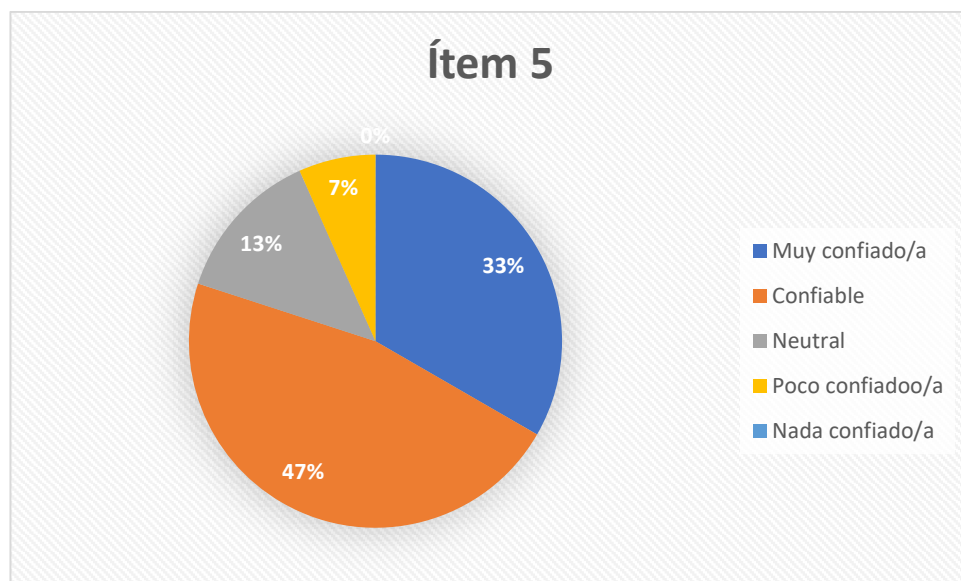


Figura 5. Confianza en el sistema de monitoreo de rehabilitación

Realizado por: Miguel Bonilla

Análisis e interpretación de resultados

Basado en el resultado obtenido en la **Figura 5.** se identifica que:

- Un 33% indicó que ve el sistema cómo muy confiable, mientras que el 47% percibe el sistema como confiable. Logrando mostrar opiniones positivas en la efectividad del sistema.

- Un 17% no muestra una opinión a favor o en contra del sistema y un 7% no confía en la efectividad del mismo.

Ítem 6: ¿Qué tan cómodo/a se sentiría utilizando un sistema de rehabilitación virtual que monitoree el movimiento realizado durante su proceso de rehabilitación? (Seleccionar una opción en una escala del 1 al 5), donde 1 es "Nada cómodo/a" y 5 es "Muy cómodo/a".

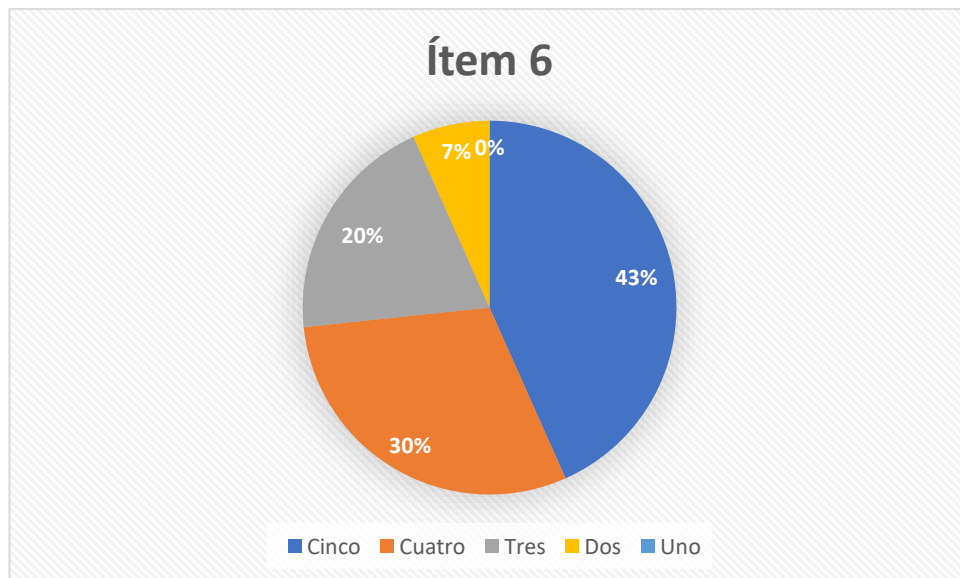


Figura 6. Comodidad al usar el sistema de rehabilitación

Realizado por: Miguel Bonilla

Análisis e interpretación de resultados

Según la información recopilada en la **Figura 6.** se concluye que:

- El 73% de los encuestados manifestó una comodidad al utilizar el sistema. Esto puede deberse a varios factores como la implementación de tecnología.

- El 20% se mantuvo al margen al dar una opinión al respecto y expreso neutralidad respecto a la comodidad. Finalmente, un 7% demostró incomodidad al tener que usar el sistema.

Ítem 7: ¿Qué beneficios espera obtener de un sistema de rehabilitación virtual que monitoree el movimiento realizado en su proceso de rehabilitación de la mano derecha? (Seleccionar todas las opciones que apliquen)

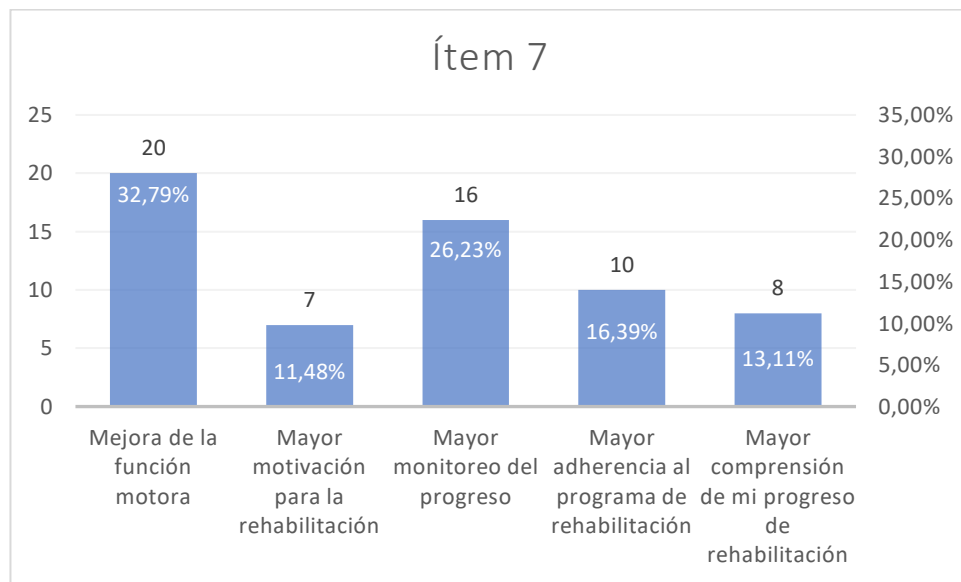


Figura 7. Beneficios de un sistema de monitoreo de la rehabilitación

Elaborado por: Miguel Bonilla

Análisis e interpretación de resultados

Conforme a la **Figura 7.** se recopiló la siguiente información:

- De todas las respuestas obtenidas se obtuvo que el 32.79% sostiene que el sistema ayuda en la mejora de la función motora. Logrando mejorar significativamente las actividades realizadas día a día. El 11.48% de las respuestas sustentan que se logra una mayor motivación para continuar con el

proceso de rehabilitación. Mediante esto se logra que el paciente culmine con éxito el proceso de rehabilitación.

- Mientras que el 26.23% de los resultados obtenidos mantiene que se alcanza un mayor monitoreo en el progreso.
- El 16.39% de la información recopilada señala que se consigue una mayor adherencia al programa de la rehabilitación. Finalmente, un 13.11% de las respuestas señala que se obtiene una mayor comprensión del progreso de la rehabilitación de la mano derecha.

Ítem 8: ¿Qué desventajas ve en el uso de un sistema de rehabilitación virtual que monitoree el movimiento realizado en su proceso de rehabilitación de la mano derecha? (Seleccionar todas las opciones que apliquen)

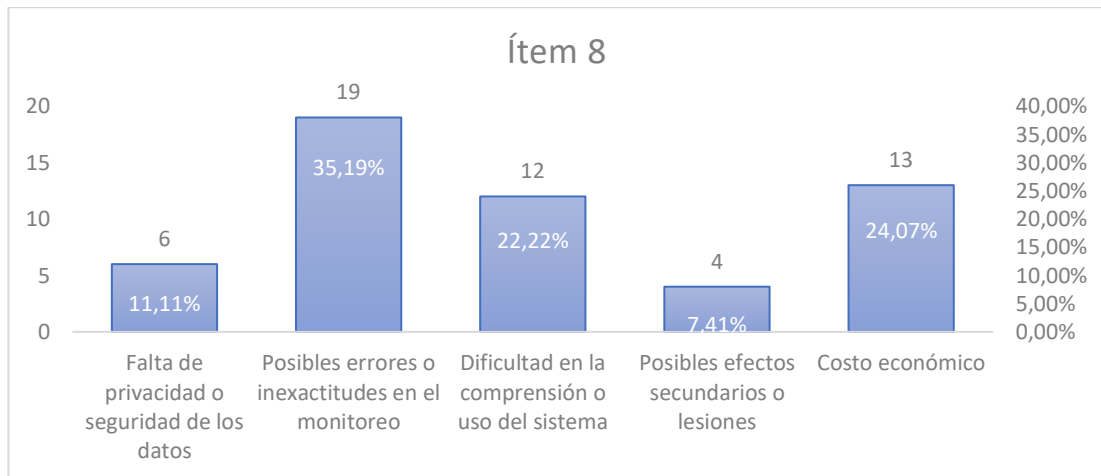


Figura 8. Desventajas de usar un sistema de rehabilitación virtual

Realizado por: Miguel Bonilla

Análisis e interpretación de resultados

Según los datos recopilados en la **Figura 8.** se obtiene:

- El 11.11% de las respuestas registrada incertidumbre con respecto al manejo de protocolos de seguridad. Esto plantea la duda sobre la protección de los datos obtenidos. Además, el 35.19% apunta a posibles fallos o inexactitud al momento de monitorear el movimiento de la mano derecha. Planteando así interrogantes en la fiabilidad del sistema.
- Otro 22.22% de las respuestas indica una dificultad en la comprensión o uso del sistema. Indicando la necesidad de innovar en el aspecto de usabilidad e interfaz del sistema. Por otro lado, el 7.41% expone inquietudes sobre posibles efectos secundarios o lesiones como secuelas del uso del sistema.
- Finalmente, el 24.07% señala angustias sobre el coste económico que posee el sistema. Para ello se evalúa la viabilidad financiera y el acceso necesario a los recursos económicos.

Ítem 9: ¿Qué características o funcionalidades considera importantes en un sistema de rehabilitación virtual que monitoree el movimiento realizado en su proceso de rehabilitación de la mano derecha? (Seleccionar todas las opciones que apliquen)



Figura 9. Características y funcionalidades del sistema

Realizado por: Miguel Bonilla

Análisis e interpretación de resultados

De acuerdo con los resultados obtenidos en la **Figura 9**, se consigue:

- El 15% de las respuestas se enfocan en que el sistema debe de poseer una interfaz de usuario amigable. Mediante esto se consigue facilitar el uso del sistema. Así se logra resolver las dudas que se tenía sobre la dificultad en la comprensión o uso del sistema especificado en el ítem 8.
- Un conjunto equivalente al 10% de las respuestas señala que se debe de tener precisión en el monitoreo de los movimientos. Se logra comprender que se debe de tener una gran precisión para que el sistema tenga una alta fidelidad. Esto va de mano con el 10% siguiente, el cual especifica que se debe de tener retroalimentación en tiempo real cuando se utiliza el sistema.
- El siguiente 10% de respuestas analizadas indica que es necesaria una personalización del sistema. Esto conlleva a que el usuario se sienta más cómodo utilizando el sistema. El 12.50% ve necesario un monitoreo del progreso a lo largo del tiempo. Así se obtiene un avance lineal en el proceso de rehabilitación de la mano derecha.
- Otro 10% refleja que es necesario integrar el sistema con otros métodos de rehabilitación. Obteniendo un mayor alcance en la rehabilitación al combinar diversas técnicas para la recuperación. El 9.17% de las respuestas refleja que el tener acceso a registros o reportes generados por el sistema es necesario. Así se logra tener conocimiento sobre el avance diario obtenido.
- Un porcentaje del 11.67% de las respuestas habla sobre el uso del sistema en el hogar. Si el sistema es fácil de usar se puede utilizar en diferentes entornos, en el hogar, en el centro de rehabilitación, en oficinas y otros lugares. Finalmente, otro 11.67% de las respuestas ve como indispensable el

asesoramiento o guía por parte de profesionales de la salud. El funcionamiento del sistema puede mejorar si un fisioterapeuta asesora al paciente al momento de usarlo. De esta manera se puede evitar posibles efectos secundarios o lesiones como se menciona en el ítem 8.

Ítem 10: ¿Qué desafíos potenciales identifica en el uso de un sistema de rehabilitación virtual que utilice el movimiento para la verificación de la rehabilitación de la mano derecha? (Seleccionar todas las opciones que considere válidas)

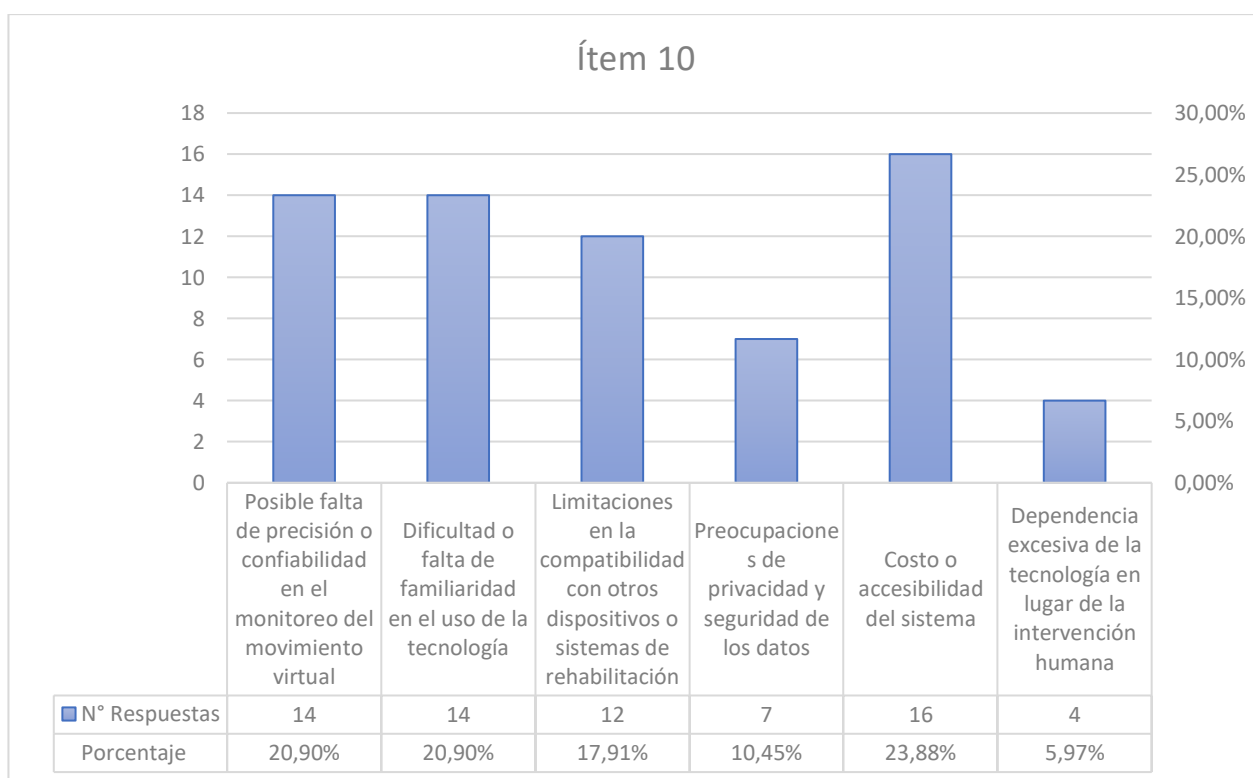


Figura 10. Desafíos potenciales en el uso del sistema

Realizado por: Miguel Bonilla

Análisis e interpretación de resultados

Según los datos recopilados en la **Figura 10.** se obtiene:

- Del total de respuestas, el 20.90% cree que puede existir una baja confiabilidad al monitorear el movimiento virtual. Mediante esto se tiene una percepción que indica una baja precisión en el monitoreo del movimiento virtual. El siguiente 20.90% destaca la dificultad o falta de familiaridad en el uso de la tecnología como un desafío potencial. Esto señala que los usuarios pueden tener dificultades al utilizar el sistema.
- El 17.91% sugiere un desafío en tener limitaciones en la compatibilidad con otros dispositivos o sistemas de rehabilitación. Esto implica que los usuarios pueden tener dificultades para integrarse con otros equipos o programas de rehabilitación. Un porcentaje del 10.45% habla sobre la privacidad y seguridad de los datos. Estos datos sobre el movimiento virtual son de carácter personal ya que es utilizado con fines médicos. Por lo cual se implementan estándares para resguardar la información.
- El 23.38% resalta el costo o accesibilidad al sistema. Este desafío indica que los usuarios encuentran problemas al acceder al sistema por su costo u obstáculos para hacer uso del sistema. Por último, un 5.97% tiene como desafío la dependencia excesiva de la tecnología en lugar de la intervención humana. Por ello en el ítem 9 se resalta el uso de profesionales para monitoreo del progreso que tiene el paciente en el uso del sistema.

Ítem 11: ¿Considera que el precio de recibir rehabilitación para la mano derecha en un consultorio privado es excesivo?

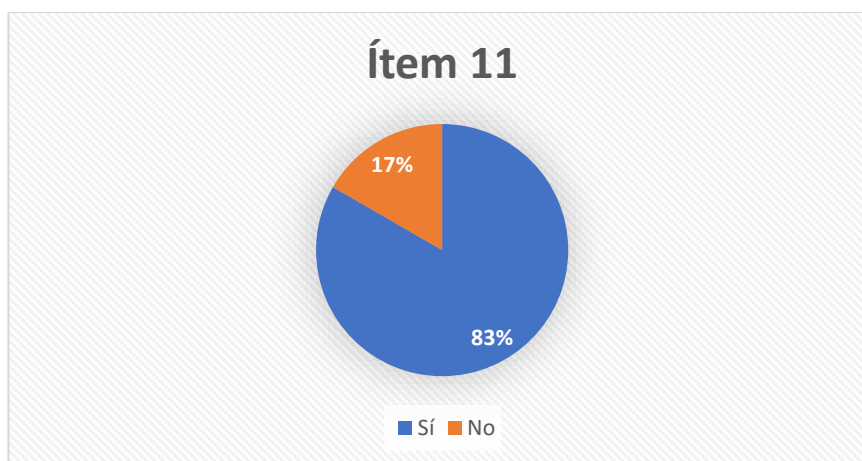


Figura 11. Costo de terapias en consultorios privados

Elaborado por: Miguel Bonilla

Análisis e interpretación de resultados

De acuerdo con los resultados obtenidos en la **Figura 11**, se obtiene que:

- El 83% concuerda con que el costo de realizar rehabilitación de la mano derecha en un centro privado es alto. Por ello muchos de los pacientes no pueden realizar el adecuado tratamiento de rehabilitación en la mano derecha.
- El 17% restante considera que el precio para acceder a rehabilitación de la mano derecha en un consultorio privado es razonable. Comparado con el 83% que no pueden costearlo el acceder a un servicio puede resultare excesivamente difícil.

Tras haberse aplicado la entrevista se obtuvo los siguientes resultados:

Resultado de la entrevista aplicada a fisioterapeutas

Análisis Exploratorio

El análisis exploratorio realizado permitió la identificación de las palabras más frecuentes en las entrevistas realizadas sobre la rehabilitación de la mano derecha. En el top 10 de palabras más utilizadas se encuentran ejercicios, movimiento, paciente, realizar, movilidad, dolor, proceso, tendinitis, patología ya plazo. Mediante esto se puede ver la importancia que tiene el ejercicio y la movilidad en el proceso de rehabilitación, así como la adaptación de los ejercicios al paciente y su patología específica. De igual manera, se resalta la importancia que debe de tener el paciente para cumplir con el proceso adecuadamente y la relevancia que tiene el fisioterapeuta en el transcurso de la rehabilitación.

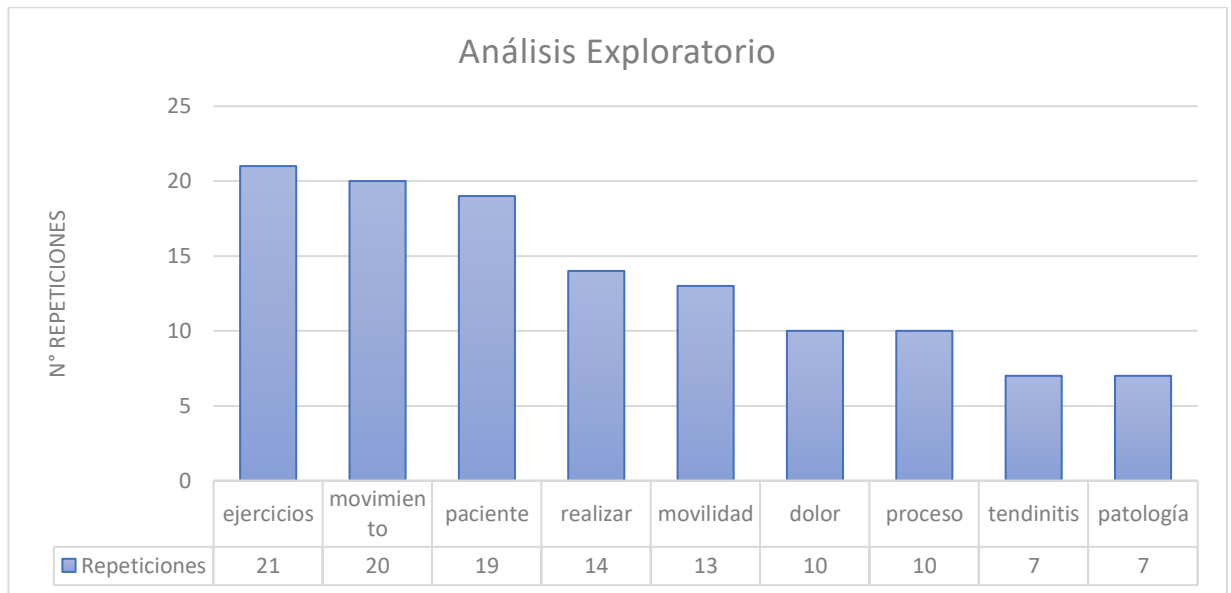


Figura 12. Análisis Exploratorio

Realizado por: Miguel Bonilla

Codificación

Se crean diferentes categorías relevantes basadas en el análisis exploratorio. En primer lugar, se creó una categoría para los ejercicios mencionados en la entrevista y la relación que está tiene con las otras categorías. Otros grupos encontrados fueron el de dolor, tendinitis, movilidad de la mano, proceso de rehabilitación y la patología a la que pertenece. De esta manera se logra la identificación de diferentes temas, conceptos o patrones en la información recopilada. Logrando facilitar la relación entre las categorías identificadas.

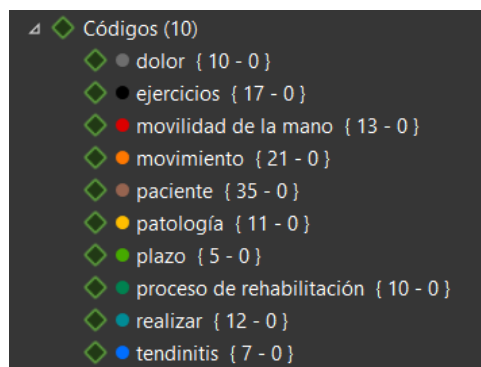


Figura 13. Códigos de codificación

Realizado por: Miguel Bonilla

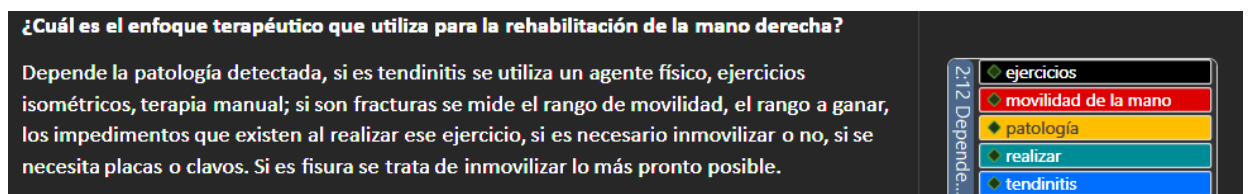


Figura 14. Codificación de la entrevista

Realizado por: Miguel Bonilla

CAPÍTULO III – RESULTADOS Y DISCUSIÓN

3.1. Análisis y discusión de los resultados

Realizado las encuestas a los usuarios del sistema se obtiene la factibilidad sobre el proyecto y su uso en el campo de la rehabilitación física aplicado a la mano derecha. Mediante las entrevistas se obtuvo información acerca del tratamiento a seguir y el proceso que conlleva la recuperación en la mano derecha. Seguido se elabora una tabla para identificar las fases necesarias en el programa de rehabilitación con el fin de alcanzar una recuperación satisfactoria. Después, se realiza la adquisición de datos en pacientes que presentan un diagnóstico que limite la movilidad en la mano derecha, con los datos recopilados se procede a utilizar algoritmos de machine learning para su clasificación. Finalmente, se investiga una metodología y se empieza el desarrollo del sistema de monitoreo de la rehabilitación.

3.1.1. Proceso de la rehabilitación de la mano derecha

Con la finalidad de recobrar la capacidad funcional y el movimiento en la mano derecha se lleva a cabo un proceso de rehabilitación. Con el fin de ejecutar este proceso es necesario seguir diversas etapas que permiten al paciente recuperar su fuerza, movilidad y destreza de la mano derecha.

Tabla 8. Etapas de la rehabilitación

Etapas	Procedimiento
1º Etapa	Revisión para diagnóstico del paciente.
2º Etapa	Ejecución de técnicas para controlar el dolor y la inflamación o enfocarse en la realización de ejercicios para mejorar la movilidad de la mano derecha.
3º Etapa	Mejora de la movilidad de la mano derecha mediante ejercicios.

Elaborado por: Miguel Bonilla

Etapa 1

Durante el proceso de revisión, se recopila una variedad de información con el objetivo de analizarla en profundidad. Esta información incluye los antecedentes médicos personales y familiares del paciente, los síntomas actuales que presenta, el diagnóstico de fisioterapia, la evaluación física, el nivel de dolor que experimenta, así como la evaluación de la salud ósea y muscular. Esta información es esencial para establecer un diagnóstico preciso y desarrollar un plan de tratamiento individualizado y efectivo, lo que puede mejorar significativamente la funcionalidad de la mano derecha. La funcionalidad está estrechamente ligada a las habilidades de una persona para realizar una acción o tarea. [35]

Al realizar la recolección de información se obtuvo un conocimiento empírico sobre las características utilizadas para realizar el diagnóstico. En la **Tabla 9**. se detalla lo que se debe de considerar para una buena revisión.

Tabla 9. Características utilizadas en la revisión patológica

Antecedentes patológicos	Enfermedad Actual	Diagnóstico de fisioterapia	Evaluación física	Nivel de dolor	Evaluación ósea	Evaluación muscular
Identificar factores de riesgo.	Limitación de movilidad.	Lesiones de ligamentos de las manos.	Evaluación de la amplitud de movimiento, fuerza muscular.	Ubica dolor en tejidos blandos mediante palpación.	Se considera normal si no se detectan deformidades, fracturas u otras anormalidades en la estructura ósea de la mano.	Se determina si la musculatura de la mano se encuentra hipotónica (debilidad muscular) o hipertónica (aumento de la tensión muscular).
Identificar enfermedades previas.	Debilidad muscular.	Lesiones musco-esqueléticas en la mano.	Valoración de la sensibilidad táctil, reflejos.	Identifica dolor al mover la mano o articulaciones en una dirección específica.	Se entiende como rigidez a la falta de movilidad en las articulaciones de la mano.	La musculatura de la mano no presenta hipotonía ni hipertonía si la fuerza y la resistencia de los músculos de la mano están dentro del rango normal.
Identificar predisposición genética.	Dolor en las articulaciones de la mano.	Historia familiar de enfermedades congénitas.	Examinación de la postura, equilibrio.	El dolor en área no afectada directamente es un dolor referido.	Se denomina como hipomovilidad a una movilidad reducida.	Se detecta si hay una disminución de la masa muscular, lo que se conoce como hipotrofia, o un aumento de la masa muscular, lo que se conoce como hipertrofia en la musculatura de la mano.
Identificar enfermedades crónicas.	Inflamación en las articulaciones de la mano.	Síndrome del túnel carpiano, tendinitis, bursitis, artritis.	Evaluar la alineación de la mano, coordinación.	Existe presencia de dolor en la mano sin actividad física	Se identifica como hipermovilidad a una movilidad excesiva en las articulaciones de la mano.	No se presenta hipotrofia ni hipertrofia si se establece según el tamaño y la apariencia de los músculos en

				o movimiento en curso.		relación con los patrones normales.
Identificar lesiones previas.	Dolor en la mano al hacer actividades repetitivas.	Lesiones traumáticas previas.	Apreciación de la flexibilidad, estabilidad de la mano.	Se ve presencia de dolor en la mano durante o después de realizar movimientos específicos.	La amplitud se considera normal si no hay limitaciones en los movimientos de la mano y alterada si hay limitaciones en la movilidad de la mano.	Si la fuerza de los músculos se encuentra dentro de los límites normales o presentan alguna alteración.

Elaborado por: Miguel Bonilla

Etapa 2

Existen diversas opciones para mitigar el dolor y reducir la inflamación en la mano derecha. Entre ellas se encuentran la magnetoterapia, la electroterapia y el uso de compresas frías y calientes [35]. La magnetoterapia es útil para consolidar huesos fracturados, mientras que la electroterapia estimula los músculos y ayuda en la recuperación. Combinando estas terapias con el uso de compresas se puede reducir la inflamación y el dolor, lo que permitirá al paciente realizar ejercicios de manera efectiva.

Para lograr una recuperación efectiva de la funcionalidad en la mano derecha, es necesario llevar a cabo una serie de ejercicios específicos de terapia que permitan mejorar la movilidad de la mano y la muñeca. Con el propósito de mejorar la destreza de la mano derecha, es importante seguir un programa de ejercicios específicos que ayuden a fortalecer los músculos y mejorar la flexibilidad de la mano y la muñeca. Se pueden incluir ejercicios como la apertura y cierre de la mano, la flexión y extensión de la muñeca, la elevación y descenso de la mano, y la pinza realizada en la mano [35]. También se pueden usar dispositivos como pelotas de goma y bandas elásticas para mejorar la resistencia y la fuerza de agarre. Es importante realizar estos ejercicios de forma regular y constante para obtener resultados efectivos y duraderos. Además, se recomienda realizarlos bajo la supervisión de un fisioterapeuta para asegurar la correcta ejecución y evitar posibles lesiones.

Etapa 3

Después de recuperar la movilidad en la mano derecha, se procede a fortalecer los músculos para mejorar su funcionamiento y acelerar la recuperación. Los ejercicios que se realizan son personalizados para cada paciente, según su caso particular, y pueden incluir estiramientos, fortalecimiento, coordinación y agarre. Existen varios tipos de ejercicios a realizar como:

- Ejercicios de estiramiento.

- Ejercicios de fortalecimiento.
- Ejercicios de coordinación.
- Ejercicios de agarre.
- Ejercicios de resistencia progresiva [35].

Después de detectar alguna patología que requiera tratamiento, es fundamental realizar los ejercicios adecuados para recuperar la funcionalidad de la mano derecha de manera efectiva. Los ejercicios de estiramiento se utilizan para aliviar la rigidez muscular y restablecer la amplitud de movimiento, mientras que los ejercicios de fortalecimiento contribuyen a reconstruir la fuerza muscular perdida [35]. Por otro lado, se trabaja en la coordinación y el agarre para lograr una mayor precisión y destreza de la mano al realizar actividades cotidianas, como vestirse, escribir y sostener objetos con seguridad.

3.1.2. Ejercicios de rehabilitación para la mano derecha

El realizar ejercicios es parte fundamental de la rehabilitación de la mano. Estos permiten mejorar la flexibilidad, movimiento, fuerza, coordinación, agarre y resistencia al realizar actividades cotidianas. Los ejercicios más comunes son basados en la flexión y extensión de la muñeca, apertura y cierre de la mano y una pinza realizada entre el dedo mediano y el dedo pulgar. De esta manera se trabajan los músculos, tendones, articulaciones y ligamentos involucrados en los movimientos que realiza la mano.

Flexión y extensión de la muñeca

La flexión y extensión de la muñeca son prácticas fundamentales para estirar los músculos y tendones de la muñeca y los antebrazos, lo que previene la rigidez y mantiene una adecuada amplitud de movimiento. Asimismo, mediante estos ejercicios,

se fortalecen los músculos flexores y extensores de la muñeca. Para llevar a cabo estos movimientos, se utiliza resistencia en forma de pesas o bandas elásticas. El fortalecimiento de estos músculos mejora significativamente el agarre y la coordinación de la mano. Además, ayuda a reducir la fatiga muscular, permitiendo realizar esfuerzos prolongados como levantar pesas o practicar deportes.

La amplitud de los movimientos articulares se mide a partir de la posición anatómica de una articulación específica. En el caso de la muñeca, la amplitud se refiere al rango de movimiento en el que la articulación puede moverse desde su posición anatómica hacia la flexión. Para este caso, la amplitud de la flexión activa es de 80° a 85° , lo que significa que la articulación puede moverse en un rango angular que va desde los 80° hasta los 85° respecto a su posición anatómica. Por otro lado, la extensión, también conocida como flexión dorsal, es el movimiento opuesto a la flexión activa. En este caso, la amplitud de la extensión es también de 80° a 85° , lo que indica que la articulación puede moverse en un rango angular que va desde los 80° hasta los 85° en dirección opuesta a la flexión activa, regresando a su posición anatómica [36].

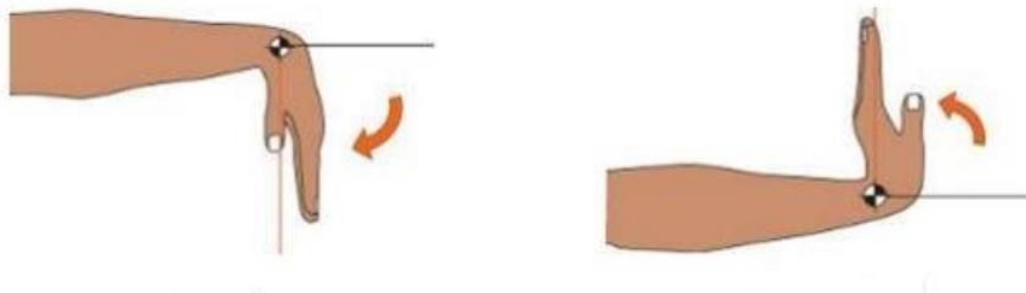


Figura 15. Flexión y Extensión de la Mano

Realizado por: Miguel Bonilla

Apertura y cierre de la mano

Este ejercicio es considerado muy versátil debido a la implicación de diversos músculos y tendones de la palma y los dedos. Al realizarlo, se previene la rigidez en los dedos, lo que evita que el agarre de la mano se debilite. Asimismo, se fortalecen

los músculos flexores de los dedos, utilizando para ello una pelota de goma o bandas elásticas. Esto proporciona mayor confianza al mejorar la coordinación y destreza para llevar a cabo actividades cotidianas. Además, al realizar varias repeticiones, se desarrolla la resistencia muscular y la capacidad de mantener el agarre de manera continua.

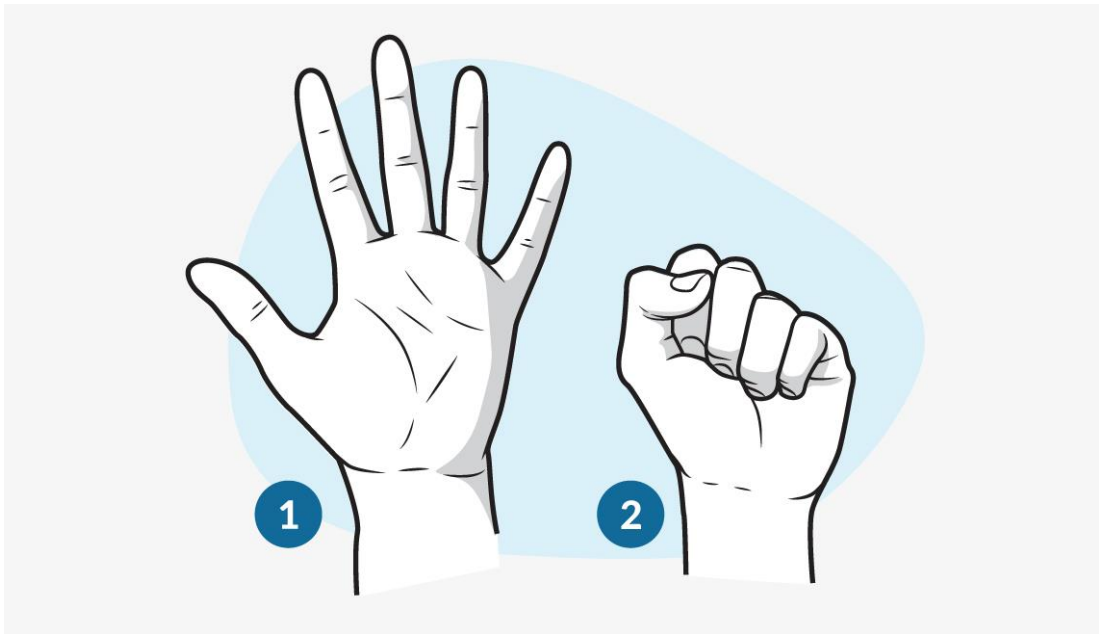


Figura 16. Apertura y cierre de la mano

Realizado por: Miguel Bonilla

Pinza entre el dedo mediano y el dedo pulgar

Este movimiento es considerado esencial, ya que permite el manejo preciso de objetos pequeños. Al realizar esta acción, se estiran los músculos flexores del pulgar y de la base del pulgar. Se recomienda mantener el gesto durante unos segundos para obtener mayores beneficios. Para fortalecer dichos músculos, se utilizan bandas elásticas o un objeto con forma de pinza. De esta manera, se mejora el agarre de objetos y se adquiere una coordinación fina y una adecuada fuerza en los dedos. Al incrementar el número de series en el ejercicio, se logra una mayor resistencia, lo que permite sostener objetos pesados durante períodos prolongados.



Figura 17. Pinza de la mano

Realizado por: Miguel Bonilla

3.1.3. Patologías que necesitan rehabilitación

La rehabilitación de la mano derecha es efectiva para tratar una amplia gama de patologías que afectan su funcionalidad. Entre las patologías más habituales se encuentran las lesiones traumáticas como fracturas, esguinces y luxaciones. De igual manera entran enfermedades degenerativas como la artritis, osteoartritis, enfermedad de Dupuytren, esclerosis múltiple y Parkinson. Por último, se toma en cuenta afecciones que dañan los tendones, músculos y articulaciones de la mano. La rehabilitación temprana es fundamental para obtener mejores resultados al mejorar la movilidad de la mano[35].

Lesiones traumáticas

Cuando se trata de lesiones traumáticas se debe de obtener información detallada sobre el tipo de lesión que se tiene, puede ser fracturas, esguinces, luxaciones, contusión,

mecanismo de rotación, corte, aplastamiento, lesiones químicas o quemaduras, etc. De igual manera el lugar dónde ocurrió y la posición de la mano cuando ocurrió el trauma. El recopilar estos datos ayudan a tener un diagnóstico preciso y determinar adecuadamente el tratamiento adecuado y determinar la frecuencia y duración con la que se debe de realizar el ejercicio [35].

Enfermedades degenerativas

Las patologías degenerativas que con más frecuencia afectan a la mano derecha son la artrosis u osteoartrosis, las cuales pueden provocar dolor, inflamación y rigidez en la mano derecha. De igual manera, la artritis comparte características con estas patologías al ser una enfermedad inflamatoria. Otra patología común de enfermedades degenerativas es el Parkinson, la cual produce síntomas como movimientos involuntarios, temblor, rigidez, bradicinesia, alteración de los reflejos posturales, etc. La rehabilitación es eficaz para abordar de manera correcta los síntomas mencionados y así mejorar la funcionalidad de la mano derecha [35].

Afecciones Múltiples

Otras afecciones que afectan la funcionalidad de la mano derecha se originan especialmente cuando existe daño en los tendones, músculos y articulaciones. Entre ellas se encuentran la tendinitis, desgarros o distensiones musculares, síndrome del túnel carpiano, distensión del mangito rotador, inflamación de los tendones, etc. Diagnosticado cualquiera de estas afecciones se puede llegar a experimentar síntomas de dolor, debilidad muscular, hinchazón y dificultad de movimiento [35].

3.2. Desarrollo de la propuesta

Durante la elaboración del proyecto, se llevó a cabo una evaluación de diversos lenguajes para el desarrollo de algoritmos de Machine Learning. Se analizaron diversas opciones como Python y Matlab, considerando sus características, bibliotecas y frameworks disponibles para el Machine Learning.

En cuanto al desarrollo del sistema se consideraron diferentes metodologías para el desarrollo del sistema. Se evaluaron enfoques para seleccionar la metodología más adecuada y que permita un desarrollo eficiente y exitoso.

3.2.1. Adquisición de datos mediante video cámaras

El adquirir datos de los movimientos realizados por la mano derecha es esencial para analizar información precisa sobre el movimiento y posición que tiene la mano en tiempo real. Se puede utilizar diversos tipos de cámaras, como cámaras RGB, Leap Motion Controller y Kinect. Cada tipo de cámara tiene sus características y aplicaciones particulares. Para decidir qué cámara es mejor se llevará a cabo una comparación entre ellas. Basándose en la comparación se identificará cuál es la más adecuada acorde a los requisitos específicos del sistema.

Tabla 10. Tabla comparativa de video cámaras

Características	Cámaras RGB	Leap Motion Controller (LMC)	Kinect
Resolución	640x480	640x240	640x480
Campo de visión	90°	150x120°	57°
Profundidad	27 mm	60 a 80 cm	1,2 m a 3,5 m
Captación del espectro electromagnético	380 nm – 760 nm	850 nm	830 nm

Velocidad de captura	30 FPS	200 FPS	30 FPS
Compatibilidad	Windows, Linux, Mac OS.	Windows, Linux, Mac OS [37].	Windows, experimental Mac OS y Linux [38].
Conectividad	USB 2.0	USB 2.0	USB 2.0
Tecnología	Sensores de imagen CMOS.	Cámaras de infrarrojo cercano [37].	Sensor CMOS de infrarrojos, motor de inclinación, cámara RGB [38].
Integración a Software	Se utilizan para videoconferencia, streaming y edición de video.	Se emplea realidad virtual y aumentada, simulación, diseño interactivo, juegos y más [37].	Se aplica en juegos interactivos, seguimiento corporal, animación [38].
Datos que Captura	Imágenes	27 posiciones de la mano e imágenes	Imágenes
Costo	\$66	\$139	\$166

Elaborado por: Miguel Bonilla

Después del estudio comparativo de la **Tabla 10**, se seleccionó el LMC como la cámara con la cual se procede a la obtención de datos. El LMC sobresale por el uso de infrarrojos y su capacidad de efectuar un seguimiento preciso y en tiempo real de los movimientos de la mano. A diferencia de otras cámaras el LMC proporciona una experiencia más fluida y espontánea. Esta posee una integración sencilla con software y entornos de programación. Convirtiéndolo en la elección ideal para desarrollar

aplicaciones de realidad virtual, realidad aumentada, diseño interactivo y juegos aplicados a la rehabilitación.

Sistema de adquisición de datos

Para realizar la adquisición de datos se utiliza la combinación de LMC y Matlab. Mediante Matlab se procesan y analizan los datos recolectados a través del LMC. En el caso actual, Matlab también se utiliza para el preprocesamiento de datos, extracción de características y clasificación de los gestos de la mano. Además, Matlab es muy utilizado para el análisis de grandes conjuntos de datos y realizar cálculos complejos. Para un correcto análisis de los datos, se compila librerías de C++, MEX en combinación con Matlab.

Las librerías compiladas de C++ ayudan al desarrollo de una estructura a partir de cada uno de los fotogramas obtenidos. Estos datos se almacenan y guardan en una matriz de datos. El uso de librerías MEX permite la integración de lenguajes de programación como C++ y Matlab. Al combinar estos lenguajes se puede hacer uso de la librería Leap Motion Controller en C++, esto mediante la API MEX. De esta manera se aprovecha las capacidades de ambos entornos y facilita el análisis y procesamiento de los datos.

Los datos capturados mediante LMC incluyen la posición espacial, dirección, velocidad e imágenes de la mano. Estos datos son necesarios para generar el dataset y obtener información demográfica y analítica de los participantes. Los datos obtenidos son nombre, edad, género, etnia, ocupación, lesiones en la mano y nivel de iluminación. Este tipo de información es útil para recopilar datos precisos y evaluar la condición previa de los participantes en el estudio.

El sistema de adquisición de datos consta de 4 módulos vinculados: INPUTS, OUTPUTS, Get the frame y Get an image. El módulo INPUTS establece el LMC como el mecanismo de recolección de datos y el protocolo a seguir para la toma de datos. En cambio, el módulo OUTPUTS es la toma de los datos demográficos mediante preguntas directas al usuario. Estos datos se completan con la obtención del nivel de iluminación mediante el uso de una aplicación. El módulo Get the frame obtiene los fotogramas mientras el LMC rastrea la mano del usuario. Cada fotograma se evalúa en la entrada con datos constantes como la marca de tiempo, mano rastreada e identifica si es la mano derecha o izquierda. El último módulo Get an image se encarga de construir la imagen en relación a los puntos espaciales de la mano. Esto se consigue procesando cada fotograma y determinando el tiempo. Logrando establecer el fotograma actual y obteniendo cada toma del sensor los datos de seguimiento. Finalmente, estos datos se almacenan en un archivo .mat [39].

3.2.2. Lenguaje para el desarrollo de algoritmos de Machine Learning

Existen varios lenguajes que permiten trabajar con algoritmos de Machine Learning. Cada lenguaje muestra beneficios y limitaciones en cuanto a su funcionalidad, eficiencia y factibilidad de uso. Por eso, es importante conocer las características, capacidades, ventajas y desventajas de los lenguajes a elegir conforme a las necesidades del proyecto. Se efectuó un estudio comparativo entre el lenguaje de programación Python y Matlab. Para ello se elaboró una tabla comparativa, para tomar una decisión fundamentada sobre el lenguaje que mejor se adapte a los requerimientos del proyecto.

Python

Python es un lenguaje interpretado ampliamente utilizado en el campo de la ciencia de datos y el machine learning. Con sus librerías especializadas, como Pandas, Scikit-Learn y TensorFlow, ofrece herramientas poderosas para el desarrollo de algoritmos. Su portabilidad en diferentes plataformas, como Linux, Windows y Mac OS, lo

convierte en una elección flexible. Gracias a su amplia gama de librerías y su capacidad multiplataforma, Python es una opción para Machine Learning [40].

Matlab

Matlab es un software creado con el propósito para resolver problemas en el ámbito de la ciencia y la ingeniería. Cuenta con una extensa gama de herramientas incorporadas para el análisis de datos, control de sistemas, procesamiento de imágenes y señales. Adicionalmente, Matlab es un lenguaje de programación basado en matrices, lo que lo hace el método más adecuado para expresar y realizar cálculos matemáticos computacionales [41].

En la **Tabla 11.** se realiza una comparación entre los lenguajes propuestos para el desarrollo de algoritmos de machine learning.

Tabla 11. Tabla comparativa de lenguajes de desarrollo para Machine Learning

Características	Python	MATLAB
Lenguaje de programación	Interpretado, multipropósito	Interpretado, enfocado en cálculos numéricos
Sintaxis	Fácil de aprender y leer	Sintaxis más orientada a las matemáticas
Librerías y módulos	Amplia variedad de librerías y frameworks como TensorFlow, PyTorch, Scikit-learn, Keras entre otros.	MATLAB Machine Learning Toolbox, Neural Network Toolbox, Statistics and Machine Learning Toolbox
Costo	Gratuito y de código abierto (Open source)	Requiere licencia para uso comercial
Compatibilidad de sistemas operativos	Funciona en Windows, macOS y Linux	Soporte para múltiples sistemas operativos

Exploración y manipulación de datos	Librerías como Pandas y NumPy para una exploración y manipulación eficiente de datos	Herramientas incorporadas de MATLAB para visualización y manipulación de datos
Visualización de datos	Librerías como Matplotlib y Seaborn para visualización rica y personalizable	Herramientas incorporadas de MATLAB para visualización y presentación de resultados

Elaborado por: Miguel Bonilla

A pesar de que Matlab cuenta con funciones muy similares a Python, este último se selecciona como la opción preferida para el desarrollo de machine learning. En primer lugar, Python posee una amplia variedad de librerías y módulos diseñados para machine learning. Python también posee herramientas para el manejo de datos, lo cual permite realizar acciones como filtrado, agrupamiento, transformación, limpieza de datos entre otros. Además, Python es un lenguaje de código abierto y gratuito, lo que lo hace accesible y reduce los costos en el desarrollo del proyecto.

3.2.3. Metodología de desarrollo

El uso de una metodología es fundamental para garantizar el éxito del proyecto. Una metodología ágil permite trabajar en ciclos cortos. De esta manera se prioriza las funcionalidades clave y se entrega avances del sistema. Igualmente, esta metodología permite interactuar con los usuarios finales y obtener retroalimentaciones. Esto permite tener una respuesta rápida a los cambios y manejar de forma eficiente los recursos y procesos. Para determinar la metodología que mejor se ajuste al desarrollo del proyecto se realizará una comparativa entre tres metodologías ágiles.

Scrumban

Scrumban es una metodología derivada de la unión entre Scrum y Kanban. Esta metodología busca proporcionar un punto medio al utilizar la estructura de Scrum y la

flexibilidad de Kanban. Al emplear Scrumban, se logra tener una rápida evolución y cambios constantes en el desarrollo del proyecto [42].

Extreme Programming (XP)

La metodología XP es ampliamente utilizada en el desarrollo de software. Esta misma ofrece soluciones efectivas y simples. La metodología XP se basa en retroalimentación continua entre el cliente y el equipo de desarrollo. Esta retroalimentación permite adaptar el desarrollo de acuerdo a las necesidades del cliente [42].

Crystal

La metodología Crystal se basa en 7 principios:

- **Entregas frecuentes.** – se basa en un ciclo de vida iterativo e incremental.
- **Mejora reflexiva.** – Las iteraciones permiten una mejora continua del proyecto.
- **Comunicación osmótica.** – El equipo tiene una comunicación directa y fluida.
- **Seguridad personal.** – El equipo puede dar su opinión sin restricciones.
- **Enfoque.** – Períodos de no interrupción al equipo.
- **Fácil acceso al cliente.** – Como mínimo se realizan reuniones semanales y los clientes deben de estar accesibles.
- **Entorno técnico.** - Pruebas automatizadas, gestión de la configuración e integración continua [42].

A continuación, en la **Tabla 12.** se realiza una comparación de cada una de las metodologías descritas.

Tabla 12. Tabla comparativa de metodologías

Características	Scrumban	Extreme Programming	Crystal
Descripción	Un tablero Scrumban es un tablero Kanban que integra elementos de Scrum como un backlog del sprint, las etapas de flujo de trabajo del equipo, y una columna para las tareas finalizadas.	Metodología ágil que gestiona los proyectos que se centran en la velocidad y la simplicidad de los ciclos.	Permite desarrollo de software ágil que se centra en las personas y la interacción de ellas mientras se trabaja en el proyecto.
Implementación	Desde proyectos pequeños hasta de gran magnitud o a largo plazo.	Proyectos a corto plazo.	Proyectos de pequeña y mediana escala.
Tamaño del equipo	Varía según el proyecto.	Máximo de 12 personas.	Hasta 8 personas o menos.
Estilo de desarrollo	Es flexible e incrementa su capacidad para adaptarse según las necesidades del proyecto.	Desarrollo iterativo, pruebas constantes, integración frecuente e incremental aportando pequeñas mejoras.	Ve el desarrollo como una serie de partidos que consisten en inventar y comunicar. Cada partido es diferente y tiene como objetivo entregar software y prepararse para el siguiente juego.
Estabilidad	Combina la estabilidad de Scrum con la flexibilidad y adaptabilidad de Kanban.	No contiene código duplicado, menor número posible de métodos y clases.	Se mide mediante hitos clave y la estabilidad de las fases.
Ciclo de vida	Iteraciones cortas y entrega incremental.	Ciclos cortos.	Iterativo e incremental.
Flexibilidad	Mayor flexibilidad del método Kanban.	Modelos de implementación y disponibilidad del usuario.	El paradigma orientado a objetos ayuda a conseguir flexibilidad.

Adaptabilidad	Se adapta rápidamente a los cambios.	Ofrece más énfasis en la adaptabilidad.	Por la vista del sistema algo cambiante se posee una gran adaptabilidad.
Calidad	Mejora la calidad de los resultados y de la toma de decisiones.	Ofrece rendimiento y resultados de calidad.	Una alta calidad en menos tiempo.
Rendimiento	Monitorea regularmente la eficiencia del proceso.	Deja las optimizaciones al final.	Mejora el rendimiento del equipo al disponer más recursos de los necesarios.
Diseño	Combina los principios de Scrum y Kanban en un enfoque ágil.	Enmiendas puntuales y funcionalidad mínima.	Adapta el proceso y las practicas según las características únicas del equipo.
Pruebas	Trabajo posterior al desarrollo del sprint (pruebas del sistema, empaquetado y despliegue)	Implantación y pruebas de aceptación, protección contra fallos y unidades de prueba.	Pruebas automatizadas, gestión de la configuración e integración continua.

Realizado por: Miguel Bonilla

Según el análisis comparativo, se considera a Crystal como la mejor opción para el desarrollo del proyecto. Crystal ofrece flexibilidad y adaptabilidad, lo que permite gestionar adecuadamente los cambios en el proceso. Además, Crystal se destaca por su interacción y comunicación del equipo de trabajo con los usuarios del sistema. Lo que permite un claro entendimiento de las necesidades y requisitos del sistema. Igualmente, el desarrollo en Crystal se enfoca en entregar el software y mejorar continuamente, todo esto en menor tiempo y con una alta calidad.

3.2.4. Aplicación de la metodología de desarrollo

3.2.4.1. Proceso de desarrollo de la propuesta

El proceso a seguir en la metodología se encuentra detallado en la **Figura 15**.

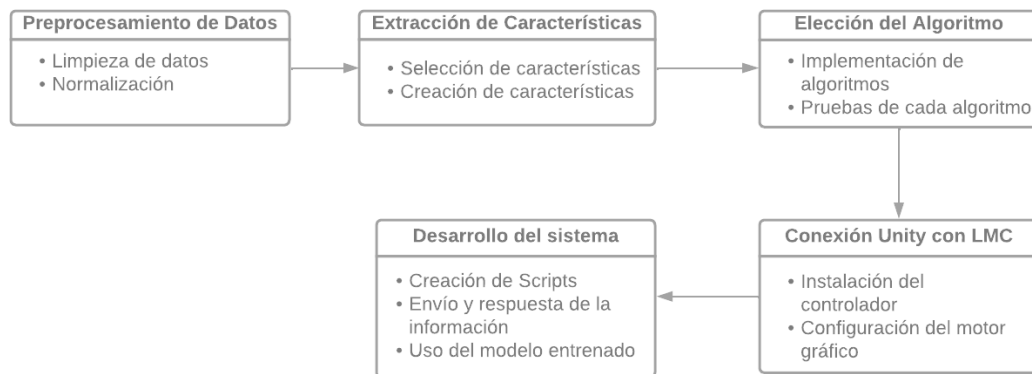


Figura 18. Esquema de la metodología

Elaborado por: Miguel Bonilla

El proceso comienza con el preprocesamiento de datos, en el cual se lleva a cabo la corrección de inconsistencias presentadas en el dataset. Luego, se realiza la normalización de los datos, asegurando que las características tengan la misma escala. A continuación, se procede con la extracción de las características. Esta sección incluye la selección de características relevantes para mejorar la precisión del modelo. Después de eso, se realiza la elección del algoritmo adecuado. Para ello se implementan los algoritmos elegidos y se realizan pruebas para evaluar su rendimiento. Posteriormente, se establece la conexión entre Unity y LMC, esto conlleva a la instalación del controlador y la configuración del motor gráfico. Con la conexión realizada se procede al desarrollo del sistema. El sistema contiene scripts que facilitan la captura y procesamiento de los datos capturados por LMC. Después de capturar esta información se la envía y recibe utilizando web socket, utilizando el modelo entrenado

3.2.4.2. Preprocesamiento de datos

Para el preprocesamiento se trabajó con un conjunto inicial de celdas que se obtuvo durante la adquisición de datos. Este conjunto de celdas está compuesto de 30 filas y 75 columnas. Las 30 filas corresponden al número de repeticiones realizadas por cada gesto, mientras que las 75 columnas representan a los 15 pacientes que llevaron a cabo

los 5 gestos realizados. Cada celda dentro del conjunto contiene una matriz de 70 filas por 30 columnas. Las 70 filas representan las instancias de tiempo en que se capturo los datos. Mientras que las 30 columnas correspondían a las 15 posiciones espaciales (X, Y, Z) y las 15 direcciones (X, Y, Z) de cada dedo de la mano.

Posteriormente, se procedió a reorganizar las celdas para obtener una matriz resultante de 2250 filas por 2100 columnas. Las 2250 filas corresponden a las observaciones obtenidas del conjunto de celdas de 30 x 75. Por otro lado, las 2100 columnas son las características extraídas de cada una de las celdas de 70 x 30. El reacomodar las celdas de esta manera permitió obtener un dataset más adecuado para continuar con el preprocesamiento de datos.

A continuación, se utilizó la fórmula de normalización de datos. Esta fórmula fue aplicada tanto en columnas como en filas. El primer paso de esta fórmula es obtener el valor máximo y valor mínimo correspondiente a cada fila o columna, según corresponda. Esta fórmula permitió ajustar las características a la misma escala, lo que es importante para el correcto análisis de los datos. Esto resulta fundamental para evitar que valores más grandes o pequeños dominen el análisis. Logrando que el clasificador realice una mejor comparación entre las diferentes características.

3.2.4.3. Extracción de características

Para la extracción de características se utilizó la matriz de datos de 2250 observaciones por 2100 características. Con el fin de analizar los datos se dividió la matriz en ventanas de un tamaño de 70. Cada ventana se asocia con una posición o dirección específica (X, Y, Z) de cada dedo de la mano. El propósito de dividir en ventanas es trabajar en fragmentos más manejables la función generada por el gesto de la mano. Dado que cada observación es una función el dividir en ventanas permite realizar operaciones para obtener valores de referencia que representen a la función original.

A continuación, cada ventana de tamaño de 70 se divide en 5 subventanas más pequeñas de 14 datos cada una. En cada subventana se calculan 5 valores como características: media, desviación estándar, raíz cuadrada media, derivada y varianza. Se obtienen 5 características de cada subventana y un total de 25 características por ventana. En la **Tabla 13** se detalla las características obtenidas.

Tabla 13. Características del dataset

Características		
Nombre	Descripción	Cantidad por Ventana
Media	Proporciona una medida de tendencia central de los gestos.	5
Desviación Estándar	Se calcula la medida de dispersión de los datos con respecto a la media.	5
Raíz Cuadrada Media	Permite caracterizar la intensidad o tamaño promedio de los gestos, identificando aquellos con mayor o menor magnitud.	5
Derivada	Se puede ver como los datos van cambiando a lo largo del tiempo. Esto permite identificar los movimientos realizados y analizar patrones de movimiento en los gestos.	5
Varianza	Se mide la variabilidad de los gestos y sirve para identificar aquellos con una mayor dispersión.	5
Total de características por ventana		25

Elaborado por: Miguel Bonilla

Finalmente, al tener 30 ventanas se obtienen 750 características en total (30 ventanas y 25 características por ventana). Estas características se combinan en un dataset final de 2250 observaciones por 750 características. Este dataset final se puede utilizar para análisis y entrenamiento de algoritmos de Machine Learning. Dichos algoritmos de Machine Learning usan las características extraídas para aprender los patrones y realizar predicciones.

Dispersión de clases

La dispersión de clases representada por la matriz inicial de 2250 x 2100 fue realizada utilizando t-SNE. El t-SNE es un algoritmo de reducción de dimensionalidad no lineal para visualizar patrones y estructuras. Al aplicar t-SNE en dicha matriz se pudo observar que no existía una dispersión significativa en las clases. De igual forma se puede observar que no existe una separación o clasificación de las etiquetas asociadas. El gráfico de la dispersión de todos estos datos se encuentra reflejado en la **Figura 16**.

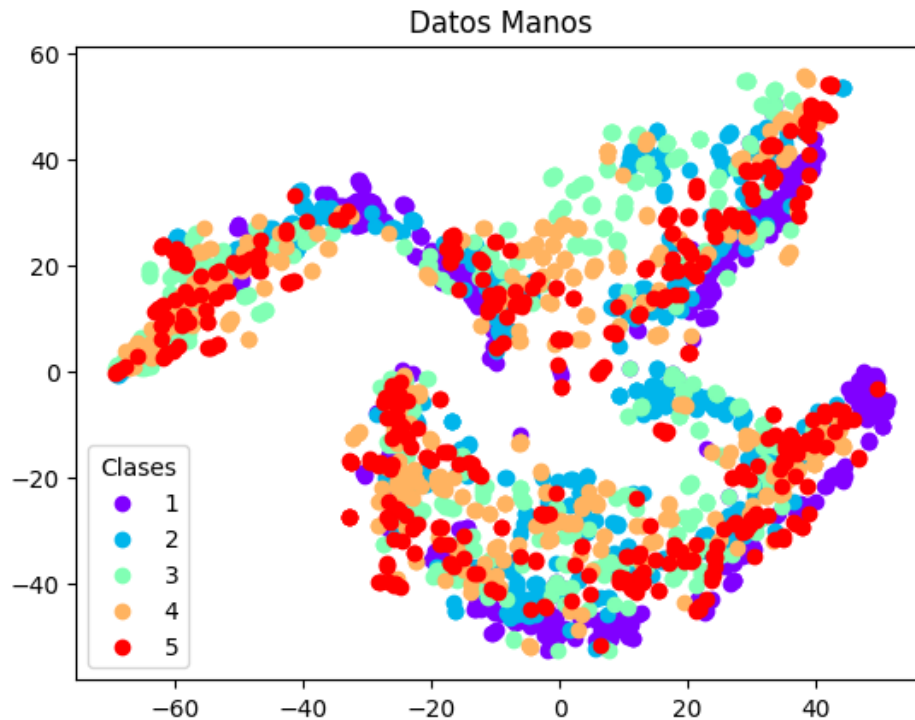


Figura 19. Gráfico t-SNE del dataset sin extraer características

Elaborado por: Miguel Bonilla

Al realizar la extracción de características se obtuvo una matriz de 2250 x 750. Esta matriz constituye el dataset que se utilizará para el entrenamiento y testeo de los datos. Para visualizar la dispersión y clasificación de dicho dataset se aplicó t-SNE. Al graficar se pudo visualizar que existen similitudes entre las características y se agrupan para la clasificación. La visualización resultante del dataset final mediante t-SNE se presenta en la **Figura 17**.

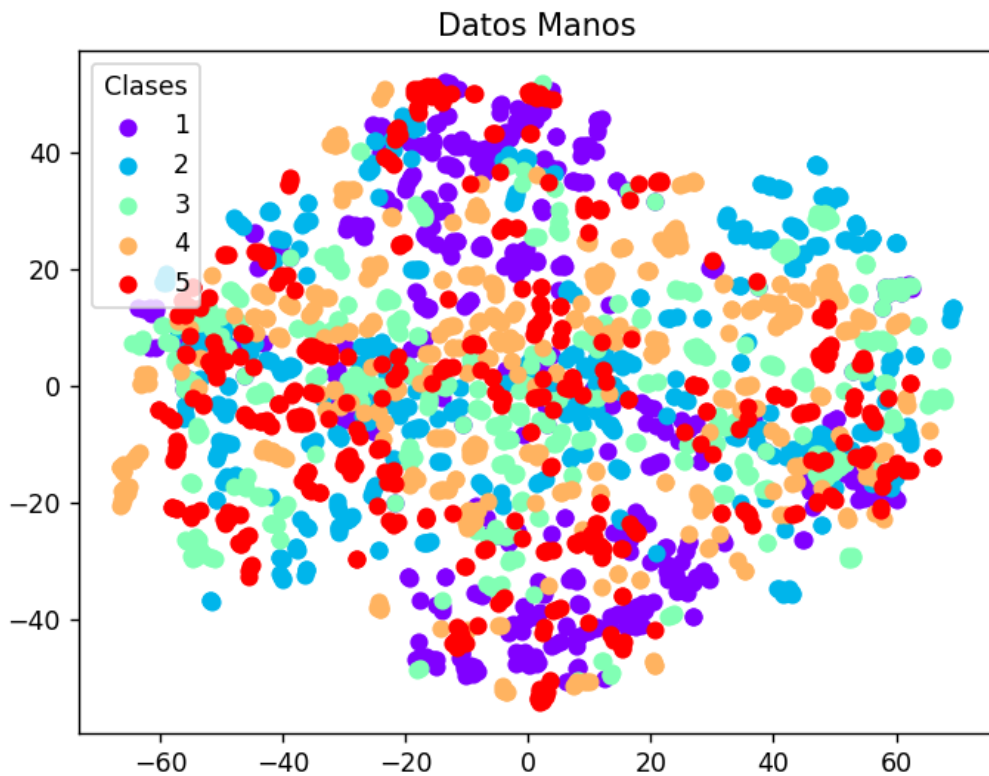


Figura 20. Gráfico t-SNE dataset final

Elaborado por: Miguel Bonilla

3.2.4.4. Elección del algoritmo

Para lograr una correcta clasificación de los datos se analizaron dos algoritmos: Red Neuronal Artificial (RNA) y K-Nearest Neighbors (KNN).

- **Red Neuronal Artificial**

Las redes neuronales artificiales son un algoritmo de aprendizaje que se basa en emular el funcionamiento del cerebro humano para resolver problemas complejos. Una red neuronal está formada por un conjunto de neuronas que

realiza operaciones para producir una salida. Estas neuronas se encuentran conectadas y organizadas en capas, donde las neuronas de una capa se conectan a la capa siguiente. Se inicia el entrenamiento con una serie de pesos, los cuales se ajustan a medida que los datos fluyen entre las capas. Conforme los pesos pasan entre las capas estos se ajustan y la red neuronal aprende a realizar tareas específicas [43].

- **K-Nearest Neighbors (KNN)**

Es un algoritmo de aprendizaje automático supervisado. Se basa en que los elementos similares entre si tienden a agruparse en el mismo espacio. El algoritmo KNN busca los vecinos más cercanos en función de la distancia y asigna la clase más común entre los k vecinos. El valor de k es importante, ya que este afecta la precisión y generalización del modelo. Un valor de k más pequeño puede resultar en una mejor clasificación, mientras que un valor de k grande puede suavizar la clasificación [43].

A continuación, en la **Tabla 14.** se muestra una comparación entre los algoritmos previamente entrenados. Se incluye la división que tuvo el dataset en datos de entrenamiento y prueba, la estructura del modelo y su precisión.

Tabla 14. Tabla comparativa de algoritmos de Machine Learning

Algoritmo	División del dataset	Estructura	Exactitud
Red Neuronal Artificial	El dataset se divide en conjuntos de entrenamiento y prueba utilizando una validación cruzada de 3 kfold. Esto significa que	Red neuronal de 4 capas. La capa de entrada consta de 25 neuronas, siguen dos capas ocultas de 15 y 5 neuronas respectivamente y una capa de salida con un	Entrenamiento: 96.88% Testeo: 93.55%

	el dataset se divide en 3 partes: 2 partes para entrenamiento y 1 para pruebas en cada iteración.	número de neuronas igual al número de clases. La capa de entrada y las capas ocultas utilizan una función de activación relu. La capa de salida utiliza una función de activación softmax. Además, se aplica una regularización L2 en la última capa oculta para evitar el overfitting.	
K-Nearest Neighbors (KNN)	Se divide en 80% para entrenamiento y 20% para prueba.	Se crea y entrena un clasificador KNN, especificando el número de vecinos más cercanos en 1.	Entrenamiento: 90.65 % Testeo: 87.33%

Elaborado por: Miguel Bonilla

Tras realizar la comparación en la **Tabla 14**, se seleccionó la RNA como el algoritmo a utilizar. Esta elección se basa en varios factores que respaldan su adecuación en el proyecto. En primer lugar, la RNA alcanzó una exactitud del 93.55%, lo que indica que tiene una alta capacidad para reconocer los gestos realizados. Además, las RNA son conocidas por su alto manejo de datasets extensos y características complejas. Dado que el dataset final consta de 2250 observaciones y 750 características, el alto manejo de datos es significativo para la clasificación de gestos. Específicamente por la variedad de movimientos y combinaciones que se obtienen al realizar los gestos.

3.2.4.5. Conexión de Unity con LMC

Para establecer la conexión entre Unity y LMC se descargó el SDK proporcionado por Leap Motion. El SDK facilita la captura y seguimiento preciso del movimiento de la mano derecha. Todo esto gracias a que LMC utiliza cámaras infrarrojas para captura del movimiento y posiciones de la mano. El software de seguimiento de LMC procesa los datos en tiempo real, incluyendo la identificación de los dedos, incluso si están ocultos a la vista o parcialmente obstruidos [44].

SDK de Unity para Leap Motion Controller

El SDK de Unity para LMC es una herramienta de código abierto que se encuentra en el repositorio Github de Ultraleap. Para que el SDK funcione correctamente requiere el software de reconocimiento de Ultraleap. El SDK proporciona una amplia gama de recursos y componentes para integrar y utilizar LMC en proyectos de Unity. Mediante esto se tiene una interacción más inmersiva y realista con aplicaciones y juegos.

Tabla 15. Contenido incluido en el SDK de Unity

Core	Proporciona una interfaz mínima entre Unity y el LMC. Mediante el Core se puede renderizar un conjunto de manos y adjuntar objetos a las articulaciones de la misma.
Motor de Interacción	Ofrece representaciones físicas de manos y controladores de realidad virtual.
Hands	Facilita herramientas para la manipulación de los recursos propios de manos 3D.

Elaborado por: Miguel Bonilla

Captura de datos en Unity

Una vez que el SDK de Unity está funcionando, se procede a la captura de datos en Unity. En este proceso se utilizaron 5 de los 27 puntos referenciales proporcionados por LMC para predecir el gesto realizado. Además, se utilizaron 2 puntos referenciales para mostrar el avance obtenido en la rehabilitación. De los 5 puntos seleccionados para la predicción del gesto. De los puntos se capturaron tanto las posiciones como las direcciones de cada dedo, lo que da un total de 15 posiciones y 15 direcciones. Estos datos se registran en n instancias de tiempo, lo que permite obtener datos secuenciales que realiza la mano cuando ejecuta el gesto.

En cuanto al avance de la rehabilitación, se utilizaron las posiciones del dedo medio y de la muñeca. Para cada uno de estos puntos, se capturaron las coordenadas X, Y, Z en n instancias de tiempo. Estas posiciones se utilizan para monitorear y visualizar el progreso que se tiene durante el proceso de rehabilitación. Estos datos capturados brindan información valiosa para el análisis y seguimiento de gestos y para el monitoreo de la rehabilitación.

3.2.4.6. Desarrollo del sistema

Para el desarrollo del sistema de rehabilitación de la mano derecha se procedió a realizar lo siguiente:

Creación de Scripts

Se desarrollaron dos tipos de scripts para diferentes propósitos en el sistema. El primer script se encarga de capturar y procesar los datos recolectados mediante LMC. Mientras que el segundo script es utilizado para visualizar un video de ejemplo del gesto que debe realizar el paciente.

En el script de captura y envío de datos se crean dos vectores de tipo `GameObject`. Estos vectores son utilizados para guardar los objetos correspondientes a los dedos de la mano. En el primer vector se almacenan los datos para realizar la predicción utilizando posiciones y direcciones (X, Y, Z) de cada dedo de la mano. Por lo contrario, el segundo vector almacena información para verificar el avance que posee el paciente. En este vector se guarda posiciones X, Y, Z del dedo medio y la muñeca. Los datos capturados de cada vector son guardados en un `DataTable`. Luego, se procesa y convierte la `DataTable` en una matriz para su posterior preprocesamiento y su uso en la predicción del gesto.

De manera similar, el segundo script está configurado para visualizar un video de ejemplo que muestra el gesto que el paciente debe realizar. Esto permite que el paciente observe de forma gráfica como se realiza el gesto antes de realizarlo. La visualización del gesto antes de realizarlo proporciona retroalimentación inmediata. Mediante esto el paciente puede comprender y replicar el gesto de manera más precisa. Además, al tener una referencia visual el paciente ejecuta el gesto de manera precisa y efectiva.

Envío y tratamiento de la información

La conexión entre Unity y Python se estableció mediante un Web Socket. El Web Socket permite el envío y tratamiento de la información. En el lado de Python se recibe dos matrices y se realiza el correspondiente preprocesamiento. El preprocesamiento se detalla a continuación.

Primero, se realizó una reducción en el número de instancias de tiempo. Siendo inicialmente n instancias de tiempo se las redujo a 70. Para ello se tomó cada matriz y se la dividió en 70 grupos de n filas y en cada grupo se obtuvo un valor medio. En la primera matriz, se obtuvo una matriz resultante de dimensiones 70 filas por 30 columnas. Cada fila corresponde a una instancia de tiempo, mientras que cada columna representa las posiciones y direcciones en los ejes X, Y, Z de cada dedo de la mano.

La segunda matriz resultante tiene dimensiones de 70 filas por 6 columnas. De igual manera, las filas representan las instancias de tiempo y las 6 columnas corresponden a las posiciones X, Y, Z del dedo medio y la muñeca.

A continuación, se procede a trabajar con la matriz resultante de dimensiones 70x30. Esta matriz se transforma en un vector de dimensiones 1x2100, obteniendo las 2100 características que posee el dataset antes de la extracción de características. Seguido, se aplicó el preprocesamiento de datos a este vector, comenzando con la normalización. Después, se realiza la división de ventanas y subventanas correspondientes al vector en cuestión. Luego, se extrajeron las correspondientes características: media, desviación estándar, raíz cuadrada media, derivada y varianza. Estas características se utilizaron para generar un nuevo vector de tamaño 1x750, que posteriormente se envió al clasificador para su predicción.

Por otro lado, con la matriz resultante de dimensiones 70x6 se realizó una división en dos matrices de tamaño 70x3 cada una. Estas matrices representan las instancias de tiempo y las posiciones X, Y, Z del dedo medio y la muñeca. Utilizando estas matrices, se calculó la distancia euclidiana entre los puntos de cada matriz. De manera similar, se calculó el ángulo correspondiente. Estos valores de distancia y ángulo permiten evaluar el avance del paciente en su proceso de rehabilitación.

Al finalizar el tratamiento de los datos en Python, se crea un diccionario que almacena la información relevante para la rehabilitación del paciente. Este diccionario incluye la predicción realizada por el modelo entrenado, así como las distancias y ángulos calculados durante el análisis. El diccionario de datos se envía a Unity utilizando el Web Socket establecido previamente. Esta comunicación inmediata permite una transferencia rápida y en tiempo real de la información procesada. Como resultado, el paciente puede recibir una retroalimentación inmediata y en tiempo real sobre su progreso en el proceso de rehabilitación.

Al recibir el diccionario de datos en Unity, se pueden utilizar las predicciones, distancias y ángulos calculados para proporcionar una retroalimentación al paciente. Esto permite al paciente conocer instantáneamente el estado de su rehabilitación. La retroalimentación instantánea y en tiempo real resulta valiosa para el paciente, ya que proporciona información en tiempo real sobre su rendimiento y progreso. Esto, a su vez, facilita la mejora continua y fomenta la motivación durante todo el proceso de rehabilitación.

3.2.4.7. Funcionalidad del sistema

El sistema se basa en la detección y seguimiento de gestos utilizando Leap Motion Controller. Al iniciar el sistema, se muestra la pantalla de inicio que incluye un botón para iniciar el proceso.



Figura 21. Pantalla de inicio

Elaborado por: Miguel Bonilla

Una vez que el usuario presiona el botón, se accede a una pantalla de reproducción de video de ejemplo, donde se muestra el gesto específico que se debe realizar.



Figura 22. Video de ejemplo del gesto

Elaborado por: Miguel Bonilla

Al finalizar el video, de forma automática se pasa a la pantalla de captura de datos. En esta pantalla, el usuario realiza el gesto correspondiente utilizando Unity.

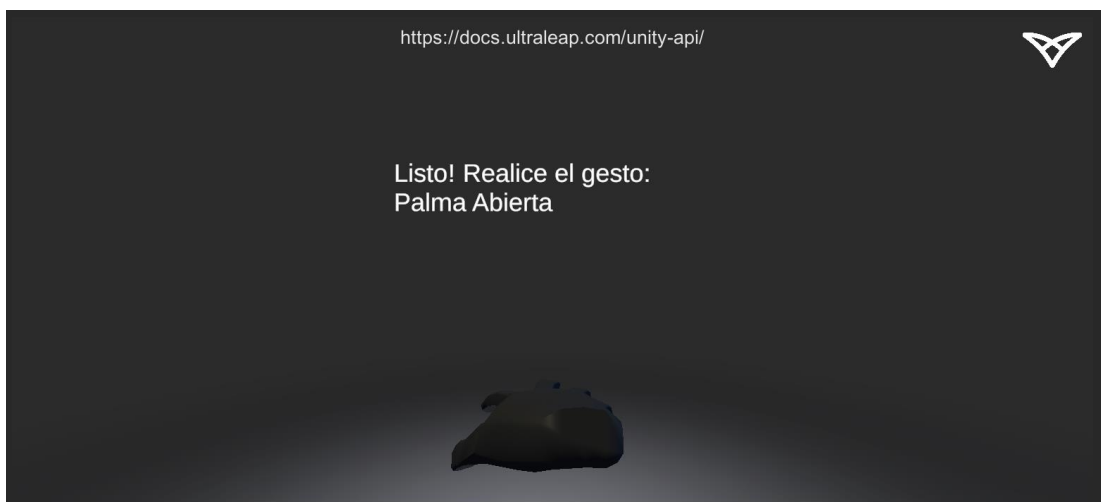


Figura 23. Pantalla de captura de datos

Elaborado por: Miguel Bonilla

El sistema procesa y analiza los datos capturados para evaluar el rendimiento del gesto realizado. Posteriormente, se muestra un mensaje que indica el avance de la rehabilitación basado en el análisis de los datos capturados.

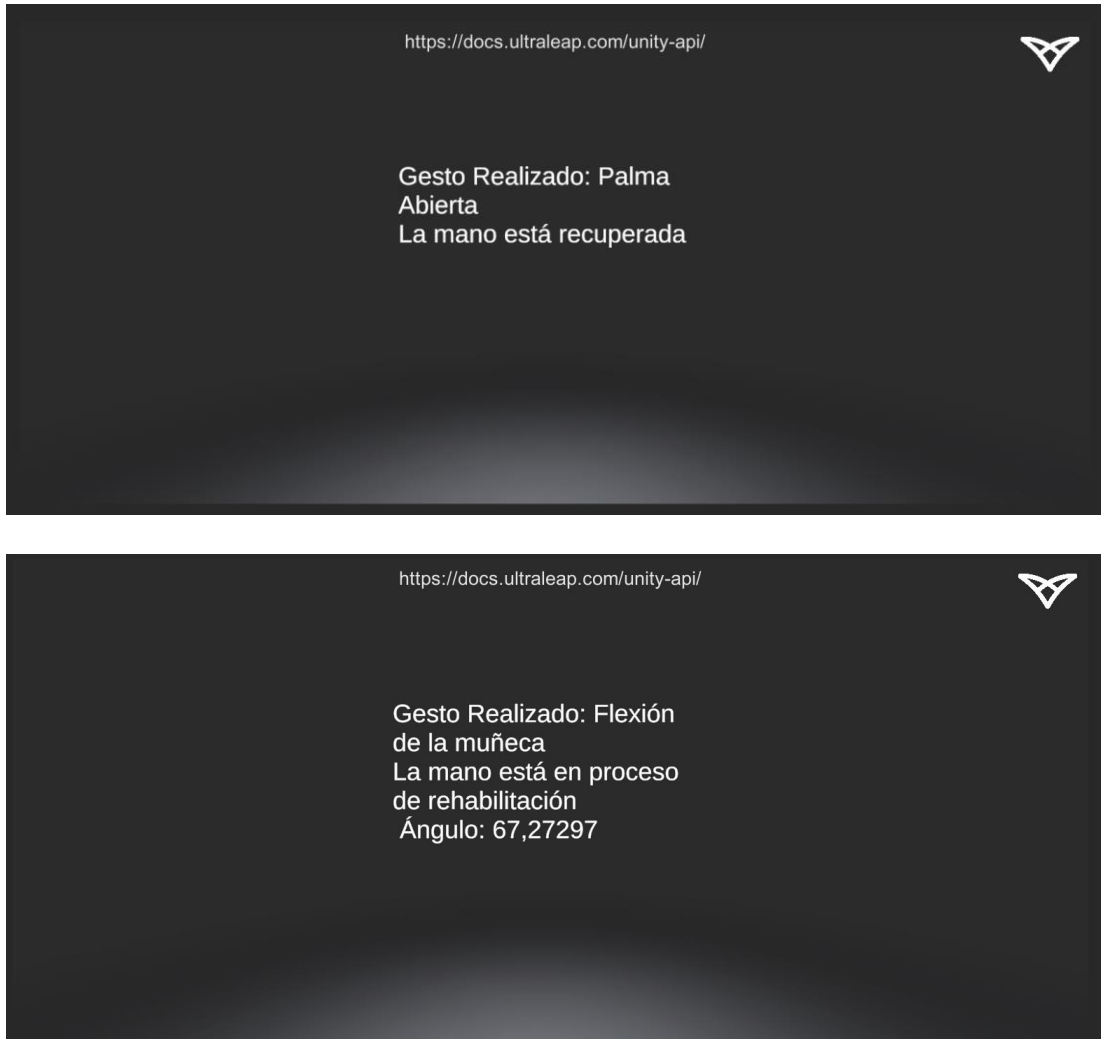


Figura 24. Avance de la rehabilitación

Elaborado por: Miguel Bonilla

Este mensaje proporciona información sobre el progreso del usuario en su proceso de rehabilitación en relación con el gesto específico objetivo. Este proceso se repite para cada uno de los cinco gestos, asegurando una evaluación y seguimiento individualizado para cada gesto específico.

3.2.4.8. Prueba del sistema

Para evaluar adecuadamente el progreso en la rehabilitación, se optó por obtener distancias y ángulos. El uso de estos valores permite determinar la efectividad de la rehabilitación en los gestos utilizados en el sistema. Para ello, se realizaron pruebas utilizando el 20% de los datos del dataset final, es decir, 450 datos. Estos datos contienen valores útiles para calcular las distancias y ángulos entre posiciones. Con el fin de obtener valores referenciales, se realizaron gráficos descriptivos para cada uno de los gestos, mostrando cómo las distancias y ángulos funcionan como puntos referenciales para medir el proceso de rehabilitación.

Palma abierta

La relación entre la distancia y el ángulo en el gesto de la palma abierta se encuentra en la **Figura 22**.

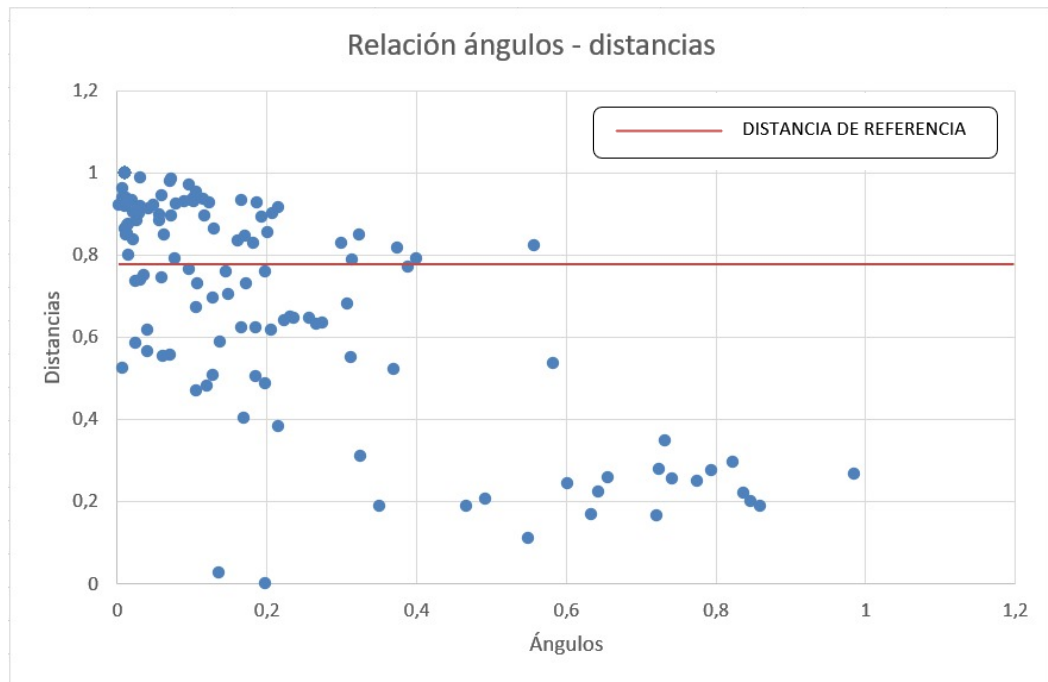


Figura 25. Relación entre distancia y ángulo del gesto palma abierta

Realizado por: Miguel Bonilla

El gráfico muestra la relación existente entre la distancia y el ángulo al realizar el gesto “Palma Abierta”. En el eje vertical se representa la distancia, mientras que en el eje horizontal se muestra el ángulo de inclinación de la mano. Mediante esto se puede analizar que:

- Cuando se observan distancias altas con muy poca inclinación o sin inclinación, es posible inferir que la mano está recuperada o ha alcanzado un buen nivel de funcionalidad. Esto significa que el paciente puede extender su mano considerablemente sin necesidad de realizar movimientos excesivos de la muñeca.
- Si se encuentran distancias y ángulos bajos, esto puede indicar que la persona posee una mano más pequeña, ya que los movimientos de extensión son limitados. También existe el caso en que el paciente se encuentre en las etapas iniciales de la rehabilitación o aún esté en proceso de mejorar la funcionalidad de la mano.
- En cuanto a las distancias y ángulos altos, esto sugiere que hay personas que tienen una distancia mayor entre su muñeca y dedo medio. En este caso, pueden estar en las primeras etapas de la rehabilitación, donde se trabaja en mejorar la extensión de la mano y la muñeca para lograr un rango completo de movimiento.

Puño

La relación entre la distancia y las muestras en el gesto del puño se encuentra en la **Figura 23.**

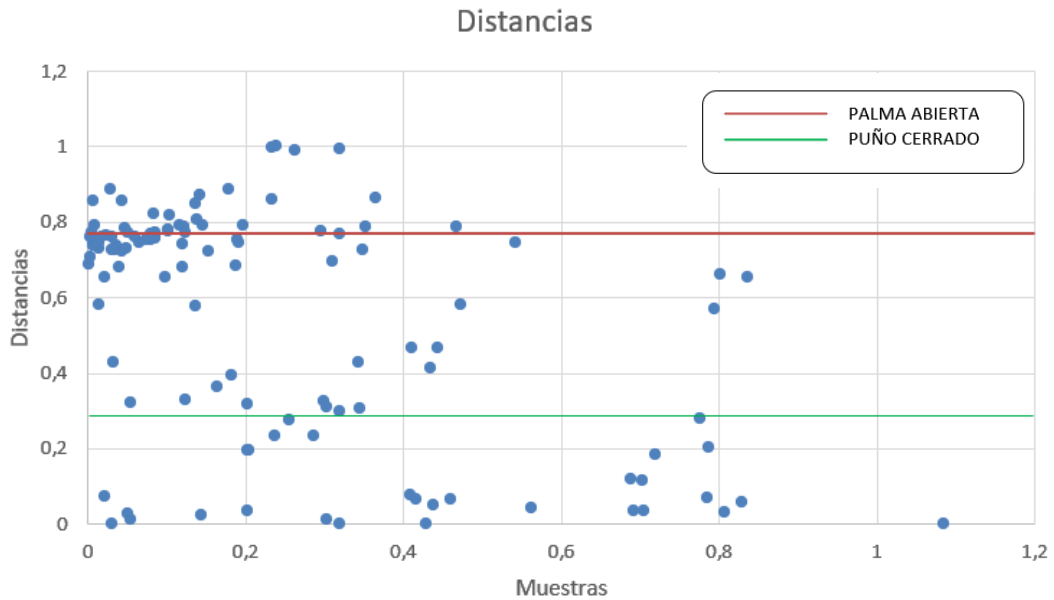


Figura 26. Relación distancia - muestras del puño

Realizado por: Miguel Bonilla

Basándose en la información proporcionada, el gráfico muestra una relación entre distancias y muestras. Después del análisis se han establecido tres rangos específicos para interpretar el estado de rehabilitación de los pacientes.

- **Valores por debajo del límite del valor de puño cerrado:** Estas muestras corresponden a pacientes cuyo proceso de rehabilitación se considera finalizado. La distancia entre las muestras en este rango indica que los pacientes han alcanzado un nivel de recuperación que les permite realizar el gesto de puño cerrado con una distancia consistente y cercana al límite definido.

- **Valores en medio del valor de la palma abierta y el puño cerrado:** Las muestras en este rango representan pacientes que se encuentran en proceso de rehabilitación. La distancia entre las muestras sugiere que están en un punto intermedio entre la palma abierta y el puño cerrado, lo que indica una mejora gradual en su capacidad para realizar ambos gestos.
- **Valores iguales o superiores al valor de la palma abierta:** Las muestras en este rango corresponden a pacientes que están empezando su proceso de rehabilitación. La distancia entre las muestras indica que su capacidad para realizar el gesto de palma abierta está aún en desarrollo y se encuentra por encima del umbral establecido.

Flexión de la muñeca

El uso de ángulos para verificar el avance al realizar la flexión de la muñeca se lo puede ver en la **Figura 24**.

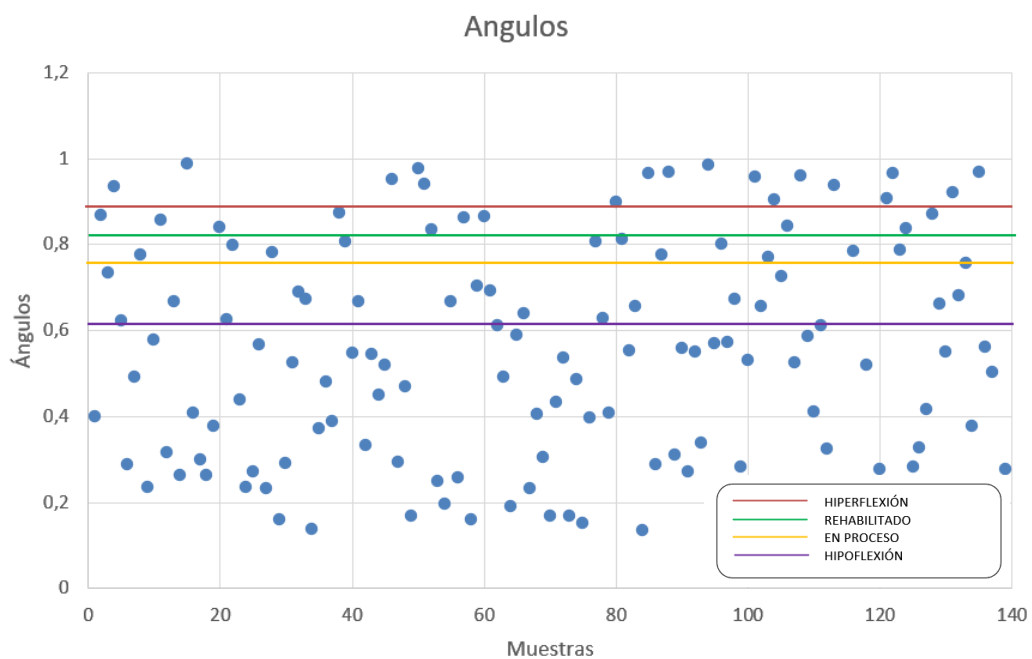


Figura 27. Ángulos de flexión de la mano

Realizado por: Miguel Bonilla

Al realizar una rehabilitación, se puede utilizar el valor del ángulo de la mano derecha como un indicador para evaluar la recuperación y la funcionalidad motora de un paciente. Para ello se estableció valores para establecer el rango que el paciente debe de tener.

- Cuando el valor del ángulo es menor a 60 grados, se considera que el paciente está en una etapa inicial de recuperación. Esto sugiere que la movilidad de la mano derecha puede estar limitada y se requiere una atención y terapia adecuadas para mejorar la funcionalidad y fortalecer los músculos afectados.
- Si el valor del ángulo se encuentra en el rango de 60 a 80 grados, se indica que el paciente está en proceso de recuperación. En esta fase, es probable que haya progresos significativos en la movilidad y funcionalidad de la mano derecha, aunque aún pueden existir ciertas limitaciones y áreas de mejora.
- Cuando el valor del ángulo se encuentra entre 80 y 85 grados, se considera que el paciente se ha recuperado correctamente. Esto indica que la movilidad de la mano derecha ha mejorado sustancialmente y el paciente puede realizar una variedad de actividades cotidianas con mayor facilidad.
- Sin embargo, si el valor del ángulo supera el umbral de 85 grados, se sugiere que el paciente puede estar enfrentando problemas en la movilidad de la mano derecha. Es posible que persistan limitaciones o dificultades funcionales, lo que podría requerir una evaluación y terapia más específica para abordar las áreas problemáticas.

Extensión de la muñeca

El uso de ángulos para verificar el avance al realizar la extensión de la muñeca se lo puede ver en la **Figura 25**.

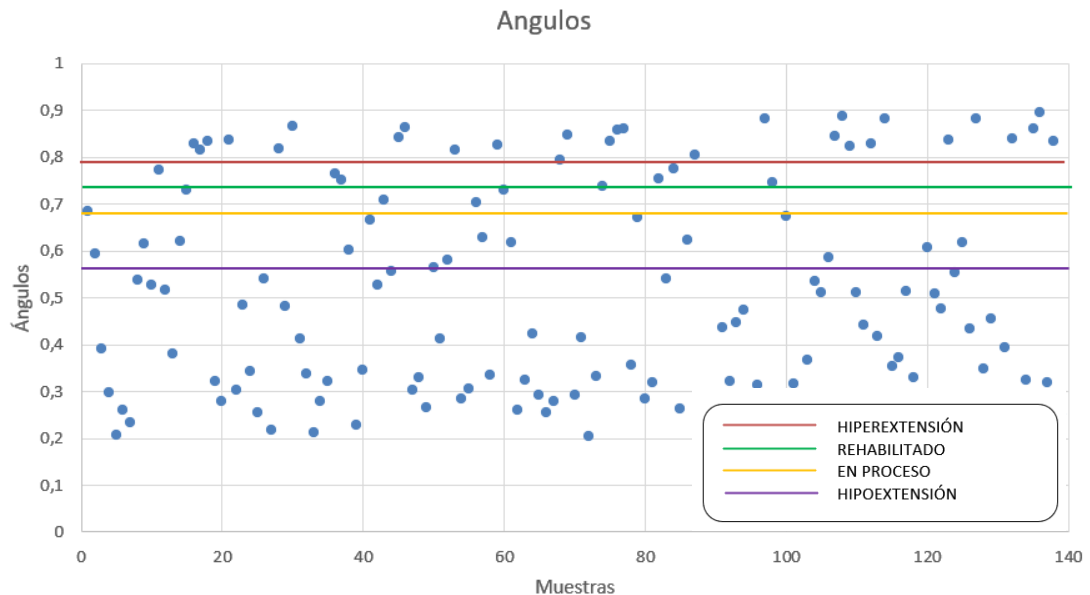


Figura 28. Ángulos de extensión por muestra

Realizado por: Miguel Bonilla

En esta representación gráfica, se muestra la relación entre el ángulo de extensión de la muñeca y los datos de muestra correspondientes. La gráfica permite observar que, al igual que en el caso de la flexión, el ángulo de extensión cuenta con valores que permiten verificar si la extensión es realizada correctamente o no. Los valores de la extensión se basan en el ángulo de extensión:

- El ángulo de extensión es menor a 60 grados, esto indica que el paciente se encuentra en una etapa inicial de rehabilitación.

- Cuando el ángulo de extensión se encuentra en el rango de 60 a 80 grados, esto sugiere que el paciente está en proceso de recuperación.
- Una vez que el ángulo de extensión alcanza el rango de 80 a 85 grados, esto indica que el paciente ha logrado una recuperación adecuada en la movilidad de la muñeca.
- Sin embargo, si el ángulo de extensión supera el umbral de 85 grados, esto podría indicar posibles dificultades en la movilidad de la mano derecha.

Pinza entre dedo medio y pulgar

La realización entre la distancia y la muestra de la pinza se encuentra en la **Figura 26**.

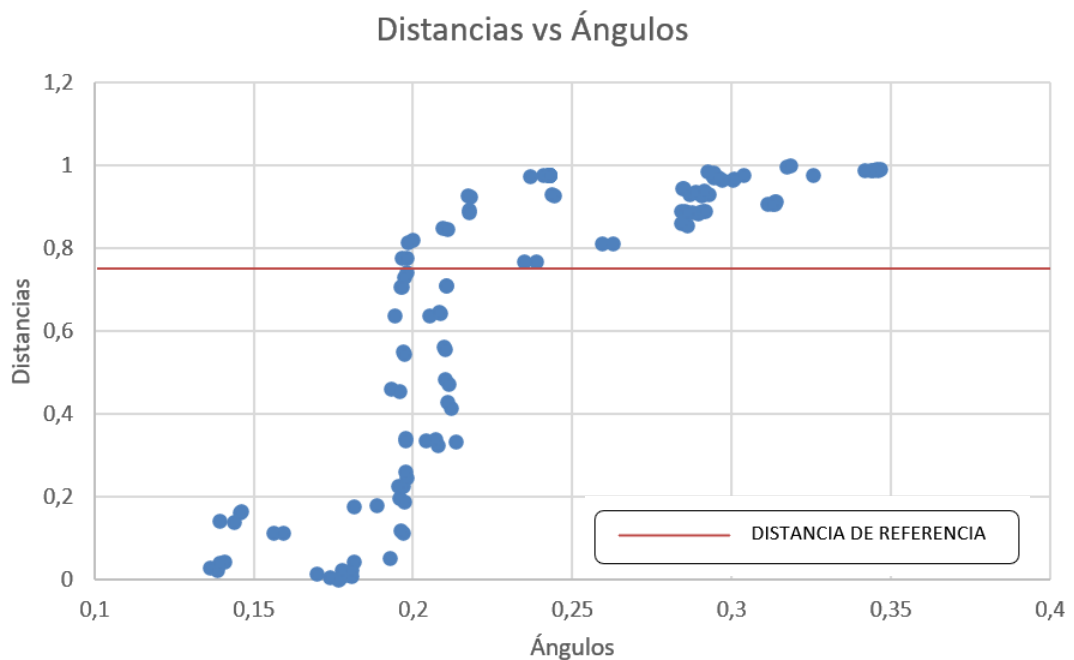


Figura 29. Relación distancia con ángulos del gesto de la pinza

Elaborado por: Miguel Bonilla

El gráfico muestra una relación entre distancias y ángulos en el contexto de la rehabilitación de pacientes. A continuación, se presenta una interpretación técnica considerando diferentes combinaciones de distancias y ángulos:

- **Angulo bajo y distancia alta:** Si el ángulo es bajo y la distancia es alta, es un indicativo de progreso positivo en la rehabilitación del paciente.
- **Angulo bajo y distancia también baja:** Cuando tanto el ángulo como la distancia son bajos, se infiere que el paciente se encuentra en proceso de rehabilitación.
- **Angulo y distancias altos:** Si el ángulo es alto y los valores de distancia son también altos, esto sugiere que el paciente ha finalizado su proceso de rehabilitación.

CAPÍTULO IV. – CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- En cuanto a la identificación de procesos para la rehabilitación de la mano derecha, el uso de instrumentos como las entrevistas resulta esencial. Estas entrevistas permiten conocer acerca del progreso del paciente a medida que avanza en su rehabilitación. La información obtenida de estas entrevistas facilita cuantificar el avance logrado, proporcionando así una medida tangible del progreso alcanzado.
- Para concluir, la elección de utilizar un sistema de reconocimiento de gestos ya existente ha sido acertada. Su puesta en marcha ha demostrado ser altamente beneficiosa para la recolección de datos de los gestos de la mano derecha. La eficiencia con la que este sistema captura y almacena la información ha simplificado el proceso de obtención de datos. Esto permite obtener un conjunto de datos estructurado y preciso.
- En conclusión, la elección de una RNA como modelo para reconocimiento de gestos de la mano derecha ha sido acertada debido a su alta precisión, capacidad de adaptación a los patrones de los datos y su potencial para brindar enfoques más personalizados en los tratamientos. La RNA ha demostrado ser una herramienta valiosa para verificar los gestos realizados por el paciente.
- En definitiva, el diseño de un sistema con una interfaz interactiva para el monitoreo de la rehabilitación de la mano derecha ofrece un enfoque innovador. Este sistema proporciona una manera innovadora de abordar el proceso de recuperación. Gracias a la visualización de mediciones precisas y la retroalimentación en tiempo real, se logra motivar al paciente. De manera adicional, se mantiene un seguimiento continuo que promueve una participación más activa en el proceso de recuperación.

4.2. Recomendaciones

- Para una segunda versión del sistema se recomienda realizar una exploración más profunda en el campo de los gestos utilizados en la rehabilitación de la mano derecha. Esto permitirá incluir una mayor variedad de gestos y mejorar la precisión del sistema al momento de dar el resultado sobre el avance del paciente.
- Es recomendable realizar pruebas del sistema de adquisición de datos para asegurar su funcionamiento óptimo. Estas pruebas son esenciales para asegurar que el sistema está operando de manera adecuada y sin errores. Si no se encuentra funcionando correctamente puede surgir problemas como captura incompleta o incorrecta de los datos, pérdida de información, entre otros.
- Es altamente recomendable utilizar la librería Keras para el desarrollo del modelo neuronal. Keras ofrece una interfaz intuitiva y simplifica en gran medida la construcción, entrenamiento y evaluación del modelo. Su uso amigable y su amplia gama de funciones integradas permiten una implementación eficiente y eficaz del modelo neuronal. Al aprovechar las capacidades de Keras, los desarrolladores pueden enfocarse en el diseño y la optimización del modelo sin preocuparse por detalles complejos de implementación.
- Se recomienda que el sistema sea utilizado en entornos clínicos y terapéuticos. Esto permitirá a los profesionales de la salud ofrecer una atención más personalizada y efectiva a sus pacientes. Además, posibilitará que el fisioterapeuta esté al tanto de las actividades realizadas por el paciente. Así podrá verificar el correcto uso del sistema.

REFERENCIAS BIBLIOGRÁFICAS

- [1] L. F. Sánchez Taco and I. Iglesias Navarro, “SISTEMA DE REHABILITACIÓN PARA PERSONAS CON DEFICIENCIA DEL CONTROL MOTOR DE MIEMBRO SUPERIOR EN UN ENTORNO DE REALIDAD AUMENTADA,” 2022.
- [2] A. F. Flores Taramuel, “ESTIMACIÓN DE POSICIÓN DE LA MUÑECA A TRAVÉS DE SEÑALES ELECTROMIOGRÁFICAS,” Universidad Técnica del Norte, Ibarra, 2019.
- [3] P. B. Shull, S. Jiang, Y. Zhu, and X. Zhu, “Hand Gesture Recognition and Finger Angle Estimation via Wrist-Worn Modified Barometric Pressure Sensing,” *IEEE TRANSACTIONS ON NEURAL SYSTEMS AND REHABILITATION ENGINEERING*, vol. 27, no. 4, 2019, doi: 10.1109/TNSRE.2019.2905658.
- [4] C. García Iván, O. Cruz Héctor, S. López Marco Antonio, and O. E. Cigarroa-Mayorga, “Sistema de detección Omnidireccional de Gestos Corporales para Entornos Virtuales,” *rvp.ieee.org.mx*, 2019, Accessed: Jan. 26, 2023. [Online]. Available: <http://rvp.ieee.org.mx/assets/ponencia-rocc-152.pdf>
- [5] Organización Panamericana de la Salud, “Dispositivos Médicos,” <https://www.paho.org/es/temas/dispositivos-medicos>, 2022.
- [6] N. Carvajal Tello, W. González Marmolejo, and A. Segura Ordoñez, “Design and technological development of a therapeutic device for physical rehabilitation in a healthcare environment,” *Salud Uninorte*, vol. 35, no. 2, pp. 250–263, May 2020, doi: 10.14482/sun.35.2.617.1.
- [7] G. Pardo *et al.*, “Seguimiento del Paciente en Rehabilitación con Kinect 2.0,” 2019. [Online]. Available: <http://hdl.handle.net/10757/648838>
- [8] B. Toalumbo and R. Nogales, “Hand Gesture Recognition using Leap Motion Controller, Infrared Information, and Deep Learning Framework,” 2022.
- [9] H. Avilez Carpinteri, M. López Sánchez, and J. G. González Serna, “Generación de código para identificar gestos de la mano mediante la cámara Intel RealSense SR 300 y Project Gesture,” 2022.
- [10] S. Ameer, A. Ben Khalifa, and M. S. Bouhlel, “Hand-Gesture-Based Touchless Exploration of Medical Images with Leap Motion Controller,” in *Proceedings of the 17th International Multi-Conference on Systems, Signals and Devices, SSD 2020*, Institute of Electrical and Electronics Engineers Inc., Jul. 2020, pp. 1116–1121. doi: 10.1109/SSD49366.2020.9364244.
- [11] H. Brock, S. Sabanovic, K. Nakamura, and R. Gomez, “Robust Real-Time Hand Gestural Recognition for Non-Verbal Communication with Tabletop Robot Haru,” Sep. 2020.

- [12] H. Walugembe, C. Phillips, J. Requena-Carrión, and T. Timotijevic, “Gesture Recognition in Leap Motion Using LDA and SVM,” 2019.
- [13] D. G. Alonso, A. Teyseyre, A. Soria, and L. Berdun, “Hand gesture recognition in real world scenarios using approximate string matching,” *Multimed Tools Appl*, vol. 79, no. 29–30, pp. 20773–20794, Aug. 2020, doi: 10.1007/s11042-020-08913-7.
- [14] M. Weiss Cohen and D. Regazzoni, “Hand rehabilitation assessment system using leap motion controller,” *AI Soc*, vol. 35, no. 3, pp. 581–594, Sep. 2020, doi: 10.1007/s00146-019-00925-8.
- [15] R. E. Nogales and M. E. Benalcázar, “Hand Gesture Recognition Using Automatic Feature Extraction and Deep Learning Algorithms with Memory,” *Big Data and Cognitive Computing*, vol. 7, no. 2, p. 102, May 2023, doi: 10.3390/bdcc7020102.
- [16] M. Chávez, “TECNOLOGÍA DE INFORMACIÓN Y COMUNICACIÓN (TICS),” 2019.
- [17] L. Z. M. I. Joyanes Aguilar, *PROGRAMACION EN C C++ JAVA Y UML 2ED*, Segunda. México: Programas Educativos S.A., 2014.
- [18] L. Lopez Roman, *Programacion Estructurada En Lenguaje C*, Primera Reimpresión. México D.F: ALFAOMEGA Grupo Editor, 2006.
- [19] S. Badaró, L. Javier Ibañez, and M. J. Agüero, “Sistemas Expertos: Fundamentos, Metodologías y Aplicaciones,” 2013.
- [20] S. I. Mariño and C. R. Primorac, “PROPUESTA METODOLÓGICA PARA DESARROLLO DE MODELOS DE REDES NEURONALES ARTIFICIALES SUPERVISADAS PROPOSED METHODOLOGY FOR DEVELOPING SUPERVISED ARTIFICIAL NEURAL NETWORKS MODELS,” 2016.
- [21] P. Meseguer González and R. López de Mántaras Badia, *Inteligencia artificial*. Madrid: Consejo Superior de Investigaciones Científicas, 2017.
- [22] F. R. M. L. S. Herrera, “Sistema inteligente de selección de componentes y presupuesto de computadoras (PC-WORLD),” Oct. 2018.
- [23] T. A. BYRD and R. D. HAUSER, “Expert systems in production and operations management: research directions in assessing overall impact,” *Int J Prod Res*, vol. 29, no. 12, pp. 2471–2482, 1991, doi: 10.1080/00207549108948097.
- [24] K. Krishnakumar, “Intelligent Systems For Aerospace Engineering-An Overview,” 2002.
- [25] J. F. Imre J. Rudas, “Intelligent Systems,” 2006.

- [26] J. Carlos *et al.*, “Parametrización de terapia física para falanges de la mano humana Parameterization of Physical Therapy for Phalanges of the Human Hand,” *Research in Computing Science*, vol. 149, no. 8, pp. 2020–555.
- [27] S. M. Hall, F. G. Lee, S. J. Wallace, and N. F. Miller, “Technique for Reconstruction of Midcarpal Instability Associated with Lunotriquetral Coalition,” *Plast Reconstr Surg Glob Open*, vol. 10, no. 7, p. E4430, Jul. 2022, doi: 10.1097/GOX.0000000000004430.
- [28] M. González-Hernández, J. Gudiño-Lau, J. Guedea-Salazar, E. Hernández-Barón, M. Durán-Fonseca, and D. Vélez-Díaz, “Diseño y construcción de un dispositivo mecatrónico para terapia ocupacional para niño,” Nov. 2018, [Online]. Available: <https://repository.uaeh.edu.mx/revistas/index.php/xikua/issue/archive>
- [29] M. Khayatzadeh, A. H. Mehraban, and M. Kamali, “Facilitators of implementing occupation based practice among Iranian occupational therapists: A qualitative study,” 2016. [Online]. Available: <http://mjiri.iums.ac.ir>
- [30] M. Ferre, I. Galiana, R. Wirz, and N. Tuttle, “Haptic Device for Capturing and Simulating Hand Manipulation Rehabilitation,” Nov. 2011.
- [31] J. L. Rueda-Arreguín, C. R. Torres-San Miguel, V. Ramírez-Vela, G. Urriolagoitia-Sosa, and L. Martínez-Sáez, “Simulation by finite element method of intramedullary telescopic systems for rehabilitation of patients with osteogenesis imperfecta,” *Revista Mexicana de Ingeniería Biomedica*, vol. 40, no. 2, May 2019, doi: 10.17488/RMIB.40.2.2.
- [32] A. Torres Velásquez, C. A. Díaz León, L. F. García Muriel, and M. L. Toro Hernandez, “El aporte de la biomecánica y la Ingeniería en rehabilitación en la Ingeniería Biomédica de la eia-ces,” vol. 2, Nov. 2002.
- [33] A. D. Gaines and R. Davis-Floyd, “ON BIOMEDICINE,” 2003.
- [34] J. Goecks, V. Jalili, L. M. Heiser, and J. W. Gray, “How Machine Learning Will Transform Biomedicine,” *Cell*, vol. 181, no. 1. Cell Press, pp. 92–101, Apr. 02, 2020. doi: 10.1016/j.cell.2020.03.022.
- [35] R. Cantero Téllez, *Terapia de mano basada en el razonamiento y la práctica clínica*. Andalucía: UNIVERSIDAD INTERNACIONAL DE ANDALUCÍA, 2020.
- [36] A. Kapandji, *Fisiología Articular Esquemas Comentados de Mecánica Humana I*, 6th ed. Madrid: Editorial Panamericana, 2006.
- [37] Ultraleap, “Leap Motion Controller TM,” 2023. [Online]. Available: <https://www.ultraleap.com/product/vr-developer-mount/>
- [38] E. Montijano Muñoz, “La cámara Microsoft Kinect,” 2017.
- [39] R. Nogales, M. E. Benalcazar, B. Toalumbo, A. Palate, R. Martinez, and J. Vargas, “Construction of a Dataset for Static and Dynamic Hand Tracking Using

- a Non-invasive Environment,” in *Advances in Intelligent Systems and Computing*, Springer Science and Business Media Deutschland GmbH, 2021, pp. 185–197. doi: 10.1007/978-981-33-4565-2_12.
- [40] J. S. Nolasco Valenzuela, *Python Aplicaciones Prácticas*, Ra-Ma. Madrid: RA-MA Editorial, 2018.
- [41] E. M. Rojas, “Machine Learning: análisis de lenguajes de programación y herramientas para desarrollo,” 2020.
- [42] M. José and G. Rodríguez, “Estudio comparativo entre las metodología ágiles y las metodologías tradicionales para la gestión de proyectos software,” 2015.
- [43] S. Raschka and V. Mirjalili, *Python machine learning : machine learning and deep learning with python, scikit-learn, and tensorflow 2*, 3rd ed. 2019.
- [44] ultraleap, “Ultraleap,” https://docs.ultraleap.com/unity-api/?_gl=1*1xkszhe*_ga*MTYzNzY5NDc4MC4xNjgyMTc0NTE4*_ga_5G8B19JLWG*MTY4NjYwMjEzNS4xMy4wLjE2ODY2MDIxNDIuNTMuMC4w, 2021.

ANEXOS

Anexo A

Entrevista realizada

N°	Pregunta
1	¿Cuál es el enfoque terapéutico que utiliza para la rehabilitación de la mano derecha?
2	¿Cuál es el papel de la educación del paciente en el proceso de rehabilitación de la mano derecha y cómo la integra en su enfoque terapéutico?
3	¿Qué consejos o pautas proporciona a los pacientes para promover la rehabilitación de la mano derecha fuera de las sesiones de terapia?
4	¿Cuáles son algunas de las condiciones o lesiones comunes que pueden afectar el movimiento de extensión y flexión de la muñeca en la mano derecha?
5	¿Con qué frecuencia ha tratado a pacientes con restricciones en la extensión y flexión de la muñeca en la mano derecha en su práctica clínica?
6	¿Qué estrategias o ejercicios específicos recomienda para fortalecer y mejorar la función de los músculos y articulaciones involucradas en la extensión y flexión de la muñeca en la mano derecha?
7	¿Cuál es la importancia de la pinza realizada entre el dedo pulgar y el dedo medio en la funcionalidad de la mano derecha y la realización de actividades cotidianas?
8	¿Qué tipos de lesiones o condiciones médicas pueden afectar el movimiento de pinza realizada entre el dedo pulgar y el dedo medio en la mano derecha y cuáles son los desafíos más comunes que enfrentan los pacientes en su proceso de rehabilitación?
9	¿Qué desafíos comunes enfrentan los pacientes durante la rehabilitación del movimiento de puño en la mano derecha y cómo aborda esos desafíos en su práctica clínica?

10	¿Cómo aborda la rehabilitación del movimiento de puño en la mano derecha en pacientes con diferentes niveles de habilidad o limitaciones físicas, como aquellos con movilidad reducida o debilidad muscular en esa área?
11	¿Cuáles son algunos de los indicadores o señales que utiliza para evaluar la progresión de la rehabilitación de la mano derecha en sus pacientes y ajustar su enfoque terapéutico en consecuencia?
12	¿Cómo se adapta el plan de rehabilitación de la mano derecha de un paciente a medida que progresa en su proceso de recuperación?

Elaborado por: Miguel Bonilla

Anexo B

Encuesta realizada

Nº	Ítem				
1	¿Ha participado alguna vez en un programa de rehabilitación que utilice movimientos virtuales para la recuperación de su mano derecha?				
	Sí		No		
2	¿Cuál es su nivel de familiaridad con el concepto de rehabilitación virtual en el contexto del movimiento de la mano derecha?				
	Muy familiarizado/a	Familiarizado/a	Neutral	Poco familiarizado/a	Nada familiarizado/a
3	¿Cuál es su nivel de conocimiento sobre el uso de sistemas informáticos?				
	Experto	Alto	Medio	Bajo	Ninguno
4	¿Considera que un sistema de rehabilitación virtual que monitoree el movimiento realizado sería beneficioso para su proceso de rehabilitación de la mano derecha?				
	Totalmente de acuerdo	De acuerdo	Neutral	En desacuerdo	Totalmente en desacuerdo
5	¿Qué nivel de confianza tendría en un sistema inteligente para monitorear su rehabilitación de la mano derecha?				
	Muy confiado/a	Confiable	Neutral	Poco confiado/a	Nada confiado/a

6	¿Qué tan cómodo/a se sentiría utilizando un sistema de rehabilitación virtual que monitoree el movimiento realizado durante su proceso de rehabilitación? (Seleccionar una opción en una escala del 1 al 5), donde 1 es "Nada cómodo/a" y 5 es "Muy cómodo/a".				
	1	2	3	4	5
7	¿Qué beneficios espera obtener de un sistema de rehabilitación virtual que monitoree el movimiento realizado en su proceso de rehabilitación de la mano derecha? (Seleccionar todas las opciones que apliquen)				
	Mejora de la función motora	Mayor motivación para la rehabilitación	Mayor monitoreo del progreso	Mayor adherencia al programa de rehabilitación	Mayor comprensión de mi progreso de rehabilitación
8	¿Qué desventajas ve en el uso de un sistema de rehabilitación virtual que monitoree el movimiento realizado en su proceso de rehabilitación de la mano derecha? (Seleccionar todas las opciones que apliquen)				
	Falta de privacidad o seguridad de los datos	Posibles errores o inexactitudes en el monitoreo	Dificultad en la comprensión o uso del sistema	Posibles efectos secundarios o lesiones	Costo económico
9	¿Qué características o funcionalidades considera importantes en un sistema de rehabilitación virtual que monitoree el movimiento realizado en su proceso de rehabilitación de la mano derecha? (Seleccionar todas las opciones que apliquen)				
	• Interfaz de usuario amigable				
	• Precisión en el monitoreo de los movimientos				
	• Retroalimentación en tiempo real				
	• Personalización del programa de rehabilitación				
	• Monitoreo del progreso a lo largo del tiempo				
• Integración con otros sistemas de rehabilitación					

	<ul style="list-style-type: none"> • Acceso a registros o reportes de progreso 	
	<ul style="list-style-type: none"> • Facilidad de uso en el hogar 	
	<ul style="list-style-type: none"> • Asesoramiento o guía por parte de profesionales de la salud 	
10	¿Qué desafíos potenciales identifica en el uso de un sistema de rehabilitación virtual que utilice el movimiento para la verificación de la rehabilitación de la mano derecha? (Seleccionar todas las opciones que considere válidas)	
	<ul style="list-style-type: none"> • Posible falta de precisión o confiabilidad en el monitoreo del movimiento virtual 	
	<ul style="list-style-type: none"> • Dificultad o falta de familiaridad en el uso de la tecnología 	
	<ul style="list-style-type: none"> • Limitaciones en la compatibilidad con otros dispositivos o sistemas de rehabilitación 	
	<ul style="list-style-type: none"> • Preocupaciones de privacidad y seguridad de los datos 	
	<ul style="list-style-type: none"> • Costo o accesibilidad del sistema 	
11	¿Considera que el precio de recibir rehabilitación para la mano derecha en un consultorio privado es excesivo?	
	<table border="1" style="width: 100%; text-align: center;"> <tr> <td>Sí</td> <td>No</td> </tr> </table>	Sí
Sí	No	

Elaborado por: Miguel Bonilla

Anexo C

Alfa de Cronbach en personal de salud

ENCUESTADOS	ITEMS					SUMA
	1	2	3	4	5	
E1	5	3	5	5	5	23
E2	4	4	4	4	3	19
E3	3	3	3	3	4	16
E4	4	3	5	5	5	22
E5	3	2	4	4	4	17
E6	3	2	2	3	3	13
VARIANZA	0,556	0,472	1,139	0,667	0,667	
SUMATORIA DE VARIANZAS	3,500					
VARIANZA TOTAL DEL INSTRUMENTO	11,889					

Elaborado por: Miguel Bonilla

Aplicación de Alfa de Cronbach

Fórmula:

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma s_1^2}{s_T^2} \right]$$

Dónde:

α = Coeficiente de confiabilidad = 0,882125 → 88,21 %

K = Número de ítems del instrumento = 5

Σs_1^2 = Sumatoria de las varianzas = 3,5

s_T^2 = Varianza total del instrumento = 11,889

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma s_1^2}{s_T^2} \right] = \frac{5}{5-1} \left[1 - \frac{3,5}{11,889} \right] = \frac{5}{4} [1 - 0,2943] = \frac{5}{4} [0,7057] = 0,882125$$

Alfa de Cronbach en personal que utiliza computadoras

ENCUESTADOS	ITEMS					SUMA
	1	2	3	4	5	
E1	3	3	4	4	3	17
E2	3	4	5	4	5	21
E3	4	5	5	5	5	24
E4	4	3	5	5	4	21
E5	3	5	5	5	4	22
E6	4	5	4	5	5	23
E7	2	3	3	4	3	15
E8	4	3	5	4	4	20
E9	2	4	5	5	5	21
VARIANZA	0,617	0,765	0,469	0,247	0,617	
SUMATORIA DE VARIANZAS	2,716					
VARIANZA TOTAL DEL INSTRUMENTO	7,984					

Elaborado por: Miguel Bonilla

Aplicación de Alfa de Cronbach

Fórmula:

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma s_1^2}{s_T^2} \right]$$

Dónde:

α = Coeficiente de confiabilidad = 0,824875 → 82,48%

K = Número de ítems del instrumento = 5

Σs_1^2 = Sumatoria de las varianzas = 2,716

s_T^2 = Varianza total del instrumento = 7,984

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma s_1^2}{s_T^2} \right] = \frac{5}{5-1} \left[1 - \frac{2,716}{7,984} \right] = \frac{5}{4} [1 - 0,3401] = \frac{5}{4} [0,6599] = 0.824875$$

Alfa de Cronbach en pacientes

ENCUESTADOS	ITEMS					SUMA
	1	2	3	4	5	
E1	2	4	4	3	2	15
E2	3	5	3	4	3	18
E3	4	5	5	4	4	22
E4	3	4	4	2	2	15
E5	4	4	5	4	5	22
E6	5	5	5	5	5	25
E7	4	5	5	5	5	24
E8	3	2	3	2	4	14
E9	5	5	4	4	4	22
E10	4	4	5	4	5	22
E11	5	4	5	4	5	23
E12	4	4	5	5	5	23
E13	4	3	4	3	5	19
E14	5	5	5	4	3	22
E15	3	4	3	4	4	18
VARIANZA	0,782	0,693	0,622	0,827	1,129	

SUMATORIA DE VARIANZAS	4,053	
VARIANZA DE LA SUMA DE LOS ÍTEMS	11,529	

Elaborado por: Miguel Bonilla

Aplicación de Alfa de Cronbach

Fórmula:

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma s_1^2}{s_T^2} \right]$$

Dónde:

α = Coeficiente de confiabilidad = 0,810625 → 81,06%

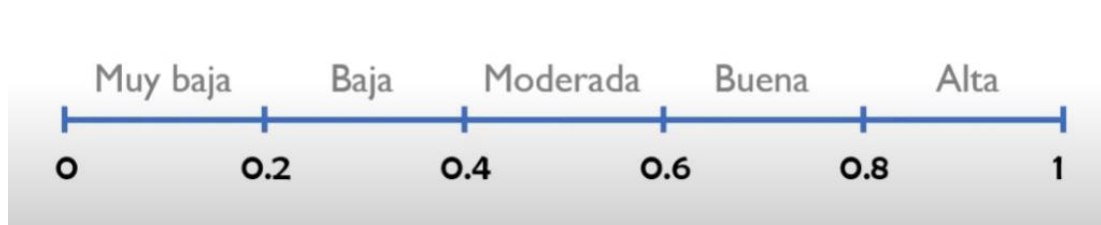
K = Número de ítems del instrumento = 5

Σs_1^2 = Sumatoria de las varianzas = 4,053

s_T^2 = Varianza total del instrumento = 11,529

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma s_1^2}{s_T^2} \right] = \frac{5}{5-1} \left[1 - \frac{4,053}{11,529} \right] = \frac{5}{4} [1 - 0,3515] = \frac{5}{4} [0,6485] = 0.810625$$

Línea de referencia para verificar el valor del coeficiente Alfa de Cronbach



Elaborado por: Miguel Bonilla

Kuder-Richardson en personal de salud

Individuos	PREGUNTAS		
	P1	P2	
1	1	1	2
2	0	0	0
3	0	1	1
4	0	1	1
5	1	1	2
6	1	1	2
Totales	3	5	
p	0,50	0,83	
q	0,50	0,17	
p*q	0,25	0,14	
$\Sigma(p*q)$	0,39		
σ^2	0,67		
K	2		

Elaborado por: Miguel Bonilla

Aplicación de Kuder-Richardson

Fórmula:

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma pq}{\sigma^2} \right]$$

Dónde:

α = Coeficiente de confiabilidad = 0,8358 → 83,58%

K = Número de ítems del instrumento = 2

q = Porcentaje de personas que respondieron afirmativamente = 0,5

p = Porcentaje de personas que respondieron negativamente = 0,5

σ^2 = Varianza total del instrumento = 0,67

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma pq}{\sigma^2} \right] = \frac{2}{2-1} \left[1 - \frac{0,39}{0,67} \right] = \frac{2}{1} [1 - 0,5821] = 2[0,4179] = 0,8358$$

Kuder-Richardson en personal que usa computador

	PREGUNTAS		
Individuos	P1	P2	
1	0	1	1
2	1	1	2
3	0	0	0
4	1	1	2
5	0	0	0
6	0	1	1
7	1	1	2
8	0	1	1
9	0	0	0
Totales	3	6	
p	0,33	0,67	
q	0,67	0,33	
p*q	0,22	0,22	
$\Sigma(p*q)$	0,44		
σ^2	0,75		
K	2		

Elaborado por: Miguel Bonilla

Aplicación de Kuder-Richardson

Fórmula:

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma pq}{\sigma^2} \right]$$

Dónde:

α = Coeficiente de confiabilidad = 0,8268 → 82,68%

K = Número de ítems del instrumento = 2

q = Porcentaje de personas que respondieron afirmativamente = 0,67

p = Porcentaje de personas que respondieron negativamente = 0,33

σ^2 = Varianza total del instrumento = 0,75

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma pq}{\sigma^2} \right] = \frac{2}{2-1} \left[1 - \frac{0,44}{0,75} \right] = \frac{2}{1} [1 - 0,5866] = 2[0,4134] = 0,8268$$

Kuder-Richardson en pacientes

Individuos	PREGUNTAS		
	P1	P2	
1	0	1	1
2	0	1	1
3	1	1	2
4	0	1	1
5	1	1	2
6	0	0	0
7	0	1	1
8	0	1	1
9	1	1	2
10	0	1	1
11	0	1	1
12	0	1	1
13	0	1	1
14	0	1	1
15	1	1	2
Totales	4	14	
p	0,27	0,93	
q	0,73	0,07	
p*q	0,20	0,06	
$\Sigma(p*q)$	0,26		
σ^2	0,44		
K	2		

Realizado por: Miguel Bonilla

Aplicación de Kuder-Richardson

Fórmula:

$$\alpha = \frac{K}{K - 1} \left[1 - \frac{\Sigma pq}{\sigma^2} \right]$$

Dónde:

α = Coeficiente de confiabilidad = 0,8182 → 81,82%

K = Número de ítems del instrumento = 2

q = Porcentaje de personas que respondieron afirmativamente = 0,73

p = Porcentaje de personas que respondieron negativamente = 0,27

σ^2 = Varianza total del instrumento = 0,44

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\Sigma pq}{\sigma^2} \right] = \frac{2}{2-1} \left[1 - \frac{0,26}{0,44} \right] = \frac{2}{1} [1 - 0,5909] = 2[0,4091] \\ = 0,8182$$

Anexo D

MANUAL DE USUARIO

INTRODUCCIÓN

La rehabilitación de la mano es un proceso crucial para aquellos que han experimentado lesiones o trastornos en esta área. Con el avance de la tecnología, surgen nuevas herramientas y enfoques para mejorar la eficacia y la calidad de la rehabilitación. En este contexto, la combinación de un sistema inteligente utilizando el Leap Motion Controller y motion capture ha demostrado ser una solución prometedora.

El Leap Motion Controller es un dispositivo de seguimiento de mano que utiliza sensores y cámaras infrarrojas para capturar y reconocer los movimientos de los dedos y la mano con gran precisión. Por otro lado, la tecnología de Motion capture permite registrar y analizar los movimientos en tiempo real, brindando una retroalimentación inmediata y detallada sobre el rendimiento del paciente.

Al combinar estos dos elementos, se puede desarrollar un sistema inteligente que monitorea y registra el movimiento de la mano durante las sesiones de rehabilitación.

Esto permite realizar un seguimiento preciso de los movimientos, evaluar el progreso del paciente y proporcionar una retroalimentación instantánea y personalizada.

El sistema inteligente utilizando el Leap Motion Controller y la captura de movimiento ofrece varias ventajas. En primer lugar, proporciona una representación visual del movimiento de la mano, lo que permite a los pacientes comprender y corregir su técnica de manera más efectiva. Además, al registrar y analizar los datos de movimiento, se pueden identificar patrones, evaluar el progreso a lo largo del tiempo y adaptar el plan de rehabilitación de manera más precisa.

REQUERIMIENTOS

HARDWARE

- Procesador: CPU con soporte para instrucciones SSE2.
- Memoria RAM: 4 GB o más.
- Espacio en disco: 10 GB de espacio libre.
- Tarjeta gráfica: Tarjeta gráfica compatible con DirectX 10 o posterior.
- Resolución de pantalla: 1280x720 o superior.

SOFTWARE

- Sistema operativo: Windows 7 SP1+, macOS 10.12+, Ubuntu 16.04+ (64-bit).

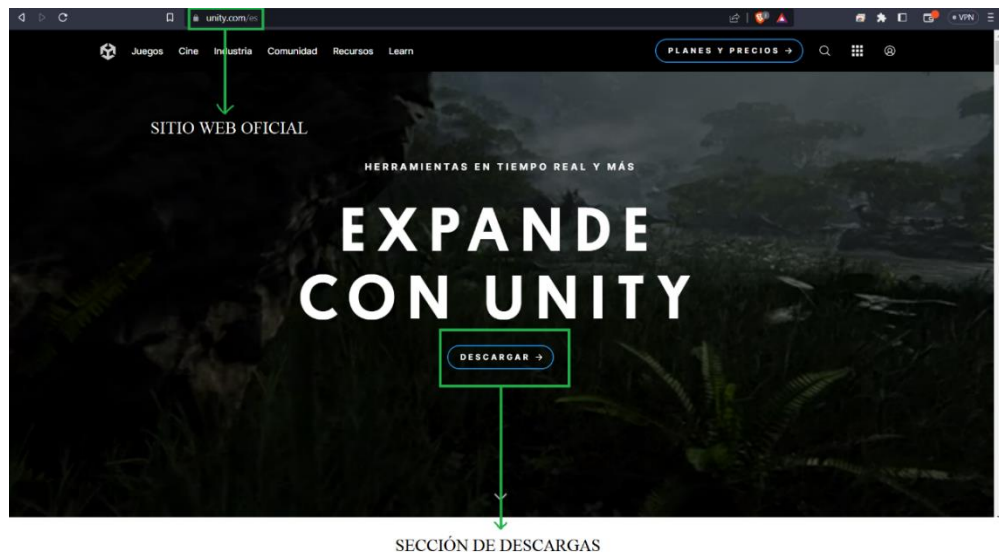
INSTALACIÓN

Para que el sistema funcione correctamente, es necesario descargar e instalar el software de seguimiento de Ultraleap, así como tener instalado Unity en el equipo. El software de seguimiento proporciona la capacidad de reconocer y rastrear los movimientos de la mano utilizando el Leap Motion Controller. Este software permite capturar datos precisos sobre la posición y los gestos de la mano, los cuales son

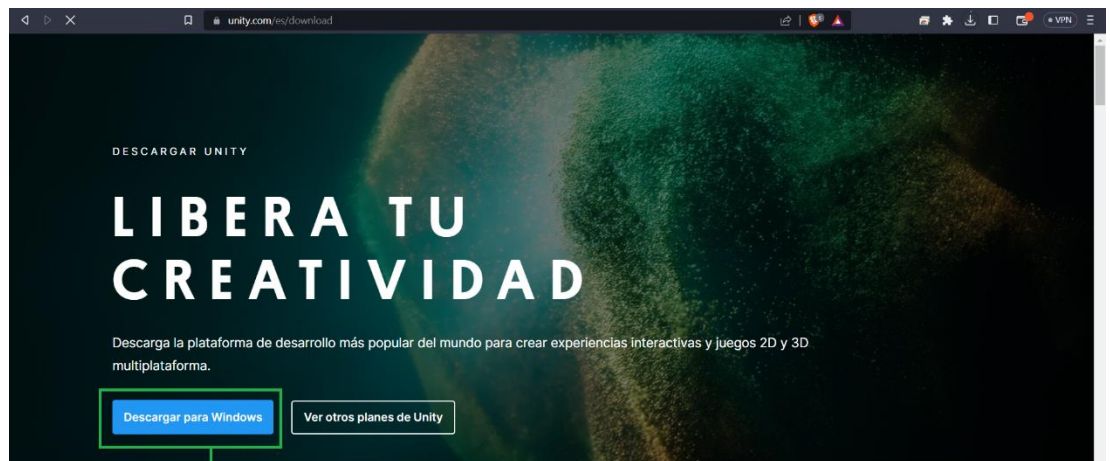
fundamentales para el funcionamiento del sistema. Por otro lado, Unity es una plataforma de desarrollo de juegos y aplicaciones ampliamente utilizada, que proporciona las herramientas necesarias para crear y visualizar entornos interactivos. Al tener instalado Unity, se puede aprovechar su funcionalidad y capacidades para integrar el seguimiento de Ultraleap en proyectos y aplicaciones, lo que permite interactuar y controlar el entorno virtual utilizando gestos y movimientos de la mano. En resumen, tanto el software de seguimiento de Ultraleap como Unity son componentes esenciales para el funcionamiento del sistema, ya que brindan las herramientas necesarias para capturar y utilizar los datos de seguimiento de la mano en entornos virtuales.

INSTALAR UNITY

- Para descargar Unity, accede al sitio web oficial de Unity y dirígete a la sección de descargas.

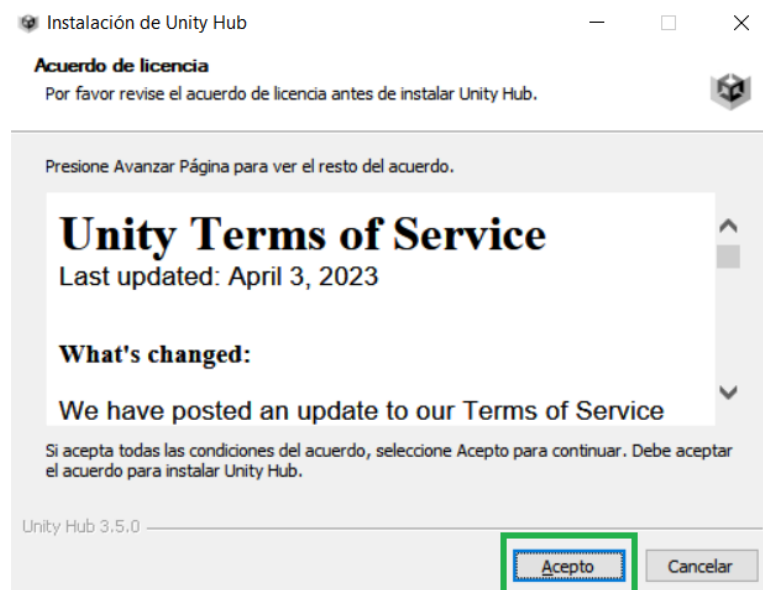


- Accede a la sección de descargas donde encontrarás el instalador disponible. Para un uso óptimo del sistema, se recomienda descargar Unity Hub 3.4.2.



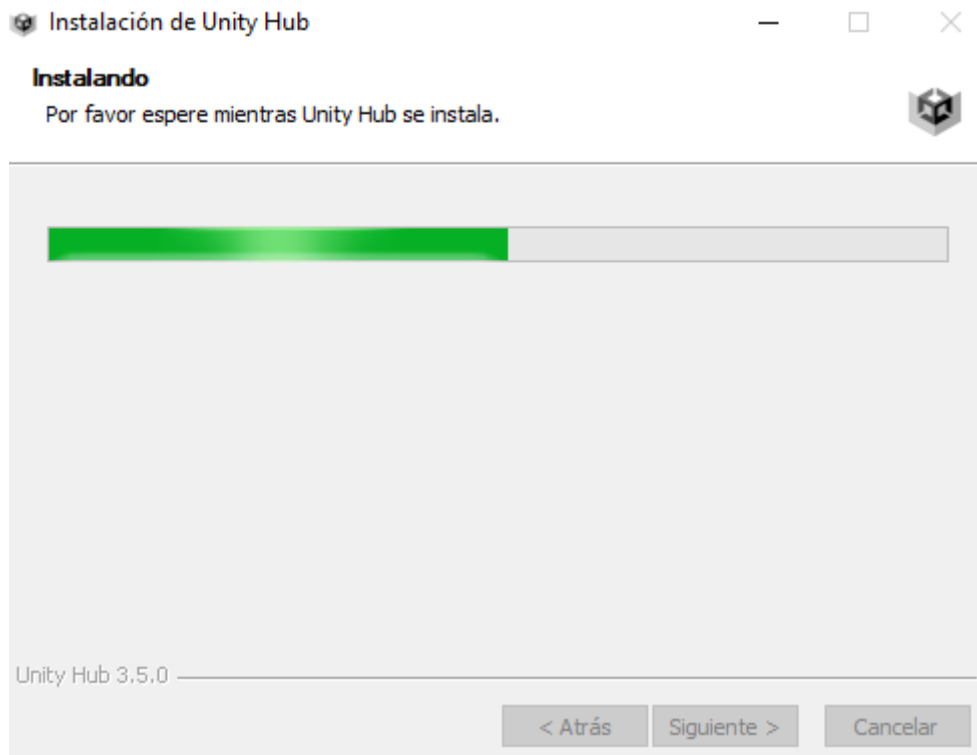
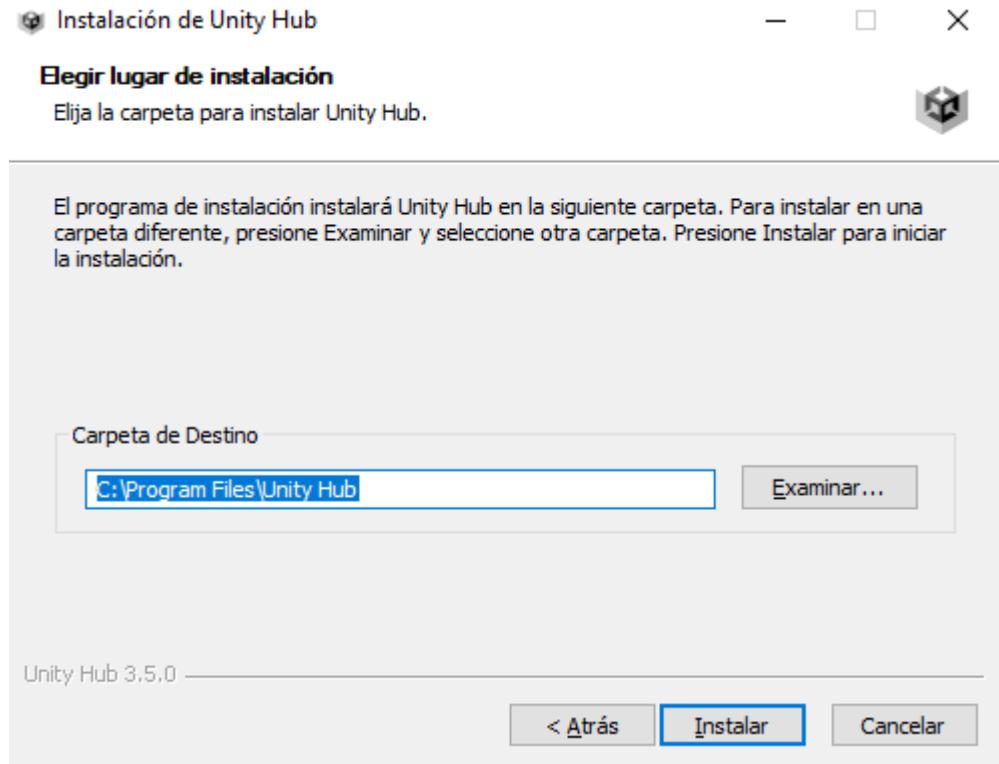
DESCARGAR UNITY

- Al descargar Unity, se presentará un instalador personalizado que se adapta a tu sistema operativo. Para continuar con la instalación, es necesario aceptar las condiciones del sistema.

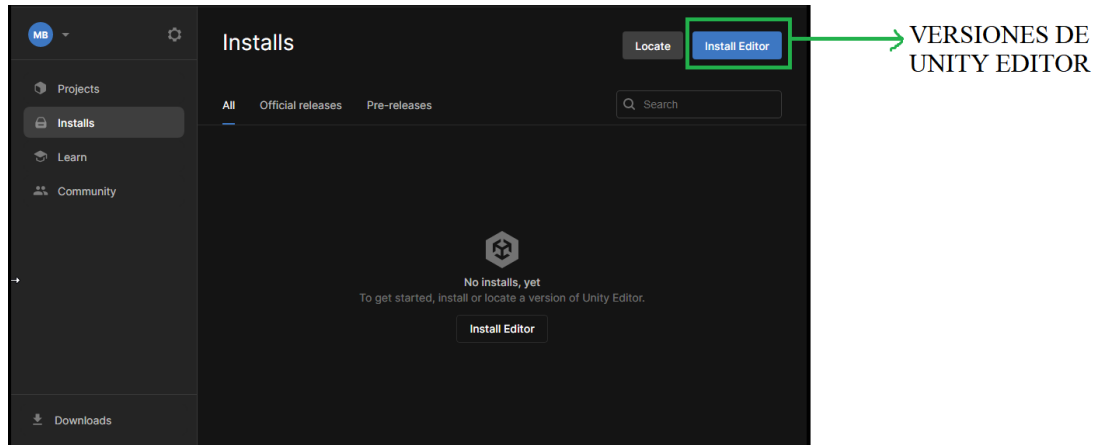


ACEPTAR CONDICIONES

- Instalación de Unity Hub siguiendo el proceso de instalación.



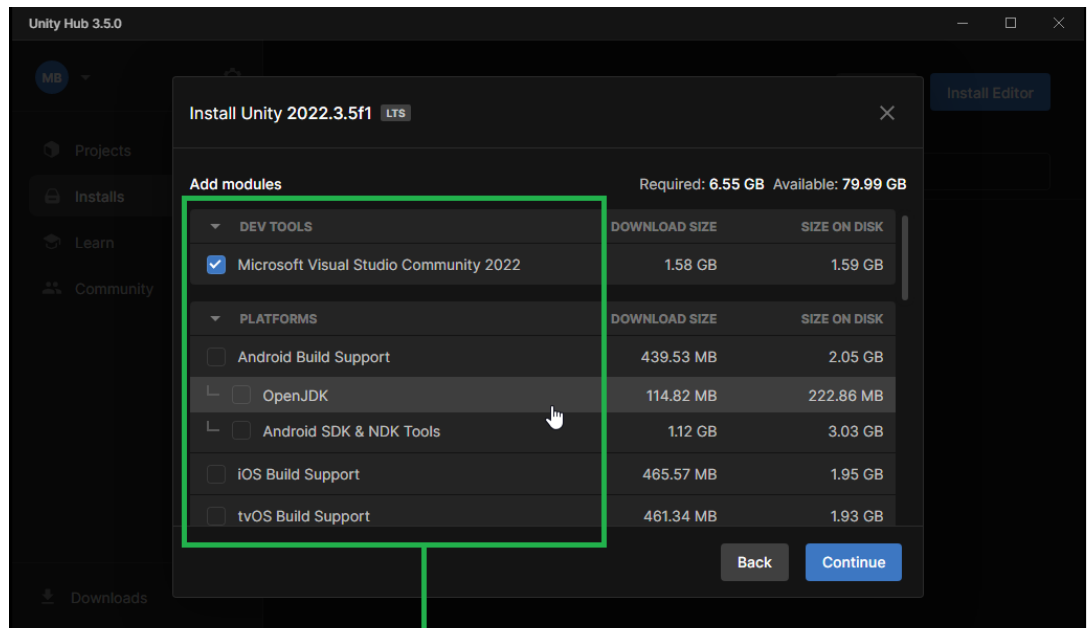
- Una vez completada la instalación de Unity Hub, se procede a instalar Unity Editor. En la sección de "Installs" se encuentra la opción para desplegar las versiones de Unity Editor disponibles para su instalación.



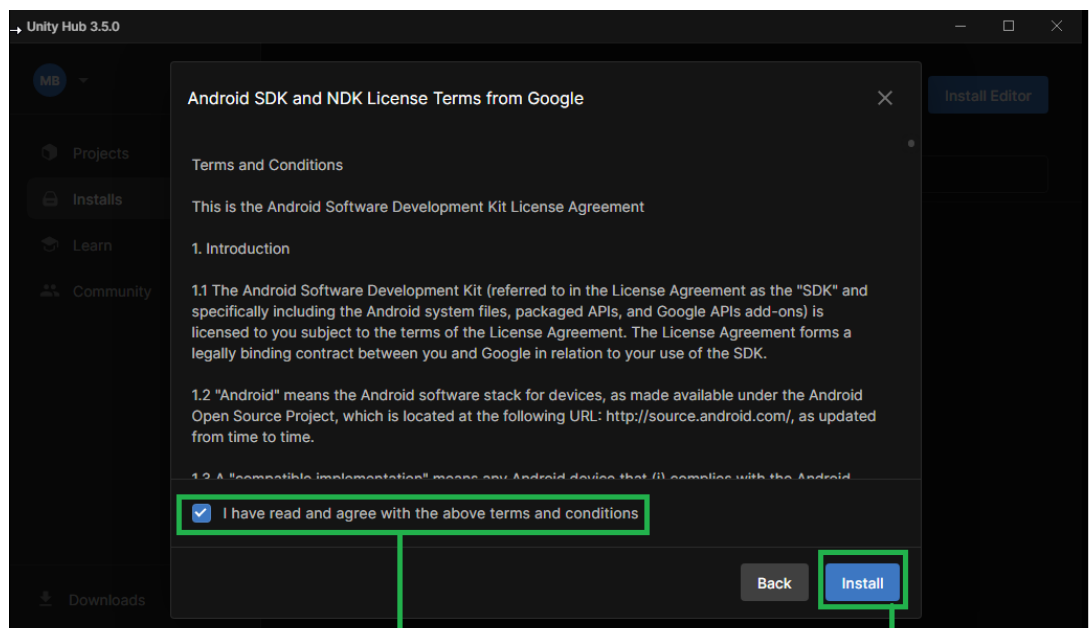
- Se debe seleccionar la versión específica de Unity Editor a instalar. Para garantizar la ejecución óptima de este proyecto, se recomienda instalar la versión 2021.3.26f1 de Unity Editor.



- Debes seleccionar la versión de Unity Editor y los componentes necesarios. Puedes elegir entre módulos, documentación y otros recursos adicionales según tus requerimientos.



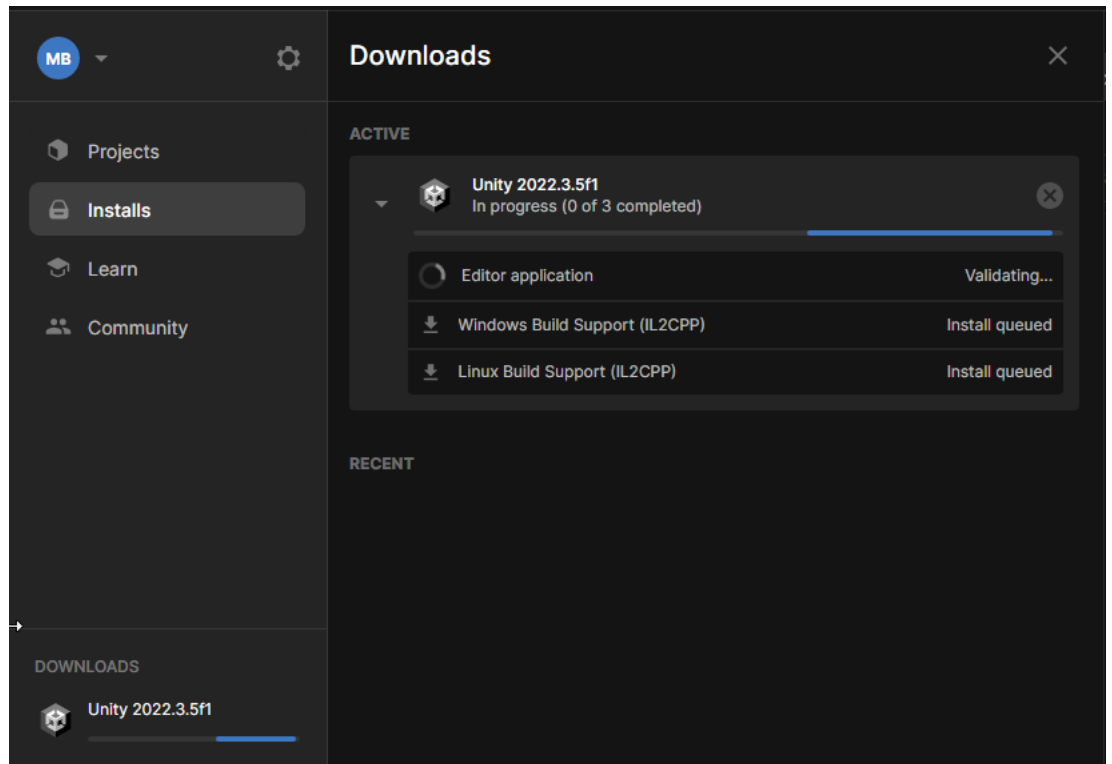
MÓDULOS DE INSTALACIÓN



ACEPTAR TÉRMINOS Y CONDICIONES

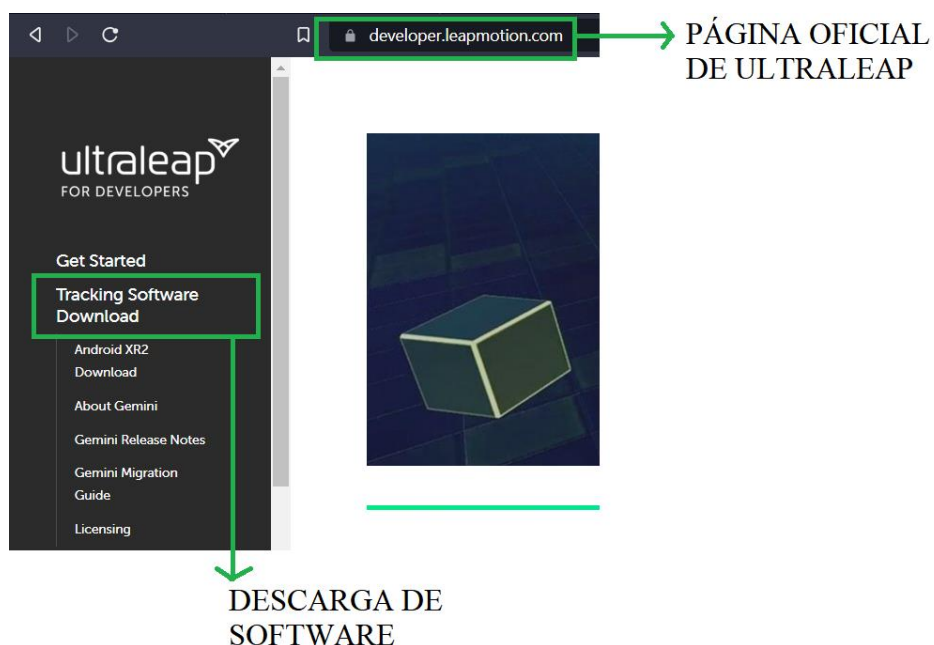
BOTÓN DE INSTALACIÓN

- Una vez finalizada la instalación de Unity Editor, es posible ejecutar los proyectos desarrollados en Unity sin ningún problema.

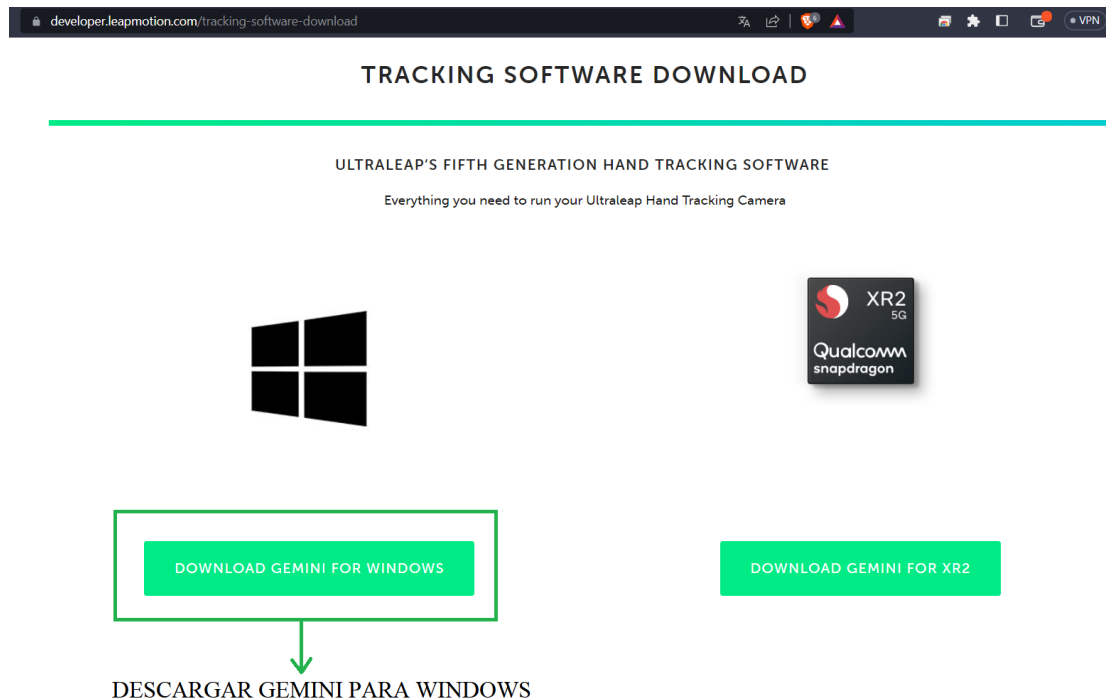


INSTALAR ULTRALEAP

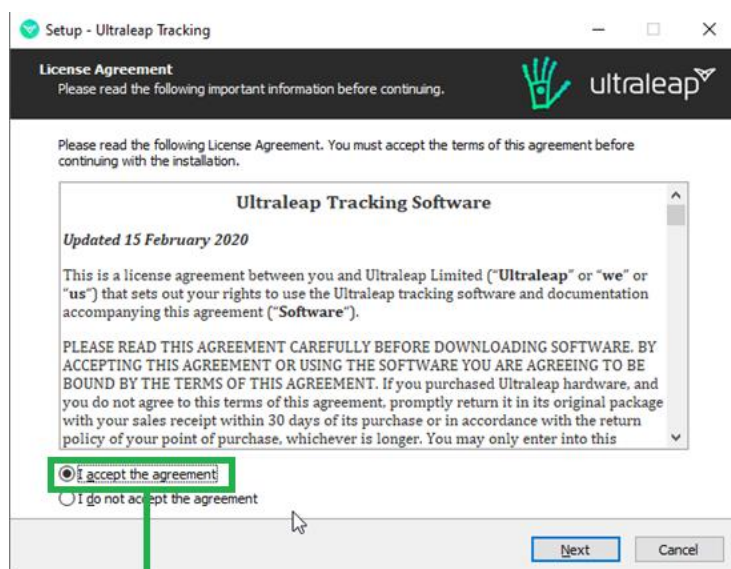
- Accede al sitio web oficial de Ultraleap y navega hacia la sección de descargas para obtener los recursos necesarios.



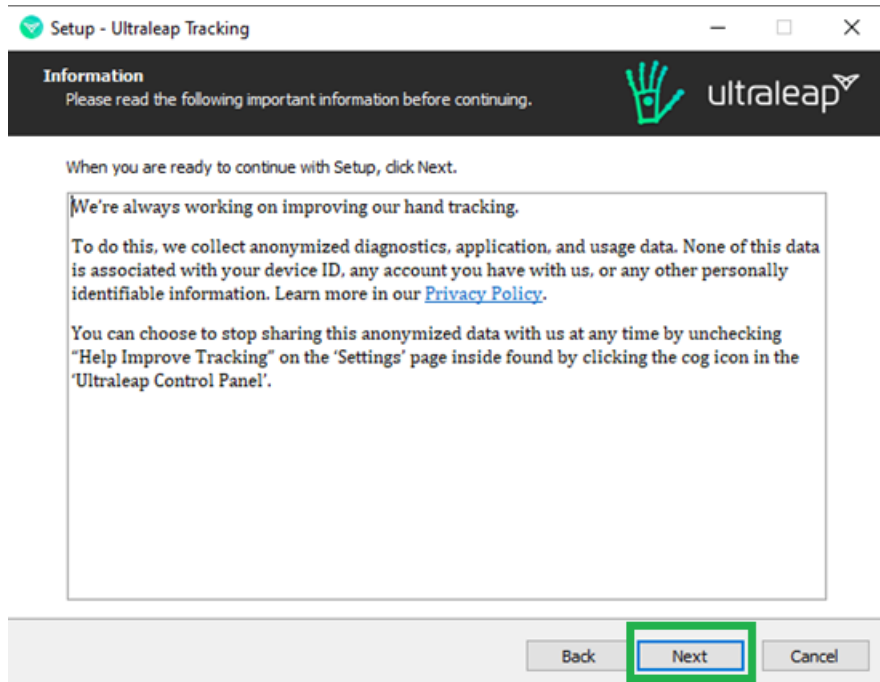
- Descarga la versión de Gemini específica para Windows desde el sitio web oficial de Ultraleap.



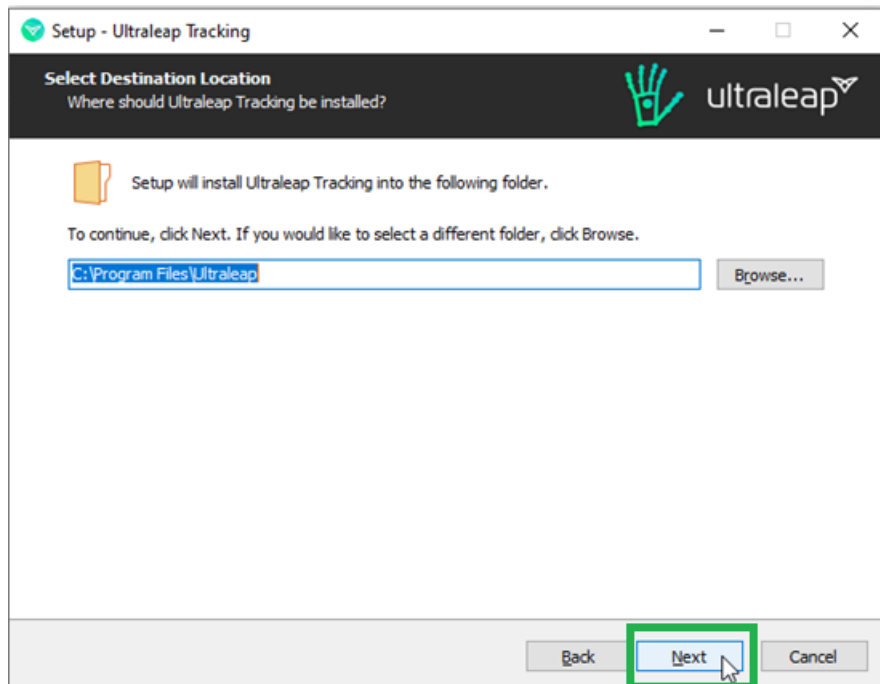
- Una vez descargado el instalador, ejecútalo y sigue las instrucciones proporcionadas para completar la instalación de Gemini.



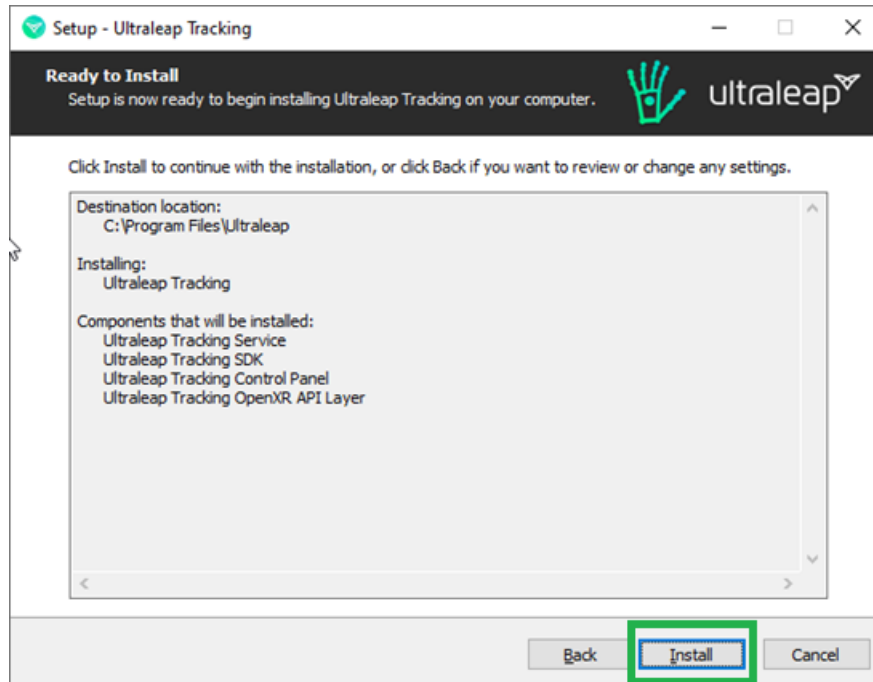
ACEPTAR TÉRMINOS Y CONDICIONES



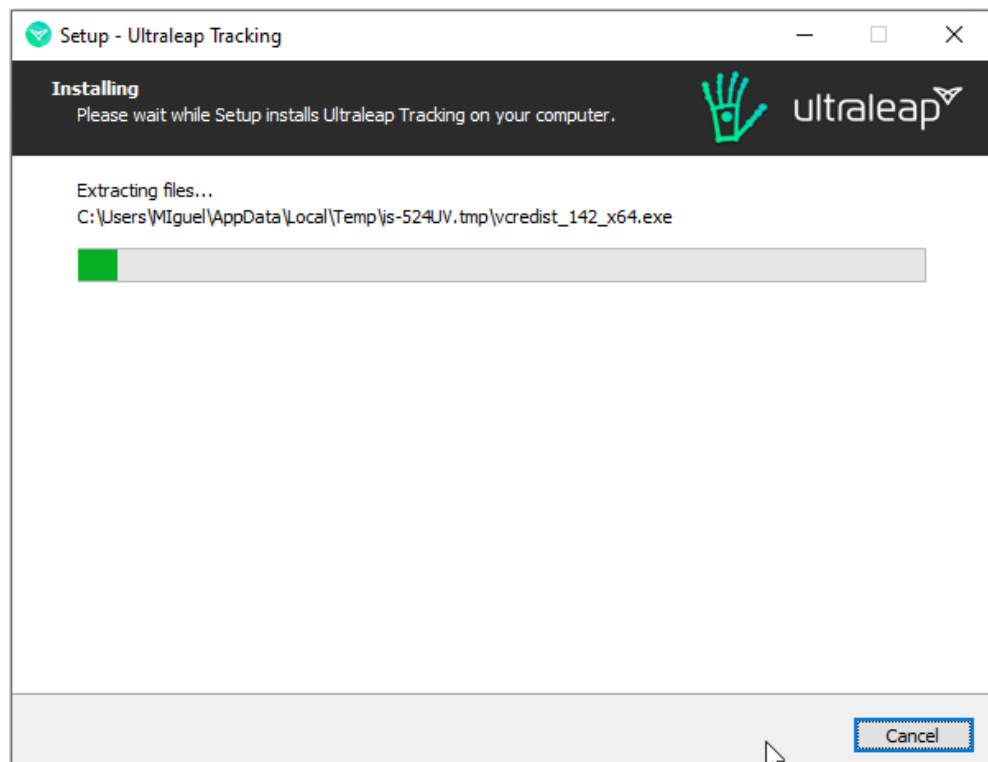
CONTINUAR



SIGUIENTE



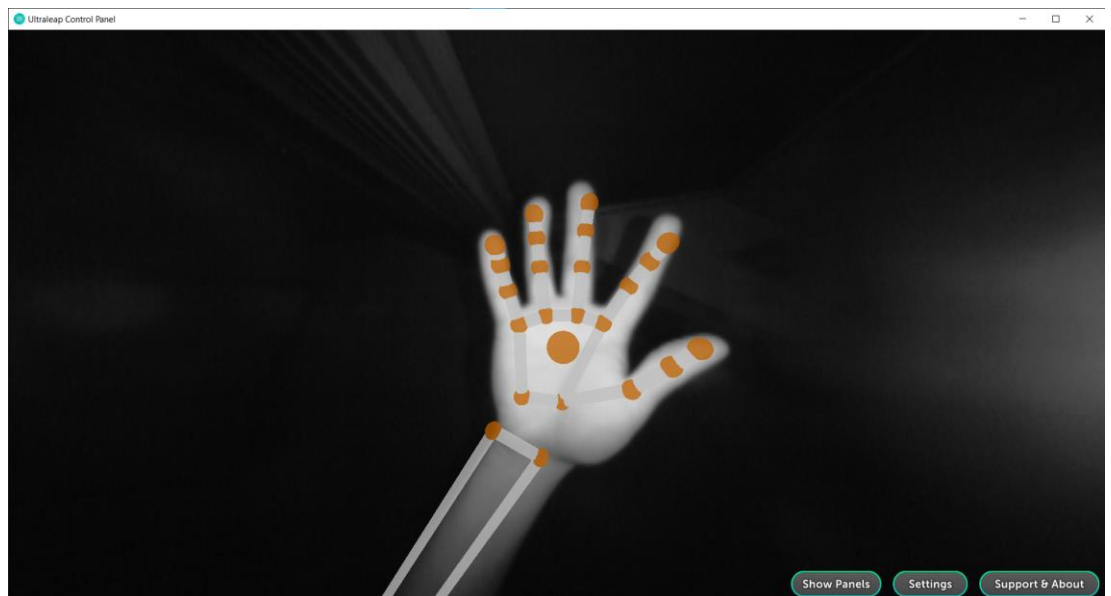
COMENZAR INSTALACIÓN



- Conecta el Leap Motion Controller a tu computador mediante un cable USB.



- Verifica la correcta instalación de Ultraleap utilizando Ultraleap Tracking para confirmar su funcionamiento adecuado.



Una vez finalizada la instalación de Unity y Leap Motion, es posible ejecutar el sistema de rehabilitación, lo que permitirá llevar a cabo el monitoreo de la rehabilitación de manera efectiva. Con esta configuración, los pacientes podrán interactuar con el sistema y realizar los gestos diseñados para su recuperación.

Anexo E

MANUAL TÉCNICO

Preprocesamiento de datos

- Importación de librerías necesarias para el preprocesamiento de datos.

```
#Importación de librerías
import pandas as pd
import numpy as np
import scipy.io
from scipy.ndimage import sobel
```

- Cargar el dataset mediante el uso de la librería pandas.

```
#Carga del dataset
datos = pd.read_csv('tabla.csv', sep=',')
```

- Se aplica la fórmula de normalización en cada fila del dataset.

```
#normalizado por filas
min_vals_1 = np.min(datos, axis=1)
max_vals_1 = np.max(datos, axis=1)
normalizado_filas = (datos.T - min_vals_1) / (max_vals_1 - min_vals_1)
```

Extracción de características

- Para la extracción de características se realizaron dos fases, en la fase 1 se realiza la división y subdivisión de ventanas. En la fase 2 se realiza los cálculos correspondientes a las características extraídas.

```
#Extracción de características
total_columnas = normalizado_filas.T.shape[1]
grupos = []
div = []
cont = 0
resultados_filas = pd.DataFrame()
# FASE 1
grupos = np.array_split(normalizado_filas.T, total_columnas // 70, axis=1)
divisiones = [np.array_split(grupo, 5, axis=1) for grupo in grupos]

# FASE 2
for i in range(len(grupos)):
    for j in range(np.shape(divisiones)[1]):
        divisiones_arr = np.array(divisiones[i][j])
        media = np.mean(divisiones_arr, axis=1)
        desviacion_estandar = divisiones_arr.std(axis=1)
        raiz_cuadrada_media = np.sqrt(np.abs(divisiones_arr)).mean(axis=1)
        varianza = np.sum(np.square(divisiones_arr), axis=1) / (len(divisiones_arr) - 1)
        resultados_filas[f'media_{i+1}_{j+1}'] = media
        resultados_filas[f'desviacion_{i+1}_{j+1}'] = desviacion_estandar
        resultados_filas[f'raiz_cuadrada_media_{i+1}_{j+1}'] = raiz_cuadrada_media
        resultados_filas[f'derivada_{i+1}_{j+1}'] = np.mean([np.diff(row) for row in divisiones_arr], axis=1)
        resultados_filas[f'varianza_{i+1}_{j+1}'] = varianza
```

- Con el dataset de las características extraídas se guarda un archivo delimitado por comas para el entrenamiento en el algoritmo.

```
#Guardado del dataset final
resultados_filas.to_csv('resultados_filas2.csv', sep=";", index=False)
```

Elección del Algoritmo

Red Neuronal Artificial (RNA)

- Se importan las librerías necesarias para el entrenamiento de la red neuronal.

```
#importar librerías necesarias
from sklearn.model_selection import KFold
from sklearn import datasets
from sklearn import metrics
import pandas as pd
import numpy as np
from keras.models import Sequential
from keras.layers import Dense
from keras.utils import to_categorical
from keras import regularizers
import os
```

- Se cargan los datos para el entrenamiento y las etiquetas.

```
# Carga del conjunto de datos y las etiquetas
dataset = pd.read_csv('resultados_filas2.csv', sep=';')
y = pd.read_csv('y.csv', sep=',')
X = dataset.values
y = to_categorical(y) |
```

- A continuación, se inicializa la validación cruzada con un KFold de 3 pliegues y se establece una lista para almacenar los resultados del entrenamiento. También se crea un directorio para guardar los modelos del entrenamiento.

```
# Creación de un objeto KFold para la validación cruzada con 3 pliegues (k = 3)
kfold = KFold(n_splits=3, shuffle=True, random_state=42)

# Lista para almacenar los resultados de las métricas de evaluación
accuracy_scores = []

# Carpeta para almacenar los modelos
model_folder = 'saved_models'
os.makedirs(model_folder, exist_ok=True)
```

- Se empieza con la validación cruzada y se establece el conjunto de entrenamiento y de prueba

```
# Realiza la validación cruzada
for i, (train_index, test_index) in enumerate(kfold.split(X)):
    # Divide el conjunto de datos en conjuntos de entrenamiento y prueba
    X_train, X_test = X[train_index], X[test_index]
    y_train, y_test = y[train_index], y[test_index]
```

- Se crea el modelo de la red neuronal y se realiza el entrenamiento. En el modelo también se aplica un regularizador L2 de 5.

```
# Crea un modelo de red neuronal
model = Sequential()
model.add(Dense(25, input_dim=X_train.shape[1], activation='relu'))
model.add(Dense(15, activation='relu'))
model.add(Dense(5, activation='relu', kernel_regularizer=regularizers.l2(5)))
model.add(Dense(y.shape[1], activation='softmax'))
```

- Se compila el modelo y se entrena en algoritmo en 1000 épocas, obteniendo la exactitud en el conjunto de entrenamiento.

```
# Compila el modelo
model.compile(optimizer='adam', loss='categorical_crossentropy')

# Entrena el modelo en el conjunto de entrenamiento
model.fit(X_train, y_train, epochs=10000, batch_size=10, verbose=0)

# Obtiene las predicciones en el conjunto de entrenamiento
y_train_pred = model.predict(X_train)
y_train_pred_classes = np.argmax(y_train_pred, axis=1)

# Calcula la exactitud en el conjunto de entrenamiento
train_accuracy = metrics.accuracy_score(np.argmax(y_train, axis=1), y_train_pred_classes)
```

- Finalizado el entrenamiento se realiza predicciones con el conjunto de pruebas y se obtiene la exactitud en el entrenamiento y en la prueba.

```
# Realiza predicciones en el conjunto de prueba
y_pred = model.predict(X_test)
y_pred_classes = np.argmax(y_pred, axis=1)

# Calcula la exactitud en el conjunto de prueba
test_accuracy = metrics.accuracy_score(np.argmax(y_test, axis=1), y_pred_classes)

# Imprime la exactitud de entrenamiento y prueba
print("Exactitud en entrenamiento {}: {:.2f}".format(i+1, train_accuracy))
print("Exactitud en prueba {}: {:.2f}".format(i+1, test_accuracy))
```

- Se calcula la métrica de evaluación y se guarda el modelo para cada validación.

```
# Calcula la métrica de evaluación (exactitud en este caso)
accuracy = metrics.accuracy_score(np.argmax(y_test, axis=1), y_pred_classes)
accuracy_scores.append(accuracy)

# Guarda el modelo para esta validación
model_filename = os.path.join(model_folder, 'model_{}.h5'.format(i))
model.save(model_filename)
print("Modelo {} guardado".format(i))
```

- Se calcula el promedio de las exactitudes y se imprime la exactitud promedio.

```
# Calcula el promedio de las métricas de evaluación
mean_accuracy = sum(accuracy_scores) / len(accuracy_scores)

# Imprime el resultado promedio
print("Exactitud promedio: {:.2f}".format(mean_accuracy))
```

K-Nearest Neighbors KNN

- Importar librerías necesarias para el entrenamiento.

```
#importar librerías necesarias
import numpy as np
import pandas as pd
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import joblib
from keras.utils import to_categorical
```

- Cargar el dataset y las etiquetas.

```
#Carga del dataset
dataset = pd.read_csv('resultados_filas2.csv', sep=';')
y = pd.read_csv('y.csv', sep=',')

x = dataset.values
y = to_categorical(y)
```

- Dividir en datos de entrenamiento y testeo. Seguido, instanciar KNN con un valor de vecino de 1. Al final, entrenar el modelo.

```
#División de datos de entrenamiento y testeo
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

#Instancia de KNN con un valor de 1 vecino
knn = KNeighborsClassifier(n_neighbors=1)

#entrenamiento del modelo
knn.fit(X_train, y_train)
```

- Guardar el modelo entrenado y realizar predicciones.

```
#Guardar el modelo entrenado
joblib.dump(knn, 'knn_model.pkl')

#realizar predicciones
y_pred = knn.predict(X_test)
```

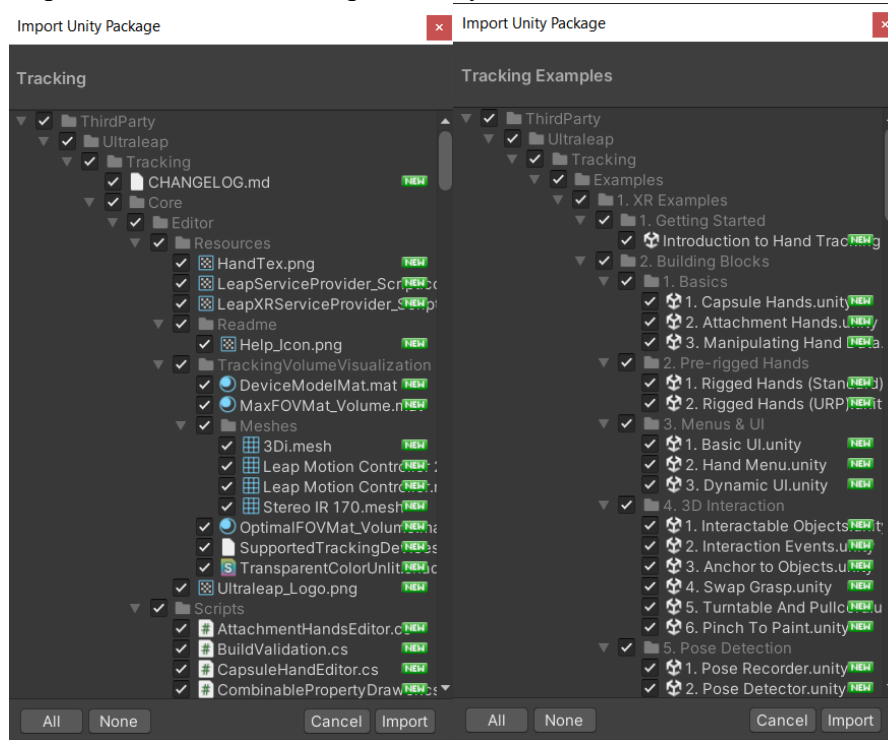
- Imprimir la precisión de entrenamiento y testeo.

```
#imprimir la precisión de entrenamiento
train_accuracy = accuracy_score(y_train, y_pred)
print("Training Accuracy:", train_accuracy)

#imprimir la precisión de testeo
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)
```

Conexión de Unity con LMC

- Importar el SDK de LMC para Unity.



Desarrollo del Sistema

- Creación de los datatable para la captura de datos.

```
//Creación de DataTable para captura de datos
private DataTable positionData;
private DataTable directionData;
private DataTable palmAndHandData;

//Instanciar los DataTable
positionData = new DataTable();
directionData = new DataTable();
palmAndHandData = new DataTable();
```

- Captura de datos de posición y dirección de cada dedo de la mano.

```
//Captura de datos de la posición y dirección de cada dedo de la mano
DataRow positionRow = positionData.NewRow(); // Nueva fila para posición
DataRow directionRow = directionData.NewRow(); // Nueva fila para dirección

int index = 0;

foreach (GameObject finger in fingers)
{
    Vector3 position = finger.transform.position;
    Vector3 direction = finger.transform.forward;

    positionRow[index] = position.x;
    positionRow[index + 1] = position.y;
    positionRow[index + 2] = position.z;

    directionRow[index] = direction.x;
    directionRow[index + 1] = direction.y;
    directionRow[index + 2] = direction.z;

    index += 3; // Avanzar al siguiente conjunto de valores (x, y, z)
    Debug.Log(manoDetectada);
}

positionData.Rows.Add(positionRow);
directionData.Rows.Add(directionRow);
```

- Capturar los datos de la posición del dedo medio y la muñeca.

```
//Captura de datos de la posición de la palma y el dedo medio
DataRow palmAndHandPositionRow = palmAndHandData.NewRow();

int palmAndHandIndex = 0;

foreach (GameObject palmOrHandPosition in fingerAndPalmPositions)
{
    Vector3 position = palmOrHandPosition.transform.position;

    palmAndHandPositionRow[palmAndHandIndex] = position.x;
    palmAndHandPositionRow[palmAndHandIndex + 1] = position.y;
    palmAndHandPositionRow[palmAndHandIndex + 2] = position.z;

    palmAndHandIndex += 3; // Avanzar al siguiente conjunto de valores (x, y, z)
}

palmAndHandData.Rows.Add(palmAndHandPositionRow);
```


- Convertir los DataTable a una matriz de float.

```
//Convertir los DataTables a una array de tipo float[,]
float[,] positionArray = ConvertDataTableToArray(positionData);
float[,] directionArray = ConvertDataTableToArray(directionData);
float[,] unido = ConcatenateArrays(positionArray, directionArray);

float[,] palmAndHandPositionArray = ConvertDataTableToArray(palmAndHandData);
```

- Función para convertir de DataTable a Matriz de tipo float.

```
private float[,] ConvertDataTableToArray(DataTable dataTable)
{
    int numRows = dataTable.Rows.Count;
    int numCols = dataTable.Columns.Count;
    float[,] dataArray = new float[numRows, numCols];

    for (int i = 0; i < numRows; i++)
    {
        for (int j = 0; j < numCols; j++)
        {
            dataArray[i, j] = Convert.ToSingle(dataTable.Rows[i][j]);
        }
    }

    return dataArray;
}
```

- Función de envío de las matrices y recepción de los datos obtenidos en Python.

```
private void SendParameters(float[,] param1, float[,] param2)
{
    // Crear un diccionario para almacenar los parámetros
    Dictionary<string, float[,]> parameters = new Dictionary<string, float[,]>();
    parameters["param1"] = param1;
    parameters["param2"] = param2;

    // Convertir el diccionario a JSON
    string json = JsonConvert.SerializeObject(parameters);

    // Convertir el JSON a bytes
    byte[] bytesToSend = Encoding.ASCII.GetBytes(json);

    // Enviar los bytes a través del socket
    socket.SendFrame(bytesToSend);

    // Recibir la respuesta del servidor
    byte[] response = socket.ReceiveFrameBytes();
    string responseString = Encoding.ASCII.GetString(response);

    ResponseData responseData = JsonConvert.DeserializeObject<ResponseData>(responseString);
}
```

- Creación de la clase ResponseData.

```
private class ResponseData
{
    public int prediccion { get; set; }
    public float[] distancias { get; set; }
    public float[] angulos { get; set; }
}
```

- Recibir la información proveniente del Web Socket.

```
//Obtener predicción, distancias y ángulos
int prediccion = responseData.prediccion;
float[] distancias = responseData.distancias;
float[] angulos = responseData.angulos;

float distanciaMinima = distancias.Min();
float anguloMaximo = angulos.Max();
```

- Uso de los datos obtenidos mediante el Web Socket y envío a la siguiente ventana del video de ejemplo.

```
//uso de los datos obtenidos en el web socket
if (prediccion == 1)
{
    if (distanciaMinima >= 11.5f && distanciaMinima >= 15f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Palma Abierta\nLa mano está recuperada";
    }

    if (distanciaMinima >= 8.5f && distanciaMinima <= 11.5f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Palma Abierta\nLa mano está en proceso de rehabilitación";
    }

    if (distanciaMinima <= 8.4f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Palma Abierta\nLa mano ha comenzado la rehabilitación";
    }
}

ejecutado = false;

PlayerPrefs.SetInt("score", score);
StartCoroutine(LoadNextScene("Video2"));
}
```

```

//uso de los datos obtenidos en el web socket
if (prediccion == 2)
{
    if (distanciaMinima >= 11.5f && distanciaMinima >= 15f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Puño\nLa mano ha comenzado la rehabilitación";
    }

    if (distanciaMinima >= 8.5f && distanciaMinima <= 11.5f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Puño\nLa mano está en proceso de rehabilitación";
    }

    if (distanciaMinima <= 8.4f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Puño\nLa mano está recuperada";
    }

    ejecutado = false;

    PlayerPrefs.SetInt("score", score);
    StartCoroutine(LoadNextScene("Video3"));
}

```

```

//uso de los datos obtenidos en el web socket
if (prediccion == 3)
{
    if (anguloMaximo >= 0f && anguloMaximo >= 60f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Flexión de la muñeca\nLa mano ha comenzado la rehabilitación \n Ángulo: " + anguloMaximo;
    }

    if (anguloMaximo >= 60f && anguloMaximo <= 80f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Flexión de la muñeca\nLa mano está en proceso de rehabilitación\n Ángulo: " + anguloMaximo;
    }

    if (anguloMaximo >= 80f && anguloMaximo <= 85f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Flexión de la muñeca\nLa mano está recuperada\n Ángulo: " + anguloMaximo;
    }

    if (anguloMaximo >= 80f && anguloMaximo <= 85f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Flexión de la muñeca\nEl ángulo de flexión supera los valores para una flexión normal\n Ángulo: " + anguloMaximo;
    }

    ejecutado = false;

    PlayerPrefs.SetInt("score", score);
    StartCoroutine(LoadNextScene("Video4"));
}

```

```

//uso de los datos obtenidos en el web socket
if (prediccion == 4)
{
    anguloMaximo = Mathf.Abs(anguloMaximo);

    if (anguloMaximo >= 0f && anguloMaximo >= 60f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Extensión de la muñeca\nLa mano ha comenzado la rehabilitación \n Ángulo: " + anguloMaximo;
    }

    if (anguloMaximo >= 60f && anguloMaximo <= 80f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Extensión de la muñeca\nLa mano está en proceso de rehabilitación\n Ángulo: " + anguloMaximo;
    }

    if (anguloMaximo >= 80f && anguloMaximo <= 85f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Extensión de la muñeca\nLa mano está recuperada\n Ángulo: " + anguloMaximo;
    }

    if (anguloMaximo >= 80f && anguloMaximo <= 85f)
    {
        textoVariable.text = "Gesto Realizado: Extensión de la muñeca\nEl ángulo de flexión supera los valores para una flexión normal\n Ángulo: " +
        anguloMaximo;
    }

    ejecutado = false;

    PlayerPrefs.SetInt("score", score);
    StartCoroutine(LoadNextScene("Video5"));
}

```

```

if (prediccion == 5)
{
    if (distanciaMinima >= 11.5f && distanciaMinima >= 15f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Pinza\nLa mano ha comenzado la rehabilitación";
    }

    if (distanciaMinima >= 8.5f && distanciaMinima <= 11.5f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Pinza\nLa mano está en proceso de rehabilitación";
    }

    if (distanciaMinima <= 8.4f)
    {
        textoVariable.text = "";
        textoVariable.text = "Gesto Realizado: Pinza\nLa mano está recuperada";
    }

    ejecutado = false;
}

```

- Función para detectar si la mano derecha se encuentra en el LMC.

```
private bool LeapMotionDetectaMano()
{
    if (leapMotionController != null)
    {
        Frame frame = leapMotionController.GetLeapController().Frame();

        if (frame.Hands.Count > 0)
        {
            Hand firstHand = frame.Hands[0];

            if (firstHand.IsRight)
            {
                return true;
            }
            else
            {
                return false;
            }
        }
    }

    return false;
}
```

- Ejecución del método para guardar los datos mientras se ejecuta el sistema.

```
private void Update()
{
    bool validar = LeapMotionDetectaMano();
    StartCoroutine(GuardarDatos(validar));
}
```

Utilización del modelo entrenado y Web Socket

- Cargar el modelo y crear el servidor web socket.

```
#Cargar el modelo
model = load_model('model_1.h5')

#crea servidor web socket
context = zmq.Context()
socket = context.socket(zmq.REP)
socket.bind("tcp://*:5555")

#imprimir si el web socket se conecta correctamente
print("Funciona")
```

- Recepción de datos capturados en Unity y creación de los array para su respectivo uso.

```
#recibir los datos obtenidos en unity
json_data = socket.recv().decode('utf-8')
matrices = json.loads(json_data)

#crear los array de cada dato obtenido
array1 = np.array(matrices['param1'])
array2 = np.array(matrices['param2'])
```

- Dividir la segunda matriz en dos matrices de posiciones del dedo y posiciones de la muñeca. Siguiendo si se encuentran valores nulos reemplazarlos por 0.

```
#dividir la segunda matriz en las posiciones del dedo medio y muñeca
num_columnas = array2.shape[1]
mitad_columnas = num_columnas // 2

medio = np.empty((array2.shape[0], mitad_columnas))
muneca = np.empty((array2.shape[0], mitad_columnas))

medio = array2[:, :mitad_columnas]
muneca = array2[:, mitad_columnas:]

#verificar si existen datos nulos y reemplazarlos por 0
medio = np.nan_to_num(medio)
muneca = np.nan_to_num(muneca)
```

- Creación de dataframes para su correspondiente división en ventanas.

```
#crear dataframes de cada matriz para su normalización
combinar = pd.DataFrame(array1.astype(float))
combinar_medio = pd.DataFrame(medio.astype(float))
combinar_muneca = pd.DataFrame(muneca.astype(float))

#dividir en ventanas de 70 datos
particiones = np.array_split(combinar, 70)
particiones_medio = np.array_split(combinar_medio, 70)
particiones_muneca = np.array_split(combinar_muneca, 70)
```

- Obtener los valores medios de cada división creada

```
#obtener los valores medios de cada ventana para obtener una matriz de 70 filas
medias = []
for particion in particiones:
    media_particion = particion.mean()
    medias.append(media_particion)

medias_medio = []
for particion in particiones_medio:
    media_particion2 = particion.mean()
    medias_medio.append(media_particion2)

medias_muneca = []
for particion in particiones_muneca:
    media_particion3 = particion.mean()
    medias_muneca.append(media_particion3)
```

- Obtener matrices y vectores a utilizar para identificar el gesto realizado y mostrar el avance que posee la mano derecha en la rehabilitación.

```
#convertir la matriz de medias en un vector
matriz_medias = np.array(medias)
vector_medias = matriz_medias.T.flatten()

#obtener las matrices de dedo medio y muñeca
matriz_medias_medio = np.array(medias_medio)
matriz_medias_muneca = np.array(medias_muneca)
```

- Cálculo de distancias y ángulos para verificar el avance de la rehabilitación

```
#calculo de distancia
distancias = []
for fila1, fila2 in zip(matriz_medias_medio, matriz_medias_muneca):
    distancia = euclidean(fila1, fila2) * 100
    distancias.append(distancia)

#calculo de ángulos
angulos = []
for fila1, fila2 in zip(matriz_medias_medio, matriz_medias_muneca):
    distancia_x = fila2[0] - fila1[0]
    distancia_y = fila2[1] - fila1[1]
    distancia_z = fila2[2] - fila1[2]

    angulo_radianes = math.atan2(distancia_x, math.sqrt(distancia_z**2 + distancia_y**2))
    angulo_grados = math.degrees(angulo_radianes)
    angulos.append(angulo_grados)
```

- Normalizado del vector para la predicción

```
#normalizado del vector de medias
min_vals_1 = np.min(vector_medias)
max_vals_1 = np.max(vector_medias)

normalizado_filas = (vector_medias.T - min_vals_1) / (max_vals_1 - min_vals_1)
```

- Realizar la creación de ventanas y subventanas para la extracción de características en el vector a utilizar para la predicción.

```
#Extracción de características del vector ingresado
total_columnas = normalizado_filas.T.shape[0]
tamano_parte = 70
grupos = [normalizado_filas[i:i+tamano_parte] for i in range(0, len(normalizado_filas), tamano_parte)]

divisiones = [np.array_split(grupo, 5) for grupo in grupos]

resultados_vector = []

for i in range(len(grupos)):
    for j in range(np.shape(divisiones)[1]):
        divisiones_arr = np.array(divisiones[i][j])
        media = np.mean(divisiones_arr)
        desviacion_estandar = divisiones_arr.std()
        raiz_cuadrada_media = np.sqrt(np.abs(divisiones_arr)).mean()
        varianza = np.sum(np.square(divisiones_arr)) / (len(divisiones_arr) - 1)
        derivada = np.mean([np.diff(np.atleast_1d(row)) for row in divisiones_arr])
        resultados_vector.extend([media, desviacion_estandar, raiz_cuadrada_media, derivada, varianza])
```


- Preparar el modelo para la predicción impidiendo el ingreso de valores nulos y hacer que el vector resultado sea una tupla. La tupla del vector resultado se ingresa a la predicción del modelo. Realizada la predicción del modelo se obtiene el valor de predicción.

```
#verificar que los datos del vector de resultados no tengan campos nulos
resultados_vector = np.nan_to_num(resultados_vector)

#hacer que el resultado del vector sea una tupla para predecir
resultados_vector = resultados_vector.tolist()

#prediccion del modelo
pred = model.predict([resultados_vector])

#obtener valor de predicción
y_test_pred_category = np.argmax(pred, axis=1)
```

- Preparar los datos y enviar a través del Web Socket

```
#resultado a enviar desde el web socket
resultado = {
    'prediccion': str(y_test_pred_category[0]),
    'distancias': distancias,
    'angulos' : angulos
}

#envío de datos desde web socket
bytes_to_send = json.dumps(resultado).encode()
socket.send(bytes_to_send)
```

Anexo F

Certificados de aprobación obtenidos por los fisioterapeutas de centros de rehabilitación autorizados.

 UNIVERSIDAD TÉCNICA DE AMBATO  FACULTAD DE CIENCIAS DE LA SALUD

UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD CIENCIAS DE LA SALUD
CARRERA DE FISIOTERAPIA
LABORATORIO DE TERAPIA FÍSICA 

Ambato, 18 de Julio de 2023

CERTIFICADO

Quien suscribe Licenciada, Magister Angela Priscila Campos Moposita portadora de la cédula 1803276979, en calidad de Responsable del Laboratorio de Terapia Física tengo a bien certificar que el Sr. Miguel Enrique Bonilla Quispe portador de la cédula de ciudadanía No. 1805326343, estudiante de la carrera de Tecnologías de la Información de la Facultad de Ingeniería en Sistemas Electrónica e Industrial de la Universidad Técnica de Ambato, realizó su proyecto de titulación denominado “SISTEMA INTELIGENTE USANDO VIDEO CÁMARAS Y MOTION CAPTURE PARA MONITOREO DE LA REHABILITACIÓN DE LA MANO DERECHA”, en el Laboratorio de Laboratorio de Terapia Física ubicado en el campus Ingahurco, según los estándares requeridos para la rehabilitación de la mano derecha.

Es todo cuanto puedo certificar en honor a la verdad y autorizo al interesado/a hacer uso del presente documento como a bien tenga.

Atentamente,



Lic. Mg. Angela Priscila Campos Moposita
RESPONSABLE DEL LABORATORIO DE TERAPIA FÍSICA





UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD CIENCIAS DE LA SALUD
CARRERA DE FISIOTERAPIA
LABORATORIO DE TERAPIA FISICA

Ambato, 18 de Julio de 2023

CERTIFICADO

Quien suscribe Licenciada, Evelyn Johanna Pacha Jara portadora de la cédula 1850360205, en calidad de Técnico de Laboratorio 1 tengo a bien certificar que el Sr. Miguel Enrique Bonilla Quispe portador de la cédula de ciudadanía No. 1805326343, estudiante de la carrera de Tecnologías de la Información de la Facultad de Ingeniería en Sistemas Electrónica e Industrial de la Universidad Técnica de Ambato, realizó su proyecto de titulación denominado "SISTEMA INTELIGENTE USANDO VIDEO CÁMARAS Y MOTION CAPTURE PARA MONITOREO DE LA REHABILITACIÓN DE LA MANO DERECHA", en el Laboratorio de Terapia Física ubicado en el campus Ingahurco, según los estándares requeridos para la rehabilitación de la mano derecha.

Es todo cuanto puedo certificar en honor a la verdad y autorizo al interesado/a hacer uso del presente documento como a bien tenga.

Atentamente,



Lic. Evelyn Johanna Pacha Jara
TÉCNICO DE LABORATORIO 1