



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA ELECTRÓNICA
MAESTRÍA EN INGENIERÍA ELECTRÓNICA,
OPCIÓN INSTRUMENTACIÓN ELECTRÓNICA

Tesis para obtener el grado de:

Maestro en Ingeniería Electrónica

DETECCIÓN DE CAÍDAS EN ADULTOS MAYORES

USANDO INFORMACIÓN TÉRMICA E INTELIGENCIA ARTIFICIAL

Presenta:

JESÚS MANUEL ROA ESCALANTE*

Asesor:

DR. ALDRIN BARRETO FLORES

Co-asesor:

DR. SALVADOR E. AYALA RAGGI

Agradecimientos

Agradezco al Consejo Nacional de Ciencias y Tecnología (CONACyT) por haberme apoyado con una beca de investigación con la cual me fue posible concluir esta etapa de mi desarrollo académico. También agradezco a la Benemérita Universidad Autónoma de Puebla (BUAP) y a la Facultad de Ciencias de la Electrónica (FCE) por haberme permitido estudiar esta Maestría.

Quiero agradecer a mis directores de tesis el Dr. Aldrin Barreto Flores y el Dr. Salvador E. Ayala Raggi por permitirme trabajar en esta investigación y por todas las atenciones que tuvieron conmigo, brindando sus consejos y conocimiento a pesar de las dificultades vividas durante la pandemia, gracias por siempre atender mis dudas y otorgarme su tiempo que es tan valioso. Gracias por ser personas ejemplares y siempre alentarme a superarme.

Gracias a todos mis profesores que me acompañaron en la culminación de mis estudios de maestría en la FCE, reconozco todo su profesionalismo para impartir las clases y por siempre dar lo mejor en sus cátedras. Asimismo, agradezco a la M.C. Ana María Rodríguez Domínguez por apoyarme con todas las dudas y tramites durante la maestría.

También agradezco a mis sinodales el M.C. Rodrigo Lucio Maya Ramírez, Dra. Josefina Castañeda Camacho y la M.C. Selene Edith Maya Rueda por atender mis revisiones de tesis durante la maestría y por sus valiosos comentarios que lograron mejorar este trabajo de investigación. Todo mi respeto a ustedes y gracias por su tolerancia.

Agradezco a todos mis amigos Annel, Sandra, Saúl y Sergio, que me alentaron a nunca rendirme y brindarme su apoyo en momentos difíciles, les deseo lo mejor en la vida.

Dedicatoria

Dedico este trabajo a mis padres y familia, Manuel, María Elena, Claudia, Lorena y Memo. Gracias por estar siempre a mi lado y por brindarme su tiempo, su amor y su sabiduría sin medida, por transmitirme sus valores y enseñanzas, ustedes han dejado una huella indeleble en mi corazón. Este gran logro es también de ustedes.

Contenido

Índice de figuras	6
Lista de tablas	9
1. Introducción.....	10
1.1. Objetivos.....	12
1.1.1. Objetivo general.....	12
1.1.2. Objetivos específicos	12
1.2. Justificación	13
1.3. Organización del trabajo	14
2. Estado del arte.....	14
2.1. Sistemas de detección de caídas.....	14
2.2. Sensores térmicos en la detección de caídas.....	18
2.3. Registro de imágenes.....	19
3. Marco teórico.....	21
3.1. Termómetros de radiación.....	21
3.1.1. El sensor MLX90640.....	22
3.2. Raspberry Pi.....	23
3.3. Protocolo de Comunicación I ² C.....	25
3.4. Inteligencia Artificial.....	26
3.5. Machine Learning.....	27
3.5.1. Clasificador KNN Ponderado.....	27
3.6. Procesamiento digital de imágenes	29
3.6.1. Filtro de mediana	29
3.7. Análisis de Componentes Principales.....	30
4. Método propuesto	31
4.1. Descripción del sistema	31
4.2. Diseño de un algoritmo de almacenamiento.....	33
4.3. Construcción de una base de datos térmicos.....	35
4.4. Preprocesamiento de los datos.....	37
4.4.1. Normalización en rotación, traslación y escala del conjunto de entrenamiento.....	37
4.5. Propuesta de clasificación de secuencias de fotogramas térmicos usando un clasificador k-NN.....	46
4.6. Optimización de la propuesta de clasificación de secuencias de fotogramas térmicos	49
5. Resultados.....	51
6. Conclusiones.....	56
6.1 Contribuciones	57

6.2 Trabajo futuro	57
Referencias	58
Anexos	63
a) Algoritmo de almacenamiento en lenguaje Python	63
b) Algoritmo de clasificación KNN	64

Índice de figuras

Figura 3.1 Sensores MLX90640 con resolución de 32×24 pixeles en diferente configuración del campo de visión, siendo uno de $55^\circ \times 35^\circ$ y el segundo de $110^\circ \times 75^\circ$ [12].....	22
Figura 3.2 Diagrama de bloques del sensor MLX90640 donde se muestra el acceso a la información de los sensores almacenados en la RAM mediante el protocolo I ² C extraído de [12].....	23
Figura 3.3 Tarjeta Raspberry Pi 4B 8 GB la cual provee al usuario de un incremento en la velocidad de los procesos de multimedia, procesador y conectividad [33].....	24
Figura 3.4 Localización de pines de alimentación y de funciones especiales, además de pines de comunicación I ² C de la Raspberry Pi 4B [34].....	24
Figura 3.5 Diagrama de bloques de un ejemplo de conexión maestro y esclavo para el protocolo de comunicación I ² C [35].....	25
Figura 3.6 Esquema demostrativo de la metodología para utilizar un filtro de mediana usando una ventana de 3×3 pixeles [41].....	30
Figura 4.1 Esquema del funcionamiento del sistema utilizando secuencias térmicas e imágenes de entrenamiento de una persona estando acostada y parada para construir un vector de probabilidades que permita clasificar si la secuencia pertenece a una caída o no caída.	32
Figura 4.2 Esquema de la optimización del sistema propuesto, utilizando las características resultantes de aplicar PCA para construir un vector de características y utilizarlo en la clasificación de una caída y no caída.....	33
Figura 4.3 Diagrama de flujo del algoritmo diseñado para el almacenamiento de las secuencias térmicas.....	34
Figura 4.4 Integración del sensor térmico mediante el protocolo de comunicación I ² C usando los pines SDA Y SCL en una Raspberry Pi 4.....	34
Figura 4.5 Captura de datos térmicos del sensor MLX90640, con el sujeto de prueba una distancia de a 2 m (a) y 4 m (a).....	35
Figura 4.6 Montaje del dispositivo integrado de captura de secuencias térmicas en el techo de la habitación.....	35
Figura 4.7 Datos obtenidos del sensor MLX90640 simulando una caída.	36
Figura 4.8 Datos obtenidos del sensor MLX90640 durante una caminata.....	36
Figura 4.9 Procesamiento digital de imagen aplicado.....	37

Figura 4.10 Diagrama general del proceso de normalización en rotación, traslación y escala para una imagen térmica de una caída.....	38
Figura 4.11 Proceso de búsqueda y ajuste del centroide comenzando de izquierda a derecha sobre la imagen para encontrar la región de interés en una caída.	39
Figura 4.12 Proceso de cálculo de la densidad para normalizar la escala del objeto de interés en una imagen de caída.....	41
Figura 4.13 Recorte del objeto de interés utilizando el algoritmo de normalización de escala.	42
Figura 4.14 Obtención de los gradientes de una imagen de caída, a) gradientes horizontales, b) gradientes verticales.....	43
Figura 4.15 Histograma polar correspondiente a las direcciones y la sumatoria de magnitudes de las imágenes, a) histograma polar original, b) histograma polar suavizado.....	43
Figura 4.16 Imágenes resultantes de la separación de la imagen Imm, n en dirección $hmax1$, a) mitad denominada $C1(m, n)$, b) mitad denominada $C2(m, n)$	44
Figura 4.17 Imagen resultante del proceso de normalización en rotación, traslación y escala.....	44
Figura 4.18 Diseño de interfaz para normalizar en rotación, traslación y escala las imágenes del set de entrenamiento.	46
Figura 4.19 Imágenes de ejemplo de un fotograma de una persona acostada y una persona parada que se encuentran en dentro del nuevo conjunto normalizado.....	47
Figura 4.20 Propuesta del sistema de reconocimiento de caídas utilizando secuencias térmicas e imágenes de entrenamiento de una persona estando acostada y parada.	48
Figura 4.21 Construcción del vector de características de la secuencia térmica utilizando las características resultantes de la reducción mediante PCA de cada fotograma de la secuencia térmica.....	49
Figura 4.22 Propuesta optimizada del sistema de reconocimiento de caídas utilizando secuencias térmicas e imágenes de entrenamiento de una persona estando acostada y parada.....	50
Figura 5.1 Ejemplo de un vector de probabilidad de estar acostado para una secuencia de no caída.....	51
Figura 5.2 Ejemplo de un vector de probabilidad de estar acostado para una secuencia de una caída.....	52

Figura 5.3 Resultados de la validación cruzada del clasificador K-NN ponderado con $k=11$, 352 secuencias equilibradas (caídas y no caídas) de entrenamiento y un conjunto de validación equilibrado de 60 secuencias.....	53
Figura 5.4 Resultados de la validación cruzada del sistema optimizado con un clasificador K-NN ponderado usando $k = 9$ y un porcentaje de reconstrucción de las características de un 80%.....	54
Figura 5.5 Resultados de la validación cruzada del sistema optimizado con un clasificador K-NN ponderado usando $k = 9$ y un porcentaje de reconstrucción de las características de un 70%.....	55
Figura 5.6 matriz de confusión construida con los resultados de las 20 pruebas mediante validación cruzada.....	55

Lista de tablas

Tabla 2.1 Trabajos realizados para la detección de caídas y principales aportaciones.....	17
Tabla 3.1: Características del sensor MLX90640 obtenidas de la referencia [12].	.22

1. Introducción

La tecnología de vigilancia para adultos mayores ha evolucionado en las últimas décadas para incluir dispositivos de seguimiento de actividad, sistemas de alerta de emergencia, sensores de movimiento y cámaras de vigilancia. Estos dispositivos ayudan a los cuidadores y familiares a supervisar el bienestar de un adulto mayor y detectar cualquier signo de problema de salud o de seguridad. También pueden ayudar a los adultos mayores a mantener su independencia y seguridad en su hogar. Aunque entendemos que la tecnología no debe ser vista como una solución única y debe ser utilizada en conjunto con el cuidado humano y la atención médica, debemos resaltar la gran importancia y la valiosa ayuda que esta puede proporcionar en un mejor cuidado y manejo de los pacientes. En el mundo, la cantidad de adultos mayores aumenta cada año, de acuerdo con datos de la Organización Mundial de la Salud (OMS), se prevé que entre 2015 y 2050 existirá un aumento del 12% al 22% de la población mundial de este gremio, por ende, conlleva a necesitar cada vez una sociedad más preparada en temas de la vejez, tales como cuidados especiales y atención médica [1]. La comunidad científica está preocupada por este desafío y está trabajando en el desarrollo de tecnologías innovadoras para abordarlo.

Los sistemas de detección de caídas se pueden agrupar en tres categorías principales: sensores medio ambientales, portátiles y una combinación de ellos [2]. En algunas investigaciones hacen uso de sensores portátiles, los cuales miden la aceleración de segmentos específicos del cuerpo, y utilizando inteligencia artificial lograr detectar una caída, este tipo de dispositivos deben estar siempre en contacto con la persona a observar [3]. El estudio de la cinemática de una caída también es muy importante, ya que permite entender las características necesarias y suficientes para desarrollar un algoritmo de detección de estas, tal como en [4], donde se estudia la secuencia de una caída mediante acelerómetros colocados en la cabeza y el torso de un maniquí de prueba, de esta manera, encontraron los picos de aceleración que existen durante una caída.

La detección de caídas utilizando imágenes tradicionales puede ser difícil debido a factores como la iluminación cambiante, el movimiento y el ángulo de la cámara. Aún en un ambiente controlado como el de una habitación, la detección de caídas mediante las imágenes que proporciona una cámara común es un problema desafiante porque la persona puede caer en cualquier zona dentro de ésta con una infinidad de posturas diferentes y dentro de una escena donde puede haber una gran variedad de objetos de diversos tipos [5]. En contraposición a las

dificultades existentes en los métodos que usan imágenes normales, la utilización de imágenes térmicas es un camino diferente a la hora de implementar la detección de caídas. Estas imágenes capturan la radiación infrarroja emitida por un objeto, y a diferencia de los sistemas que utilizan imágenes normales en el rango visible, la segmentación del objeto de interés, es decir la persona, es mucho más sencilla, pues es el único objeto caliente en la escena [6]. Por otra parte, las cámaras infrarrojas no tienen la capacidad de capturar la escena a detalle, lo que supone una garantía de privacidad para el usuario [7].

Existen algunos trabajos que usan imágenes térmicas que hacen uso de técnicas geométricas y de umbralización para determinar si una persona está de pie o en el suelo [8]. El problema de este tipo de enfoque es que pueden fallar cuando las personas realizan la acción de acostarse sin existir una caída genuina. Para poder discriminar entre caídas reales y falsos positivos este trabajo propone usar aprendizaje automático entrenado con una gran cantidad de ejemplos tanto de caídas reales como de otra variedad de acciones que incluyen sentarse, deambular y acostarse en la cama o en el suelo.

Algunos trabajos han desarrollado modelos de aprendizaje automático que utilizan redes neurales convolucionales y SVM para detectar patrones que indican una caída [9], [10]. El problema de estos métodos es que la toma de las imágenes es frontal, lo que puede ocasionar oclusiones debido a objetos que se interpongan en la escena o a siluetas que se hacen más pequeñas o se desvanecen debido a la distancia variable entre la cámara térmica y el sujeto a monitorear.

En este trabajo lo que se propone es utilizar aprendizaje automático en imágenes térmicas tomadas desde el techo de una habitación, lo cual tiene como ventaja, con respecto a los trabajos anteriormente mencionados, que el tamaño del sujeto en la escena es aproximadamente siempre el mismo y la distancia entre la cámara y el sujeto es siempre la misma, lo cual nos brinda estabilidad en la silueta de la persona, pues esta no se hace más pequeña ni se diluye con la distancia como suele ocurrir en los métodos descritos propuestos por otros autores.

Por otro lado, dado que el individuo en la escena capturada desde el techo cambia de posición dentro de ésta y también de postura constantemente, es necesario el desarrollo de un método capaz de segmentar al individuo de la escena con el propósito de poder extraer de esta región de interés un conjunto de características apropiadas para distinguir entre caídas y no caídas.

Para ello en este trabajo desarrollamos un algoritmo capaz de extraer el objeto de interés de cada fotograma térmico, un segundo algoritmo que normaliza

la escala del objeto y, por último, un método para normalizar el ángulo de dicho objeto. Una vez teniendo las imágenes de la región de interés alineadas, es posible obtener un conjunto más pequeño de características para clasificación cuando se aplica el método de Análisis por Componentes Principales (PCA).

1.1. Objetivos

1.1.1. Objetivo general

Desarrollar un sistema de detección de caídas usando inteligencia artificial e información infrarroja.

1.1.2. Objetivos específicos

- Estudiar los sensores MLX90640 (matriz 24x32) para generar datos térmicos.
- Analizar los tiempos de creación de las matrices de datos térmicos para evaluar la integración del sensor vía I²C.
- Generar una base de datos térmicos donde se capture información de la caída de una persona.
- Preprocesamiento de la región de interés de la base de datos.
- Diseñar un algoritmo basado en Aprendizaje Máquina que reconozca una caída.
- Validación del algoritmo propuesto.

1.2. Justificación

Un padecimiento importante asociado con las personas mayores son las caídas, estas pueden definirse como una situación inesperada donde una persona pasa de estar realizando una actividad física a estar en el suelo, el porcentaje de sufrir este fenómeno se incrementa con la edad y el grado de restricción de la motricidad [11]. La pérdida de masa muscular y el deterioro del equilibrio son factores importantes en la incidencia de caídas en personas mayores, gracias a la identificación de estas características, se han podido proponer estrategias para la prevención de las caídas, tales como la reducción del uso de algunos medicamentos, eliminar peligros en los hogares, terapias para mejorar el equilibrio y corrección de la vista [12]. Las caídas y el miedo a caerse dependen uno del otro, debido a esto, cuando un individuo tiene un historial de caídas, es propenso a desarrollar posteriormente miedo a caer y, esta condición adquirida también conduce a una mala calidad de vida, depresión y un decremento de la vida social [13].

La visión estadística planteada por la OMS nos muestra un futuro donde la población mundial tendrá cada vez un porcentaje mayor de personas en edad avanzada. Como resultado, es importante tener en cuenta el impacto de las caídas en esta población y la importancia de tomar medidas preventivas y proporcionar atención rápida cuando se producen. Por lo tanto, es importante desarrollar y utilizar tecnologías como los sistemas de detección de caídas para ayudar a mitigar estos riesgos y mejorar el cuidado de la salud de las personas mayores. De acuerdo con información de la OMS [14], las caídas están posicionadas en el segundo lugar de causa de muerte en el mundo, teniendo un valor aproximado de 646,000 casos anuales de este fenómeno, colocando a los adultos mayores de 60 años como los principales afectados. En países con alto desarrollo como Finlandia, el sistema sanitario ha gastado alrededor de US\$ 3,611 en atender cada caso relacionado con caídas en personas de edad avanzada.

Partiendo de sistemas que pueden detectar caídas con video cámaras, encontramos muchas ventajas a la hora de implementar algoritmos, pero desventajas cuando la iluminación no es controlada, además se invade la privacidad de los usuarios mientras están dentro de sus hogares. Los dispositivos portátiles también cuentan con desventajas, la más evidente es que su autonomía está limitada por el uso de baterías, además, corren el riesgo de dañarse durante su uso.

Analizando los casos anteriores, surge la necesidad de desarrollar un sistema capaz de identificar caídas entre distintos comportamientos cotidianos, asegurando la privacidad de los usuarios, por este motivo, se propone un manejo de datos en el

rango infrarrojo, obteniendo la información mediante un sensor térmico para identificar una caída utilizando inteligencia artificial.

1.3. Organización del trabajo

Este trabajo de tesis consta de 6 capítulos. El Capítulo 1 se presentó la Introducción, que describe los aspectos generales sobre las caídas en adultos mayores. El Capítulo 2 se hará revisión del Estado del arte de los algoritmos de detección de caídas, en el Capítulo 3 se dará el Marco teórico del trabajo. El Capítulo 4 aborda la metodología propuesta para la detección de caídas. En el Capítulo 5 se muestran los Resultados y finalmente en el Capítulo 6, las Conclusiones.

2. Estado del arte

Este capítulo se enfoca en los antecedentes relacionados con la detección de caídas y los sistemas de aprendizaje automático utilizados para resolver esta problemática. Se presentan diferentes enfoques para identificar caídas, así como modelos de clasificación utilizados para reconocer diferentes tipos de comportamientos. Se discuten tanto las limitaciones como los beneficios de cada método y se explica cómo estos informes han influenciado el enfoque adoptado en este trabajo.

2.1. Sistemas de detección de caídas

En la detección de comportamientos específicos, el uso de redes neuronales es crucial para diferenciar entre una caída y otros comportamientos. Utilizando un acelerómetro de 3 ejes y algoritmos basados en redes neuronales multicapa, se ha logrado una sensibilidad del 98.4% y una especificidad del 98.6%. Esto demuestra la efectividad de utilizar redes neuronales para detectar caídas [15]. Otro método relacionado es mediante dispositivos colocados en el ambiente [16], los cuales obtienen información por medio de cámaras de video y utilizando de manera conjunta métodos de procesamiento de imagen como Histogramas de Gradientes

Orientados (HOG), Patrones Binarios Locales LBP y características extraídas con el software Caffe, lograron una detección de caídas con 93.7 % de sensibilidad y 92 % de especificidad. Es posible resaltar que, al utilizarse cámaras de video, se presentan retos a resolver según cada ambiente en el que se trabaje, estos pueden ser desde la iluminación del lugar, como la posición que tendrá dicha cámara, además, se destaca una invasión a la privacidad del usuario.

Los algoritmos de seguimiento de objetos en imágenes térmicas son muy importantes tanto en aplicaciones civiles como militares, especialmente en el ámbito de la vigilancia y monitoreo. En comparación con el seguimiento de objetos visuales, el seguimiento por imágenes infrarrojas tiene sus ventajas, una de ellas es poder medir la temperatura de una persona sin requerir el contacto directo, tal como en [17], donde desarrollaron un dispositivo de bajo costo capaz de visualizar en tiempo real imágenes térmicas de la piel, utilizando el sensor AMG8833, los autores realizaron una comparativa entre sensores infrarrojos de gamas más altas; los cuales tienen mayor resolución pero un precio notablemente alto, de modo que, pudieron concluir en que su sistema a pesar de tener una baja resolución de la imagen, logra mediciones con variaciones muy pequeñas comparadas con sensores de gama alta.

El trabajo presentado en [18], se desarrolló una solución integral para la gestión remota de caídas dentro de un entorno con múltiples sujetos de prueba, su investigación presentó una solución de detección de bajo costo, que ha sido desarrollada mediante el uso de sensores térmicos de baja resolución. Su configuración de los sensores permite capturar los datos de manera discreta e integrar datos en una plataforma de sensores escalable donde se implementa un enfoque innovador para el procesamiento de imágenes térmicas. La clasificación de caída o no caída se calcula en tiempo real utilizando descomposición de imágenes y clasificación con una red neuronal (NN), que entrenó mediante minimización el Error de generalización localizado con características extraídas por Redes neuronales convolucionales (CNN).

El uso de cámaras térmicas también tiene atribuidas propiedades indeseables, que hacen del seguimiento una tarea más desafiante. En [19] observamos un algoritmo de seguimiento basado en Filtros de Correlación Kernelized (KCF) que demuestra una alta precisión y robustez en las tareas de seguimiento visual, con una complejidad computacional muy baja. El algoritmo se ha probado en varios ejemplos elegidos, utilizando dos representaciones de características diferentes los cuales son píxeles sin formato y características HOG.

El rendimiento de la detección de caídas basada en video puede verse afectado por diferencias en el fondo. Esto logra volverse más prominente en las cámaras térmicas donde las intensidades pueden cambiar debido a las diferencias de calor, en su trabajo [20], plantean la hipótesis del aprendizaje de las características espacio-temporales utilizando la conciencia de la región y el movimiento, esto con la finalidad de mejorar las secuencias de video y de igual forma la detección de las caídas, por lo que desarrollaron una red contradictoria que se compone de autocodificadores convolucionales 3D de dos canales, los cuales reconstruyen los datos térmicos y las secuencias de entrada de flujo óptico. Esto les permite rastrear la región de interés, logrando utilizar al error de reconstrucción como indicador de ocurrencia para una caída. Otro trabajo de investigación que también hace uso del enfoque anterior, utiliza un método de aprendizaje que consiste en un autocodificador convolucional espacio-temporal (3DCAE) que reconstruye una secuencia de fotogramas y usan una red neuronal convolucional espacio-temporal (3DCNN) como clasificador, así cuando ingresan una secuencia de imágenes a su red, si el error en la reconstrucción es alto, además de tener baja probabilidad del discriminador, esto indicará una secuencia de video anómala, de esta manera, logran mostrar anomalías para identificar caídas invisibles durante sus pruebas [9].

Muchos sensores infrarrojos que existen en el mercado son utilizados de en aplicaciones tanto domesticas como industriales. En [7], determinan que muchas de las soluciones que se logran mediante estos sensores tienen en común el uso de procesamiento y transferencia de los datos, además que consumen una gran cantidad de energía y, esto los obliga a requerir una fuente de alimentación por cable, además, resaltan que una instalación de estos sensores, utilizando medidas que no perturben la comodidad resulta costosa, por ultimo mencionan que estos sistemas pueden recibir perturbaciones si en la habitación se encienden aparatos eléctricos.

En [21], se presenta un método novedoso para la detección de caídas, llamado DeepFall, este consiste en abordar el problema de detección de caídas como un problema de detección de anomalías. El proceso en que se desarrolla el DeepFall hace un uso novedoso de autocodificadores convolucionales espaciotemporales profundos para aprender características espaciales y temporales de actividades normales utilizando modalidades de detección no invasivas. En este trabajo consideran la gran diferencia entre los datos que pueden obtenerse de las actividades normales de la vida diaria y los datos de una caída, remarcan una notoria escases de datos de caídas. En consecuencia, proponen el entrenamiento de

un codificador automático en actividades normales y usar su error de reconstrucción para identificar caídas invisibles durante la prueba.

Finalmente, aprovechando grandes características de usar sensores infrarrojos, se plantea en [6] un sistema capaz de identificar cuatro comportamientos, los cuales son: caminar, estar recostado, parado y finalmente una caída, al comparar diferentes algoritmos de aprendizaje máquina, encontraron que el algoritmo por mayoría de votos tuvo el mejor desempeño al reconocer los comportamientos propuestos, obteniendo un 97.75% de confiabilidad al momento de detectar una caída, además, se concluye que puede mantenerse la privacidad del usuario mientras el sistema está activo.

A continuación se describe en la tabla 2.1 trabajos que utilizan diferentes sensores para lograr la detección de caídas. Se incluye el autor o autores del trabajo, el tipo de sensor utilizado, la metodología utilizada sus resultados y las limitaciones que presenta su sistema.

Tabla 2.1 Trabajos realizados para la detección de caídas y principales aportaciones.

Autores	Sensor utilizado	Método de clasificación utilizado	Resultados	Limitaciones
S. Yoo y D. Oh [3]	Acelerómetro	Red neuronal artificial basada en aprendizaje profundo.	Colocando el sensor en la muñeca, se logró obtener un 100 % de confiabilidad al detectar una caída.	Debido a que el dispositivo es portátil, el sistema tendrá autonomía limitada, y puede dañarse durante su uso.
M. Vallejo, C. Isaza y J. López [15]	Acelerómetro	Red neuronal artificial con 3 capas ocultas.	Detección de caídas con un 98.4 % de sensibilidad y 98.6 % de especificidad.	Autonomía determinada por la batería del dispositivo.
K. Wang, G. Cao, D. Meng, W. Chen y C. Cao [16]	Video cámara	Combinación de Histogramas de Gradientes Orientados (HOG), Patrones Binarios Locales LBP y características extraídas con el software Caffe.	Lograron un 93.7 % de sensibilidad y 92 % de especificidad para identificar una caída.	El procesamiento digital de imagen requiere ambientes controlados, donde la iluminación no genere interferencias, además, el uso de ropa de diferentes colores requeriría un banco extenso de patrones de entrenamiento. La privacidad del usuario se ve afectada.

Y. Ogawa y K. Naito [6]	Sensor de temperatura	Múltiples algoritmos basados en aprendizaje máquina, tomados de la biblioteca Scikit-Learn, utilizando una Raspberry Pi.	Se logró un 97.75 % de confiabilidad, utilizando un algoritmo de clasificación por votación para determinar una caída entre comportamientos como: caminar, estar parado y acostado.	La resolución de los datos es pequeña, contando solo con una matriz de 4x16 pixeles. La cantidad de dispositivos para el funcionamiento del sistema es alta.
A. Hayashida, V. Moshnyaga y K. Hashimoto [7]	Panasonic Grid eye	Algoritmo de extracción de primer plano	Obtuvieron un reconocimiento de caídas del 94% de confiabilidad en un entorno con temperatura ambiente de 24°C	Este porcentaje disminuyó a medida que la temperatura se hizo más alta, colocado en el techo, esta posición podría dificultar el reconocimiento a la hora de reconocer caídas en personas con gorros y ropa robusta.

2.2. Sensores térmicos en la detección de caídas

El uso de sensores térmicos para entrenar clasificadores inspirados en Aprendizaje Automático está siendo cada vez más utilizado, dado las ventajas que pueden tener contra otro tipo de variables físicas, en el trabajo [22], utilizan dos sensores infrarrojos que entregan individualmente una matriz de 16×4 , colocándolos sobre la misma pared, pero a diferentes lugares, logran ubicar la posición del cuerpo humano un plano tridimensional, ubicándolo mediante un algoritmo de posicionamiento basado en el ángulo de llegada (AOA). Mientras que ambos sensores capturan la acción al mismo tiempo, se elige el sensor que contenga la región más grande para el proceso de extracción de características, obteniendo una recopilación de 80 caídas y 80 comportamientos normales. Finalmente, obtuvieron una precisión del 93% al emplear un clasificador KNN (K-Nearest Neighbors).

Gran parte de los trabajos encontrados, donde se utilizan las imágenes térmicas, tratan de eliminar interferencias de ruido térmico y sólo identificar al sujeto de interés, tal como [8], donde es utilizada una Raspberry Pi 3 para adquirir los datos de un sensor infrarrojo que entrega matrices de 80×64 pixeles, en su investigación realizan procesos para determinar la altura del cuerpo humano, además de suavizar la fluctuación en el cálculo de esta. Los resultados obtenidos en esta investigación logran un 99% de precisión aplicando un criterio de umbral a la altura obtenida del sujeto para identificar una caída.

Existen investigación donde evalúan el uso de visión térmica para identificar las posibles debilidades del método, en [23] implementan un conjunto de soluciones facilitadas por el Big Data y presentan resultados prometedores para el uso de este enfoque, los autores señalan dos procesos de detección de caídas que muestran gran desempeño en la identificación de caídas. Otras investigaciones orientadas a la detección de estos comportamientos en sitios específicos generan soluciones para diferenciar actividades normales de una persona al entrar a un baño y caída simuladas, logrando una tasa de discriminación del 97.8%, destacando que las caídas se pueden detectar en menos de un segundo mientras que el sistema puede proteger la privacidad del sujeto [24].

2.3. Registro de imágenes

La definición del registro de imágenes dada en [25] enuncia que el registro de imágenes es un proceso donde se realiza la superposición de imágenes que han sido tomadas en diferentes momentos, ya sea usando diferentes puntos de vista o utilizando diferentes sensores con el objetivo de ser analizadas. El registro de imágenes tiene gran importancia cuando se combinan varias fuentes de datos de imágenes, para detectar cambios y en la restauración de imágenes multicanal.

En [26] se revisan técnicas de procesamiento de imágenes y reconocimiento de patrones que son utilizadas para analizar bio-imágenes, siendo el objetivo de los autores crear una guía tutorial para los investigadores que utilizan el procesamiento de imágenes, por ello, se abordan tareas como la transformación de niveles de gris, binarización, filtrado de imágenes, segmentación de imágenes, seguimiento visual de objetos, flujo óptico y el registro de imágenes. El trabajo muestra que el propósito del registro de imágenes es superponer dos imágenes mientras se ajustan entre sí de manera flexible, de modo que, si existe alguna diferencia aún después de la adaptación, esas imágenes contienen una diferencia intrínseca. Otro propósito del registro de imágenes es comprender la deformación relativa de una imagen con respecto a la otra imagen.

El registro de imágenes se ha utilizado también en el reconocimiento de huellas palmares, como se ve en [27], donde aplican un preprocesamiento que calcula el ancho de las palmas, logrando como principal aplicación la normalización

de la escala de las huellas. Al binarizar la imagen de la mano y colocar un punto de referencia, utilizan un algoritmo de seguimiento de límites, donde basándose en la distancia entre la referencia y los bordes de la mano, encuentran los puntos mínimos para trazar una línea que una a dichos puntos, de esta manera normalizan la rotación de la mano. Para normalizar en escala utilizan los bordes inferiores de la mano para redimensionar el ancho de la palma a 300 píxeles, aplicando una detección de la línea de la vida mediante el uso de la transformada de radón finita modificada. Finalmente realizan un recorte a la mitad de la palma para aplicar diferentes métodos de reconocimiento como PCA y LDA.

Otro trabajo donde se utiliza la normalización de imágenes de manos es [28], donde se emplea una base de datos de radiografías de manos para realizar una predicción de la edad ósea, usando la red Xception para entrenar un regresor con un conjunto nuevo de imágenes previamente segmentado y normalizado en rotación. La normalización de las radiografías es planteada en tres niveles, el primero realiza una segmentación de la región de la mano en las imágenes originales utilizando DeepLab V3 plus. El segundo paso, realiza la alineación del conjunto nuevo de imágenes segmentadas utilizando cuatro puntos clave de la imagen, obtenidos al aplicar un regresor separable de redes neuronales convolucionales MobileNet V1. Al trazar una recta entre los puntos clave del centro de la muñeca y la punta del dedo medio, se calcula el ángulo necesario para rotar la mano.

El uso de métodos como PCA y la normalización de imágenes ha contribuido a mejorar la detección de objetos, tal como en [29] donde adquieren imágenes de manos con variaciones en su ubicación y posición de los dedos, por ello, los autores proponen una normalización en la traslación y rotación de las manos para una correcta identificación. El método consta de segmentar primero la región de las manos, luego se identifica la posición de las puntas de los dedos y sus valles utilizando el contorno de la mano. Cada dedo se lleva a su orientación normalizada girándolo alrededor de la articulación metacarpiano-falange. Con este método obtuvieron un 99.59% de precisión en el reconocimiento de manos.

En [30] se propone XAOM, el cual es un método de alineación y orientación de imágenes de radiografías de 21 regiones corporales diferentes. Se conocen otros métodos para este propósito que se adaptan a una región del cuerpo específica y una forma de uso. Este método propuesto es capaz de alinear y orientar imágenes

de rayos X de cualquier región del cuerpo. Se experimentaron enfoques como la transformada de Hough, el algoritmo de detección de línea rápida y PCA. Siendo el Análisis de Componentes Principales el método con mejores resultados, dando una diferencia media del ángulo detectado y el obtenido previamente de 1.65° .

Una vez revisados los diferentes enfoques que se tienen para la detección de caídas. En contraste con los sistemas de detección que no utilizan algoritmos de aprendizaje automático, los que sí lo emplean muestran un mejor desempeño a la hora de identificar estos comportamientos, debido a que es posible entrenar al sistema para mitigar los falsos positivos que podrían generarse al realizar comportamientos muy parecidos a una caída, como recostarse, sentarse, etc. Nuestro sistema utiliza la ventaja de utilizar secuencias de imágenes que contienen comportamientos de caídas y no caídas, dichas secuencias son trabajadas utilizando el método PCA para extraer su vector de características y lograr separar más la línea entre los comportamientos que puedan ser muy parecidos a una caída.

3. Marco teórico

3.1. Termómetros de radiación

La temperatura se conoce como la medida física asociada con todos los campos de la ciencia. Los sensores térmicos se encuentran en muchos elementos, desde elementos comunes dentro de cualquier hogar de aplicaciones más sofisticadas. Todas las sustancias y objetos emiten radiación térmica cuando se encuentran a una temperatura superior al cero absoluto ($-273,15^\circ\text{C}$). Existe una relación entre la temperatura y la energía de radiación emitida que se puede utilizar para calcular la temperatura de la superficie del objeto, los termómetros de radiación se utilizan principalmente para medir la temperatura de un objeto desde cierta distancia [31].

Los termómetros de radiación realizan mediciones casi exclusivamente indirectas sin contacto con el objeto. Sin embargo, los termómetros de termopar y resistencia también pueden realizar una medición indirecta si no se colocan exactamente en la ubicación de interés.

3.1.1. El sensor MLX90640

El MLX90640 es un sensor térmico que entrega una imagen de 32x24 píxeles totalmente calibrada de fábrica. Este sensor contiene 768 píxeles FIR. Se integra un sensor ambiental para medir la temperatura ambiente del chip y un sensor de suministro para medir el VDD. Las salidas de todos los sensores IR, Ta y VDD se almacenan en la RAM interna y son accesibles a través de I²C, es muy utilizado en aplicaciones de detección de movimiento, detección de personas, en sistemas de control de temperatura de aires acondicionados, en hornos de microondas, en la industria es usado para el control de temperatura de partes en movimiento, entre otras [32].

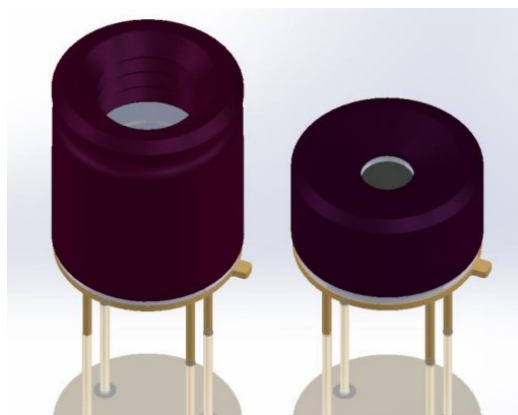


Figura 3.1 Sensores MLX90640 con resolución de 32×24 píxeles en diferente configuración del campo de visión, siendo uno de 55°×35° y el segundo de 110°×75° [12].

Tabla 3.1: Características del sensor MLX90640 obtenidas de la referencia [12].

Especificaciones del sensor MLX90640	
Voltaje de trabajo	3.3 V
Corriente de consumo	23 mA
Frecuencia de actualización	0.5 Hz – 64 Hz
Temperatura de operación	-40°C a 85°C
Rango de Temperatura del objetivo	-40°C a 300°C
Campo de visión	55°×35° y 110°×75°

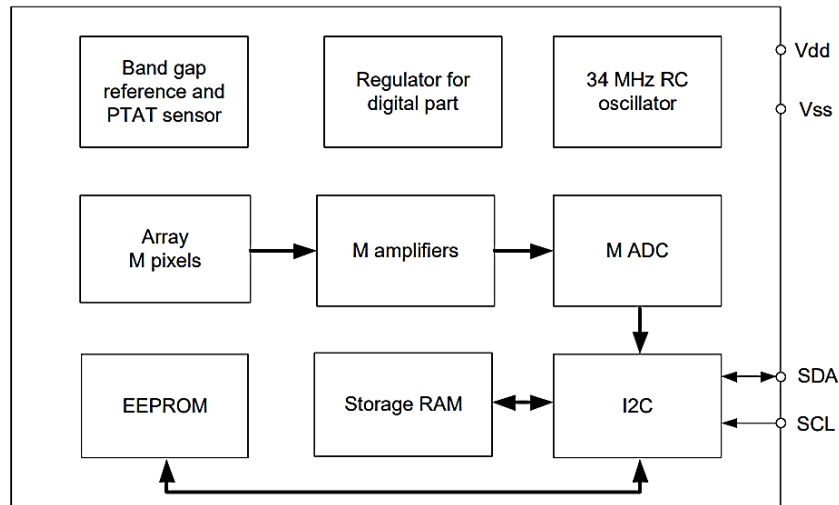


Figura 3.2 Diagrama de bloques del sensor MLX90640 donde se muestra el acceso a la información de los sensores almacenados en la RAM mediante el protocolo I²C extraído de [12].

3.2. Raspberry Pi

El hardware de código abierto se refiere a todo lo que se lanza al público y se requiere para producir un producto específico, como diagramas de circuitos, listas de materiales y diagramas de placas de circuitos impresos. En otras palabras, no existe una licencia de patente para la tecnología, y los recursos necesarios para el desarrollo están a disposición del público, lo que facilita la configuración del sistema. Por lo tanto, se puede utilizar como una nueva aplicación de código abierto utilizando sensores y modificando la programación del código abierto para que se ajuste a la función deseada.

La minicomputadora Raspberry Pi es un dispositivo popularizado por ser una computadora de bajo costo, actualmente en su última versión (Modelo Pi 4B), es posible adquirirla con la cantidad de memoria RAM que el usuario mejor requiera, siendo posible elegir entre 2 GB, 4 GB y 8 GB. La velocidad de su procesador Broadcom BCM271 Quad-Core de 64 bits de alto rendimiento, puede alcanzar velocidades de 1,5 GHz. También cuenta con dos puertos micro HDMI, los cuales puede decodificar resoluciones de hasta 4K, cuenta con una LAN inalámbrica de banda dual para 2.4 y 5.0 GHz, Gigabit Ethernet, 2 puerto 3.0 y Bluetooth 5.0. [33].

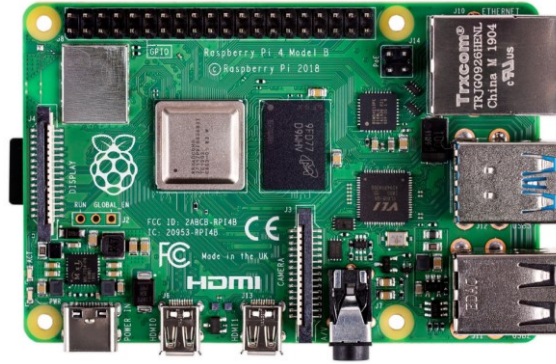


Figura 3.3 Tarjeta Raspberry Pi 4B 8 GB la cual provee al usuario de un incremento en la velocidad de los procesos de multimedia, procesador y conectividad [33].

Dentro de sus pines GPIO es posible encontrar el protocolo de comunicación I²C. El sistema operativo que utiliza esta minicomputadora es Linux, este es conocido por ser gratuito y cuenta con mantenimiento por parte de voluntarios que trabajan de forma gratuita.

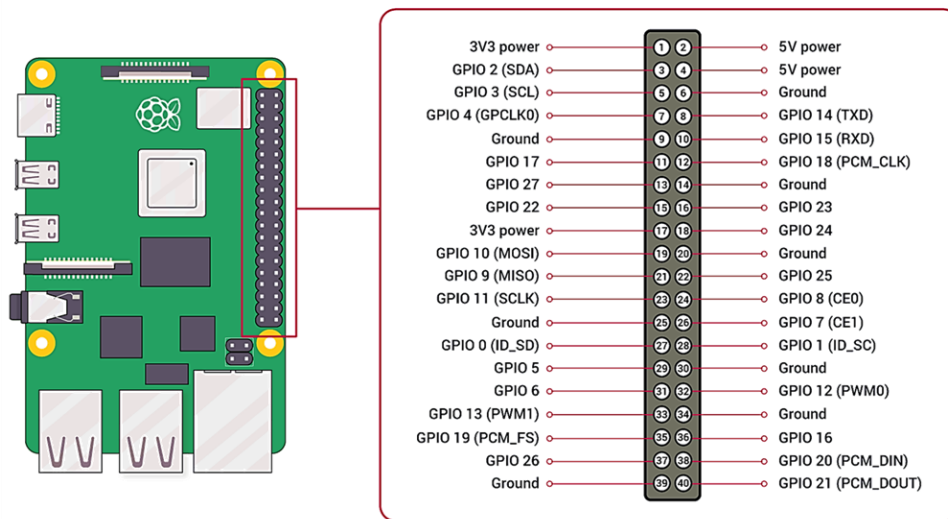


Figura 3.4 Localización de pines de alimentación y de funciones especiales, además de pines de comunicación I²C de la Raspberry Pi 4B [34].

3.3. Protocolo de Comunicación I²C

El I²C (Inter-Integrated Circuit) se define como un protocolo digital rápido de comunicación que requiere dos cables referentes a la comunicación, SDA (Serial Data) y SCL (Serial Clock) y los cables de voltaje y tierra para alimentar al dispositivo a utilizar, Este bus es un protocolo en serie síncrono desarrollado por Philips Semiconductor (ahora NXP Semiconductors) a principios de la década de 1980 para admitir la interconexión a nivel de placa de módulos IC y periféricos [35].

El protocolo I²C también es conocido como IIC ó TWI (Two Wire Interface), es quizás el protocolo más común que se encuentra en las placas de circuito impreso (PCB). La mayoría de las placas, como Arduino y Raspberry Pi, admiten este protocolo. Además, esta comunicación permite conectar hasta 127 dispositivos esclavos con esas dos líneas, con hasta velocidades de 100, 400 y 1000 kbits/s [36].

La conexión de tantos dispositivos al mismo bus es una de las principales ventajas. Además, si comparamos al I²C con otro protocolo serial, como Serial TTL, este incluye más bits en su trama de comunicación que permite enviar mensajes más completos y detallados. El I²C contiene dos elementos básicos, un maestro y un esclavo, El maestro puede comunicarse con el esclavo porque cada dispositivo del esclavo tiene su propia dirección. Similar a la comunicación serial UART, I²C tiene bits de inicio y parada. Para generar el bit de inicio, el SDA debe reducirse manteniendo SCL alto y para generar el bit de parada, el SDA se debe bajar manteniendo el SCL alto. Y todos los demás bits relacionados con los datos, la dirección del esclavo, el reconocimiento de los datos y la dirección del flujo de datos (maestro a esclavo o esclavo a maestro) se señalarán entre los bits de inicio y parada.

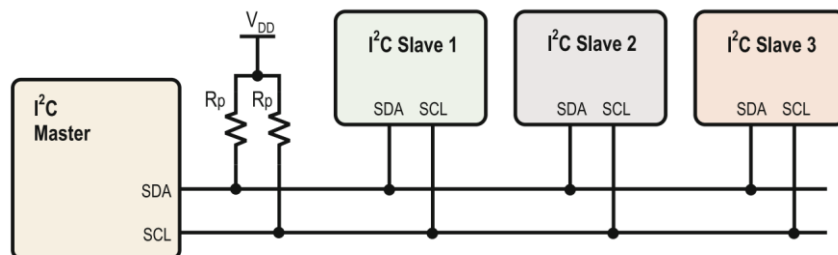


Figura 3.5 Diagrama de bloques de un ejemplo de conexión maestro y esclavo para el protocolo de comunicación I²C [35].

3.4. Inteligencia Artificial

Para comprender lo que es la inteligencia artificial, es necesario entender primero el concepto de inteligencia. La inteligencia es una característica que nos identifica como especie *Homo sapiens*. A lo largo de la evolución, el ser humano ha buscado comprender cómo funciona nuestro cerebro, cómo pensamos y percibimos el entorno, y cómo podemos predecir y manipular el mundo que nos rodea. La inteligencia artificial se enfoca en crear entidades inteligentes, lo cual es un camino aún más grande y desafiante que comprender simplemente cómo funciona la inteligencia humana. El desarrollo de investigaciones en relación con la Inteligencia Artificial es relativamente nuevo, y se sabe que tuvo un gran impulso después de la Segunda Guerra Mundial, dando lugar al nombre con que hoy en día se conoce en el año 1956. Actualmente, es conocido que la IA se encuentra presente en gran variedad de subcampos, donde pueden resolverse problemas desde puntos de vista generales como aprendizaje y percepción, hasta tareas específicas, como jugar al ajedrez, demostraciones matemáticas, escribir poesía, conducir un automóvil en una calle concurrida y diagnosticar enfermedades. Se pueden encontrar en las aplicaciones de la Inteligencia Artificial varias áreas de estudio [37], las cuales podemos describir como:

- Procesamiento de lenguaje natural: son sistemas capaces de reconocer, procesar y emular el lenguaje humano.
- Sistemas expertos: son sistemas que imitan el comportamiento de un experto humano realizando tareas de toma de decisiones a partir de información suministrada previamente.
- Robótica: Es el diseño e implementación de máquinas capaces de percibir el entorno y desempeñar actividades que requieren de inteligencia humana.
- Machine Learning: esta área otorga la capacidad a las computadoras de “aprender” imitando el aprendizaje humano.

3.5. Machine Learning

Machine Learning o Aprendizaje Automático es un campo de investigación que se enfoca en la extracción de conocimientos y patrones a partir de datos mediante técnicas estadísticas, inteligencia artificial y tecnologías de información. Es conocido también como análisis predictivo o aprendizaje estadístico. El objetivo es desarrollar modelos y algoritmos que puedan aprender de los datos y mejorar su habilidad para realizar tareas específicas sin ser programados explícitamente. La aplicación de métodos de aprendizaje automático se ha vuelto omnipresente en los últimos años en la vida cotidiana, estos algoritmos se utilizan en una amplia variedad de aplicaciones prácticas en la vida cotidiana, como recomendaciones automáticas de películas, comida y productos, personalización de la radio en línea y reconocimiento de caras en fotos. Muchos sitios web y dispositivos modernos tienen algoritmos de aprendizaje automático en su núcleo, lo que permite una experiencia personalizada para el usuario y mejora la eficiencia en las tareas. Esta herramienta poderosa que se ha aplicado a una variedad de problemas científicos importantes, como la comprensión de estrellas, la búsqueda de planetas distantes y el análisis de secuencias de ADN. Sin embargo, no es necesario que su aplicación sea tan grande para beneficiarse del aprendizaje automático [38].

3.5.1. Clasificador KNN Ponderado

El modelo de Aprendizaje Automático KNN (K-Nearest Neighbors), es un algoritmo de aprendizaje no paramétrico, esto quiere decir que no se puede caracterizar por un conjunto limitado de parámetros, de modo que se basa en la idea de que un punto dado es similar a los puntos que lo rodean en el espacio de características. En KNN, datos de entrenamiento se utilizan como "vecinos" para clasificar un punto de datos desconocido. El algoritmo determina la clase de un punto desconocido basándose en la clase de los puntos más cercanos en el conjunto de entrenamiento. El número de vecinos más cercanos que se consideran se define mediante el parámetro "k". Este algoritmo a menudo requiere un gran número de datos de entrenamiento y un tiempo de cálculo más largo para procesar nuevos datos [37].

Al utilizar KNN para clasificar una serie de entradas consiste en encontrar los ejemplos más cercanos para cada consulta, por esto se le denomina búsqueda de k vecinos más cercanos. Para hacer la clasificación, asumimos que todas las instancias corresponden a puntos en el espacio n -dimensional, si tomamos la instancia x y está descrita por el vector de características: $(a_1(x), a_2(x), \dots, a_n(x))$, donde $a_r(x)$ denota el valor de la i -ésima característica de x , de esta manera, la distancia entre las instancias x_i y x_j está definida como $d(x_i, x_j)$, la cual corresponde a la distancia Euclidiana, que está definida como:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2}$$

Ecuación 3.1

Para demostrar el criterio que se emplea en clasificación definamos 1 vecino más cercano ($k = 1$), entonces el algoritmo K-NN asigna a la muestra x_j el valor x_i más cercano de entrenamiento. Para valores mayores de k , el algoritmo asigna el valor más común entre los k ejemplos de entrenamiento más cercanos [39].

Hasta ahora se ha explicado el funcionamiento un clasificador K-NN normal, de modo que al agregar una función gaussiana mostrada en la ecuación 3.2, que permita optimizar nuestras distancias al multiplicarlas por un peso obtendremos la versión ponderada del clasificador K-NN. Estos valores de distancias se verán afectados de forma que se le dé más peso a los vecinos que se encuentren más cercanos de la muestra y mientras la distancia crezca se reduzca el peso de los vecinos.

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

Ecuación 3.2

3.6. Procesamiento digital de imágenes

Una imagen digital es una función de un número finito de elementos, cada uno de los cuales tiene una ubicación y un valor particulares. Estos elementos se conocen como elementos de imagen, elementos de imagen y píxeles. El término píxel se usa ampliamente para representar una imagen digital. El procesamiento de imágenes implica cambiar la naturaleza de una imagen para poder adecuar su información y poder interpretarla de mejor manera, asimismo, permite mejorar el desempeño de otros algoritmos que requieran encontrar características dentro de estas imágenes. Los pasos básicos del procesamiento de imágenes involucrados son la adquisición de imágenes, la mejora de imágenes, la restauración de imágenes, el procesamiento de imágenes en color, el procesamiento de ondas y múltiples resoluciones, la compresión, el procesamiento morfológico, la segmentación, la representación y descripción, y el reconocimiento de objetos [40].

3.6.1. Filtro de mediana

Los filtros de mediana tienen la capacidad de minimizar la pérdida de nitidez en imágenes, además del nivel de desenfoco. El filtro almacena cada píxel con el valor del píxel medio obtenido después de que los píxeles de la vecindad se hayan clasificado en orden creciente. El píxel mediano tiene el valor más alto de la primera mitad de los píxeles de la vecindad y el valor más bajo de la otra mitad de los píxeles restantes. Recreando un ejemplo sencillo donde se elige una máscara de 3×3 , el método de filtrado se muestra en la figura 3.6.

El valor del píxel que se examina tiene un valor de 200, este es reemplazado por la mediana dentro de la máscara utilizada, que es 87. En general, para ventanas de dimensiones $L \times L$. El filtro mediano es muy eficaz para la reducción del ruido impulsivo de los llamados sal y pimienta (puntos dispersos blancos y negros). El modo de funcionamiento del filtro de mediana es similar al de los filtros lineales, en el sentido de que, de manera análoga a la máscara de convolución, para cada píxel de la imagen se coloca una ventana, para identificar los píxeles a colocar en aumento. pedido.

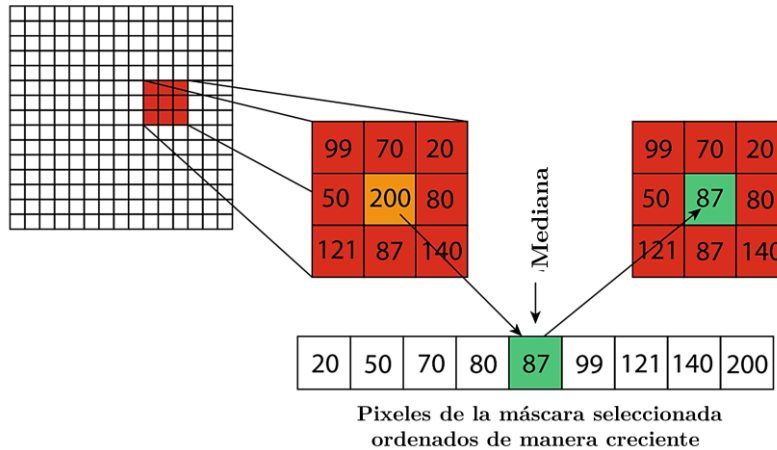


Figura 3.6 Esquema demostrativo de la metodología para utilizar un filtro de mediana usando una ventana de 3×3 pixeles [41].

3.7. Análisis de Componentes Principales

El método de Análisis de Componentes Principales (ACP) o mejor conocido en inglés como Principal Components Analysis (PCA), es utilizado cuando se necesita disminuir la cantidad de dimensiones en las que una base de datos está representada, de modo que se logra mantener las características importantes de los datos. Esta transformación que sufre el conjunto de estudio permite obtener un nuevo conjunto que no está correlacionado, debido a que se toman de forma ordenada las características de mayor variación encontradas en el conjunto original [42].

El PCA es un análisis de la estructura de covarianza de datos multivariados. En busca encontrar las direcciones en los datos con la mayor varianza. Dada un set de datos M podemos encontrar su matriz de covarianza de la siguiente manera:

$$C = \frac{M^T M}{n}$$

La matriz de covarianza hace referencia a una matriz $(d \times d)$ en la que la (i, j) ésimas entrada es la covarianza entre las dimensiones i y j [43].

4. Método propuesto

Este capítulo contempla el proceso para desarrollar un método de adquisición de datos térmicos, el cual requiere de la sinergia entre software capaz de configurar la minicomputadora para funcionar con los sensores térmicos y elementos físicos que permitan una captura de imágenes térmicas óptimas.

4.1. Descripción del sistema

Se ha diseñado un sistema utilizando aprendizaje automático implementado con lenguaje Python, dicho sistema propuesto utiliza diferentes etapas para llegar a la detección de una caída, primero es necesario realizar una normalización en rotación, traslación y escala del objeto de interés, esto es logrado utilizando 3 algoritmos que en conjunto permiten identificar al objeto de mayor interés en los fotogramas térmicos, esto es realizado obteniendo el centroide del objeto que irradia más calor en la escena, después se determina el tamaño utilizando un cálculo de la densidad dentro de 8 anillos, los cuales parten del centroide previamente calculado y aumentan su radio con la finalidad de encontrar el radio máximo del objeto. Una vez encontrado el centroide de la persona y su tamaño, se realiza un recorte en la imagen, para después utilizar un algoritmo que permite encontrar el ángulo dominante de esta persona y así poder realizar una corrección angular. De esta manera se obtiene un nuevo set de fotogramas de las secuencias ahora normalizadas. Este sistema se muestra en la figura 4.1.

Al contar con un nuevo set de imágenes normalizado podemos aplicar el método de análisis de componentes principales (PCA), este método tiene la ventaja de funcionar de manera óptima cuando se utiliza en imágenes similares entre sí. Después, se utilizan las características reducidas mediante PCA para clasificar los datos utilizando K-NN ponderado. El resultado es un vector de probabilidad de que el sujeto se encuentre acostado en los fotogramas. Este vector es clasificado nuevamente con K-NN utilizando una base de datos de vectores de probabilidad pertenecientes a secuencias de caídas y no caídas reducida en dimensionalidad. El resultado final es una clasificación de caída o no caída.

Para desarrollar las etapas del sistema propuesto fue necesario desarrollar previamente un método para el almacenamiento de secuencias térmicas utilizando el sensor MLX90640 mediante el protocolo de comunicación I²C [32]. Para esto, se seleccionó el dispositivo Raspberry Pi [44], ya que es una minicomputadora que puede trabajar con el lenguaje de programación necesario y acelerar el proceso para determinar la frecuencia óptima de lectura, permitiendo trabajar en el mismo bus de comunicación. Esto se realiza con el fin de utilizar las lecturas para la identificación de caídas. Para construir una base de datos sobre el comportamiento a identificar, se diseñó un algoritmo para almacenar las lecturas del sensor térmico, después, se realizaron simulacros de caídas, con el fin de obtener los ejemplos suficientes para entrenar un clasificador preliminar.

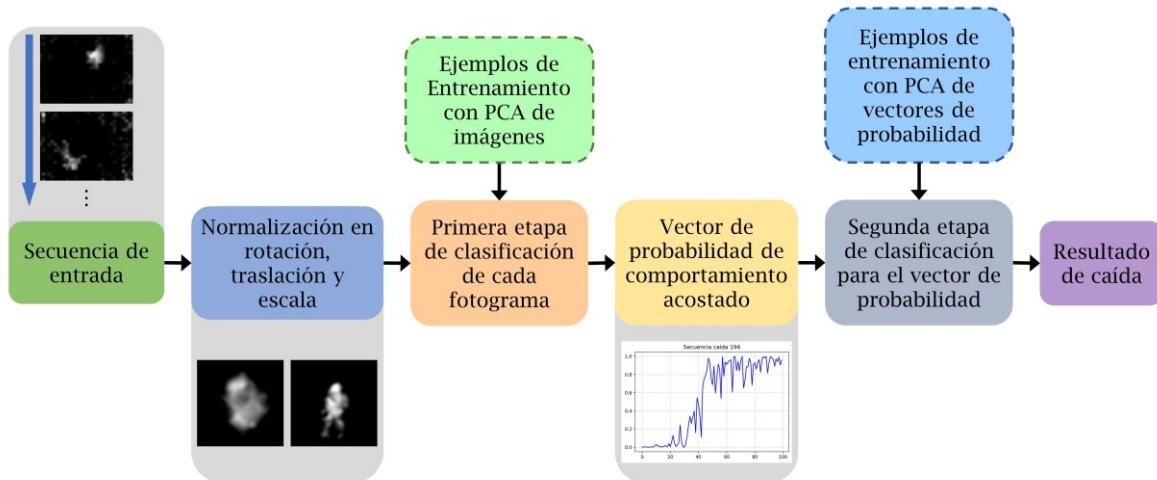


Figura 4.1 Esquema del funcionamiento del sistema utilizando secuencias térmicas e imágenes de entrenamiento de una persona estando acostada y parada para construir un vector de probabilidades que permita clasificar si la secuencia pertenece a una caída o no caída.

Después de realizó una mejora al sistema propuesto, dando un foque diferente del uso de los vectores reducidos de características, para lograr describir de una manera más exacta las secuencias de caídas y no caídas, se unieron los 100 vectores de características resultantes de cada fotograma de la secuencia, de esta forma se creó un nuevo set de datos, el cual fue reducido nuevamente con el método PCA. De esta manera, obtenemos las características más importantes de toda la secuencia térmica. Dicha propuesta se observa en la figura 4.2, en la cual observamos que del

vector de probabilidades son concatenados para obtener un solo vector que describa de mejor manera la secuencia de una caída y una no caída.

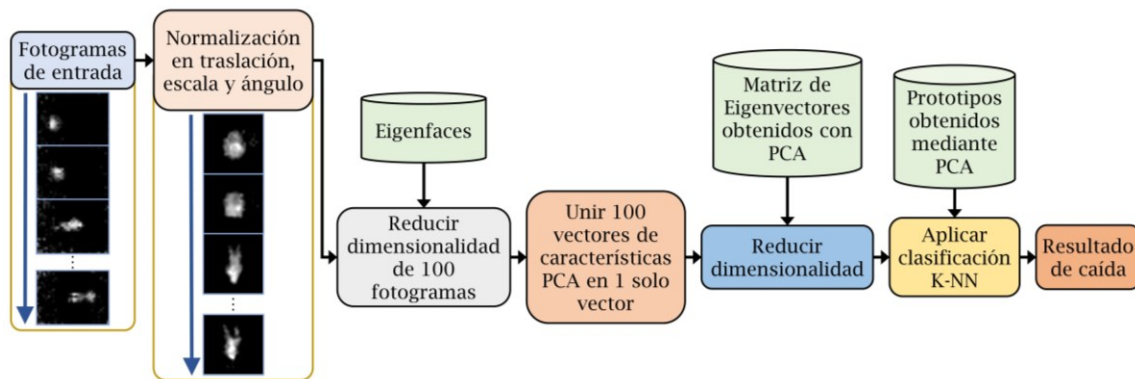


Figura 4.2 Esquema de la optimización del sistema propuesto, utilizando las características resultantes de aplicar PCA para construir un vector de características y utilizarlo en la clasificación de una caída y no caída.

4.2. Diseño de un algoritmo de almacenamiento

Una vez enlazados los cables de comunicación entre la Raspberry Pi y el sensor térmico, es necesario desarrollar un algoritmo que permita almacenar los datos tal cual llegan del sensor, esto con la finalidad de aplicar un procesamiento de ellos posteriormente. Por lo tanto, se inició proponiendo el diagrama de flujo de la figura 4.3, cabe mencionar que se requirieron varias pruebas de calibración de las temperaturas obtenidas del sensor térmico, realizando un ensamble previo mostrado en la figura 4.4, estas pruebas tuvieron como resultado la obtención de un rango de distancias ideales para la captura del cuerpo humano a temperatura ambiente, como se observa en la figura 4.5. El código generado en lenguaje Python se muestra en el Anexo (a).

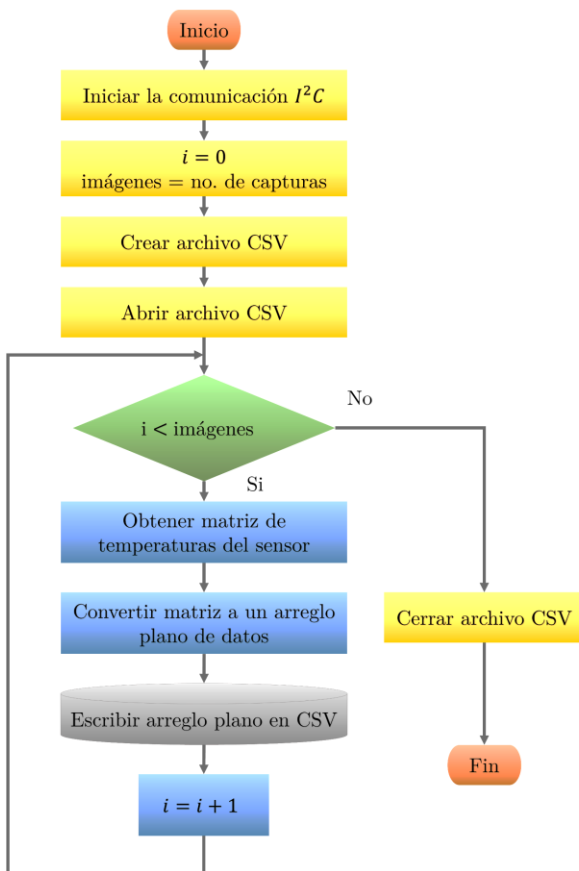


Figura 4.3 Diagrama de flujo del algoritmo diseñado para el almacenamiento de las secuencias térmicas.

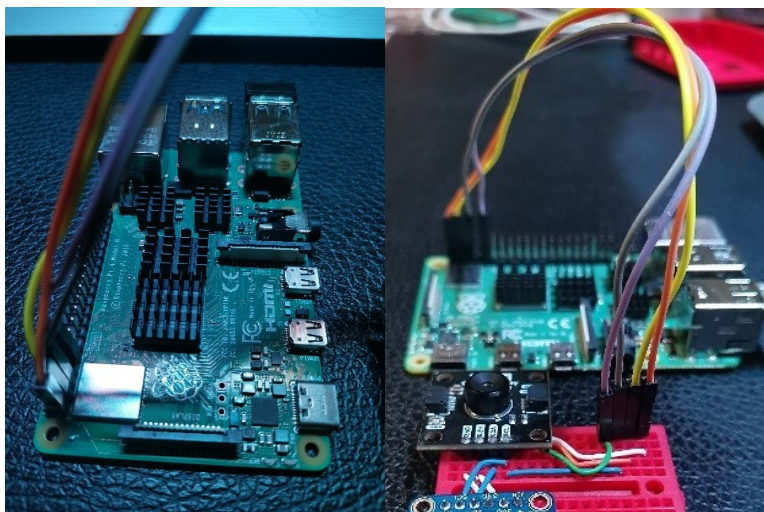


Figura 4.4 Integración del sensor térmico mediante el protocolo de comunicación I²C usando los pines SDA Y SCL en una Raspberry Pi 4.

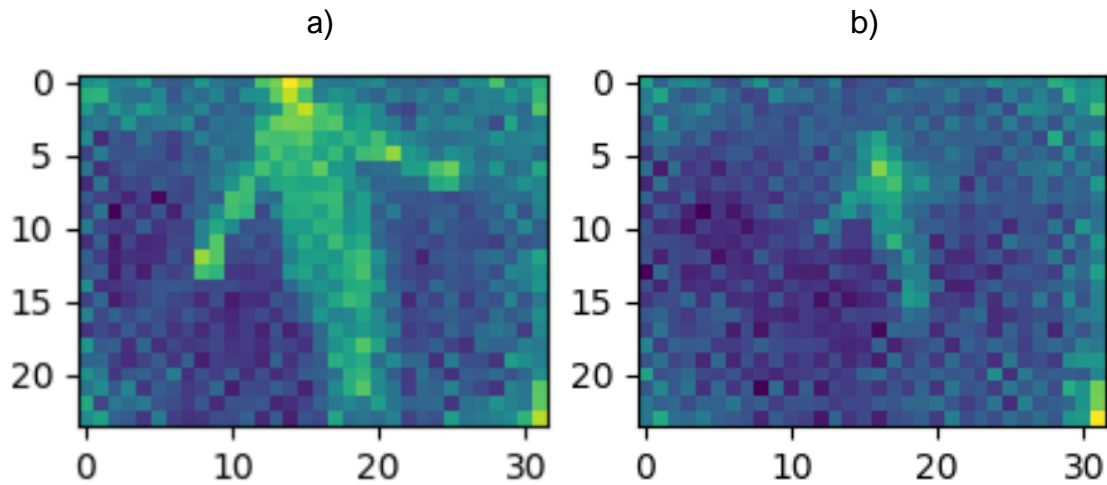


Figura 4.5 Captura de datos térmicos del sensor MLX90640, con el sujeto de prueba una distancia de a 2 m (a) y 4 m (a).

4.3. Construcción de una base de datos térmicos

El lugar óptimo de captura visto en muchos trabajos de investigación fue la vista aérea, dado que el cuerpo humano tendrá un perfil de mayor dimensión en la caída que en comparación con otros comportamientos como caminar o estar sentado. Para esto se diseñó un sistema de montaje de techo para la minicomputadora y el sensor térmico mostrado en la figura 4.6.

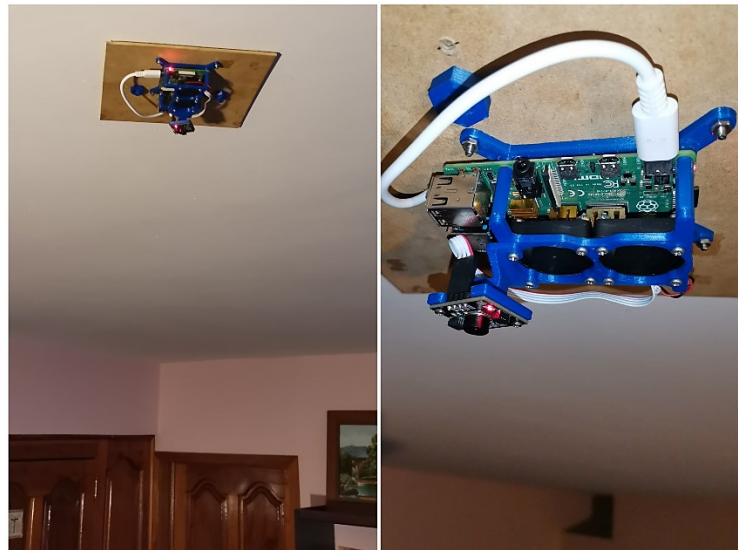


Figura 4.6 Montaje del dispositivo integrado de captura de secuencias térmicas en el techo de la habitación.

Para comenzar a capturar los datos necesarios para identificar una caída, es necesario simular un gran número de caídas como se muestra en la figura 4.7, esto con el fin de aportar los suficientes ejemplos para el entrenamiento del clasificador a desarrollar posteriormente.

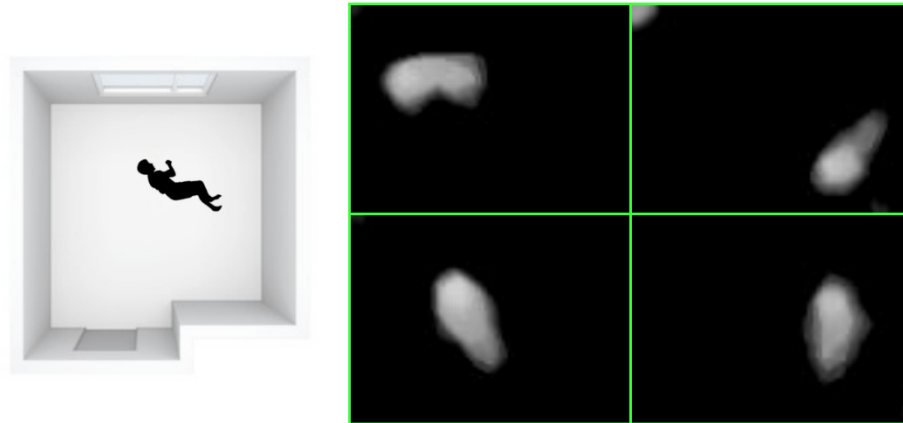


Figura 4.7 Datos obtenidos del sensor MLX90640 simulando una caída.

Se obtuvieron un total de 4000 ejemplos del comportamiento acostado y 4000 ejemplos del comportamiento parado, con los cuales se procederá al entrenamiento del clasificador.

Las capturas del cuerpo humano de la figura 4.8, mientras se realiza una caminata son bastante diferenciables a la vista, este dato nos permite corroborar que existen características suficientes para que un algoritmo de aprendizaje maquina pueda aprender a clasificar estos comportamientos.

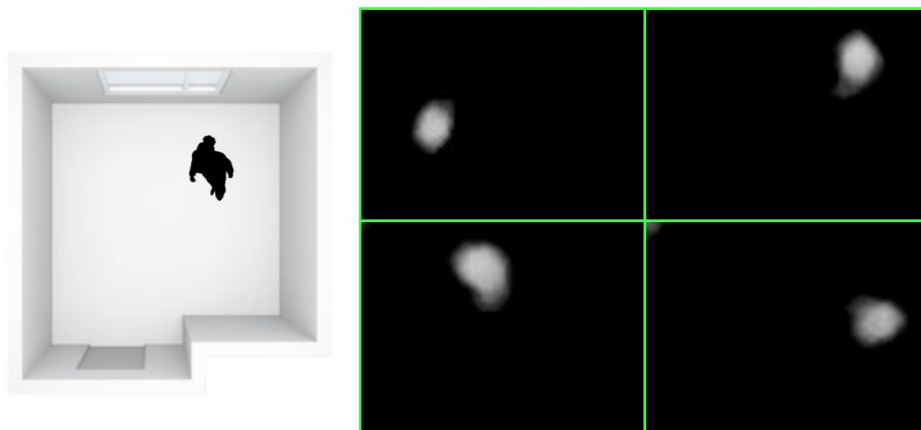


Figura 4.8 Datos obtenidos del sensor MLX90640 durante una caminata.

4.4. Preprocesamiento de los datos

Para obtener imágenes con reducción de ruido térmico, se propone un método de procesamiento digital de imágenes, donde la primera tarea es aplicar un filtro de umbral, donde se extrae gran parte del ruido de fondo de la imagen térmica, como se puede observar en la figura 4.9, el umbral no es del todo suficiente, ya que quedan aún píxeles de ruido, lo cual nos lleva a aplicar un segundo filtro, el cual es se realiza usando el método de la mediana. De esta manera obtenemos imágenes más claras y enfocadas a detectar un cuerpo humano.

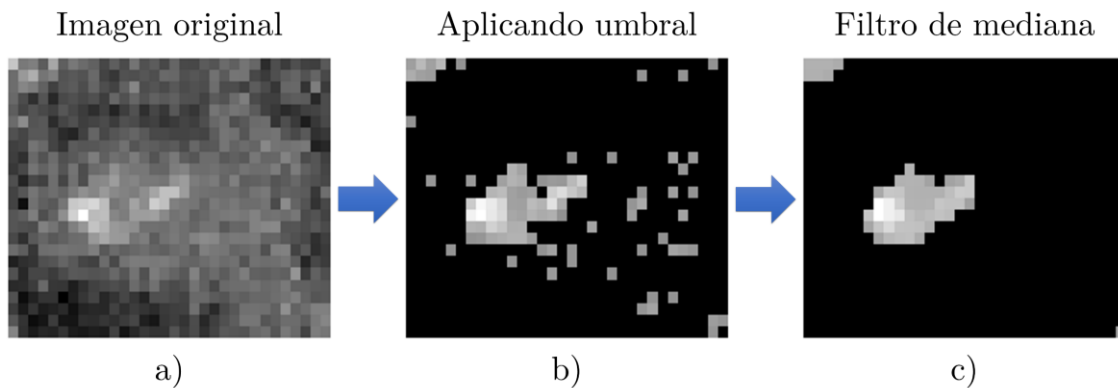


Figura 4.9 Procesamiento digital de imagen aplicado.

4.4.1. Normalización en rotación, traslación y escala del conjunto de entrenamiento

Para lograr mejores resultados a la hora de utilizar un clasificador, dadas las condiciones de no contar con extensas bases de datos acerca de caídas con visión térmica, se propone utilizar un método capaz de centrarse en la región de interés y así, obtener un nuevo conjunto de entrenamiento donde se ha eliminado la variación en la rotación, traslación y escala. El diagrama descrito en la figura 4.10, muestra el tipo de imagen de entrada y la salida que se obtiene después de aplicar el algoritmo de normalización.

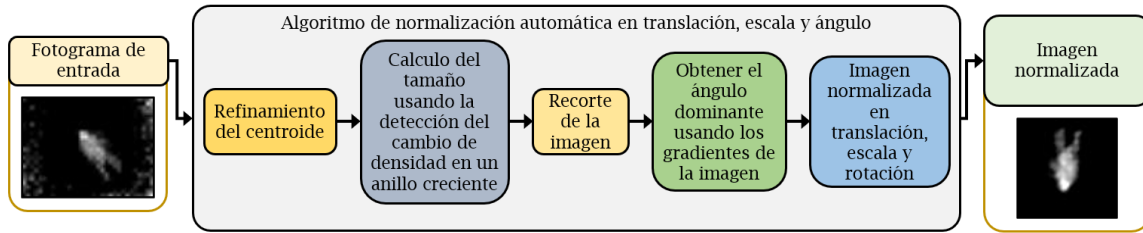


Figura 4.10 Diagrama general del proceso de normalización en rotación, traslación y escala para una imagen térmica de una caída.

El primer paso utilizado es el redimensionamiento de la imagen, donde se espera que, si el conjunto de entrenamiento cuenta con imágenes en diferentes resoluciones, esto no afecte a la hora de emplear el algoritmo. Además, si se tuviera un cambio de sensor continuaría el buen funcionamiento del método desarrollado.

Después, se procede a obtener el centro de masa de la imagen, debido a que se ha demostrado de forma experimental que el lugar con los píxeles de valor más alto de temperatura pertenece al cuerpo de la persona, dando lugar a que algunos ruidos térmicos en la muestra se terminen eliminando. Al ubicar el centro de masa, se procede a colocar una circunferencia de radio r_m en dicho punto, siendo por primera vez igual a 120 píxeles de radio. con el fin de volver cero todos los píxeles fuera de dicha circunferencia, posteriormente se realiza un ciclo donde el centroide es recalculado para la nueva imagen obtenida, utilizando tres variables que definen el criterio de paro para el algoritmo, **V1** que está relacionada a los ciclos máximos esperados del algoritmo completo, **V2** que define la cantidad máxima de ciclos en que la distancia euclidiana entre el centroide actual y anterior sea menor o igual a 1 para romper el ciclo, **V3** define la cantidad máxima de ciclos que pueden realizarse mientras se tenga que la distancia entre centroides sea mayor que 1.

En cada ciclo transcurrido habrá un decremento del radio de la máscara circular de 5 píxeles, siendo el radio mínimo otro criterio de paro, definido por un radio de no menos de 40 píxeles, el cual se conoce de manera experimental. Este proceso de refinamiento del centroide está descrito en el algoritmo mostrado en el Algoritmo 1 y se aprecia su funcionamiento en la figura 4.11, donde se aprecia que se obtiene solo el centro de masa del objeto más importante en la imagen.

Algoritmo 1: Algoritmo para el refinamiento del centroide usando un círculo binario de radio decreciente.

Entradas: imagen $I(m,n)$, mascara circular $M(m, n)$

Salidas: Radio final r

COMIENZO

INICIALIZAR

$r_{min} = 40 ; r_{max} = 120 ; \Delta r = 5 ; V1 = 0 ; V2 = 0 ; V3 = 0$

CALCULAR

Centroide de $I(m, n)$ y asignarlo a (x_{cprev}, y_{cprev})

ASIGNAR

$r \leftarrow r_{max}$

MIENTRAS ($V1 < 45$ y $V2 < 20$ y $V3 < 20$ y $r > 40$) **HACER:**

Crear imagen binaria $M(m, n)$ con un círculo en (x_{cprev}, y_{cprev}) y radio r

Hacer $I_m(m, n) \leftarrow I(m, n)M(m, n)$

Calcular el centroide de $I_m(m, n)$ y asignarlo a (x_c, y_c)

Calcular la distancia Euclidiana d entre (x_{cprev}, y_{cprev}) y (x_c, y_c)

Si $d < 1$ entonces:

Hacer $r \leftarrow r - \Delta r ; V2 \leftarrow V2 + 1 ; V3 \leftarrow 0$

En otro caso:

Hacer $V2 \leftarrow 0 ; V3 \leftarrow V3 + 1$

Hacer $V1 \leftarrow V1 + 1 ; (x_{cprev}, y_{cprev}) \leftarrow (x_c, y_c)$

FIN

FIN

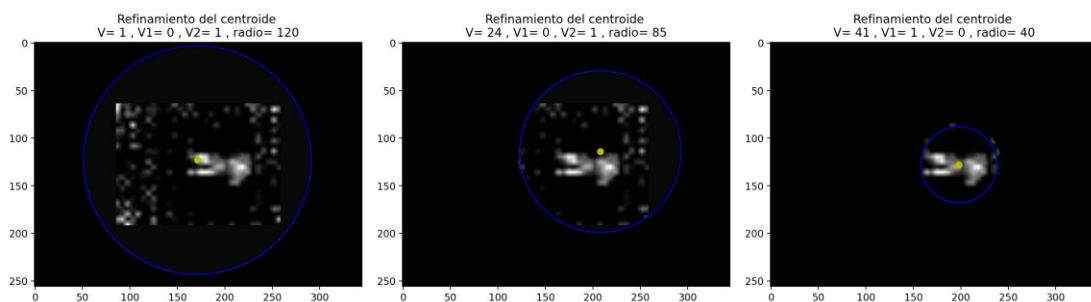


Figura 4.11 Proceso de búsqueda y ajuste del centroide comenzando de izquierda a derecha sobre la imagen para encontrar la región de interés en una caída.

Continuando con el proceso de normalización, ahora se realiza la obtención de la escala del objeto de interés, partiendo de la hipótesis de que la caída de densidad de los pixeles con forme se aleja del centroide es información suficiente para

conocer el tamaño del objeto. Por este motivo se propuso medir la densidad en región de tipo anillo cuyo radio crece gradualmente. Así, el método comienza asignando a una constante c un valor de 1,32. Después se inicializa un conjunto \mathbf{R} con valores para el radio de cada anillo, siendo en total 18 elementos, y el segundo conjunto contiene los valores de área de cada elemento del conjunto \mathbf{R} .

Para detectar la caída en el valor de densidad cuando los anillos crecen hasta llegar a los extremos del cuerpo humano, se debe normalizar los valores de densidad de 18 anillos para homogeneizar la curva de disminución de densidad y establecer un único umbral. El proceso propuesto consiste en ordenar el conjunto de densidades \mathbf{D} de mayor a menor, tomar los dos primeros valores máximos D_1 y D_2 , calcular el promedio \bar{m} entre ellos y normalizar el conjunto de densidades dividiendo cada elemento D_i por \bar{m} para obtener el conjunto normalizado \mathbf{E} . Luego, se establece un umbral experimental de 0.2 para determinar la densidad mínima a la que debe llegar el algoritmo en todas las imágenes. Este proceso se puede representar en un algoritmo como el Algoritmo 2 que menciona.

Algoritmo 2: Detección de la caída de densidad para establecer la región del cuerpo humano.

Entradas: imagen $I(m, n)$, bordes $E(m, n)$

Salidas: imagen recortada $I_E(m, n)$

COMIENZO

INICIALIZAR

$$c = 1.32 ; R = \{r_1, r_2, r_3 \dots r_{18} \mid r_i = 30c^i\}$$

CALCULAR

$$A = \{A_1, A_2, A_3 \dots A_{18} \mid A_i = \pi r_i^2\}$$

CREAR MÁSCARAS

$$M = \{M_1(m, n), M_2(m, n), M_3(m, n) \dots M_{18}(m, n)\}$$

HACEMOS

$$M_E = \{E(m, n)M_i(m, n) \mid M_i(m, n) \in M\}$$

INICIALIZAR ANILLOS Y ÁREAS

$$H = \{H_1, H_2, H_3 \dots H_{18}\}$$

$$H_A = \{H_{A1}, H_{A2}, H_{A3} \dots H_{A18}\}$$

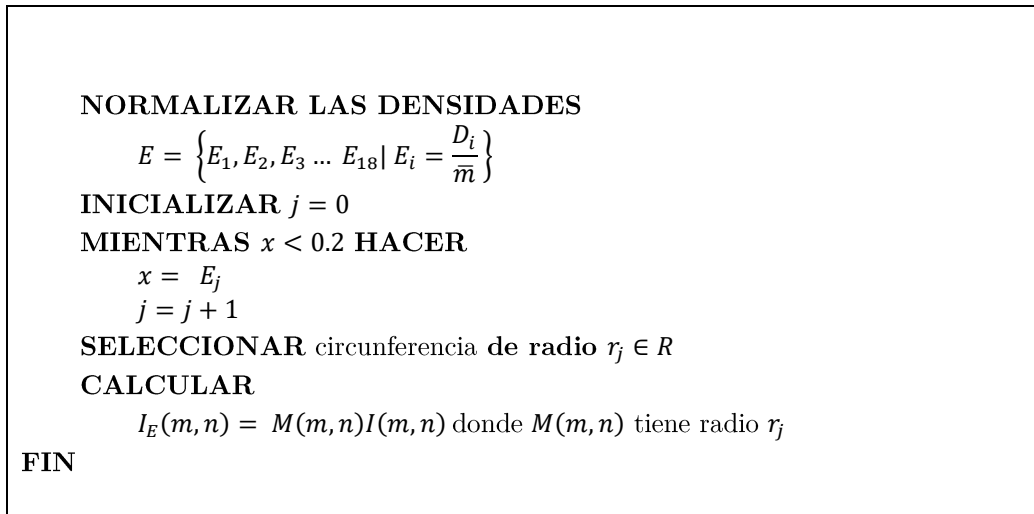
CALCULAMOS LA DENSIDAD

$$D = \{D_1, D_2, D_3 \dots D_{18} \mid D_i = \frac{H_i}{H_{Ai}}\}$$

ORDENAMOS D comenzando por el valor más grande

CALCULAMOS

$$\bar{m} = \frac{D_1 + D_2}{2}$$



El proceso del cálculo de la caída de densidad está representado en la figura 4.12, la cual su radio crece con cada iteración, permitiendo encontrar la zona donde la densidad decrece hasta un umbral dado, asegurando que el objeto está contenido en ese radio descrito.

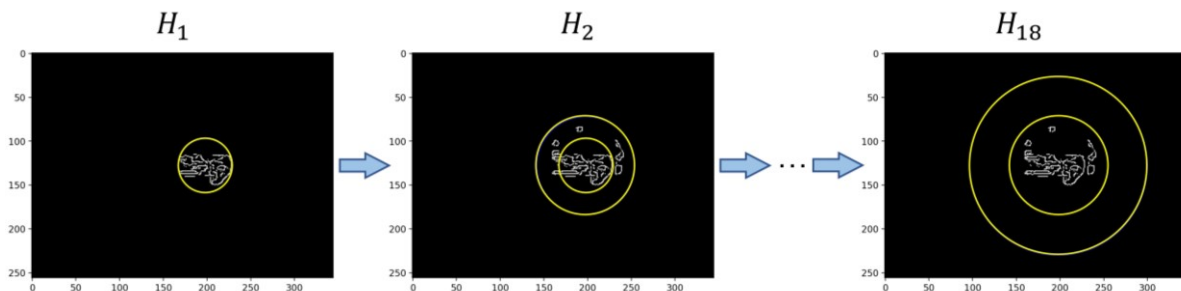


Figura 4.12 Proceso de cálculo de la densidad para normalizar la escala del objeto de interés en una imagen de caída.

Una vez terminado el Algoritmo 2, se obtiene la imagen normalizada únicamente en traslación y escala mostrada en la figura 4.13. Siendo el último paso encontrar el ángulo necesario para rotar la imagen en dirección horaria y obtener un objeto alineado verticalmente.

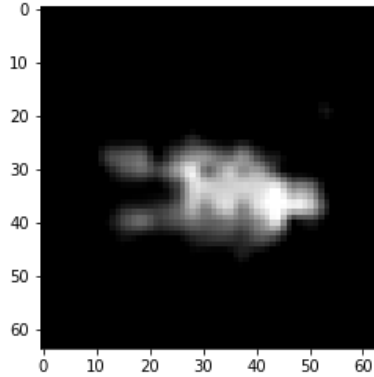


Figura 4.13 Recorte del objeto de interés utilizando el algoritmo de normalización de escala.

Proponemos una técnica que permite corregir el ángulo de las imágenes de la base de datos. El primer paso es calcular el histograma de gradientes tanto horizontales como verticales de la imagen original $I^g(m, n)$. El mapa de gradientes \mathbf{G} está compuesto por dos componentes, la primera define las magnitudes $G_m(m, n)$ y la segunda contiene las direcciones $G_a(m, n)$ de dichas magnitudes.

Cuando utilizamos las componentes G_m y G_a es posible calcular un histograma de ángulos con 360 intervalos, donde la componente en el eje ordenado se obtiene mediante la suma de todas las magnitudes de gradiente de los vectores de gradiente dentro del intervalo angular correspondiente representado por cada intervalo. Al mismo tiempo, cada columna de este histograma podría representarse como un vector cuya magnitud corresponde a la altura en el histograma y su ángulo corresponde a la etiqueta de su respectivo contenedor. Con esta información creamos un histograma polar de $I^m(m, n)$, que contiene un total de 360 vectores, mostrado en la figura 4.14, en esta etapa es posible observar muchas variaciones en el seguimiento de las direcciones y magnitudes. Con la finalidad de suavizar este histograma original, creamos un nuevo vector H_i donde: $i = 1, 2, 3, \dots, 360$, donde en cada i -ésimo índice contiene la suma vectorial del conjunto de vectores contenidos dentro de una ventana angular que va del vector i al vector $i + 90^\circ$ del histograma polar original. A partir de los nuevos vectores H_i que se muestran en la figura 4.15, se busca el vector de mayor magnitud h_{max1} .

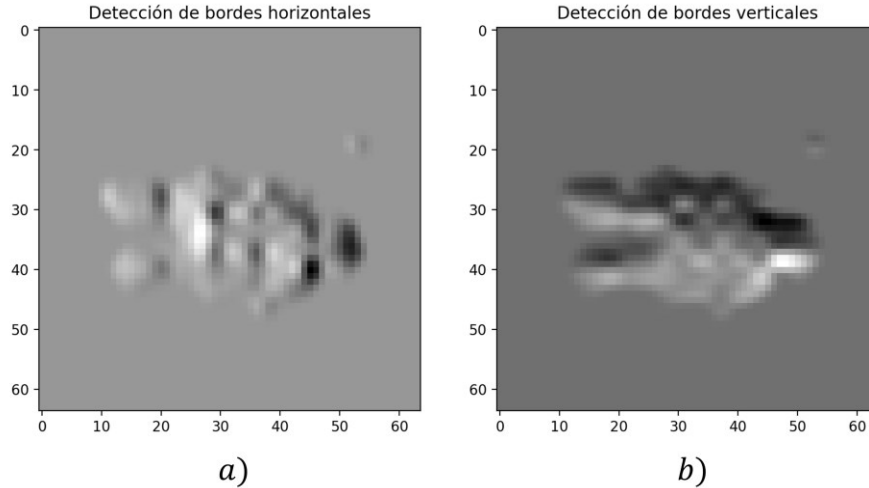


Figura 4.14 Obtención de los gradientes de una imagen de caída, a) gradientes horizontales, b) gradientes verticales.

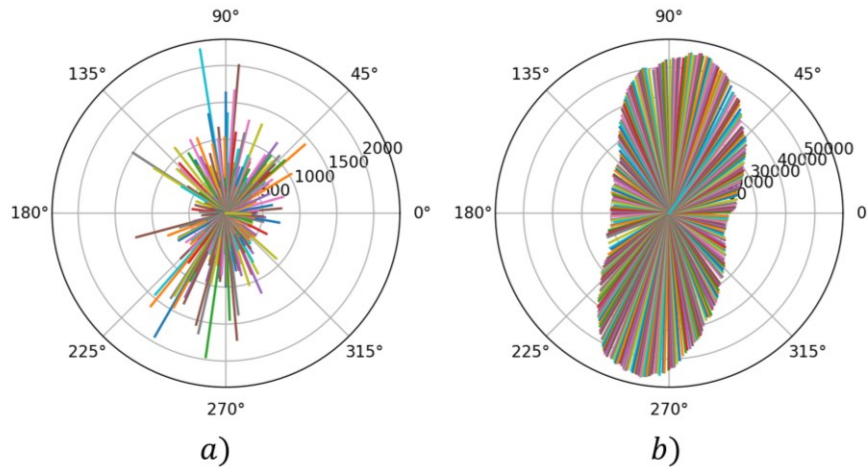


Figura 4.15 Histograma polar correspondiente a las direcciones y la sumatoria de magnitudes de las imágenes, a) histograma polar original, b) histograma polar suavizado.

Después separamos la imagen original con una línea de dirección h_{max1} y tomando en la región opuesta al primer vector máximo. Ahora, tenemos que decidir cuál de estos dos vectores se considerará el principal vector dominante. Este enfoque permite centrarnos en la propiedad particular de las dos regiones separadas, a la primera mitad le denominamos $C_1(m, n)$ y a la segunda $C_2(m, n)$ mostradas en la figura 4.16. Se define una ventana deslizante $P(m, n)$ de la mitad del tamaño de la imagen $I^m(m, n)$. Para todos los píxeles (m, n) de $C_1(m, n)$,

centramos la ventana $P(m,n)$ en (m,n) y calculamos la suma p de todos los valores de gris dentro de la ventana. Cada valor p se almacena en un vector S_1 . Se realiza el mismo proceso para $C_2(m,n)$ y se crea un vector similar S_2 . Por otra parte, calculamos los valores medios \bar{p}_1 y \bar{p}_2 a partir de los vectores S_1 y S_2 . Finalmente, si $\bar{p}_1 < \bar{p}_2$ entonces la dirección dominante será $h_M = h_{max1}$ de lo contrario $h_M = h_{max2}$, de esta manera, se gira la imagen en dirección horaria los grados asignados a h_M , dando como resultado la imagen normaliza en rotación mostrada en la figura 4.17, el pseudocódigo de este proceso se describe en el Algoritmo 3.

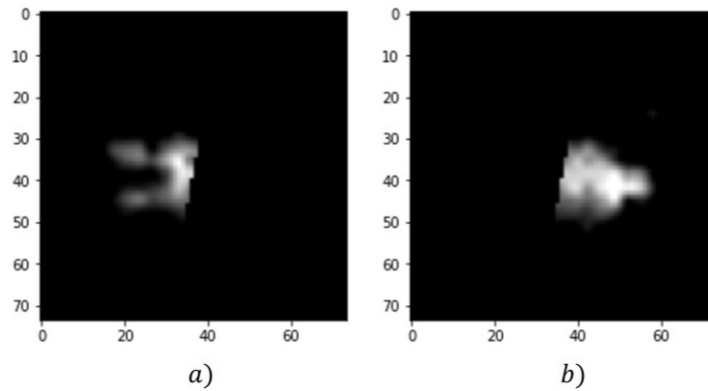


Figura 4.16 Imágenes resultantes de la separación de la imagen $I^m(m,n)$ en dirección h_{max1} , a) mitad denominada $C_1(m,n)$, b) mitad denominada $C_2(m,n)$.

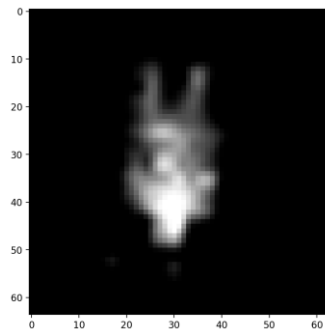


Figura 4.17 Imagen resultante del proceso de normalización en rotación, traslación y escala.

Algoritmo 3: normalización del ángulo del objeto de interés**Entradas:** imagen $I^g(m, n)$ **Salidas:** imagen corregida $I_1^s(m, n)$ **COMIENZO****HACER** $I^m(m, n) \leftarrow I^g(m, n)B(m, n)$ **OBTENER** histograma polar de $I^m(m, n)$ y **ASIGNAR** en H **OBTENER** $h_{max1} \leftarrow \max(H)$ **DIVIDIR** $I^m(m, n)$ sobre la dirección h_{max1} **ASIGNAR** las dos partes de la imagen $I^m(m, n)$ en $C_1(m, n)$ y $C_2(m, n)$ **CALCULAR** el valor medio \bar{g} de intensidad de gris en $I^m(m, n)$ **INICIALIZAR** $j = 1$ **MIENTRAS** $j \neq 2$ **PARA CADA** (m, n) en $C_j(m, n)$ **SI** $C_j(m, n) \geq \bar{g}$ **ENTONCES****USAR** ventana $P(m, n)$ centrada en (m, n) **HACER** $p = \sum_{m, n} P(m, n)$ **ALMACENAR** p en S_j **FIN****FIN** $j = j + 1$ **FIN****HACER** $\bar{p}_1 = \text{promedio}(S_1)$ y $\bar{p}_2 = \text{promedio}(S_2)$ **SI** $\bar{p}_1 < \bar{p}_2$ **ENTONCES** $h_M = h_{max1}$ $I_1^s(m, n) = \text{rotación horaria}(I^g(m, n))$ de ángulo $\angle h_M$ **SI NO** $h_M = h_{max2}$ $I_1^s(m, n) = \text{rotación horaria}(I^g(m, n))$ de ángulo $\angle h_M$ **FIN**

Con el objetivo de organizar las funciones descritas anteriormente se diseñó una interfaz capaz de normalizar en rotación, traslación y escala cualquier imagen que ingrese el usuario, además, se diseñó la opción de normalizar conjuntos grandes de imágenes contenidas en carpetas del sistema operativo, como se muestra en la imagen 4.18, se cuenta con visualización previa de la imagen ingresada y visualización del resultado, asimismo, se añadió la función de cambio de escala, con la que el usuario puede reducir o aumentar la resolución con la que se trabajan las imágenes en los procesos del programa, con el objetivo de lograr acelerar el proceso de normalización.

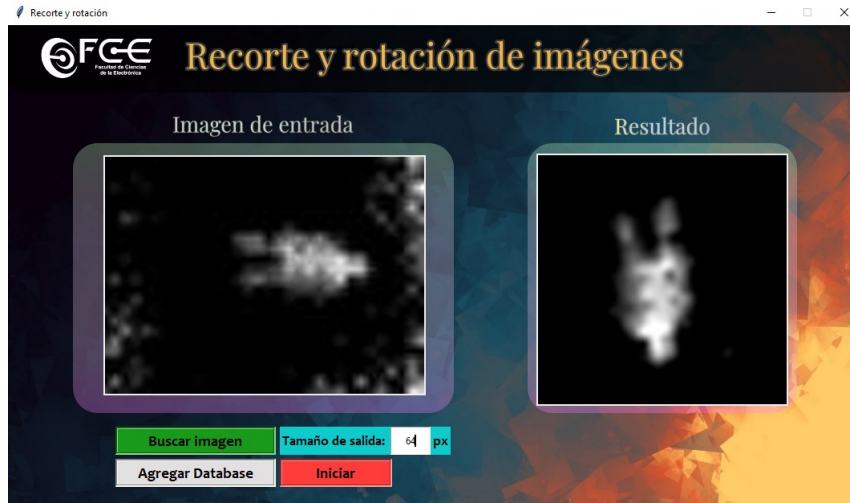


Figura 4.18 Diseño de interfaz para normalizar en rotación, traslación y escala las imágenes del set de entrenamiento.

4.5. Propuesta de clasificación de secuencias de fotogramas térmicos usando un clasificador k-NN

Para clasificar los nuevos fotogramas normalizados en rotación, traslación y escala, proponemos crear un conjunto de imágenes térmicas que muestren a una persona estando parada y acostada. Estas imágenes se obtendrán a través de nuevas capturas, con el objetivo de representar los comportamientos de la manera más precisa posible. Es importante que este conjunto de imágenes esté balanceado en cuanto a la cantidad de ejemplos de cada comportamiento para una clasificación precisa, tal como se muestra en la figura 4.19.

Estas imágenes permitirán discriminar en cada fotograma el comportamiento de la persona (parada o acostada), lo que a su vez nos permitirá clasificar toda la secuencia de entrada. Es importante destacar que este nuevo conjunto de ejemplos a utilizar ya ha sido normalizado y es similar a la secuencia de entrada ya normalizada, esta característica permite aplicar PCA para reducir la dimensionalidad de todo el conjunto, obteniendo solo las características más importantes que nos permitirán comparar y obtener la clasificación precisa de la secuencia completa.

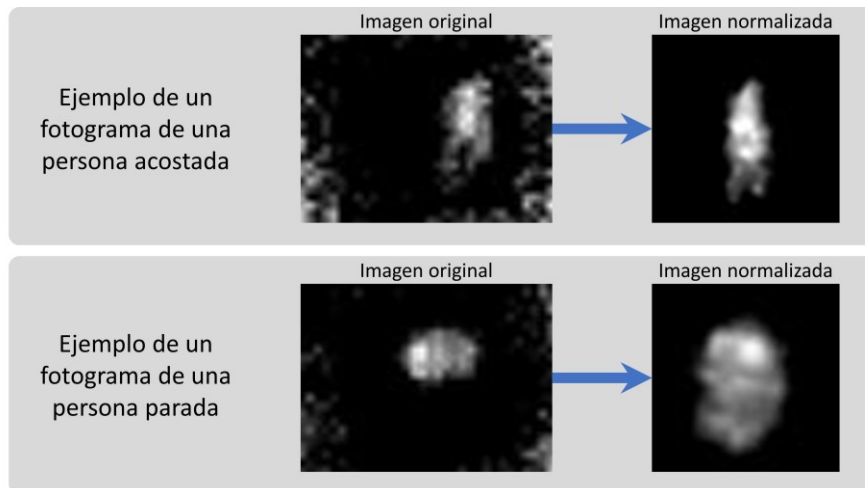


Figura 4.19 Imágenes de ejemplo de un fotograma de una persona acostada y una persona parada que se encuentran en dentro del nuevo conjunto normalizado.

La clasificación de cada fotograma nos entrega a la salida el valor de probabilidad de estar acostado de cada fotograma, esto nos permite crear un vector que corresponde a la probabilidad de cada fotograma, el cual utilizamos para una segunda clasificación. En esta nueva clasificación se utiliza un conjunto de ejemplos que han sido de igual forma reducidos mediante PCA, de modo que al realizar la clasificación de estos vectores de probabilidad podemos identificar si la secuencia de entrada pertenece a una secuencia de caída o una no caída. Este proceso es mostrado en la figura 4.20.

Usando el clasificador k-NN, el cual se encarga de encontrar entre el banco de ejemplos de una persona parada y acostada la probabilidad de que el fotograma que está en la entrada pertenezca a un fotograma de una persona acostada. Para aplicar el clasificador k-NN a nuestra base de datos térmica que consta de 4000 ejemplos del comportamiento acostado y 4000 ejemplos del comportamiento parado, con estos ejemplos se propone clasificar 412 secuencias, con la finalidad de obtener para cada secuencia un vector de probabilidad de estar acostado. El diseño del algoritmo se realizó utilizando lenguaje Python, el pseudocódigo es mostrado en el algoritmo 4.

En el algoritmo de clasificación k-NN solo los puntos que se encuentran entre los k vecinos más cercanos al punto de prueba contribuyen (de manera igual) a la estimación de la probabilidad posterior y, por lo tanto, a la clasificación. También

se pueden usar pesos que asignen importancia diferente según el rango de distancia entre los K vecinos más cercanos, lo que lleva a reglas de clasificación k -NN ponderadas.

Algoritmo 4: k-Nearest Neighbors

COMIENZO

Dada la entrada: $D = \{(x_1, c_1), \dots, (x_N, c_N)\}$
 $x = (x_1 \dots, x_n)$ caso nuevo para clasificar.

PARA todo objeto clasificado (x_1, c_1)
 CALCULAR $d_i = d(x_1, x)$

ORDENAR $d_i (i = 1, \dots, N)$ de forma ascendente.

SELECCIONAMOS los k casos D_x^k ya clasificados más cercanos a x .

ASIGNAMOS a x la clase más frecuente en D_x^k .

FIN

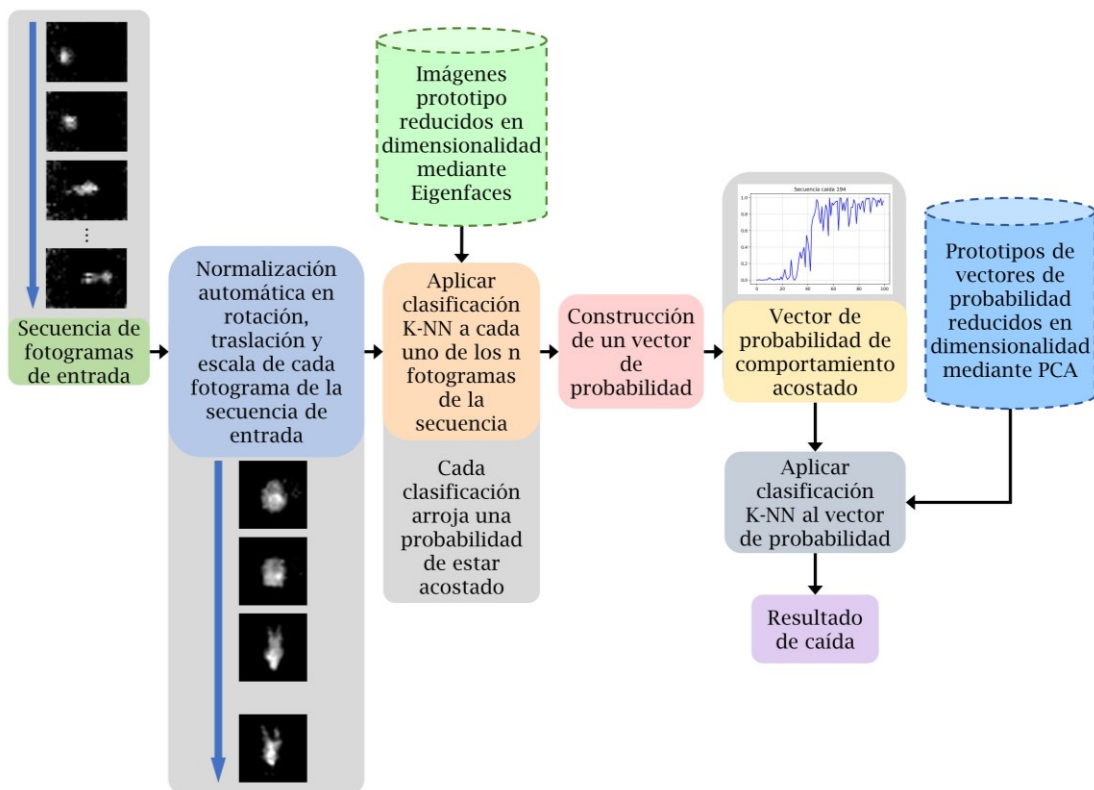


Figura 4.20 Propuesta del sistema de reconocimiento de caídas utilizando secuencias térmicas e imágenes de entrenamiento de una persona estando acostada y parada.

El resultado de la segunda etapa de clasificación está condicionado por la etapa previa de clasificación donde se construyó el vector de probabilidad, dándonos una respuesta sobre si el vector de probabilidad pertenece a una secuencia de caída o de una no caída. Este resultado es desarrollado únicamente con las premisas de conocer si la persona está acostada o parada en cada fotograma de dicha secuencia.

4.6. Optimización de la propuesta de clasificación de secuencias de fotogramas térmicos

Aplicando un nuevo enfoque a la metodología anteriormente planteada, se planteó cambiar la manera en que se construye el vector de características con que se clasifican los fotogramas de las secuencias de caídas y no caídas. De modo que, utilizando la reducción PCA sobre un conjunto balanceado de secuencias de caídas y no caídas, es posible construir un vector que contenga el resultado de todas las características reducidas de los n fotogramas de la secuencia de imágenes térmicas.

Como se muestra en la figura 4.21, al concatenar el resultado de la reducción de características de cada fotograma podemos obtener un vector que contenga las características más significativas de la secuencia completa. Esto permite que podamos separar aún más las diferencias entre una secuencia de caída una de no caída, con la finalidad de obtener clasificaciones con mayor precisión de las secuencias de entrada.

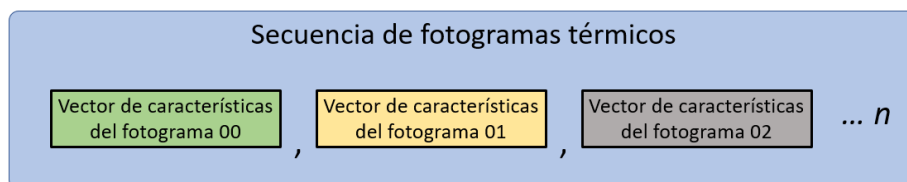


Figura 4.21 Construcción del vector de características de la secuencia térmica utilizando las características resultantes de la reducción mediante PCA de cada fotograma de la secuencia térmica.

Estas imágenes permitirán discriminar en cada fotograma el comportamiento de la persona (parada o acostada), lo que a su vez nos permitirá clasificar toda la secuencia de entrada. Es importante destacar que este nuevo conjunto de ejemplos a utilizar ya ha sido normalizado y es similar a la secuencia de entrada ya normalizada, esta característica permite aplicar PCA para reducir la dimensionalidad de todo el conjunto, obteniendo solo las características más importantes que nos permitirán comparar y obtener la clasificación precisa de la secuencia completa.

Una vez obtenido el nuevo vector de características se procede a volver a reducir su dimensionalidad usando de nuevo el método de PCA, apoyándonos de una matriz de Eigenvectores obtenida del conjunto de ejemplos de entrenamiento. Una vez reducido el vector, se aplica la clasificación mediante k-NN y a diferencia del método propuesto anteriormente, este sistema solo aplica la clasificación en una sola ocasión. Este sistema optimizado se muestra en la figura 4.22.

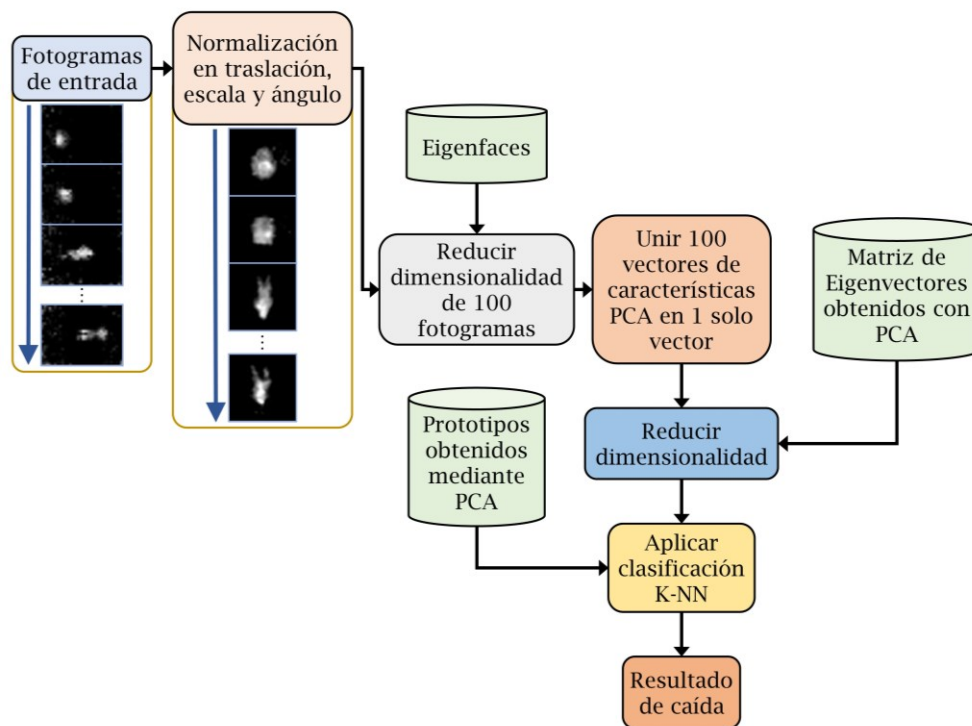


Figura 4.22 Propuesta optimizada del sistema de reconocimiento de caídas utilizando secuencias térmicas e imágenes de entrenamiento de una persona estando acostada y parada.

5. Resultados

Los vectores de probabilidad obtenidos de la primera etapa de clasificación otorgan la información necesaria para determinar mediante un criterio simple si la secuencia es o no una caída. Este vector se forma utilizando la probabilidad de que la persona se encuentre acostada en cada fotograma de la secuencia, como se observa en la figura 5.1, se muestra como la probabilidad aumenta en la secuencia a medida que la persona para de estar parada a estar acostada, este gráfico describe una forma lenta en que sucede esta transición, esto es debido a que la persona en esta secuencia se acostó de manera voluntaria.

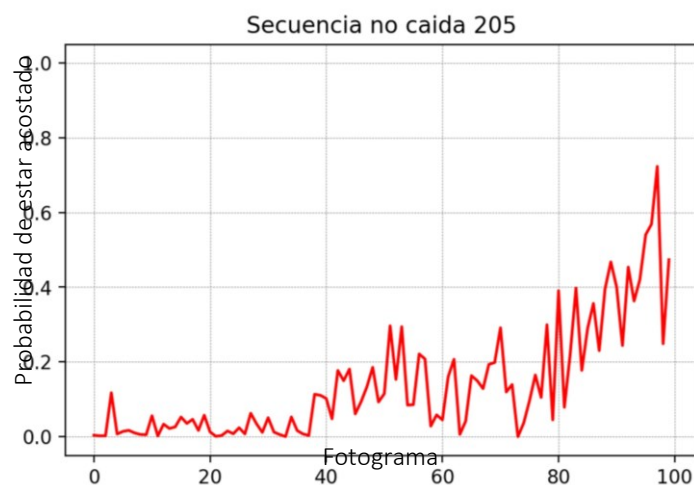


Figura 5.1 Ejemplo de un vector de probabilidad de estar acostado para una secuencia de no caída.

En la figura 5.2 puede observarse que la transición entre la probabilidad de estar parado y estar acostado fue más rápida, debido a la naturaleza de una caída, esta suele suceder en segundos, lo cual genera que el gráfico muestre un tipo de escalón en alguna zona de los fotogramas.

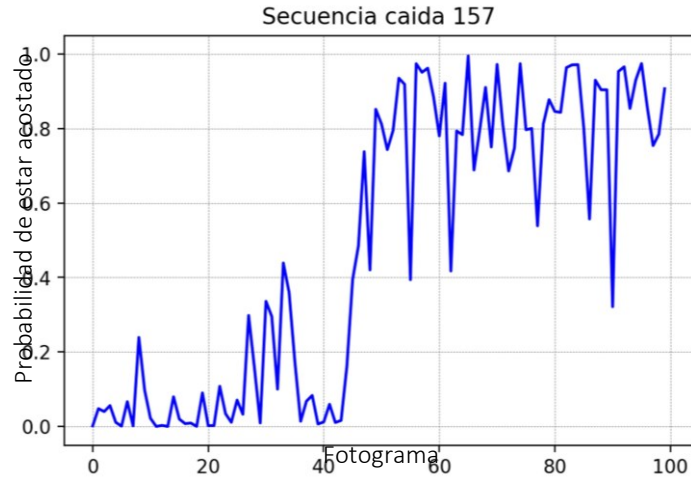


Figura 5.2 Ejemplo de un vector de probabilidad de estar acostado para una secuencia de una caída.

Al realizar la validación del segundo clasificador K-NN ponderado, se propuso realizar 100 pruebas donde los ejemplos de entrenamiento y test se eligieran siempre de manera aleatoria, obteniendo siempre una configuración de 352 secuencias de entrenamiento, las cuales están siempre equilibradas en 176 secuencias de caída y 176 secuencias de no caída, y un total de 60 secuencias de validación de forma equilibradas entre caídas y no caídas.

En la figura 5.3, se puede observar el porcentaje de precisión de cada prueba realizada, siendo un 91.48 % la precisión media y la desviación estándar de un 3.04 %, utilizando un total de 11 vecinos cercanos. Observando que a una cantidad pequeña de vecinos es posible obtener un desempeño aceptable a la hora de identificar una caída, actualmente se continúan haciendo diseños de otros tipos de clasificadores.

Validación cruzada del K-NN Ponderado usando PCA para identificar caídas y no caídas

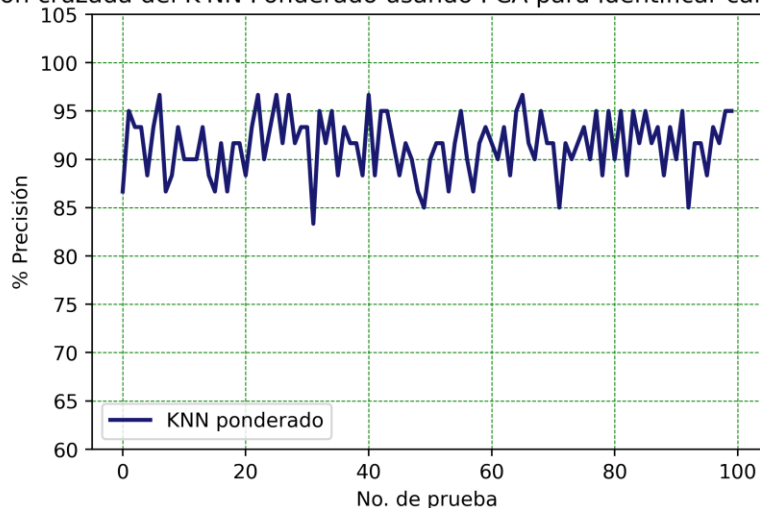


Figura 5.3 Resultados de la validación cruzada del clasificador K-NN ponderado con $k=11$, 352 secuencias equilibradas (caídas y no caídas) de entrenamiento y un conjunto de validación equilibrado de 60 secuencias.

Al utilizar el nuevo algoritmo de optimización propuesto usando los vectores de características concatenados y reducidos mediante PCA, se utilizaron ahora 206 secuencias de caídas y 206 secuencias de no caídas, de las cuales se seleccionaron 352 secuencias equilibradas en caídas y no caídas, para entrenar el clasificador y 60 secuencias que están de igual forma equilibradas para validarlo. Utilizando un porcentaje de reconstrucción de los vectores características del 80% se logró que en un promedio de 25 pruebas la precisión del clasificador resultara de un 97.86% con una desviación estándar de 1.79% como se muestra en la figura 5.4.

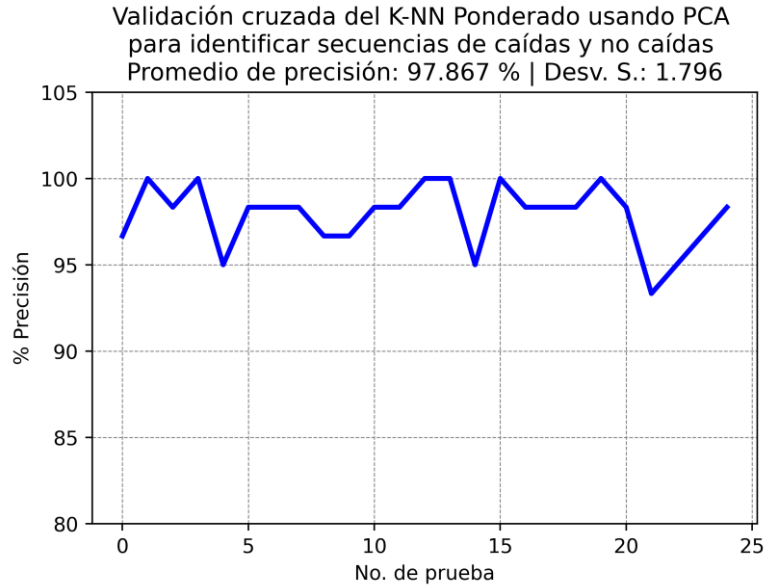


Figura 5.4 Resultados de la validación cruzada del sistema optimizado con un clasificador K-NN ponderado usando $k = 9$ y un porcentaje de reconstrucción de las características de un 80%.

Debido a que el parámetro de reconstrucción de las características usando PCA puede ser modificado, se redujo experimentalmente en diferentes valores para determinar el porcentaje óptimo, siendo un 70% donde mostró el mejor desempeño, obteniendo al final un promedio de precisión del 98.58% con una desviación estándar de 1.32% en 20 pruebas como se muestra en la figura 5.5. El tiempo de clasificación es de aproximadamente 5 ms. La matriz de confusión en la figura 5.6 muestra una precisión de predicción de caídas de 0.997, precisión de predicción de no caídas de 0.974, especificidad de 0.997 y sensibilidad de 0.974.

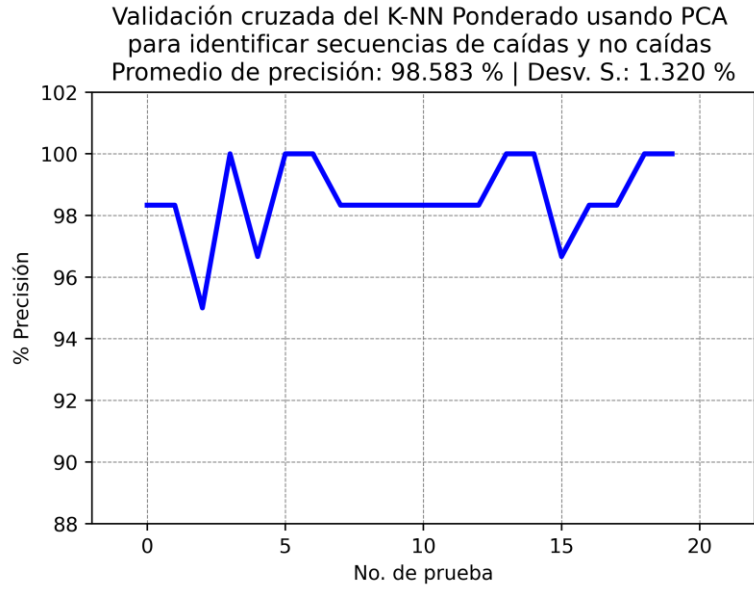


Figura 5.5 Resultados de la validación cruzada del sistema optimizado con un clasificador K-NN ponderado usando $k = 9$ y un porcentaje de reconstrucción de las características de un 70%.

		Valores reales	
		Caída	No caída
Valores de predicción	Caída	49.86 %	0.13 %
	No caída	1.28 %	48.72 %

Figura 5.6 matriz de confusión construida con los resultados de las 20 pruebas mediante validación cruzada.

6. Conclusiones

Este trabajo presenta un sistema innovador que permite detectar caídas en imágenes térmicas y radiografías de manos utilizando cuatro algoritmos principales. El primer algoritmo se encarga de ubicar el centroide del objeto de interés, mientras que el segundo y tercer algoritmo se enfocan en normalizar la escala y el ángulo del objeto, respectivamente. Por último, el sistema combina los tres algoritmos para detectar caídas mediante vectores de características reducidos en dimensionalidad mediante PCA y un clasificador K-NN.

Para lograr la creación de una base de datos con imágenes y secuencias térmicas fue necesario desarrollar una metodología que permitiera la captura de estas, ya que se investigaron diferentes bases de datos públicas en internet, pero estas no contaban con nuestras especificaciones de captura, como la posición del sensor en la habitación y que además fueran tomadas con un sensor de baja resolución. Una de las principales ventajas de este sistema es su enfoque en la normalización de prototipos, lo que permite reducir significativamente la cantidad de ejemplos de entrenamiento necesarios. Esto se traduce en un ahorro de tiempo y recursos, así como en una mayor eficiencia y precisión en la detección de caídas. En resumen, este trabajo representa un importante avance en el campo de la detección de caídas, lo cual puede tener un gran impacto en la mejora de la calidad de vida de las personas mayores y discapacitadas.

El algoritmo propuesto para la normalización en rotación, traslación y escala representa una ventaja significativa en la clasificación de patrones, ya que permite buscar y detectar características y diferencias intrínsecas en cada clase a través del registro de imágenes. La combinación de esta normalización con técnicas como PCA y K-NN ponderado ha demostrado ser altamente viable en la clasificación de secuencias, lo que es fundamental para refinar aún más el algoritmo.

Es importante destacar que el poder computacional de la tarjeta Raspberry Pi 4 ha sido suficiente para ejecutar los algoritmos desarrollados en este trabajo. Esto significa que el sistema es altamente eficiente en esta plataforma. En general, el enfoque y los resultados obtenidos en este trabajo tienen el potencial de mejorar significativamente la precisión y eficiencia de los sistemas de clasificación de patrones en diversas áreas, incluyendo la medicina, la seguridad y la industria.

6.1 Contribuciones

Nuestro trabajo aporta a la ciencia las siguientes contribuciones:

1. El desarrollo de un método que encuentra la ubicación del centroide de un objeto de interés en imágenes térmicas.
2. Un algoritmo para normalizar la escala del objeto mediante la detección de la caída de densidad del objeto de interés con respecto al fondo negro en una imagen,
3. Un algoritmo para la normalización angular del objeto de interés
4. Un sistema que utiliza los 3 algoritmos anteriores para detectar caídas utilizando vectores de características reducidos en dimensionalidad mediante PCA para detectar secuencias de caídas de imágenes térmicas usando un clasificador K-NN.
5. Una base de datos térmica de secuencias de caídas y no caídas utilizando un sensor de baja resolución.

6.2 Trabajo futuro

Nuestro trabajo puede mejorar mucho, al realizar una optimización en la reducción de características se pueden obtener vectores con un menor porcentaje de reconstrucción. Además, si se robustece el método de normalización en traslación, escala y ángulo, este puede llegar a ser aplicable a imágenes con diferentes tipos de siluetas y también a imágenes médicas. Al cambiar el clasificador k-NN por un Multilayer Perceptron podemos visualizar que el porcentaje de precisión aumente. Además de las mejoras en software, el hardware del sistema también puede ser cambiado por otro con mayores recursos de procesamiento, utilizando tarjetas que permitan el uso de los algoritmos de aprendizaje automático a mayor velocidad, con el objetivo de lograr que nuestro sistema procese el algoritmo de normalización de imágenes en menor tiempo. De esta forma, se podrá obtener resultados más rápidos y eficientes en el procesamiento de datos.

Referencias

- [1] J. Beard, A. Oficcer y A. Cassels, «Informe mundial sobre el envejecimiento y la salud,» OMS, Ginebra, 2015.
- [2] P. Caicedo, C. Rengifo y L. Rodríguez, «Contribuciones de las Ciencias Electrónicas al Problema de las Caídas en Población Adulta Mayor,» *REVISTA MEXICANA DE INGENIERÍA BIOMÉDICA*, vol. 37, pp. 251-269, 2016.
- [3] S. Yoo y D. Oh, «An artificial neural network–based fall detection,» *International Journal of Engineering Business Management*, vol. 10, pp. 1-7, 2018.
- [4] M. Tarabini, B. Saggin, M. Bocciolone, D. Scaccabarozzi y M. Magni, «Falls in older adults: kinematic analyses with a crash,» *2016 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, pp. 1-6, 2016.
- [5] T. Moeslund, A. Hilton y V. Kruger, «A survey of advances in vision-based human motion capture and analysis,» *Computer Vision and Image Understanding*, vol. 104, pp. 90-126, 2006.
- [6] Y. Ogawa y K. Naito, «Fall detection scheme based on temperature distribution with IR array sensor,» *2020 IEEE International Conference on Consumer Electronics (ICCE)*, pp. 1-5, 2020.
- [7] A. Hayashida, V. Moshnyaga y K. Hashimoto, «The use of thermal ir array sensor for indoor fall detection,» de *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Banff, Canada, 2017.
- [8] L. Yu, H. Chen, H. He, H. Nie, X. Zhai y B. Xiong, «A Fall Detection System Based on a Thermopile Imaging Array and a Back Projection Algorithm,» *2020 IEEE International Conference on Electro Information Technology (EIT)*, pp. 60-65, 2020.

- [9] S. S. Khan, J. Nogas y A. Mihailidis, «Spatio-temporal adversarial learning for detecting unseen falls,» *Pattern Analysis and Applications*, vol. 24, p. 381–391, 2021.
- [10] S. Vadivelu, S. Ganesan, O. R. Murthy y A. Dhall, «Thermal Imaging Based Elderly Fall Detection,» *Computer Vision – ACCV 2016 Workshops*, vol. 10118, pp. 541-553, 2017.
- [11] M. Schwenk, A. Lauenroth, C. Stock, R. Rodriguez, P. Oster, G. McHugh, C. Todd y K. Hauer, «Definitions and methods of measuring and reporting on injurious falls in randomised controlled fall prevention trials: a systematic review,» *BMC Medical Research Methodology*, vol. 12, n^o 50, 2012.
- [12] D. Waters, C. Qualls, M. Cesari, Y. Rolland, L. Vlietstra y B. Vellas, «Relationship of Incident Falls with Balance Deficits and Body Composition in Male and Female Community-Dwelling Elders,» *The journal of nutrition, health & aging*, vol. 23, pp. 9-13, 2018.
- [13] R. Igual, C. Medrano y I. Plaza, «Challenges, issues and trends in fall detection systems,» *BioMedical Engineering Online*, pp. 12-66, 2013.
- [14] Organización Mundial de la Salud, «Caídas,» 16 Enero 2018. [En línea]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/falls>. [Último acceso: 19 Marzo 2021].
- [15] M. Vallejo, C. Isaza y J. López, «Artificial Neural Networks as an alternative to traditional fall detection methods,» *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1648-1651, 2013.
- [16] K. Wang, G. Cao, D. Meng, W. Chen y C. Cao, «Automatic fall detection of human in video using combination of features,» *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pp. 1228-1233, 2016.
- [17] A. Bhargavi, K. Sunitha y B. Mahima, «Development of Low-cost Thermal Imaging System as a Preliminary Screening Instrument,» *Procedia Computer Science*, vol. 172, pp. 283-288, 2020.

- [18] C. Zhong, W. Ng, S. Zhang, C. Nugent, C. Shewell y J. Medina-Quero, «Multi-Occupancy Fall Detection Using Non-Invasive Thermal Vision Sensor,» *IEEE Sensors Journal*, vol. 21, nº 4, pp. 5377-5388, 2021.
- [19] M. Stojanović, V. N. M. Stanković y S. Stanković, «Object Tracking in Thermal Imaging using Kernelized Correlation Filters,» *17th International Symposium INFOTEH-JAHORINA*, pp. 1-6, 2018.
- [20] V. Mehta, A. Dhall, S. Pal y S. S. Khan, «Motion and Region Aware Adversarial Learning for Fall Detection with Thermal Imaging,» de *2020 25th International Conference on Pattern Recognition (ICPR)*, Milan, Italy, 2021.
- [21] J. Nogas, S. Khan y A. Mihailidis, «DeepFall: Non-Invasive Fall Detection with Deep Spatio-Temporal Convolutional Autoencoders,» *Journal of Healthcare Informatics Research*, vol. 4, p. 50–70, 2020.
- [22] W.-H. Chen y H.-P. Ma, «A fall detection system based on infrared array sensors with tracking capability for the elderly at home,» *2015 17th International Conference on E-health Networking, Application & Services (HealthCom)*, pp. 428-434, 2015.
- [23] J. Rafferty, J. Medina-Quero, S. Quinn, C. Saunders, I. Ekerete, C. Nugent, J. Synnott y M. Garcia-Constantino, «Thermal Vision Based Fall Detection via Logical and Data driven Processes,» *2019 IEEE International Conference on Big Data, Cloud Computing, Data Science & Engineering (BCD)*, pp. 35-40, 2019.
- [24] S. Kido, T. Miyasaka, T. Tanaka, T. Shimizu y T. Saga, «Fall detection in toilet rooms using thermal imaging sensors,» *2009 IEEE/SICE International Symposium on System Integration (SII)*, pp. 83-88, 2009.
- [25] B. Zitova, «Mathematical Approaches for Medical Image Registration,» de *Encyclopedia of Biomedical Engineering*, Oxford, Elsevier, 2019, pp. 21-32.
- [26] S. Uchida, «Image processing and recognition for biological images,» *Development, growth & differentiation*, vol. 55, nº 4, p. 523–549, 2013.

- [27] W. Jia, R.-X. Hu, J. Gui, Y. Zhao y X.-M. Ren, «Palmprint Recognition Across Different Devices,» *Sensors*, vol. 12, nº 6, pp. 7938-7964, 2012.
- [28] M. A. Zulkifley, S. R. Abdani y N. H. Zulkifley, «Automated Bone Age Assessment with Image Registration Using Hand X-ray Images,» *Applied Sciences*, nº 20, 2020.
- [29] E. Yoruk, H. Dutagaci y B. Sankur, «Hand based biometry,» *Image and Video Communications and Processing*, pp. 1106-1115, 2005.
- [30] F. Hržić, S. Tschauner, E. Sorantin y I. Štajduhar, «XAOM: A method for automatic alignment and orientation of radiographs for computer-aided medical diagnosis,» *Computers in Biology and Medicine*, vol. 132, 2021.
- [31] C. M. Jha, *Thermal Sensors Principles and Applications for Semiconductor Industries*, New York: Springer, 2015.
- [32] Melexis Inspired Engineering, «MLX90640 32x24 IR array,» 3 Diciembre 2019. [En línea]. Available: <https://www.melexis.com/en/documents/documentation/datasheets/datasheet-mlx90640>. [Último acceso: 8 Marzo 2021].
- [33] Raspberry Pi Trading Ltd, «Raspberry Pi 4 Tech Specs,» Enero 2021. [En línea]. Available: <https://datasheets.raspberrypi.org/rpi4/raspberry-pi-4-product-brief.pdf>. [Último acceso: 8 Junio 2021].
- [34] Raspberry Pi Ltd, «Raspberry Pi Documentation,» 2012. [En línea]. Available: <https://www.raspberrypi.com/documentation/computers/raspberry-pi.html>. [Último acceso: 17 febrero 2023].
- [35] M. Jiménez, R. Palomera y I. Couvertier, *Introduction to Embedded Systems*, New York, NY: Springer, 2014.
- [36] W. Gay, «I2C,» de *Beginning STM32*, Berkeley, CA, Apress, 2018, pp. 195-221.
- [37] S. Russell y P. Norvig, *Artificial Intelligence: A Modern Approach*, Upper Saddle River, New Jersey: Prentice Hall, 2010.

- [38] A. C. Müller y S. Guido, Introduction to machine learning with Python : a guide for data scientists, Sebastopol, California: O'Reilly Media, Inc, 2017.
- [39] T. M. Mitchell, Machine Learning, McGraw Hill, 1997.
- [40] R. M. Thanki y A. M. Kothari, «Introduction,» de *Digital Image Processing using SCILAB*, Cham, Switzerland, Springer, 2019, pp. 1-18.
- [41] A. D. C. Distanto, «Image Enhancement Techniques,» de *Handbook of Image Processing and Computer Vision*, Cham, Switzerland, Springer, 2020, pp. 387-484.
- [42] J. I. T., Principal Component Analysis, New York, NY: Springer, 2002.
- [43] C. C. Aggarwal, Linear Algebra and Optimization for Machine Learning, Cham, Switzerland: Springer Cham, 2020.
- [44] E. López Aldea, Raspberry Pi: fundamentos y aplicaciones, Paracuellos de Jarama, Madrid: RA-MA Editorial, 2017.
- [45] Panasonic Industry, «Infrared Array Sensor Grid-EYE,» 15 Mayo 2020. [En línea]. Available: https://b2b-api.panasonic.eu/file_stream/pids/fileversion/1819. [Último acceso: 8 Marzo 2021].
- [46] U. Braga-Neto, Fundamentals of Pattern Recognition and Machine Learning, Cham, Switzerland: Springer Cham, 2020.
- [47] M. Turk y A. Pentland, «Eigenfaces for Recognition,» *Eigenfaces for Recognition*, vol. 3, pp. 71-86, 1991.

Anexos

a) Algoritmo de almacenamiento en lenguaje Python

```
import time
import busio
import board
import adafruit_amg88xx
import time, board, busio
import pandas as pd
import numpy as np
import adafruit_mlx90640

i2c = busio.I2C(board.SCL, board.SDA, frequency=1000000) # setup I2C
mlx = adafruit_mlx90640.MLX90640(i2c) # begin MLX90640 with I2C comm
mlx.refresh_rate = adafruit_mlx90640.RefreshRate.REFRESH_16_HZ # set refresh
rate
frame = np.zeros((24*32,)) # setup array
amg = adafruit_amg88xx.AMG88XX(i2c)
t_array=[]

while True:
    t1=time.monotonic()
    pixels=[]
    for row in amg.pixels:
        pixels=pixels+row

    mlx.getFrame(frame)
    t_array.append(time.monotonic()-t1)

    pixels2=pd.DataFrame(pixels)
    frame2=pd.DataFrame(frame)
    rate=len(t_array)/np.sum(t_array)
    rate2=pd.DataFrame([rate])
    thermalframes=[pixels2, frame2, rate2]
    arreglo=pd.concat(thermalframes)

    arreglo.to_csv('DATOS-MLX-AMG', header=False, index=i++)
    #print(thermalframes)
    #print('AMG: {0:2.1f}C'.format(np.mean(pixels)))

    print('Sample rate: {0:2.1f}FPS'.format(len(t_array)/np.sum(t_array)))
    print("\n")
```

b) Algoritmo de clasificación KNN

```
from PIL import Image
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

##### FUNCIONES #####

def inicio(k,ejemplos,entradas,frecuencia):

print('\n_____ \n')
    print('No. Vecinos cercanos: %d Ejemplos empleados: %d (H= %d / M= %d)'
Entradas: %d' % (k,ejemplos,frecuencia[0],frecuencia[1],entradas))
    print('\n')

def distancia_euclidiana(x,y):
    return np.sum(np.sqrt(np.square(x-y)))

def precision(conteo,patrones):
    return (conteo/len(patrones))*100

def pesos_reciproco(distancias):
    return 1/(np.array(distancias)+1)

def gausiana(distancias):

    sigma=min(distancias)/4
    return np.exp(-((distancias**2)/(2*(sigma**2))))

def suma_pesos(caida,parado,etiqueta):

    if etiqueta[1] == 1:
        caida+=etiqueta[[0]]
    if etiqueta[1] == -1:
        parado+=etiqueta[[0]]

    return(caida,parado)

##### importar datos #####

datos = 'ejemplos-completo.csv'
entradas = pd.read_csv(datos, header=None)
datos_entrada = entradas.to_numpy()
n,m = np.shape(datos_entrada)

datos = 'test-completo.csv'
test = pd.read_csv(datos, header=None)
datos_test = test.to_numpy()
n2,m2 = np.shape(datos_test)

# # PARAMETROS
```

```

k_valores=np.array(range(3,100,2))

# k=12

ejemplos=datos_entrada[:, :-1] # ejemplos de entrenamiento
etiqueta_ejemplos=datos_entrada[:, -1] # etiquetas de ejemplos
clase, frecuencia = np.unique(etiqueta_ejemplos, return_counts=True)

entradas=datos_test[:, :-1] # entradas
etiqueta_entradas=datos_test[:, -1]

clases=np.array([1, -1])

resultado=np.zeros([len(k_valores),3])

plot=0

##### ALGORITMO KNN CLASICO #####

for k in k_valores:

    conteo, conteo2, conteo3 = 0,0,0

    inicio(k, len(ejemplos), len(entradas), frecuencia)

    for j in range(len(entradas)):
        distancias =[distancia_euclidiana(entradas[j], ejemplos[i]) for i in
range(len(ejemplos))]
        # print(distancias)
        idx = np.argpartition(distancias, k) #devuelve un array de los indices de
patrones mas cercanos de forma
        # print(idx)
        clas, freq = np.unique(etiqueta_ejemplos[idx[:k]], return_counts=True)
#ordena de manera ascendente el conteo de etiquetas de los vecinos más cercanos
        label=etiqueta_entradas[j]
        prediccion=clas[np.argmax(freq)]

        if label==prediccion:
            conteo+=1

##### ALGORITMO KNN CON PESOS #####

d=np.array(distancias)

##### RECIPROCO DE LAS DISTANCIAS #####

caida, parado = 0,0

pesos_etiqueta=np.transpose(np.array([pesos_reciproco(d[idx[:k]]), etiqueta_ejemp
los[idx[:k]]]))

for vecino in pesos_etiqueta:
    caida, parado = suma_pesos(caida, parado, vecino)

```

```

lista=[caida,parado]
lista=np.array(lista,dtype=object)

prediccion2=clases[np.argmax(lista)]

if label==prediccion2:
    conteo2+=1

##### GAUSIANA #####

caida,parado = 0,0

pesos_etiqueta2=np.transpose(np.array([gausiana(d[idx[:k]]),etiqueta_ejemplos[id
x[:k]]]))
# print(pesos_etiqueta2)
for vecino2 in pesos_etiqueta2:
    caida,parado = suma_pesos(caida,parado,vecino2)
    # print(hombre,mujer)

lista2=[caida,parado]
lista2=np.array(lista2,dtype=object)

prediccion3=clases[np.argmax(lista2)]

# print('Entrada: %d P Clasico: %d P. Recip: %d P. Gauss: %d' %
(label,prediccion,prediccion2,prediccion3)) #devuelve el indice del array con
mayor votacion entre los vecinos cercanos

if label==prediccion3:
    conteo3+=1

p=np.array([precision(conteo,entradas),precision(conteo2,entradas),precision(con
teo3,entradas)])
resultado[plot,:]=np.transpose(p)
plot+=1
print('Precisión Clasico: %d %% \nPrecisión Recíproco: %f %% \nPresición
Gauss: %f %% ' %
(precision(conteo,entradas),precision(conteo2,entradas),precision(conteo3,entrad
as)))

##### GRAFICAS DE PRESICIÓN #####

fig, ax = plt.subplots(dpi=1200)

ax.plot(k_valores,resultado[:,0],color='midnightblue',
linewidth=3,label='Clasico')
ax.plot(k_valores,resultado[:,1],color='maroon', linewidth=2,label='Reciproco')
ax.plot(k_valores,resultado[:,2],color='darkorange', linewidth=1,label='Gauss')

ax.axis('equal')

```

```
plt.title('KNN con: %d Ejemplos (caida= %d / parado= %d) y %d Entradas'  
%(len(ejemplos),frecuencia[0],frecuencia[1],len(entradas)))  
plt.xlabel('No. Vecinos')  
plt.ylabel('% Precisión')  
  
# plt.ylim(75,105)  
plt.grid(color = 'green', linestyle = '--', linewidth = 0.5)  
leg = ax.legend(loc = "lower left");
```