ESTUDIO COMPARATIVO DE ALGORITMOS DE VISIÓN COMPUTACIONAL ORIENTADOS AL RECONOCIMIENTO DE CAÍDAS

JOSÉ CAMILO ERASO GUERRERO

Trabajo de grado. Maestría en Automática

Director:

MsC: Elena Muñoz España

Codirector:

Ph.D: Mariela Muñoz Añasco

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Maestría en Automática

Popayán, *2022*

JOSÉ CAMILO ERASO GUERRERO

ESTUDIO COMPARATIVO DE ALGORITMOS DE VISIÓN COMPUTACIONAL ORIENTADOS AL RECONOCIMIENTO DE CAÍDAS

Trabajo de grado presentado a la Facultad de Ingeniería Electrónica y Telecomunicaciones de la Universidad del Cauca para la obtención del

Título de

Magíster en Automática

Director:

MsC: Elena Muñoz España

Codirector:

Ph.D: Mariela Muñoz Añasco

Popayán

2022

PAGINA DE DEDICATORIA

Este gran logro en mi vida se lo dedico a Dios Todopoderoso, mi alabanza de gratitud por bendecirme con sabiduría y entendimiento.

Para mi esposa Mary y nuestro hijo Jacob, por ser mi hoy y mi futuro, mi fuente de inspiración y el motor que me impulsa a seguir adelante, por su compañía, amor y por ser un gran soporte en la construcción de este proyecto.

A ti María, mi madre, todo mi agradecimiento, por estar ahí en todo momento, por tus consejos, tus valores y motivación para ser una persona de bien.

Dedicado a todos mis seres queridos por darme su mayor apoyo incondicional, consejos, ánimo y compañía, que han logrado forjar mi futuro. Especialmente a ti mamá Margarita que, aunque partiste en el transcurso de esta maestría, sigues aquí conmigo, en mi recuerdo y mi corazón.

PAGINA DE AGRADECIMIENTOS

Mi más sincero agradecimiento a la Universidad del CAUCA y en especial a las ingenieras Elena Muñoz España y Mariela Muñoz Añasco, por su invaluable profesionalismo, sabiduría y experiencia, con la que me aportaron conocimientos tan acertados, orientando y motivando a plasmar la presente investigación, que permite consolidar mi futuro profesional. Ahora y por siempre en cada proyecto estará la huella de lo aprendido con ustedes.

RESUMEN ESTRUCTURADO

En los últimos años, el reconocimiento de actividades humanas se ha convertido en un área de constante exploración en diferentes campos. Este trabajo incluye una revisión de la literatura enfocada en diferentes tipos de actividades humanas y dispositivos de adquisición de información para el reconocimiento de actividades, y profundiza en la detección de caídas de personas de tercera edad por medio de visión computacional, utilizando métodos de extracción de características y técnicas de inteligencia artificial, como redes neuronales convolucionales, dando a conocer la importancia del reconocimiento de caídas humanas en personas de la tercera edad, además de permitir encontrar aquellos factores que dificultan el verdadero avance en el área, como lo son las bases de datos disponibles, las cuales contienen elementos excesivamente controlados.

Por lo anterior, como parte de la presente investigación, los autores crean CAUCAFall, una base de datos con 10 participantes, que simulan cinco tipos de caídas y cinco tipos de actividades de la vida diaria (AVD). En concreto, los datos incluyen caídas hacia delante, caídas hacia atrás, caídas laterales hacia la izquierda, caídas laterales hacia la derecha y caídas producidas al sentarse. Los participantes realizaron las siguientes AVD: caminar, saltar, coger un objeto, sentarse y arrodillarse. El conjunto de datos considera individuos de diferentes edades, pesos, alturas y piernas dominantes. Los datos se adquirieron utilizando una cámara RGB en un entorno doméstico. El entorno es intencionadamente realista e incluye características no controladas, como oclusiones, cambios de iluminación (natural, artificial y nocturna), ropa diferente de los participantes, movimiento en el fondo, diferentes texturas en el suelo y en la habitación, una variedad en los ángulos de caída y diferentes distancias de la cámara a la caída. CAUCAFall es la única base de datos que contiene etiquetas para cada imagen. Los fotogramas que incluían caídas humanas registradas se etiquetaron como "caída", y las actividades AVD se marcaron como "no caída". Este conjunto de datos es útil para desarrollar y evaluar algoritmos modernos de reconocimiento de caídas, como los que aplican extracción de características, redes neuronales convolucionales con detectores YOLOv3-v4, y la ubicación y resolución de la cámara aumentan el rendimiento de algoritmos como OPENPOSE.

En el presente trabajo se comparan diferentes algoritmos de visión computacional enfocados al reconocimiento de caídas humanas, específicamente algoritmos que trabajan con extracción de características y redes neuronales convolucionales, evaluando su rendimiento en diferentes bases de datos públicas, incluyendo CAUCAFall.

Llegando a la conclusión que en la propuesta que implementa extracción de características, el desempeño es alto en aquellas bases de datos con entornos sumamente controlados; sin cambios de iluminación, sin diferentes ángulos de caída, sin oclusiones, sin cambios de escenario, sin movimiento en el fondo, sin variedad en la distancia de las caídas ante la cámara e incluso sin cambios en las texturas tanto de los entornos como en la ropa de los participantes, además se concluye que estos métodos presentan gran sensibilidad al ruido, oclusiones y cambios en el punto de vista. Por otra parte, las propuestas que basan su trabajo en Redes Neuronales Convolucionales como base de sus investigaciones, tienen un rendimiento mayor en entornos no controlados, en comparación con la extracción de características. Sin embargo, su costo computacional es mucho mayor, siendo casi imposible utilizarlo en detección de caídas en tiempo real.

STRUCTURED ABSTRACT

In recent years, the recognition of human activities has become an area of constant exploration in different fields. This work includes a literature review focused on the different types of human activities and information acquisition devices for the recognition of activities. It also delves into elderly fall detection via computer vision using feature extraction methods and artificial intelligence techniques, as convolutional neural networks, this study shows the importance of the recognition of human falls in elderly people, as well as allowing to find those factors that hinder the real progress in the area, such as the available databases, which contain excessively controlled elements.

Therefore, as part of this research, the authors created CAUCAFall, a database with 10 participants, which simulates five types of falls and five types of activities of daily living (ADLs). Specifically, the data include forward falls, backward falls, lateral falls left, lateral falls right, and falls arising from sitting. The participants performed the following ADLs: walking, hopping, picking up an object, sitting, and kneeling. The dataset considers individuals of different ages, weights, heights, and dominant legs. The data were acquired using an RGB camera in a home environment. This environment was intentionally realistic and included uncontrolled features, such as occlusions, lighting changes (natural, artificial, and night), participants different clothing, movement in the background, different textures on the floor and in the room, and a variety in fall angles and different distances from the camera to the fall. CAUCAFall is the only database that contains labels for each image. Frames including human falls recorded were labeled as "fall", and ADL activities were marked "nofall". This dataset is useful for developing and evaluating modern fall recognition algorithms, such as those that apply feature extraction, convolutional neural networks with YOLOv3-v4 detectors, and camera location and resolution increase the performance of algorithms such as OPENPOSE.

In this paper the authors compare different computer vision algorithms focused on human fall recognition, specifically algorithms working with feature extraction and convolutional neural networks, evaluating their performance on different public databases, including CAUCAFall.

The conclusion is that in the proposal that implements feature extraction, the performance is high in those databases with highly controlled environments; without lighting changes, without different fall angles, without occlusions, without changes of scenery, without movement in the background, without variety in the distance of the falls in front of the camera and even without changes in the textures of both the environments and the clothing of the participants, concluding that these methods present great sensitivity to noise, occlusions and changes in the point of view. On the other hand, the proposals that base their work on Convolutional Neural Networks as the basis of their research, have a higher performance in uncontrolled environments, compared to feature extraction. However, its computational cost is much higher, being almost impossible to use it in real-time fall detection.

CONTENIDO

LISTA DE FIGURAS

LISTA DE TABLAS

LISTA DE ACRONIMOS

HAR: Human Activity Recognition / Reconocimiento de actividades humanas.

KDA: kernel discriminant analysis / Análisis discriminante de Kernel.

STIP: Space-time Interest Points / Puntos de interés del espacio-tiempo.

CNN: Convolutional Neural Networks / Redes neuronales convolucionales.

SVM: Support Vector Machines / Maquina de vectores de soporte.

YOLO: You only look once / Solo miras una vez.

ML: Machine Learning / Aprendizaje de máquina.

ReLu: Rectified Linear Unit / unidad lineal rectificada.

ILSVRC: ImageNet Large Scale Visual Recognition Challenge / Desafío de reconocimiento visual a gran escala de ImageNet.

MOG: Gaussians of Mixtures /Mezcla de Gaussianas.

IOU: Intersection Over Union / Intersección sobre la unión

PAF: Part Affinity Fields /Campos de afinidad de piezas

HMM: Hidden Markov Model/ Modelo oculto de Markov

VGP: Virtual Grounding Point/ punto de puesta a tierra virtual

LSTM: Long short-term memory/ Memoria a corto largo término

1 Introducción

El reconocimiento de la actividad humana (HAR) permite modelar el comportamiento de un usuario mediante la identificación automática de sus tareas, por medio de la observación y análisis del comportamiento humano [1][2][3], dando como resultado, además del reconocimiento de actividades, identidades de personas, personalidades y estados psicológicos [4].

El HAR se ha convertido en los últimos años en un área de constante exploración en diferentes campos, ya que sus aplicaciones son tema de actualidad y de ayuda en la automatización de procesos y actividades, que para el ojo humano pueden pasar desapercibidas o pueden ser labores tediosas. Para [5], una pose humana transmite la configuración de las partes del cuerpo, y la información predictiva implícita sobre el movimiento subsiguiente de las personas, lo cual permite usar esta información dinámica en infinidad de aplicaciones. El HAR presenta una creciente demanda en áreas del entretenimiento [6][7][8], sistemas de video vigilancia [9][10][11][12][13][14], rescate de emergencia y robótica de emergencia [15], ciudades inteligentes, rendimiento deportivo, aplicaciones militares, monitoreo médico para el cuidado de personas de la tercera edad y diversos cuidados de la salud [16][17][18][19][20][21][22], entre otras [23][24].

Esta investigación centra su interés en el reconocimiento de caídas humanas de personas adultas, con el fin de contribuir en investigaciones futuras, cuya principal aplicación sea el reconocimiento de caídas en adultos mayores o de tercera edad, ya que según la Organización Mundial de la Salud [25] entre 2000 y 2050, la proporción de los habitantes del planeta mayores de 60 años se duplicará, pasando del 11% al 22%. En números absolutos, este grupo de edad pasará de 605 millones a 2000 millones en el transcurso de medio siglo.

Una caída se define como aquel incidente imprevisto de caer al suelo o piso causado por un empujón o tirón, características ambientales, pérdida del conocimiento o cualquier problema similar relacionado con limitaciones o trastornos de salud y que impliquen un cambio involuntario en la postura del individuo, resultando con la persona tirada en el suelo [26].

A nivel mundial, las caídas son la segunda causa más común de muertes accidentales y una de las principales causas de lesiones o discapacidades. Únicamente en Estados Unidos cada 11 segundos una persona de la tercera edad que se ha caído, es llevada a una sala de emergencias y cada 19 minutos una de estas personas muere, obviamente con el crecimiento de la población anciana esta tasa tiende a incrementar, se predice que para el 2030 en Estados Unidos cada hora podría morir 7 personas adultas a causa de una caída, en el año 2015 en dicho país el costo médico por caídas supero los USD $50 mil millones [26] [27].

Además, gran porcentaje de las personas de la tercera edad viven o permanecen solas en sus hogares, por lo que de sufrir una caída y no recibir la atención necesaria de inmediato puede incurrir en una lesión grave o incluso su muerte. Para [28] la detección automática de caídas o movimientos que puedan afectar la salud; puede reducir significativamente las consecuencias después del incidente y también puede rastrear patrones de comportamiento diarios anormales de adultos con alto riesgo de caerse, alertando las anomalías.

Por lo anterior, en la última década se han incrementado los diferentes estudios y propuestas enfocadas en la detección de caídas humanas, utilizando diversos dispositivos y tecnologías. Los

sistemas propuestos pueden ser clasificados en diferentes categorías; aquellos que usan dispositivos portátiles [29], sensores de ambiente [27], o aquellos que son abordados por visión computacional.

Es prioridad para esta investigación analizar, trabajar y comparar; técnicas de extracción de características de los métodos basados en visión computacional y modernas aplicaciones con redes neuronales convolucionales, por lo que se ha realizado una revisión de las principales técnicas utilizadas en la literatura para identificar caídas de personas adultas. Sin embargo, encontrar detecciones rápidas, precisas y en escenarios poco controlados es un desafío que aún no se ha resuelto.

1.1 Problema de investigación y pregunta de investigación

El HAR en adultos mayores con el objeto de vigilar y reconocer sus actividades diarias cobra importancia dado que en [28] se afirma que aproximadamente el 58% de las personas de más de 80 años fueron reportadas como muertas después de caídas graves, debido a traumatismos físicos, conmoción cerebral, caderas fracturadas, entre otras. De acuerdo a los resultados de una encuesta publicada en [33] se detecta que: “los ancianos independientes se caen más al caminar, tomar una ducha y subir o bajar escaleras; y caen menos al intentar levantarse o sentarse en una silla o una cama, o agacharse y que las personas mayores que viven en residencias de ancianos, se caen más al caminar y al intentar levantarse de una silla o de una cama y se caen menos al subir o bajar escaleras”, además presenta los diferentes tipos de caídas como hacia adelante al caminar provocada por un resbalón, hacia atrás al caminar provocada por un resbalón, lateral al caminar provocada por un resbalón, hacia adelante al caminar provocada por un tropiezo, hacia adelante al intentar levantarse, entre otras. Cabe mencionar que en visión computacional el reconocimiento de una caída se detecta cuando el cuerpo humano se encuentra en el suelo, sin importar de qué forma se llevó a cabo la caída, esto sumado a parámetros como la velocidad con la que cae el cuerpo al suelo o el área horizontal que delimita la silueta humana, para reconocer una caída.

Las investigaciones relacionadas con el reconocimiento de actividades en personas adultas, se centran en la identificación de actividades cortas y básicas específicas, como caminar, vomitar, caer, desmayos, sentarse, acostarse, entre otras. Independientemente de la cantidad de actividades que reconocen las investigaciones, el factor común entre los autores que trabajan HAR es la serie de problemáticas que afectan el reconocimiento de una actividad específica; como problemas de oclusión, costo computacional, similitud entre acciones, el clima, la protección de objetos, las condiciones de luz, además del hecho de que la misma actividad varía cuando la realizan diferentes personas [34][35].

Yu & Yu, et al. [31] destacan que el paso más importante en el reconocimiento de actividades humanas es la extracción de las características de un video o imagen, ya que es el punto clave para diferenciar actividades como “acostarse en la cama” de "caerse en la cama". Por lo que se han estudiado y expuesto diferentes tipos de algoritmos de reconocimiento de actividades humanas obteniendo altos porcentajes de reconocimiento. Sin embargo, se detecta que muchos de los escenarios en los que se realizan dichas investigaciones son entornos sumamente controlados, en donde se controlan los ángulos de tomas de fotografía, no existen cambios de iluminación ni movimientos en el fondo, no se involucra diferentes distancias hasta la cámara, e incluso no se han tenido en cuenta caídas en escenarios reales [36].

Diferentes autores [37][38][39] abordan conceptos como entornos no controlados o sin restricciones o poco controlados, haciendo énfasis en aquella información obtenida de escenarios en los que no se manipula de manera intencional diferentes parámetros como el movimiento significativo de la cámara, desorden en el fondo, oclusiones, escalado y resoluciones de tomas, condiciones de luz, posiciones y ángulos, etc. En [40] se destaca la importancia de las condiciones de un entorno al momento de usar

un sistema de reconocimiento y detección, lo que obliga a estudiar sistemas robustos a cambios de iluminación o ruido, tolerantes a variaciones de posición, tamaño, ángulo, etc., ya que las diferentes tomas de la actividad humana en escenarios no controlados pueden tener grandes variaciones y oclusiones, autores como [41] aseguran que las redes neuronales convolucionales son una oportuna solución a la mayoría de estos requisitos.

Teniendo en cuenta lo anterior y conociendo los grandes avances investigativos en el campo de la extracción de características de imágenes y videos y en el uso de redes neuronales convolucionales en el reconocimiento de actividades humanas, así como haciendo uso de bases de datos públicas de caídas humanas (URFD [42], Le2i [43], CMDFALL [44], FALL-UP [45], MCF [46], UCF101[47]), para el entrenamiento y optimización de los algoritmos, la pregunta de investigación que busca resolver este trabajo de grado es:

¿Qué ventajas y desventajas presentan las redes convolucionales con respecto a algoritmos de extracción de características basada en visión computacional en cuanto a eficiencia, precisión y coste computacional en el reconocimiento de caídas?

1.2 Objetivos

1.2.1 Objetivo General

Comparar el desempeño de algoritmos para el reconocimiento de caídas, que utilicen extracción de características con visión computacional y redes neuronales convolucionales, con el fin de determinar aquellos que presenten mejor eficiencia, precisión y bajo coste computacional en entornos poco controlados.

1.2.2 Objetivos Específicos

• Implementar los algoritmos para la detección de caídas, a partir de la extracción de características y el uso de redes neuronales convolucionales, en diferentes entornos de prueba.

• Establecer el protocolo de experimentación y los índices de desempeño que se utilizarán para la comparación de los algoritmos de reconocimiento de caídas.

• Determinar el desempeño de los algoritmos de reconocimiento de caídas en entornos poco controlados.

1.3 Publicaciones

Hasta el momento se han aceptado y publicado dos artículos relacionados con la presente investigación:

- Eraso Guerrero, J.C., Muñoz España, E. y Muñoz Añasco, M. 2022. Human Activity Recognition via Feature Extraction and Artificial Intelligence Techniques: A Review. Tecnura. 26, 74 (sep. 2022), 213–236. DOI: https://doi.org/10.14483/22487638.17413.

- Eraso Guerrero, J.C., Muñoz España, E., Muñoz Añasco, M. y Pinto Lopera, J. 2022. Dataset for human fall recognition in an uncontrolled environment. Data in Brief. Vol 45 (dic. 2022), DOI: https://doi.org/10.1016/j.dib.2022.108610.

1.4 Organización del trabajo

El presente trabajo se estructuró en los siguientes capítulos.

Capítulo 1: Presenta y relaciona la necesidad y la delimitación del planteamiento del problema, así como los objetivos generales y específicos.

Capítulo 2: Presenta una completa revisión de la literatura, con la actualidad del reconocimiento de caídas humanas y sus diferentes técnicas. Además del contexto de la investigación.

Capítulo 3: Contiene la metodología de la investigación, que explica el proceso que se llevó a cabo para la selección de los algoritmos comparados, presenta las diferentes bases de datos y la arquitectura propuesta para el diseño de la base de datos CAUCAFall [49] y sus diferencias respecto a las bases de datos existentes, también detalla la implementación de los algoritmos y establece los experimentos propuestos, además de especificar los índices de desempeño.

Capítulo 4: El cuarto capítulo contiene la explicación de los experimentos realizados y los resultados obtenidos.

Capítulo 5: El quinto capítulo contiene la discusión de los resultados obtenidos.

Finalmente, el último apartado está conformado por las conclusiones y posteriormente se hacen recomendaciones para trabajos futuros. Se finaliza con la bibliografía referencial.

2 Marco Teórico y revisión de literatura.

Este capítulo presenta una revisión de la literatura, primero en la clasificación de las actividades humanas, segundo en los métodos de adquisición de información para HAR y finalmente en los métodos que han sido utilizados para extraer características de videos o imágenes para el reconocimiento de actividades en adultos mayores. Por otra parte, también se presenta el marco teórico de la presente investigación.

La revisión de la literatura se elaboró con criterios de la metodología de revisión y análisis documental (RAD)[48]; dividiendo el proceso de investigación en heurística o de recolección de fuentes de información y hermenéutica o de análisis de las mismas. Por lo que se realizó una revisión y análisis de trabajos publicados hasta el año 2022, logrando un panorama actual en el reconocimiento de caídas humanas y en actuales algoritmos de reconocimiento. En total se tuvo en cuenta 153 artículos con criterios de búsqueda como: Reconocimiento de caídas humanas, reconocimiento de caídas humanas por visión computacional, Algoritmos de reconocimiento de caídas humanas, reconocimiento de caídas con redes neuronales convolucionales, caídas humanas y extracción de características, y caídas de personas de la tercera edad.

2.1 Clasificación de las Actividades Humanas

El reconocimiento de la actividad humana es un tema de investigación actual por sus múltiples aplicaciones en la industria del entretenimiento, video vigilancia, salud, robótica, ciudades inteligentes, rendimiento deportivo, aplicaciones militares y en el objetivo principal de esta revisión, el reconocimiento de actividades humanas para el cuidado de personas de la tercera edad.

Las actividades humanas se clasifican dependiendo de la complejidad y duración de dichas actividades, en [50] las acciones son divididas en actividades cortas, actividades simples y complejas ocupacionales, las primeras son actividades de corta duración, como la transición de sentarse a estar de pie, las segundas son actividades básicas como caminar y leer y las actividades complejas involucra escenarios donde se interactúa con varios objetos o personas. Por su parte [4] expone otro mecanismo de clasificación de actividades que también toma como referencia la complejidad de las mismas, el resumen de esta clasificación se muestra en la Tabla 1.

Tabla 1. Clasificación de acciones humanas

Clasificación	Descripción
Gestos	Movimientos primitivos de las partes del cuerpo de una persona, que corresponden a una acción particular.
Acciones Atómicas	Son movimientos de una persona que son parte de actividades más complejas.
Interacción humano-objeto o humano-humano	Actividades humanas que involucran dos o más personas u objetos.
Acciones grupales	Actividades realizadas por grupos de personas.
Comportamientos	Acciones físicas asociadas con emociones, personalidad y estado psicológico de un individuo.

Clasificación	Descripción
Eventos	Son actividades de alto nivel que describen acciones sociales entre individuos e indican la intención de los roles sociales de una persona.

Fuente: Tomado de [4]

Como se mencionó, las investigaciones relacionadas con el reconocimiento de actividades en personas de la tercera edad, se centran en la identificación de actividades cortas y básicas específicas. Por ejemplo [51] es una investigación que pretende hacer el reconocimiento de seis actividades determinadas (caídas de los ancianos hacia adelante, caídas hacia atrás, dolor en el pecho, desmayos, vómitos y dolores de cabeza), por su parte [52] intenta hacer el reconocimiento de otras seis actividades (caídas de personas, personas flexionadas, sentadas, en cuclillas, caminando o acostadas), sin embargo, [30] aumenta el número de acciones a reconocer (persona limpiando una mesa, tomando una bebida, soltando o agarrando un objeto, leyendo, sentándose, parándose, escribiendo, usando celular, cayéndose) lo que también aumenta la dificultad e imprecisión del sistema, debido a problemas de oclusión y la similitud entre acciones [31].

2.2 Métodos de Adquisición de Información

El primer paso para reconocer una determinada actividad humana, es obtener la información para su posterior procesamiento, dicho proceso se puede llevar a cabo por diferentes formas; el primer método se basa en el uso de sensores de ambiente, como sensor de presión, sensores acústicos, sensores de electromiografía y demás sensores que pueden ser integrados y distribuidos alrededor de todo un entorno donde se requiera la identificación de diferentes actividades [53]. En [27] se da a conocer algunos de sus usos, por ejemplo, los sensores acústicos, son utilizados para medir e identificar los sonidos de las caídas, sensores de presión son utilizados para medir los cambios de peso sobre el suelo, sensores infrarrojos para mapear el contorno de calor del individuo, entre otros; sin embargo, el autor menciona que estas tecnologías suelen ser costosas y poco prácticas ya que son muchos los factores que pueden alterar las señales de dichos sensores.

El segundo método también extrae información por medio de sensores, pero de tipo portables, es el caso de los sensores de contacto, giroscopios, acelerómetros, entre otros [54], que son ubicados en una parte especifica del cuerpo humano para almacenar información en forma de señales análogas que ayudan a identificar la actividad que está realizando el individuo. La complejidad de estos sistemas radica en la calibración de los dispositivos y en el hecho de que según [29] y [42], las personas mayores pueden olvidar usar los sensores portables y también mencionan que a la gran mayoría de personas de la tercer edad no les gusta usar estos sensores, ya que causan sentimientos de frustración por su uso en diferentes partes del cuerpo, limitando sus movimientos.

Esta investigación profundiza el tercer método de obtención de información, el cual incorpora visión por ordenador, utilizando: cámaras, sensores de profundidad, técnicas de procesamiento de imágenes y visión artificial. Según lo reportado en [30] y [31] las ventajas de este método, en comparación al método de sensores portables, radican en que no son intrusivos y puede extraer un gran volumen de información, como velocidad, ángulos, rotaciones, tiempo, distancias de objetos o personas ante la cámara, además, este método no es afectado fácilmente por ruidos en el medio ambiente. Por su parte [55] destaca el avance tecnológico para extraer imágenes de video a partir de cámaras RGB (rojo, verde y azul) o con el uso de cámaras de profundidad que otorgan imágenes de mapas de profundidad, permitiendo conocer diferentes distancias de objetos o personas ante la cámara. Por lo que [53] divide

el método basado en visión en tres categorías; métodos usando cámaras comunes RGB, métodos basados en 3D que usa múltiples cámaras, y métodos basados en 3D usando cámaras de profundidad.

Sin embargo, el método basado en visión también tiene sus limitaciones, como la falta de privacidad al tener una cámara en el entorno a todo momento, además de otros tipos de dificultades como el costo computacional, oclusiones, cambios de iluminación, posiciones de las cámaras, que son factores que condicionan directamente el rendimiento de los sistemas de reconocimiento de caídas [32].

2.3 Extracción de Características

A pesar de que HAR ha sido un tema de continua investigación en la última década, aún existen diferentes aspectos que dificultan el perfecto reconocimiento de las actividades humanas en personas de la tercera edad. Con el método de visión por computador son fundamentales características como el clima, la protección de objetos u oclusión, las condiciones de luz, la similitud entre algunas actividades, desorden en el fondo de la imagen, problemas de privacidad y demás dificultades específicas que originan falsas detecciones, por lo que es fundamental estudiar los diferentes métodos para optimizar el reconocimiento de dichas actividades, en especial las caídas.

Investigaciones como las reportadas en [31] [56] aseguran que el paso más importante para tener un reconocimiento exitoso de actividades, es el seleccionar un método de extracción de características de una imagen o video, por lo que se han planteado diferentes métodos para diferenciar efectivamente las actividades no intencionales, como caídas, de otras actividades diarias. La presente revisión de la literatura se realiza con base en la clasificación de métodos de extracción de características expuesta en [57] la cual los organiza por enfoques; características locales, características globales, representación basada en profundidad y se anexa un actual método basado en redes neuronales convolucionales. Para [58], la forma o el borde de los datos relevantes de los objetos se utilizan para determinar las características locales; sin embargo, la información global apunta hacia la descripción del flujo o movimiento de un video.

2.3.1 Extracción de Características Globales

Este método permite extraer descriptores globales de videos e imágenes, lo que según [57] permite localizar al sujeto humano y con métodos de sustracción aislarlo del fondo para adquirir su silueta y forma. Otros métodos de representación global son los llamados volúmenes espacio-tiempo 3D, que realizan seguimiento de la silueta del ser humano durante un tiempo determinado, también se expone el método de transformada de Fourier, quien basa su trabajo en el dominio de la frecuencia de las siluetas de interés para el reconocimiento de las actividades.

Existe variedad de investigaciones que utilizan extracción de características globales para el reconocimiento de actividades humanas, y en especial el reconocimiento de caídas en personas adultas, por lo general dichas investigaciones aprovechan la silueta de los cuerpos humanos para alcanzar el objetivo de reconocimiento. [51][31][29][59][60] son trabajos que tienen como objetivo principal detectar caídas en adultos mayores y centran su estudio en la extracción de la silueta humana para su posterior procesamiento, donde radican las diferencias de las investigaciones. Khan & Sohn utilizan la silueta humana en [51], para extraer información de la persona adulta, y utilizan transformada R, características invariantes de escala y rotación, y *kernel discriminant analysis* (KDA) para intentar detectar caídas humanas teniendo en cuenta las diferentes distancias de las personas ante las cámaras; por su parte [31] y [59] tienen diferentes factores en común; ambas técnicas detectan caídas de personas adultas, extraen la silueta del adulto y calculan el centro de masa de la figura humana, sin embargo, Yu & Yu, et al. [31] utilizan el método expuesto en [61] para extraer y delimitar las siluetas de las personas con *Ellipse Features*, y buscan las características de las estructuras de cada

forma de las acciones humanas para ubicar su centroide, como método de detección de caídas. Mientras que Yu & Naqui, et al. [59]; además, de calcular el centroide de la silueta humana, identifica la orientación a la que se encuentra la persona, para lo cual necesitan al menos dos cámaras sincronizadas para minimizar el problema de la oclusión. Finalmente [60] utiliza la silueta del ser humano extraída de videos o imágenes, para luego identificar histogramas de la proyección de la silueta segmentada y realizar análisis de los cambios temporales de la posición de la cabeza del anciano, para identificar una posible caída.

Los anteriores sistemas fueron implementados en diferentes áreas de trabajo, por lo que tienen diferentes rendimientos, sin embargo, los lugares en los que se realizaron las investigaciones son entornos controlados, apartamentos pequeños con múltiples cámaras y con pocos cambios de iluminación, e incluso con altos costos computacionales, por ejemplo [62] utiliza variedad de cámaras para extraer imágenes en 3D, teniendo como objetivo detectar y analizar el volumen en el espacio vertical de las siluetas de las personas de tercera edad, con el fin de activar una alarma de caída cuando la distribución del volumen esté anormalmente cerca al piso, durante un largo periodo de tiempo, el método alcanzó un 99.7% de efectividad de reconocimiento, pero lo logra con el uso de 8 cámaras simultaneas, lo que tiene un alto costo computacional en sincronización y desempeño, convirtiéndose en un sistema difícil de implementar en la cotidianidad.

Por otra parte, en [63] se realiza reconocimiento de acciones humanas en interiores y se prueba en diferentes entornos, con iluminación natural, existencia de diferentes sombras y diversas actividades diarias, lo que causa muchas fallas en el reconocimiento de las acciones, sin embargo, se basa en el uso de una única cámara RGB, por lo que es simple de ser implementado e implica bajo costo computacional. Las caídas de los adultos mayores son detectadas analizando la orientación de los movimientos, magnitud de los movimientos y cambios en la figura humana, además del movimiento en el histograma de las imágenes. El autor aconseja que en investigaciones futuras se podrían usar técnicas adicionales que incluyen la detección de la cabeza y la zona de inactividad.

El flujo óptico es una técnica de extracción global que ayuda a extraer y describir siluetas en fondos con movimiento o dinámicos, [64] utiliza esta técnica para reconocer las actividades realizadas por futbolistas, tenistas y bailarinas de ballet por medio de video transmisiones. El autor propone aplicar esta técnica para extraer el fondo dinámico y únicamente enfocarse en las siluetas de los deportistas.

A pesar de que los sistemas que utilizan extracción de características globales han mostrado buen rendimiento en entornos controlados, [57] expone que estos métodos presentan grandes dificultades por su sensibilidad al ruido, oclusiones y cambios en el punto de vista. Además, autores como [56] manifiestan que los métodos basados en siluetas y figuras carecen de solidez y generalización de la aplicación, ya que dependen de una extracción precisa de la silueta humana y de las diferentes transformaciones geométricas que pueden verse distorsionadas por la distancia y la posición del sujeto ante la cámara.

2.3.2 Extracción de Características Locales

En [57] se explica que este método de extracción de características locales se centra en parches locales específicos que se determinan mediante detectores de puntos de interés o muestreos densos, lo que quiere decir que cubre densamente el contenido de un video o imagen. El primer detector de puntos de interés fue el planteado por Harris & Stephens [65] que es conocido por ser un gran detector de esquinas, dando origen a posteriores investigaciones como la realizada por Laptev & Lindeberg [66] proponiendo puntos de interés de espacio-tiempo 3D (STIP) convirtiéndose en los principales detectores de puntos de interés y originando diferentes investigaciones, [67][68][69], que tienen como objetivo optimizar estas técnicas. Según [58] STIP son técnicas fundamentales para la extracción de puntos de interés robustos de un video o una imagen en el dominio espacio-temporal, como por

ejemplo un punto de esquina o un punto aislado donde la intensidad es máxima o mínima, o puntos finales de líneas o curvas.

La investigación presentada en [30] centra su trabajo en la simulación de un ambiente de hogar inteligente con el uso de 2 cámaras y un sensor Kinect ubicado entre las cámaras, y adopta extracción de características locales con técnicas espacio-temporales para lo cual utilizan el algoritmo Harris3D como detector de características y STIP como descriptor de características, las principales dificultades del sistema radican en problemas de oclusión y desorden en el fondo, además de que el seguimiento del cuerpo humano es una tarea desafiante y propensa a errores, también causa problemas de reconocimiento el hecho de que el sensor Kinect puede detectar información esquelética solo para objetos en el rango de 1.2 a 3.5 metros. Por su parte [70] utiliza de forma diferente el detector de puntos de esquina de Harris y la forma del histograma de las diversas imágenes, para reconocer las diferentes actividades realizadas en dos sets con ambientes controlados, los resultados obtienen tasas de reconocimiento del 95% y 88% en el set1 y set2 respectivamente.

Por su parte [71] pretende realizar reconocimiento de actividades humanas en línea, es decir sin almacenar ningún video, el sistema aprende inmediatamente las acciones en la escena y las clasifica, teniendo en cuenta la forma de las acciones humanas, utiliza técnicas de extracción de puntos de interés y a su vez analiza el histograma de la imagen para identificar las acciones realizadas, el método tiene una efectividad de reconocimiento de 87% en acciones no complejas. Por su parte [72] también obtiene tasas de reconocimiento similar, pero combinando características espacio-temporales locales y la construcción de un diccionario visual, proponiendo un sistema llamado supervector híbrido.

Otra técnica se expone en [73] quien enfoca el reconocimiento de la acción mediante la codificación de características de gradiente espacio-temporal 3D locales dentro del marco de codificación disperso. Al hacerlo, cada característica espacio-temporal local se transforma en una combinación lineal de unos pocos "átomos" en un diccionario entrenado para detectar movimiento local y características de apariencia, el método proporciona aumento en la invariancia de escala, logrando reconocer algunas actividades básicas.

Teniendo en cuenta las anteriores investigaciones y lo planteado por [74] la extracción de características locales no requiere preprocesamiento, como la segmentación de fondo o la detección humana, ofrecen invariancia de escala y rotación, son estables bajo los cambios de iluminación y son más resistentes a la oclusión en comparación con las características globales.

En [57] se resalta el hecho de que, aunque estos detectores logran buenos resultados en HAR, tienen una gran deficiencia debido a que en muchas ocasiones el cálculo de los puntos de interés estables no es el adecuado; ya que, los puntos de interés "correctos" y "discriminativos" son difíciles de identificar. Por otra parte [74] también encuentra dificultades en el presente método de extracción de características locales, porque se ven fácilmente afectados por los cambios en la vista de la cámara, el movimiento de fondo y el movimiento de la cámara.

2.3.3 Extracción de Características Basadas en Profundidad

El desarrollo de sensores de profundidad como el Microsoft Kinect [75], han permitido tener mayor acceso a mapas de profundidad y posiciones de articulaciones esqueléticas en tiempo real, lo que ha contribuido a las investigaciones en reconocimiento de actividades humanas por medio de visión computacional.

Son diversas las investigaciones, como las reportadas en [52][76][77][42][78][79][80][81], que hacen uso del sensor Kinect como instrumento de adquisición de información y que utilizan sus imágenes de profundidad para el reconocimiento de actividades humanas, la diferencia radica en las características que cada investigación desea extraer, por ejemplo, Ma & Wang, et al. [52] realizan una completa

investigación que intenta reconocer seis acciones humanas (caídas, flexiones, personas sentadas, personas en cuclillas, caminando y acostadas), combinando técnicas de extracción globales y locales de la imágenes de profundidad, analizando los cambios de la forma humana en períodos cortos de tiempo; por otra parte, [76] no prioriza el análisis de la forma de la silueta humana, y centra sus recursos en extraer de las imágenes de profundidad, la orientación del cuerpo humano y el cálculo de la altura de la columna vertebral de la persona monitoreada, dichos cálculos permiten establecer la distancia a la cual el sujeto se encuentra del suelo para determinar una posible caída, muy similar a lo planteado por [77] quien en vez de calcular la distancia de la columna vertebral hasta el suelo, lo hace con el centro de masa de la persona de tercera edad, y anexa el ángulo entre el cuerpo humano y el plano del piso, si estos datos son inferiores a unos umbrales específicos, entonces una caída es detectada.

Por su parte [42] complementa el uso de imágenes de profundidad y el cálculo de la distancia del centro de masa humano hasta el suelo, con un sensor acelerómetro, para detectar caídas de personas de la tercera edad, si la aceleración supera un valor umbral significa que la persona se encuentra en movimiento y en ese momento empieza a trabajar el sensor de profundidad que extrae la información del adulto mayor para detectar una caída. [78] es otra investigación que se basa en la velocidad de la persona y en la posición del sujeto visualizado por medio del sensor Kinect; ya que, si se detecta una alta velocidad en un corto tiempo, se asume que una caída ha ocurrido lo cual se confirma o se descarta analizando la posición del cuerpo, el sistema tiene una precisión promedio de 93.94%.

En otra investigación, [79], los autores detectan la caída de la persona mayor, utilizando el sensor Kinect para extraer la imagen 3D del entorno, con el fin de realizar un cuadro delimitador 3D que rodea a la persona de tercera edad. Cuando el cuadro delimitador tiene cambios en su ancho, alto o profundidad se analiza a qué velocidad ocurrió dicho cambio, si esa velocidad es más alta que un determinado umbral se considera como una caída. Por su parte [80] usa imágenes de profundidad para extraer información como movimiento del torso humano, las posiciones 3D de la articulación central de la cadera y la articulación central del hombro, además de la altura del centroide de la persona; se puede identificar una caída cuando las tasas de las anteriores características alcanzan valores umbrales. Este método es robusto, pero usar únicamente el sensor Kinect, hace que el sistema sea dependiente de la distancia a la que el sensor trabaja.

A diferencia de las anteriores investigaciones, en [81] no calcula distancias hasta el suelo de ninguna parte del cuerpo humano, en su lugar, combina la extracción de algunas características globales con datos de profundidad para reconocer continuamente las actividades diarias de los ancianos, se utiliza la transformación R para extraer siluetas de profundidad de partes del cuerpo de las personas mayores y posteriormente usar modelos ocultos de Markov para entrenar y reconocer las actividades diarias del hogar. Los resultados demostraron una tasa de reconocimiento promedio de 96.55%.

Según [52] una gran ventaja al usar sensores como Kinect es que la iluminación no es un problema a la hora de extraer siluetas debido a que el sensor trabaja con luz infrarroja, por lo que también es capaz de reconocer siluetas humanas en la oscuridad, otra gran ventaja que tiene el sensor Kinect es que también permite extraer la información del esqueleto humano, [82], para reconocer las acciones humanas, en la actualidad esta técnica tiene una alta aplicación en diferentes investigaciones [83][84][85][86]; sin embargo, su gran dificultad es la oclusión, ya que el reconocimiento se puede ver afectado si el cuerpo humano es ocluido por cualquier objeto, afectando el reconocimiento de diferentes actividades, por lo que investigaciones como [87][88][89] fusionan características espacio-temporales con cámaras RGB y datos de profundidad, para intentar reducir el problema de oclusión. Por supuesto, la fusión de datos ocasiona el procesamiento de volúmenes de datos más grandes, lo que aumenta las dimensiones de las características, por ende, estos factores aumentan la complejidad computacional del algoritmo de reconocimiento de acciones.

2.4 Redes Neuronales Convolucionales (CNN)

Las redes neuronales convolucionales (CNN, *Convolutional Neural Networks*) son un tipo de inteligencia artificial que se especializa en el procesamiento de pixeles y la extracción de características y patrones a partir de imágenes proporcionadas. La capa de entrada es la que contiene los valores de los pixeles de una determinada imagen, por ejemplo, si a la entrada de una CNN, se tiene una imagen de 30 pixeles de alto, por 30 pixeles de ancho, equivale a 900 neuronas a la entrada de la red (suponiendo que la imagen es a blanco y negro, si la imagen fuese a color, las neuronas de la entrada serian 30x30x3 (RGB)).

Una de las primeras arquitecturas convolucionales fue LeNet-5 [90], esta red fue utilizada por los bancos para identificar números escritos a mano en los cheques. En la actualidad existen nuevas estructuras de CNN, las principales diferencias están en el número de capas, funciones de activación y se cuenta con mejores hardware para lograr un mayor éxito en el entrenamiento cuando se trabaja con redes profundas y con grandes conjuntos de datos.

Un factor que causó el aumento en el uso de redes neuronales convolucionales fue la competencia anual ImageNet [ILSVRC] [91] (Desafío de reconocimiento visual a gran escala de ImageNet), la competencia utiliza el conjunto de datos ImageNet [92], las redes neuronales convolucionales han sido ganadores constantes de este concurso desde 2012 hasta 2017, las diferencias arquitectónicas entre los ganadores son bastante pequeñas, por eso se puede definir una estructura general. La estructura típica de una red neuronal convolucional está compuesta por capa de Entrada, capa de Convolución, capa de Pooling y capa Fully Conected [93].

El presente estado del arte destaca que en los últimos años ha tenido una creciente importancia e impacto el uso de redes neuronales convolucionales en el reconocimiento de actividades humanas, su clasificación y su optimización, por lo que diferentes autores adoptaron su uso como método de reconocimiento. Por ejemplo [94], aplica una red neuronal convolucional de retroalimentación de flujo óptico a la transmisión de video, que incorpora histogramas de estadísticas de puntos, el límite del objeto de movimiento, y el límite del sujeto para detectar caídas; además, [95] propone un modelo novedoso de esqueletos dinámicos llamado Redes Convolucionales de Gráficos Espaciales-Temporales, éste aprende automáticamente los patrones espaciales y temporales de los datos, lo que permite mayor capacidad de generalización, [96] también basa su investigación en el mapa del esqueleto humano para la predicción de caídas humanas, apoyando su investigación en la herramienta OPENPOSE para obtener el mapa óseo y convertirlo en conjunto de datos para posteriormente alimentar la red CNN. Otras investigaciones se basan en el movimiento de las personas para usar la red neuronal, [97] extrae la trayectoria que se presenta en un determinado escenario, tratando de reconocer y clasificar diferentes actividades, mientras en [98] utilizan imágenes de flujo óptico como entrada de las redes neuronales seguidas de una fase de entrenamiento para detectar caídas, similar a lo planteado por [99] quien también incorpora flujo óptico a una red neuronal convolucional, con el objetivo de que esta no solo aprenda información estática. Por otra parte, las CNN también han abordado investigaciones que incorporan mapas de profundidad extraídos de Kinect para el reconocimiento de caídas [100] [101], Adhikari. Et al. [101] concluye que combinar la sustracción de fondo de imágenes RGB e imágenes con profundidad con CNN brinda una solución posible para monitorear caídas basadas en video de interiores.

Por otro lado en [102] se implementa una red neuronal convolucional tridimensional (3D CNN), en la que se extraen las características espaciales de las imágenes 2D, y se incorpora la información del movimiento del video para detectar una caída, ayudando a disminuir los fallos de reconocimiento ocurridos por el ruido de la imagen, la variación de la iluminación y la oclusión. Por su parte [103] también incorpora 3D CNN, pero además oculta las regiones faciales ópticamente percibibles en la fase

de captura de video, contribuyendo a la protección de la privacidad al usar cámaras de vigilancia. Otra red neuronal convolucional con características propias es implementada en [104], su autor crea una CNN *multi-stream* que es una CNN con múltiples flujos, es decir construye cuatro redes neuronales convolucionales, las cuales son alimentadas con las mismas imágenes, pero extrayéndoles diferentes características (color, textura, profundidad, forma, movimiento), al final concatena las cuatro CNN para obtener una única clasificación de actividades y poder reconocer caídas.

Un método diferente de aplicación de redes neuronales convolucionales para el reconocimiento de caídas es planteado en [105], en el que se realiza extracción de características a imágenes de caídas de personas, y se anexa una CNN que es utilizada únicamente para reconocimiento facial, ya que el autor manifiesta que la expresión humana en el momento de una caída es altamente diferenciable con el uso de estas redes.

Un gran inconveniente que se tiene al trabajar con redes neuronales convolucionales es que se necesita una gran cantidad de datos para su entrenamiento, de ahí que algunos autores [106][107][108] incorporen en sus estructuras de red, diferentes redes con arquitecturas pre-entrenadas o se basen en ellas, como es el caso de las arquitecturas AlexNet [109], VGG16 [109] o ResNet [110].

Según [111] a pesar de que las aplicaciones de las CNN en el reconocimiento de actividades son exitosas, se encuentran en entornos muy restringidos y manifiesta que ninguna de estas redes es flexible para funcionar bien fuera de su dominio. Precisamente [112] y [113] son investigaciones que se han preocupado por el funcionamiento de los algoritmos de detección de actividades humanas teniendo en cuenta caídas reales, trabajando con extracción de características locales/globales y extracción de características por medio de CNN respectivamente. Dando a conocer que la gran mayoría de investigaciones que se centran en el reconocimiento de actividades humanas usan breves segmentos de datos de video capturados en entornos artificiales, condiciones óptimas y con caídas simuladas por actores. Por lo que Debard & Mertens, et al. [112] seleccionó algoritmos que presentan buen porcentaje de reconocimiento de actividades cuando han sido utilizados con bases de datos realizadas en escenarios controlados y actuados, y los utiliza para implementarlos en entornos reales, con caídas reales de personas adultas, concluyendo que no presentan la misma eficiencia, ya que en su elaboración no se tuvo en cuenta calidad de la imagen, problemas de sobreexposición, oclusiones y cambios en las condiciones de iluminación. Lo que demuestra que aún no se cumplen todas las especificaciones para que un sistema sea robusto en situaciones del mundo real.

2.4.1 You only look once (YOLO)

YOLO es un algoritmo que detecta objetos en imágenes, otorgando un cuadro delimitador con la respectiva predicción del objeto enmarcado (ver Figura 1), replanteando la detección de objetos como un único problema de regresión, directamente desde los píxeles de la imagen hasta las coordenadas del cuadro delimitador y las probabilidades de clase, siendo su principal característica que solo mira una vez (YOLO) una imagen para predecir qué objetos están presentes y dónde están.

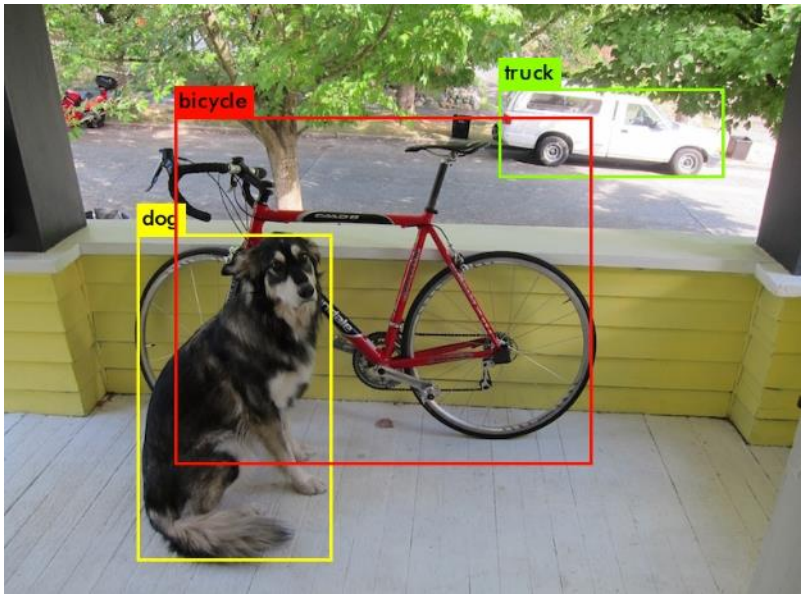

Figura 1 Cuadro delimitador y predicción otorgado por YOLO.

Fuente: Tomado de [114]

En su primera versión [114] YOLOV1 implemento una red neuronal convolucional, cuyas capas iniciales extraen características de la imagen, mientras que las capas totalmente conectadas predicen las probabilidades de salida y las coordenadas. La arquitectura de red consta de 24 capas convolucionales seguidas de 2 capas totalmente conectadas., utilizando filtros convolucionales de 3x3 y maxpool de 2x2 (ver Figura 2).

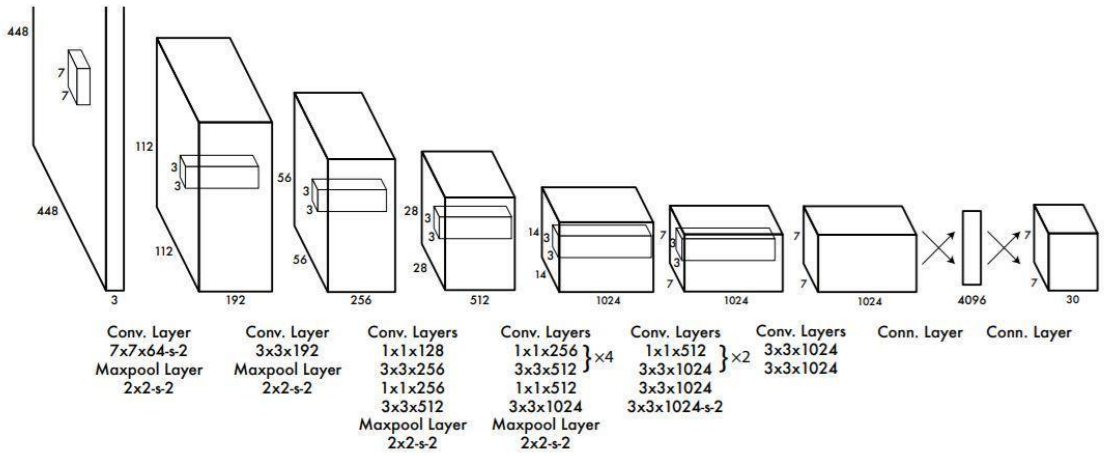

Figura 2. Estructura YOLOV1.

Fuente: Tomado de [114]

Además, utiliza una función de activación lineal rectificada (ReLu) con fugas, la cual se expresa de la siguiente forma:

$$f(x) = \begin{cases} x, & si\ x > 0 \\ & y \\ 0.1 * x & otro\ caso \end{cases} \qquad (1)$$

YOLO divide la imagen de entrada en una cuadrícula de tamaño S × S, si el centro de un objeto cae en una celda de cuadrícula, esa celda de cuadrícula es responsable de detectar ese objeto. Cada celda de la cuadrícula predice B cajas delimitadoras y las puntuaciones de confianza de esas cajas, que reflejan la confianza que tiene el modelo en que la caja contiene un objeto y también la precisión con la que cree que la caja predice, si la caja no contiene objeto la puntuación de confianza debe ser cero, de lo contrario la confianza será la intersección sobre la unión (*intersection over unión*, IOU) entre el cuadro predicho y el verdadero, entonces YOLO define confianza como:

$$Confianza = \Pr(Objeto) * IOU_{pred}^{truth} \qquad (2)$$

Entonces cada cuadro delimitador consta de 5 predicciones: x, y, w, h y confianza. Donde las coordenadas (x,y) representan el centro de la caja en relación con los límites y (w,h) representan ancho y alto de la caja respectivamente. Cada celda de la cuadrícula también predice C probabilidades de clase con respecto al objeto detectado, Pr(Class|Objeto). Estas probabilidades están condicionadas a que la celda de la cuadrícula contenga un objeto.

Al probar las predicciones del algoritmo, se multiplican las probabilidades de clase y de confianza:

$$\Pr(Class|Objeto) * \Pr(Objeto) * IOU_{pred}^{truth} \qquad (3)$$

lo que otorga una puntuación de confianza específica de la clase para cada casilla, codificando tanto la probabilidad de predicción de una clase como qué tan bien el cuadro predicho se ajusta al objeto. La Figura 3 describe, como YOLO Divide la imagen en una cuadrícula S × S y para cada celda de la cuadrícula predice cuadros delimitadores B, confianza para esos cuadros y probabilidades de clase C. Para finalmente realizar la predicción de los objetos en la imagen.

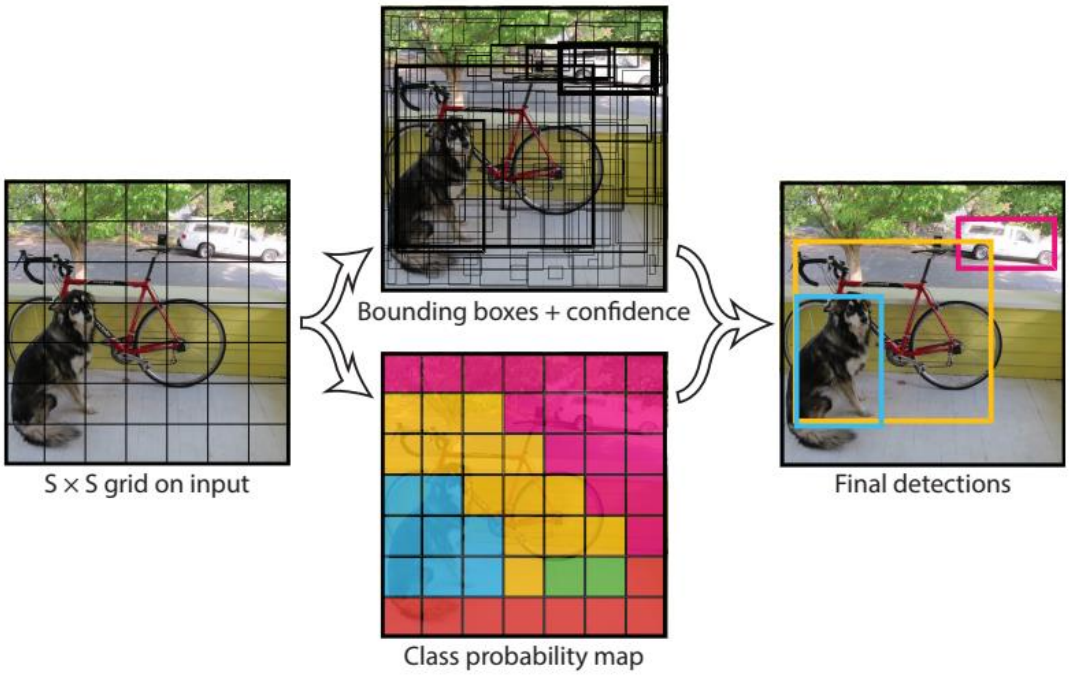

Figura 3. Proceso detección YOLO.

Fuente: Tomado de [114]

Por otra parte, el entrenamiento de YOLO, se realizó con el conjunto de datos ImageNet [92] que contiene 1000 clases diferentes, el entrenamiento duró aproximadamente una semana consiguiendo una precisión del 88%. Dicho entrenamiento se basa específicamente en regresión, optimizando el error de suma cuadrática la salida del modelo. Sin embargo, esta versión posee fuertes restricciones, si bien puede identificar rápidamente objetos en imágenes, tiene dificultades para localizar con precisión algunos objetos, especialmente los pequeños.

Por lo cual surgen nuevas versiones, como YOLOV2 o también llamado YOLO9000 [115] ya que es capaz de detectar más de 9000 clases y mantener su velocidad en tiempo real, además sus autores utilizaron pre-entrenamiento y el set de datos COCO dataset e ImageNet, para detección y clasificación respectivamente. El entrenamiento de la red se realizó con diferentes resoluciones (desde 320x320 pixeles, hasta 608x608 pixeles), por lo que a la entrada puede tener diferentes resoluciones de imágenes, contribuyendo, además, a la detección de objetos pequeños (ver Figura 4), concluyendo que a menor resolución YOLO v2 trabaja más rápido en comparación con imágenes de alta resolución.

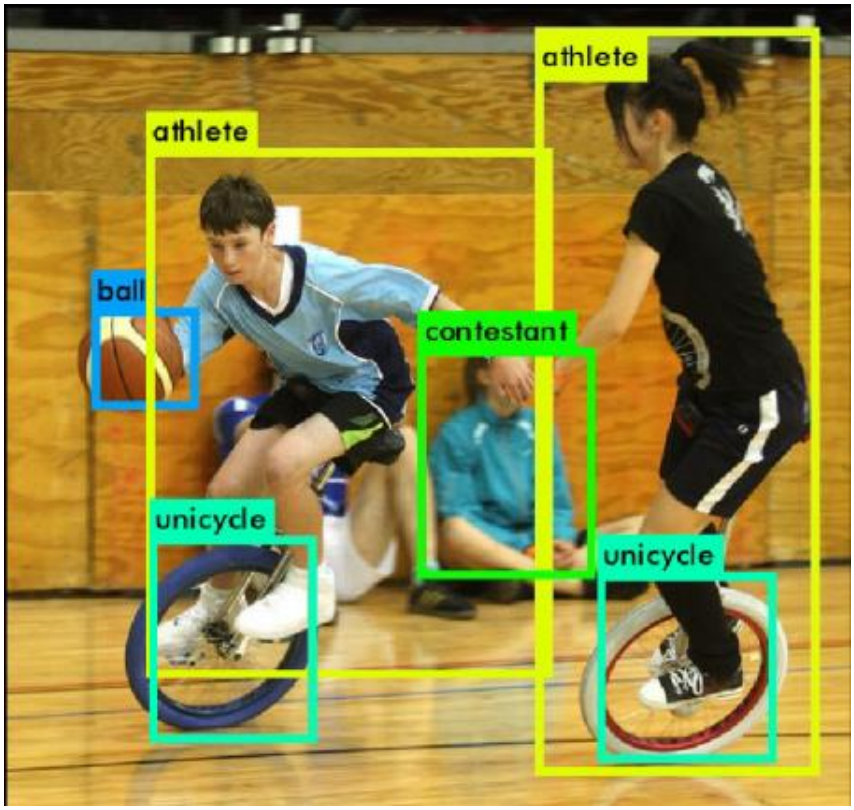

Figura 4. Detección YOLO9000.

Fuente: Tomado de [115]

Sus capas convolucionales, están basadas en la red pre-entrenada GoogleNet, la cual es una red neuronal para clasificación de imágenes, y que en comparación con otras arquitecturas como VGG-16 trabaja más rápido.

En el año 2018, Redmon & Farhadi [116] presentaron YOLOV3, considerada la mejor versión por sus avances precisión y velocidad, pudiendo ser utilizada en aplicaciones en tiempo real.

Contiene una de 53 capas de convolución y otras 53 capas adicionales dedicadas a la detección, por lo que mejoró la detección de objetos pequeños, la capacidad de detectar imágenes con diferentes aspectos, radios y escalas.

Para el año 2020, se publicó YOLOV4 [117], mostró una notable mejora en su desempeño, tanto en velocidad como en precisión, además se anexo “Data Augmentation” o aumento de datos, es la interpretación de la misma información desde diferentes puntos de vista, se basa en modificaciones pixel por pixel de las imágenes de entrenamiento, cambios de color, textura, parches negros o blancos, cortes y demás modificaciones sobre la imagen que ayudan al algoritmo a aumentar su precisión y flexibilidad, pero sin afectar su rendimiento en términos de velocidad. Este cambio de entrenamiento los autores lo denominan “Bag of freebies”, bolsa de regalos en español, pues implica un aumento en el costo de entrenamiento, pero aumentando la precisión sin costos en hardware, de forma que obtenemos un mejor desempeño [118]. A partir de la fecha se han publicado (en repositorios) diferentes versiones YOLO V5 (2020), YOLOV6(2022), YOLOV7(2022).

2.4.1.1 Deep Sort Tracking

Cuando en una imagen se realiza una detección de un objeto, en el siguiente frame se tiene que hacer una siguiente detección, así que en (n) imágenes pueden ser necesarias (n) detecciones. Deep Sort Tracking [119] evita este proceso, realiza una detección del objeto y a partir de ahí intenta seguirlo en los siguientes frames, por lo que es un método de seguimiento en línea en tiempo real. Este algoritmo agrega información de apariencia de los cuadros delimitadores, para ser robusto ante oclusiones, el mismo objetivo puede coincidir con la trayectoria anterior después de una oclusión prolongada o una detección perdida, lo que mejora el efecto del seguimiento.

En la Figura 5 se puede apreciar un ejemplo del algoritmo, en la imagen de la izquierda, se ha detectado a los sujetos 180, 129, 159, 165 y 137, un par de frames después, se muestra la imagen de la derecha, note que las detecciones tienen los mismos numerales, es decir no se realizó nuevas detecciones, si no, por el contrario, se realizó un seguimiento de la detección inicial.

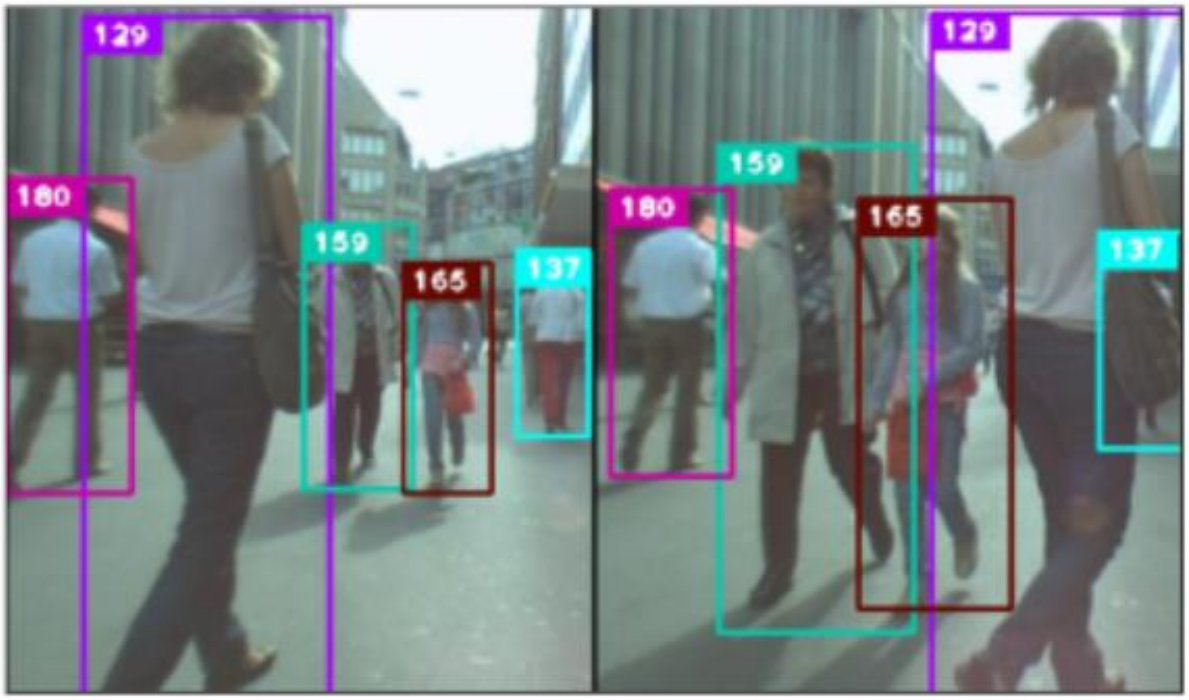

Figura 5. Seguimiento con Deep Sort.

Fuente: Tomado de [119]

En el método Deep Sort, primero predice la siguiente ubicación de cada trayectoria, y luego la distancia entre el nuevo cuadro delimitador y la ubicación predicha, este proceso se calcula mediante:

$$d^{(1)}(i,j) = (d_j - y_i)^T S_i^{-1}(d_j - y_i) \qquad (4)$$

donde se denota la proyección de la i-ésima distribución de pistas en el espacio de medición por (y_i, s_i) y la j-ésima detección del cuadro delimitador por d_j. y y S representan la matriz de covarianza S

de la media y y d como el vector variable. Luego se calcula la distancia entre el nuevo cuadro delimitador y el predicho (distancia de Mahalanobis), la cual tiene en cuenta la incertidumbre de la estimación del estado al medir cuántas desviaciones estándar se encuentra la detección de la ubicación media de la pista.

Después de calcular la distancia de Mahalanobis, se emplea una amplia red residual con dos capas convolucionales seguidas de seis bloques para calcular la información de apariencia de cada cuadro delimitador. Para hacer que el objeto aún se pueda rastrear después de un largo período de oclusión.

2.4.1.2 Atención visual guiada LSTM

Para abordar la tarea de detección de caídas en escenas concurridas, los mecanismos de atención son importantes tanto espacial como temporalmente. Por lo tanto, se introduce un LSTM (*Long short-term memory)* guiado por la atención para resolver el problema de predicción de los cuadros delimitadores rastreados que utiliza un modelo de atención visual para agrupar dinámicamente las características convolucionales capturando las regiones más importantes. Además, LSTM puede preservar la memoria temporal e incorporar la atención temporal.

Por su parte una red neuronal recurrente LSTM tiene como objetivo aprender dependencias a largo plazo; es decir, aprender las dependencias de valores futuros de una secuencia en función a los valores anteriores, por lo que están diseñadas para recordar información durante largos períodos de tiempo [120].

Todas las redes neuronales recurrentes tienen la forma de una cadena de módulos repetidos de red neuronal. Los LSTM también tienen una estructura similar a una cadena, pero en lugar de tener una sola capa de red neuronal, hay cuatro que interactúan constantemente (ver Figura 6).

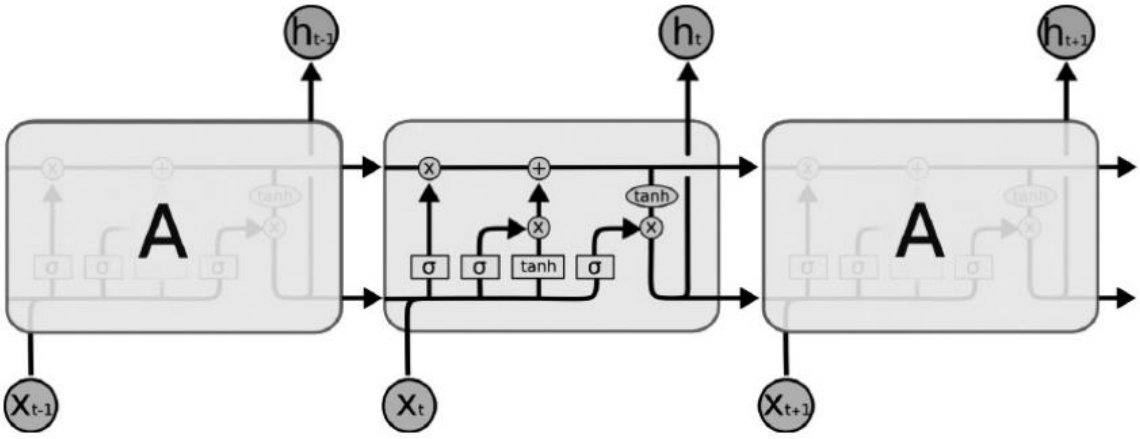

Figura 6. Estructura de una red LSTM.

Fuente: Tomado de [120]

La red LSTM modifica el estado de la celda al añadir o quitar información del estado de la celda anterior C_{t-1} para producir un nuevo estado C_{t+1}. El primer paso en una LSTM consiste en decidir qué información va a permanecer en el estado de la celda. Este procedimiento se realiza con una capa de activación, la cual toma la salida h_{t-1} y la entrada X_t , y representa con valores en el intervalo [0,1] la

necesidad de información (un valor más cercano a uno es más necesario que uno cercano a cero) para cada valor del estado C_{t-1}. Matemáticamente, este procedimiento se detalla en la ecuación (5), donde W_f representa la matriz de pesos y b_f el bias para el "forget gate" [120].

$$f_t = \sigma(W_f * [X_t, h_t - 1] + b_f) \quad (5)$$

El siguiente paso es decidir qué información retener en el estado de la celda. Para esto se utilizan dos capas, una sigmoide con la que se decide qué valores se actualizarán y una tangente hiperbólica que creará un nuevo vector de valores candidatos a ser añadidos al estado de la celda (ecuaciones (6) y (7)). Estos dos valores serán combinados más adelante para la actualización del estado de la celda [120].

$$i_t = \sigma(W_i * [X_t, h_t - 1] + b_i) \quad (6)$$

$$\widetilde{C_t} = tanh(W_c * [X_t, h_t - 1] + b_c) \quad (7)$$

Al terminar estos cálculos, el nuevo estado C_t se obtiene al multiplicar C_{t-1} por f_t, olvidando la información innecesaria y añadiendo la nueva información al multiplicar i_t por la matriz de candidatos α (ecuación (8)). Estos candidatos son escalados por i_t , ya que este representa cuánto se decidió actualizar cada valor del estado anterior

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C_t} \quad (8)$$

Finalmente, se necesita calcular la salida. Esta se debe basar en el estado actual, pero no toda la información es parte de la salida. Para realizar este filtrado, se aplica una capa sigmoidal para decidir qué valores del estado actual formarán parte de la salida. Luego, se aplica una tangente hiperbólica (para tener valores entre –1 y 1) y se multiplica estos valores por la activación de la capa sigmoidal para solo tener como output los valores seleccionados (ecuaciones (9) y (10)) [120].

$$o_t = \tanh(W_o * [X_t, h_t - 1] + b_o) \quad (9)$$

$$h_t = o_t * \tanh(h_t) \quad (10)$$

2.4.2 OpenPose

Es un algoritmo que permite obtener una visualización detallada de las personas en imágenes y vídeos: estimando la pose humana en 2D (ver Figura 7). La estimación humana se ha centrado en gran medida en la búsqueda de partes del cuerpo de los individuos. Inferir la postura de varias personas en imágenes presenta un conjunto único de desafíos. En primer lugar, cada imagen puede contener un número desconocido de personas que pueden aparecer en cualquier posición o escala. En segundo lugar, las interacciones entre las personas inducen interferencias espaciales complejas, debidas al contacto, la oclusión o las articulaciones de las extremidades, lo que dificulta la asociación de las partes. En tercer lugar, la complejidad del tiempo de ejecución tiende a crecer con el número de personas en la imagen, lo que hace que el rendimiento en tiempo real sea un reto [121].

Figura 7. Pose humana con OpenPose.

Fuente: Tomado de [121]

OpenPose reconoce diferentes puntos específicos de la silueta humana, que al interconectarse pueden describir la pose de un individuo, y también de todos los individuos a su alrededor como muestra la Figura 8, siendo el primer algoritmo con la capacidad de detectar conjuntamente un total de 135 puntos clave en el cuerpo humano, pies, manos y puntos faciales en imágenes individuales. Teniendo diferentes aplicaciones en el área de reconocimiento de actividades, robótica, seguimiento humano, aplicaciones médicas, o en la industria de los videojuegos.

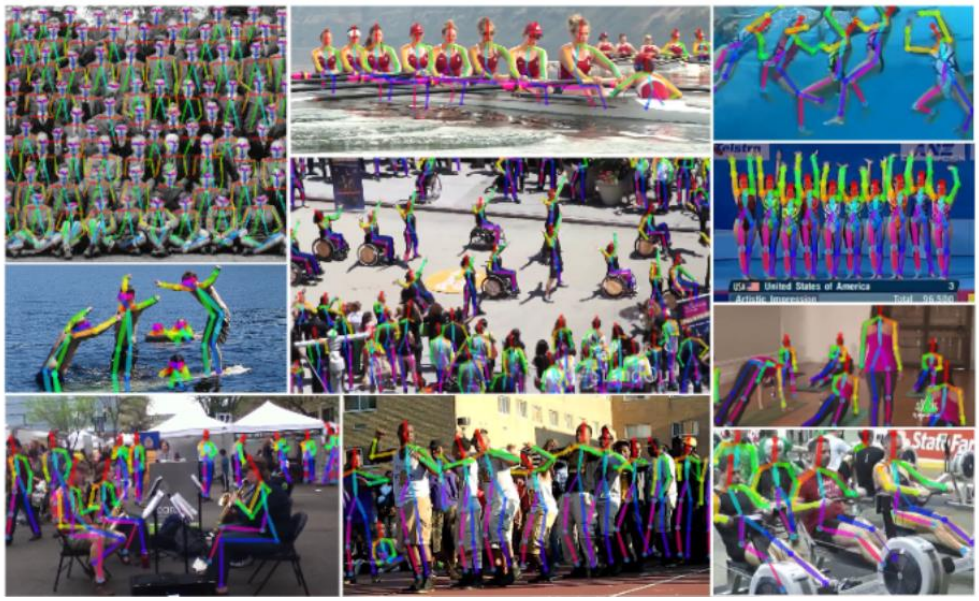

Figura 8. OpenPose con diversos individuos.

Fuente: Tomado de [121]

OpenPose presenta varias dificultades para predecir con éxito la estructura humana ya que cada imagen puede contener un número desconocido de personas que pueden aparecer en cualquier posición o escala. Además, las interacciones entre las personas inducen una interferencia espacial compleja, debido al contacto, la oclusión o las articulaciones de las extremidades, lo que dificulta la

asociación de las partes. También hay que tener en cuenta que la complejidad del tiempo de ejecución del algoritmo tiende a crecer con el número de personas.

OpenPose presenta un método eficiente para la estimación de poses utilizando la primera representación ascendente de puntajes de asociación a través de campos de afinidad de piezas (*Part Affinity Fields*, PAF), un conjunto de campos vectoriales 2D que codifican la ubicación y orientación de las extremidades en el dominio de la imagen. La Figura 9 muestra campos de afinidad de piezas (PAF) correspondientes a la extremidad que conecta el codo y la muñeca derecha. El color codifica la orientación, un vector 2D en cada píxel de cada PAF codifica la posición y orientación de las extremidades.

Figura 9. Campos de afinidad de piezas (PAF).

Fuente: Tomado de [121]

El sistema toma como entrada una imagen a color de tamaño $w * h$ (Figura 10.a), luego se predice un conjunto de mapas de confianza (S) que ubica partes del cuerpo humano (Figura 10.b), además de un conjunto de campos vectoriales (L) llamados PAF, cuya misión es codificar el grado de asociación entre las partes del cuerpo (Figura 10.c), produciendo las ubicaciones 2D de puntos anatómicos específicos para las personas de la imagen (Figura 10.e).

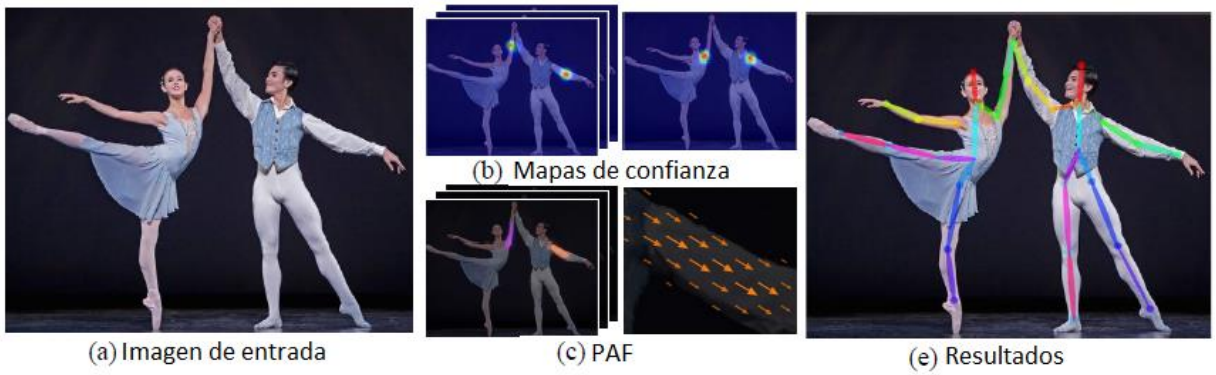

Figura 10. Proceso OpenPose.

Fuente: Tomado de [121]

El conjunto $S = (S_1, S_2, S_3, \ldots\ldots\ldots S_J)$ tiene J mapas de confianza, uno por cada parte del cuerpo. El conjunto $L = (L_1, L_2, L_3, \ldots\ldots\ldots L_C)$ tiene C campos de vectores, uno por cada extremidad, entendiendo que una extremidad es un par de partes del cuerpo. La Figura 9 es un ejemplo de un vector L_C.

Finalmente, los mapas de confianza y los PAF se analizan para generar los puntos clave 2D para todas las personas en la imagen.

Por su parte OpenPose, basa su estructura en una arquitectura de predicción iterativa (ver Figura 11), es decir cuenta con múltiples capas de redes neuronales convolucionales, el primer conjunto de etapas predice PAFs (L^t), mientras que el último conjunto predice mapas de confianza (S^t). Las predicciones de cada etapa y sus características de imagen correspondientes se concatenan para cada etapa posterior. Utilizando convoluciones de kernel de tamaño 3 su extremo.

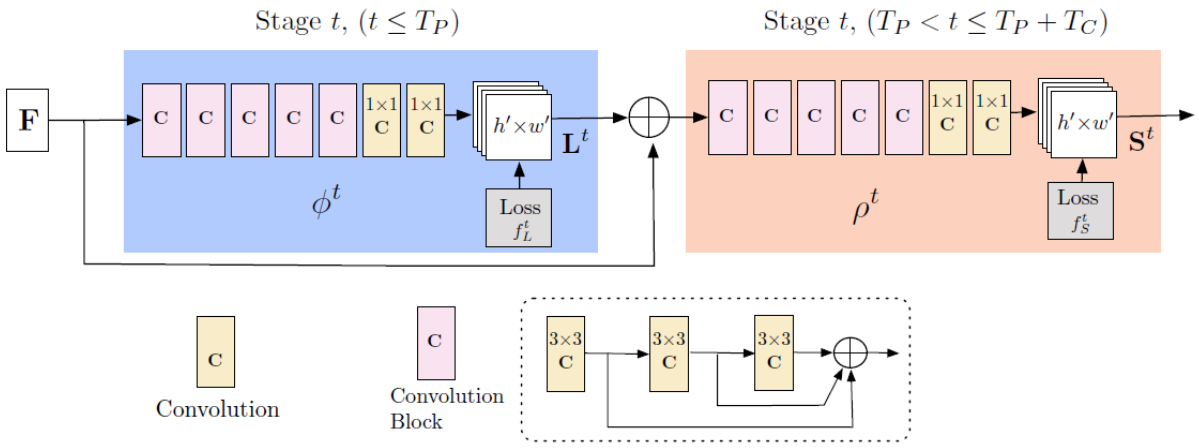

Figura 11. Estructura OpenPose.

Fuente: Tomado de [121]

El total de la estructura se compone de tres redes neuronales, la primera analiza la imagen entrante por medio de una red pre-entrenada VGG-19 que genera un conjunto de mapas de características (F), que se ingresan a la siguiente CNN (Etapa1), red que produce un conjunto de campos de afinidad parcial (PAF), $L^1 = \phi^1(F)$ donde ϕ^1 se refiere a la inferencia de la primer etapa de CNN, en cada etapa subsiguiente, las predicciones de la etapa anterior y las características F de la imagen original se concatenan y se utilizan para producir predicciones refinadas:

$$L^t = \phi^t(F, L^{t-1}), \forall 2 \leq t \leq T_P \qquad (11)$$

Donde ϕ^tse refiere a las CNN para la inferencia en la etapa t, T_P es el numero total de etapas PAF. Después de terminar con T_P iteraciones el proceso se repite para la detección de mapas de confianza, comenzando en la predicción de PAF más actualizada:

$$S^{T_P} = \rho^t(F, L^{T_P}), \forall t = T_P \qquad (12)$$

$$S^t = \rho^t(F, L^{T_P}, S^{t-1}), \forall T_P < t \leq T_P + T_c \qquad (13)$$

Donde ρ^t se refiere a las CNN para la inferencia en la Etapa t, y T_c al número de etapas del mapa de confianza total.

Para guiar a la red a predecir iterativamente PAF de partes del cuerpo en la primera rama y mapas de confianza en la segunda rama, se aplica una función de pérdida al final de cada etapa. Se calcula una pérdida L_2 entre las predicciones estimadas y los mapas y campos de verdad. La función de pérdida de la rama PAF en la etapa t_i y la función de pérdida de la rama del mapa de confianza en la etapa t_k son:

$$f_L^{t_i} = \sum_{C=1}^{C} \sum_P W(p) * ||L_C^{t_i}(p) - L_C^*(p)||_2^2 \qquad (14)$$

$$f_S^{t_k} = \sum_{j=1}^{J} \sum_P W(p) * ||S_j^{t_k}(p) - S_j^*(p)||_2^2 \qquad (15)$$

Donde L_c es la predicción correcta de PAF, S_j es el mapa de confianza de la predicción correcta y W es una máscara binaria con W(p) = 0 cuando falta la anotación en el píxel p. La máscara se utiliza para no penalizar los verdaderos pronósticos positivos durante el entrenamiento. Finalmente, la función perdida está dada por:

$$f = \sum_{t=1}^{T_P} f_L^t + \sum_{t=T_P+1}^{T_P+T_c} f_S^t \qquad (16)$$

Posteriormente OpenPose genera mapas de confianza $S_{j,k}^*$ individualmente, es decir para cada parte del cuerpo (j) en cada persona (k) que aparezca en la imagen, donde $x_{j,k}$ es la posición verdadera de la parte del cuerpo j para la persona k. El valor en la ubicación (p) en $S_{j,k}^*$, está dado por:

$$S_{j,k}^*(p) = e^{\left(-\frac{||p - x_{j,k}||_2^2}{\sigma^2}\right)} \qquad (17)$$

Donde **σ** controla la propagación de la función exponencial. El mapa de confianza predicho por la red es una agregación de los mapas de confianza individuales a través de un operador máximo,

$$S_j^*(p) = max_k * S_{j,k}^*(p) \qquad (18)$$

Entonces, al considerar la Figura 12 que muestra una sola extremidad, se obtienen los puntos $x_{j1,k}$ y $x_{j2,k}$ que pertenecen a un miembro (c) de una persona (k). Si un punto (p) se encuentra en la extremidad, el valor en $L_{c,k}^*(p)$ es un vector unitario que apunta de j1 a j2; para todos los demás puntos, el vector tiene valor cero. V es el vector unitario en la dirección de la extremidad. De esta forma se une puntos que pertenecen a una misma extremidad.

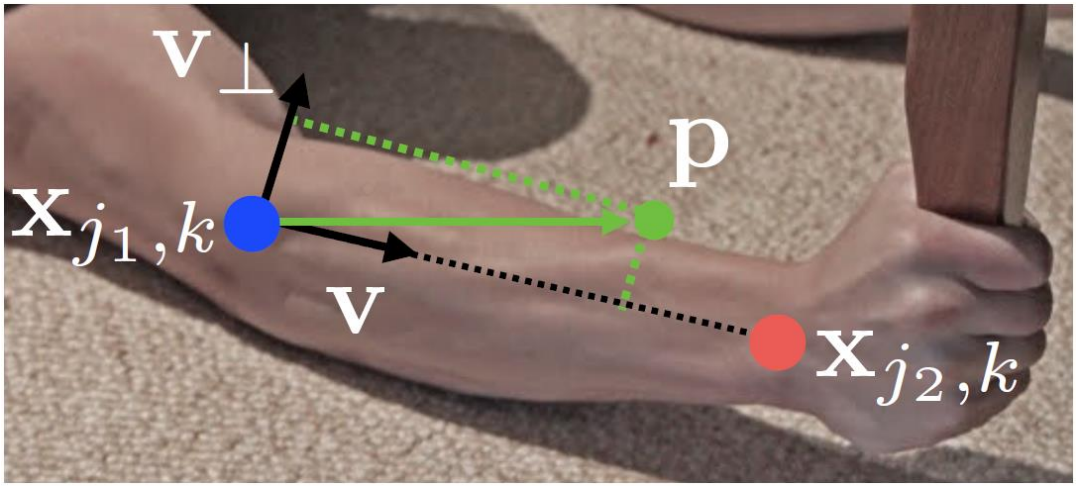

Figura 12. Formación de una extremidad con OpenPose.

Fuente: Tomado de [121]

El algoritmo ordena todas las conexiones posibles por pares según su puntuación PAF. Si una conexión intenta conectar 2 partes del cuerpo que ya han sido asignadas a diferentes personas, el algoritmo reconoce que esto contradiría una conexión PAF con mayor confianza, y posteriormente se ignora la conexión actual. De esta forma se consigue el mapa estructural de la silueta humana por medio de OpenPose.

2.5 Google Colaboratory

Google Colaboratory [122], también conocido como "Colab", es un producto de Google que permite programar y ejecutar código Python en un navegador por medio de un entorno interactivo denominado cuaderno de Colab que permite escribir y ejecutar código y que cuenta con ciertas ventajas, por ejemplo, no requiere de compleja instalación, no requiere de configuración, facilita el acceso a GPUs de forma gratuita y facilita compartir archivos.

Colab fue creado especialmente para ciencia de datos y aplicaciones de inteligencia artificial. Con Colab se puede importar conjuntos de datos de imágenes, entrenar un clasificador de imágenes y evaluar el modelo. Los cuadernos de Colab ejecutan código en los servidores en la nube de Google, lo que permite aprovechar la potencia del hardware de Google, incluidas las GPU y TPU, independientemente de la potencia del equipo que se esté usando.

2.6 Bases de Datos

A partir de los aportes realizados al desarrollo de sistemas de extracción de características desde diferentes perspectivas y a la continua evolución que tienen las redes neuronales convolucionales, se deja dilucidar la necesidad de comparar y analizar, cuál es su eficiencia en el desarrollo y funcionamiento de sistemas de reconocimiento de caídas en entornos poco controlados. Para tal objetivo es importante aclarar que se utilizan bases de datos de acceso público (URFD [42], Le2i [43], CMDFALL [44], FALL-UP [45], MCF [46], UCF101 [47]) para entrenar y optimizar los algoritmos de detección de caídas. Pero se evalúa el desempeño de algoritmos de detección de caídas en entornos que presenten cambios en iluminación, movimiento de la cámara, oclusiones y cambios de posición, con una base de datos propia (CAUCAFall [49]), que respete los estándares de las bases de datos

públicas. En la Tabla2 se especifica las principales características de las bases de datos de las que esta investigación hace uso. Lo anterior, como un siguiente paso que permita abordar sistemas de detección de caídas con un mejor rendimiento.

Tabla 2. Bases de datos que contienen caídas humanas.

Base de datos	Videos	Datos que otorga	Ambiente	Población	Tipos de caídas
URFD [42]	70 videos, 30 caídas y 40 actividades cotidianas.	Imágenes RGB, imágenes de Profundidad y señales de acelerómetro.	Interiores.	Personas adultas.	Caídas de personas estando de pie y sentadas en sillas.
LE2I [43]	191 videos, entre caídas y actividades cotidianas.	Imágenes RGB.	Interiores.	Personas adultas.	Caídas al caminar, con tropiezos y caídas desde sillas.
CMDFALL [44]	600 videos, con 20 acciones humanas, entre ellas caídas.	Imágenes RGB, imágenes de Profundidad y señales de acelerómetro.	Interiores con simulación hogareña.	30 hombres y 20 mujeres, entre los 21 y 40 años de edad.	Diferentes caídas, hacia atrás, hacia adelante, hacia la izquierda y hacia la derecha.
FALL-UP [45]	361 videos, entre caídas y actividades cotidianas.	Imágenes RGB, señales del Acelerómetro y de diferentes sensores.	Interiores.	17 personas adultas, entre los 18 y 24 años de edad.	Diferentes caídas.
Multiple Cameras Fall Dataset [46]	192 videos, entre caídas y actividades cotidianas.	Imágenes RGB.	Interiores.	Personas adultas.	Caída hacia Adelante, hacia Atrás, al Sentarse y caídas por pérdida de equilibrio.
UCF101 [47]	13.000 clips, 27 horas de datos de video con 101 acciones humanas.	Imágenes RGB.	Ambientes controlados, con movimiento en las cámaras y desorden en el fondo.	Personas adultas.	Diferentes caídas.

Fuente: El autor

Partiendo del anterior análisis y de la metodología explicada en la próxima sección del documento, la presente investigación propone la comparación de tres algoritmos ([123][124][125]) usados para el

reconocimiento de caídas humanas y que utilizan como método principal la visión computacional, evaluando su rendimiento en bases de datos públicas utilizadas entre la comunidad investigadora de caídas humanas y en una base de datos creada por los autores (CAUCAFall [49]), que intencionalmente tiene un entorno realista, con diferentes oclusiones, cambios de iluminación (natural, artificial y nocturna), diferentes ángulos de caída, diferentes texturas en el suelo y habitación, diferente ropa de los individuos, individuos de diferente edad, peso, altura y textura e incluso diferente pierna dominante, con la intensión de ver el comportamiento de los algoritmos seleccionados. Para entender el funcionamiento de la implementación de los algoritmos propuestos se debe tener en claro los conceptos de Máquinas de vectores de soporte (SVM, Support Vector Machines), Redes Neuronales Convolucionales, CNN VGG16, CNN Inception ResNet v2, YOLO (You only look once), DeepSort, Atención visual guiada LSTM, OpenPose y el entorno de programación Google Colaboratory.

3 Metodología de la Investigación

El objetivo general de esta investigación es el comparar el desempeño de algoritmos para el reconocimiento de caídas, que utilicen extracción de características con visión computacional y redes neuronales convolucionales, evaluando su rendimiento en diferentes bases de datos. Teniendo en cuenta lo anterior se llevó a cabo una metodología que consta de 8 fases. A continuación, se mencionan y se explican brevemente cada una de ellas:

Fase 1. Elaboración del estado del arte: Se realizó una revisión sistemática de la literatura especializada relacionada con el reconocimiento de actividades humanas, identificando las principales características para determinar caídas humanas.

Fase 2. Selección de algoritmos: Se realizó una discriminación de algoritmos que implementan extracción de características y redes neuronales convolucionales y que permiten reconocer caídas de personas adultas por medio de visión computacional.

Fase 3. Selección de parámetros y recolección de datos: Se seleccionaron los principales parámetros que permiten identificar caídas de personas adultas y, teniendo en cuenta los resultados, se recolectaron bases de datos que contienen caídas humanas y que hayan sido utilizadas por la comunidad científica.

Fase 4. Creación base de datos propia: Se seleccionó información de las bases de datos obtenidas en la fase 3, priorizando aquellas que fueron creadas en escenarios realistas y poco controlados, con el fin de diseñar y crear CAUCAFall, base de datos con características únicas, con un entorno hogareño y poco controlado, que respeta los parámetros de las bases de datos utilizadas por la comunidad científica.

Fase 5: Implementación de algoritmos: Se implementaron los algoritmos seleccionados en un entorno de programación y se realizaron diferentes experimentos.

Fase 6. Establecer índices de desempeño: Se seleccionaron los índices que permitan obtener una comparación de los diferentes algoritmos de reconocimiento de caídas.

Fase 7. Evaluación del desempeño de los algoritmos de reconocimiento de caídas: Se evaluó el desempeño de los algoritmos en bases de datos usadas por la comunidad científica y en CAUCAFall, realizando tablas de resultados de comparación.

Fase 8. Publicación, divulgación de resultados e informe final.

a continuación, se describe el proceso detallado de las fases de selección de los algoritmos a comparar, las bases de datos seleccionadas, el diseño de CAUCAFall, la implementación de los algoritmos y experimentos desarrollados y finalmente los índices de desempeño seleccionados.

3.1 Selección de Algoritmos a Comparar

Para el desarrollo del presente trabajo se toma como base la investigación de los autores Gutiérrez, Rodríguez y Martin publicada en el año 2021 [126], quienes realizan una revisión exhaustiva de los sistemas de detección de caídas basados en visión en bases de datos públicas como ScienceDirect, IEEE Explorer y Sensors database, complementando su búsqueda con literatura académica enfocada en salud como PubMed y MedLine. Los términos utilizados en la exploración bibliográfica fueron “detección de caídas” y “visión”, obteniendo como resultado 929 artículos, de los cuales se descartan 499 ya que los títulos de dichas investigaciones no coinciden con el contenido requerido. Las 430 investigaciones restantes son sometidas a un siguiente filtro en donde no se tienen en cuenta aquellas

que trabajen con el reconocimiento de actividades humanas diferentes a caídas, o que su temática central sea la prevención de caídas. Tampoco son de interés las investigaciones que trabajen con tecnologías mixtas; es decir que involucren sensores diferentes a cámaras, por lo que resultan 81 investigaciones que cumplen con requisitos de detección de caídas por medio de visión computacional.

Esta investigación anexa un siguiente filtro de búsqueda para las 81 investigaciones, seleccionando únicamente a aquellas que utilicen e implementen imágenes RGB, puesto que nuestro trabajo utiliza dichas imágenes, resultando 50 investigaciones, las cuales fueron usadas para determinar las bases de datos más utilizadas por la comunidad científica para la detección de caídas con imágenes RGB y poder compararlas con CAUCAFall, los resultados son mostrados en la Tabla 3.

Tabla 3. Bases de datos más utilizadas por la comunidad científica.

Base de datos	Número de investigaciones	%
UR Fall Detection [42]	21	42%
Multicam Fall Dataset [46]	9	18%
LE2I [43]	8	16%
Fall Detection Dataset [127]	7	14%
UPFall [45]	1	2%
Center For Digital Home [128]	1	2%
MOT Dataset [129]	1	2%
COCO Dataset [130]	1	2%
ntu rgb+d [131]	1	2%

Fuente: El autor

Sin embargo, no todas las bases de datos presentadas en la Tabla 3 son de acceso público. Aplicando un filtro para seleccionar artículos de revista que sean resultado de investigaciones que trabajen con mínimo una base de datos de acceso público, se obtuvieron 12 investigaciones que son presentadas en la Tabla 4, con una breve descripción de la metodología usada y la base de datos que utilizan.

Bajo criterios de selección, como la cantidad de parámetros de los algoritmos que otorgan los autores y las métricas de desempeño utilizadas, se obtuvieron 3 investigaciones ([123][124][125]) para la presente comparación.

Tabla 4. Investigaciones preseleccionadas para análisis de rendimiento.

Referencia	Algoritmos	UR Fall Detection	Multicam Fall Dataset	LE2I	UPFALL
F. Harrou et al. [132]	Background subtraction / depth characterization	✓	-	-	-
Syed F. Ali et al. [133]	Background subtraction (GMM) / global Characterization	✓	✓	-	-
D. Kumar et al. [134]	Silhouette segmentation / global characterization / silhouette center / angular velocity determined by long short-term memory (LSTM)	✓	-	-	-
F. Harrou et al. [135]	Background subtraction / global characterization	✓	-	-	-
Swe N. Htun [123]	Background subtraction / global characterization	-	-	✓	-
Yaxiang Fan et al. [136]	CNN / local characterization	-	✓	✓	-
W. Min et al. [137]	Object recognition through CNN / local characterization	✓	-	-	-
Chao Ma et al. [138]	Feature maps obtained through CNN / local characterization	✓	✓	-	-
Ricardo Espinosa et al. [139]	Global characterization / feature maps obtained through CNN / local characterization	-	-	-	✓
B. Wang et al. [140]	Human keypoints identified by OpenPose / DeepSORT / local characterization	✓	-	✓	-
Qi Feng et al. [124]	Feature maps obtained through of CNN and LSTM / local characterization	✓	✓	-	-
Qingzhen Xu et al. [125]	Human keypoints identified by OpenPose and CNN / local characterization	✓	✓	-	-

Fuente: El autor

3.1.1 Algoritmo que Incorpora Técnicas de Extracción de Características

El primer algoritmo implementado en esta investigación está basado en el trabajo realizado en [123] en donde Htun, Zin y Tin basan su algoritmo en cuatro módulos: (1) Detección del objeto, (2) extracción de características, (3) Análisis de los eventos y (4) establecimiento de reglas de toma de decisiones utilizando un modelo oculto de Markov (*Hidden Markov Model*) para diferenciar caídas, de actividades normales, tal como muestra la Figura 13.

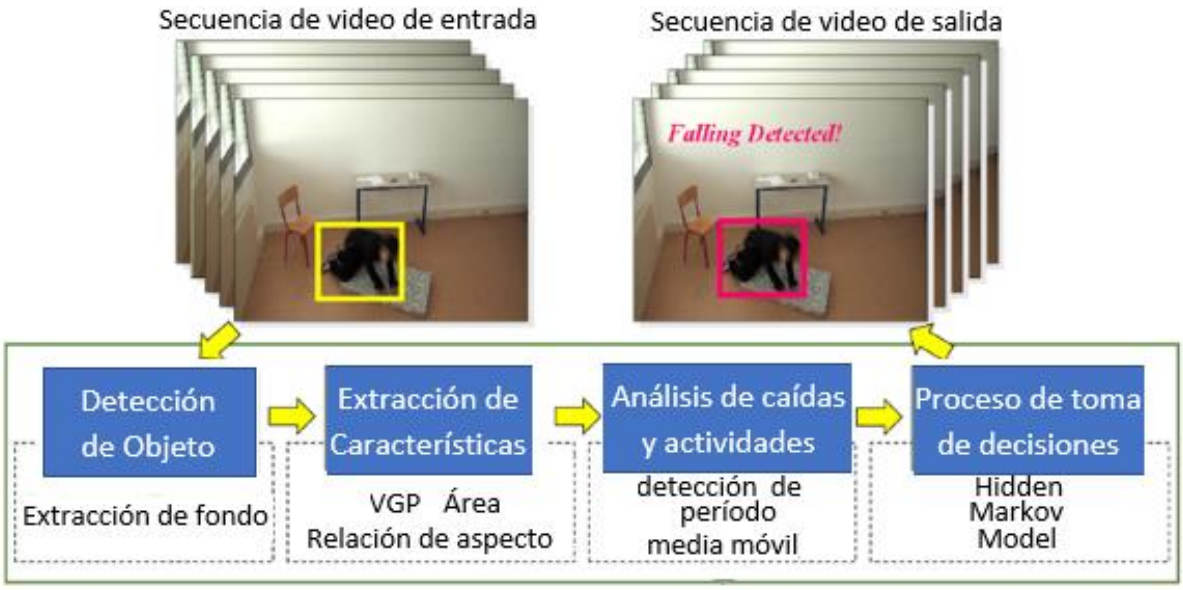

Figura 13. Módulos del Algoritmo de extracción de Características.

Fuente: Tomado de [123]

Sin embargo, los mismos autores proponen en [141] el uso de SVM (Support Vector Machine) como clasificador, tal como muestra la Figura 14. La presente investigación implementa un algoritmo basado en las dos investigaciones de los autores [123] y [141] escogiendo las características con mejores resultados como base del algoritmo a utilizar y comparar.

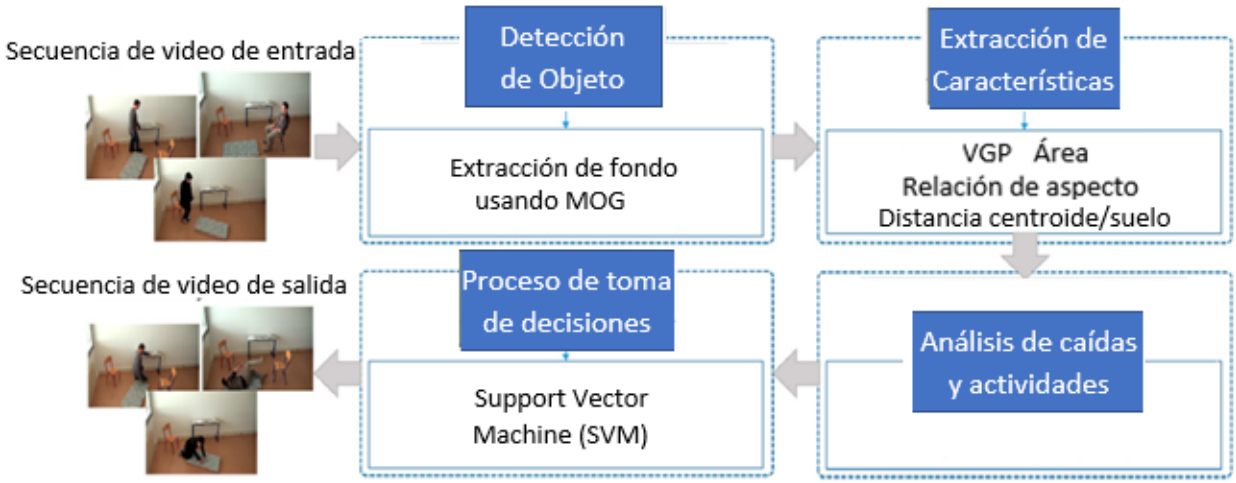

Figura 14. Algoritmo con SVM como clasificador.

Fuente: Tomado de [141]

Este algoritmo, primero realiza una detección del objeto, cuyo objetivo principal es separar correctamente los objetos de primer plano del fondo de la escena, basando su funcionamiento en seleccionar una distribución específica de mezcla de gaussianas (MoG), algoritmo que permite modelar

el primer plano, que se actualiza fotograma a fotograma. Procediendo a realizar la extracción de características de las diferentes imágenes.

Cuando se detecta un objeto, en este caso la silueta humana, se calcula el centroide (c), área, alto, ancho y la relación de aspecto (r), además de un punto de puesta a tierra virtual (*Virtual Grounding Point*, VGP) esencial para determinar una caída. Entonces la posición (p) en el momento (t) del objeto de primer plano detectado se define como:

$$p(t) = (x(t), y(t)) \qquad (19)$$

Donde el centroide del objeto esta dado por la ecuación 20:

$$C(t) = (x_c(t), y_c(t)) \qquad (20)$$

Específicamente, x_c y y_c están dados por:

$$x_c = \sum_{i=1}^{N} \frac{x_i}{N}, \quad y_c = \sum_{i=1}^{N} \frac{y_i}{N} \qquad (21)$$

Después se crea, una línea vertical desde la parte superior del objeto hasta la parte inferior, y una línea horizontal desde la parte izquierda del objeto hasta la parte derecha, y en la misma línea del centroide, pero dibujándola en la parte inferior de la silueta, ese punto es el VGP, tal como lo muestra la Figura 15.

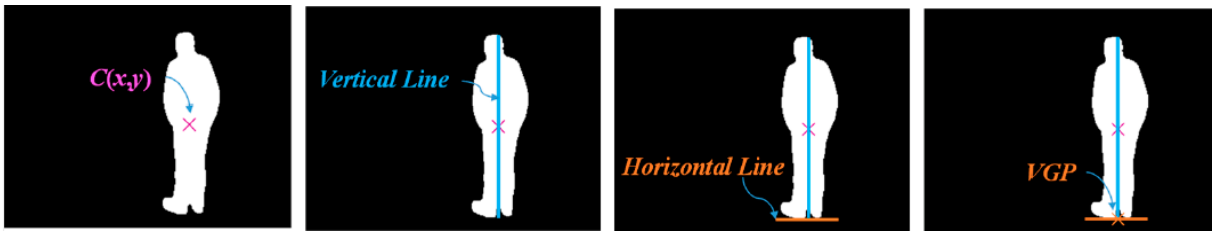

Figura 15. Proceso Calculo VGP.

Fuente: Tomado de [123]

Otras características importantes son la distancia desde el centroide del objeto y VGP, además, del área de la silueta humana y su relación de aspecto (relación entre el ancho y largo de la silueta), como se observa en la Figura16.

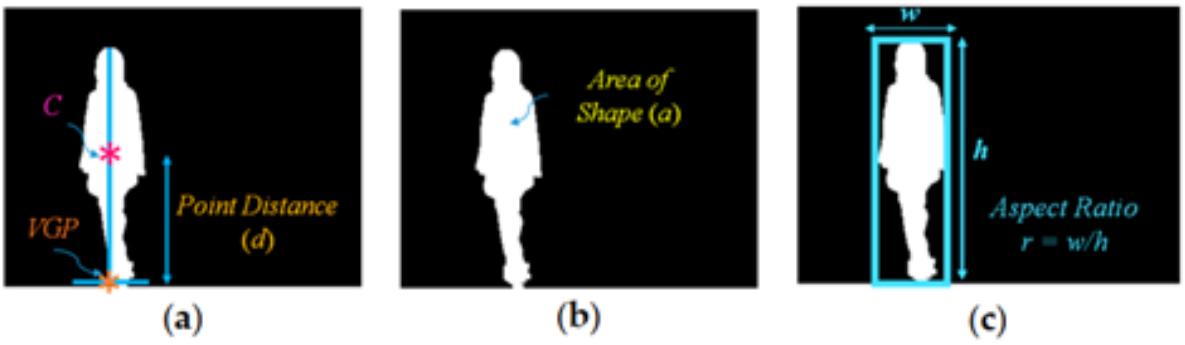

Figura 16. (a) distancia, (b) área silueta, (c) relación de aspecto.

Fuente: Tomado de [123]

A continuación, se realiza un análisis de eventos anormales y su clasificación por medio de SVM, para predecir una caída, es importante mencionar que las características extraídas tienen puntos umbrales a partir de los cuales se considera una acción de caída para las diferentes bases de datos.

3.1.2 Algoritmo que Incorpora CNN (YOLO)

El segundo algoritmo de reconocimiento de caídas escogido está basado en la investigación propuesta por Feng & Gao et.al. [124]. Sus autores también mencionan el hecho de que son pocas las investigaciones sobre detección de caídas en escenarios complejos y proponen su investigación como una solución robusta ante escenarios poco controlados.

[124] utiliza YOLO (You Only Look Once) V3 para detectar a las personas en los videos y es monitoreado por un módulo de seguimiento Deep Sort tracking, posteriormente se extraen características de los frames por medio de una red neuronal convolucional pre-entrenada VGG-16 (ver Figura 17).

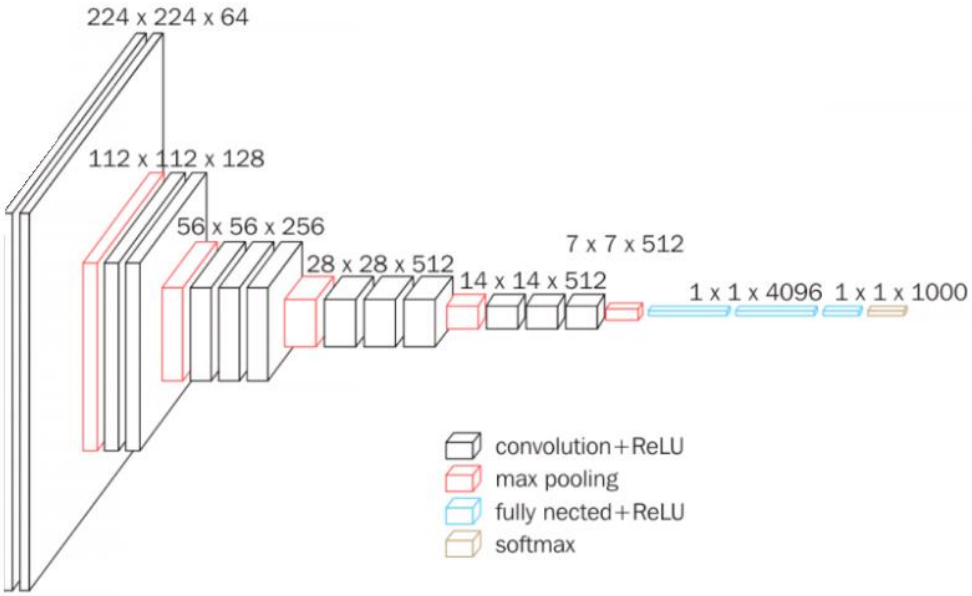

Figura 17. Estructura CNN VGG_16.

Fuente: Tomado de [124]

Finalmente se detecta los eventos de caída por LSTM guiado por atención, la Figura 18 describe el proceso mencionado. Para conocer los parámetros específicos de implementación se invita a revisar la publicación de los autores [124]. También es importante mencionar que la propuesta de los autores detecta a los peatones únicamente cuando existen eventos de caída, en el presente trabajo se detecta eventos de caída y eventos de no caída.

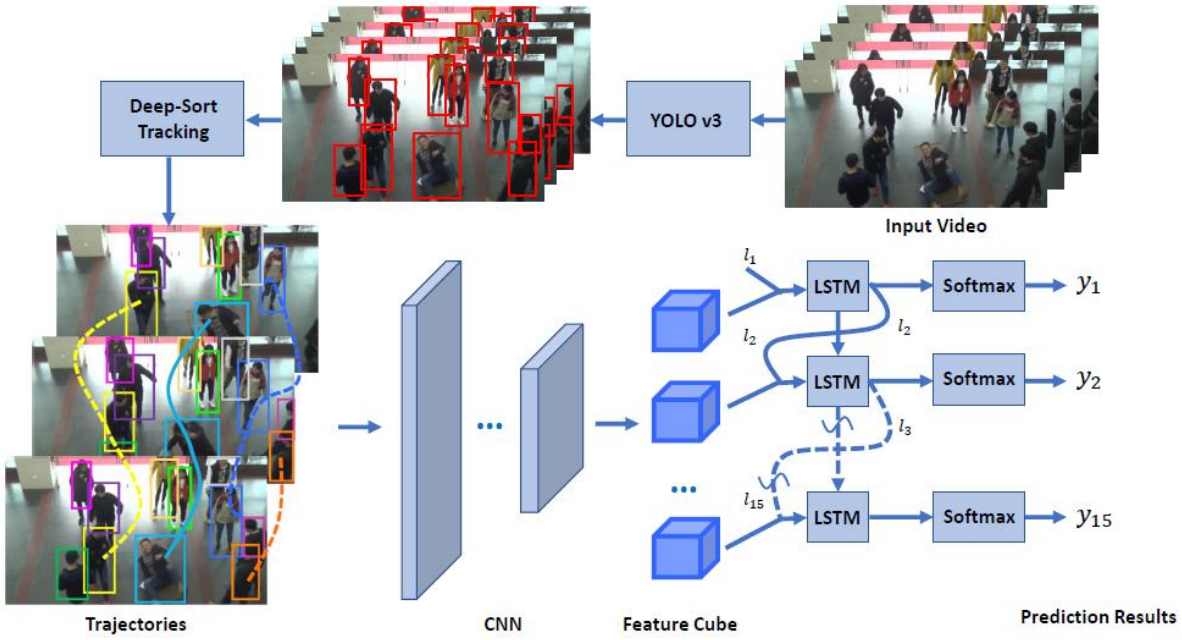

Figura 18. Arquitectura del algoritmo.

Fuente: Tomado de [124]

3.1.3 Algoritmo que Incorpora OPENPOSE

Finalmente, como tercer algoritmo de implementación se escoge una novedosa investigación presentada en [125], cuya predicción de caídas se basa en la extracción de los mapas de los esqueletos humanos de imágenes en 2D usando OPENPOSE.

Primero, partiendo de los videos de entrada y usando OpenPose, los autores crean un conjunto de datos, con los modelos del esqueleto humano para cada cuadro (Ver Figura 19).

Figura 19. Conjunto de datos obtenido de OpenPose.

Fuente: Tomado de [125]

Posteriormente realizan un pre-procesamiento de datos. Reduciendo todas las imágenes a un tamaño de 299×299 y convirtiendo las imágenes en formato TF Record, porque leer datos de TF Record, es mucho más rápido que leer datos de imágenes directamente. Los autores utilizan el 80 % de los datos como conjunto de entrenamiento y el 20 % como conjunto de prueba.

El entrenamiento del modelo se hace utilizando transferencia de aprendizaje por medio de una red neuronal convolucional pre-entrenada Inception_ResNet_V2 model (ver Figura 20), la cual es la encargada de aprender y predecir las posibles caídas humanas, la estructura general del algoritmo se muestra en la Figura 21.

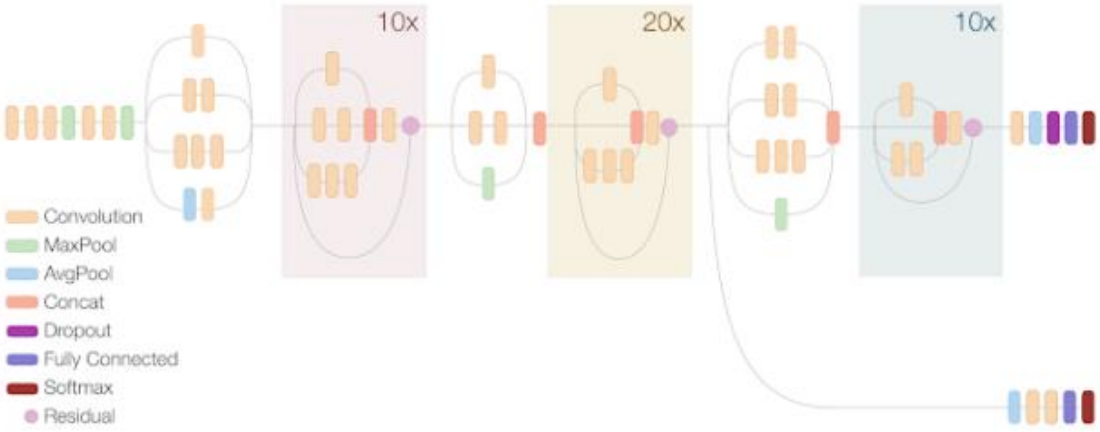

Figura 20. Estructura CNN Inception_ResNet_V2.

Fuente: Tomado de [142]

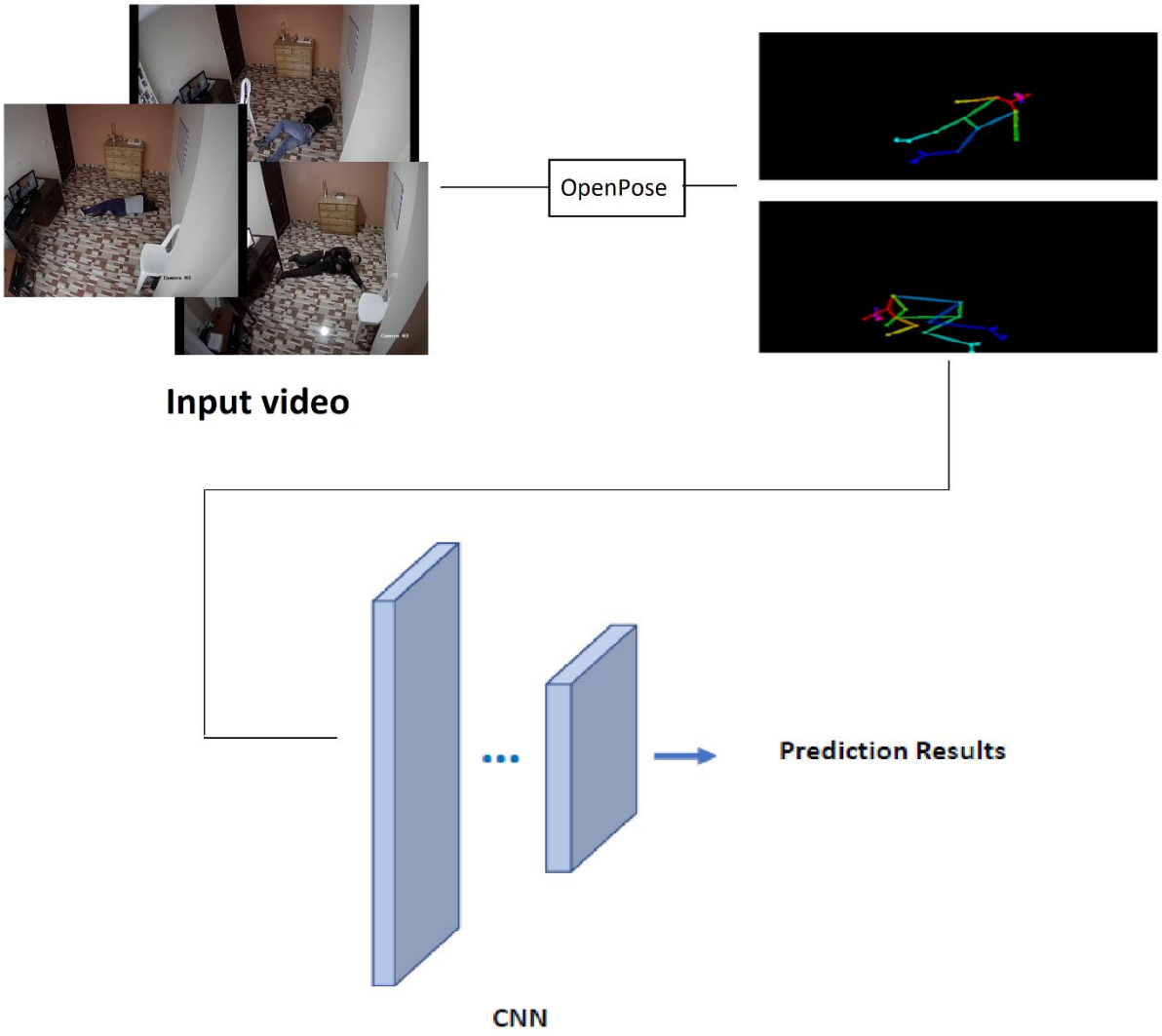

Figura 21. Arquitectura del algoritmo.

3.2 Bases de datos disponibles para el reconocimiento de caídas humanas

Las aplicaciones de reconocimiento de caídas mediante visión por ordenador han obtenido resultados satisfactorios. Sin embargo, los conjuntos de datos utilizados tienen entornos restringidos y las caídas son simuladas, lo que resulta controvertido. Los algoritmos entrenados con bases de datos muy controladas no obtienen buenos resultados en la predicción de caídas reales. Por esta razón, se propone CAUCAFall [49], este conjunto de datos puede utilizarse para analizar el progreso real del reconocimiento de caídas humanas evaluando el comportamiento de los algoritmos de reconocimiento de caídas en un entorno no controlado que simula un entorno realista. En esta sección se describe las bases de datos utilizadas en esta investigación para la comparación de los diferentes algoritmos, basándose en la información obtenida en Tabla 3.

3.2.1 UR Fall Detection

Esta base de datos [42], contiene 70 secuencias de video, entre las cuales 30 son de caídas y 40 de diferentes actividades, además de contener señales análogas obtenidas por medio de un sensor acelerómetro, los eventos de caídas se registran con 2 cámaras Kinect, es decir las caídas cuentan con dos ángulos de visión (ver Figura 22) y las diferentes actividades únicamente fueron registradas con una cámara.

La velocidad de grabación es de 30 fotogramas por segundo y participaron 5 personas, las cuales simulan caídas desde una posición de pie y desde estar sentados en una silla, las actividades normales que comprende esta base de datos son: caminar, sentarse, agacharse y acostarse.

Por otra parte, el entorno de grabación es controlado, con iluminación artificial y las caídas ocurren a la misma distancia ante la cámara.

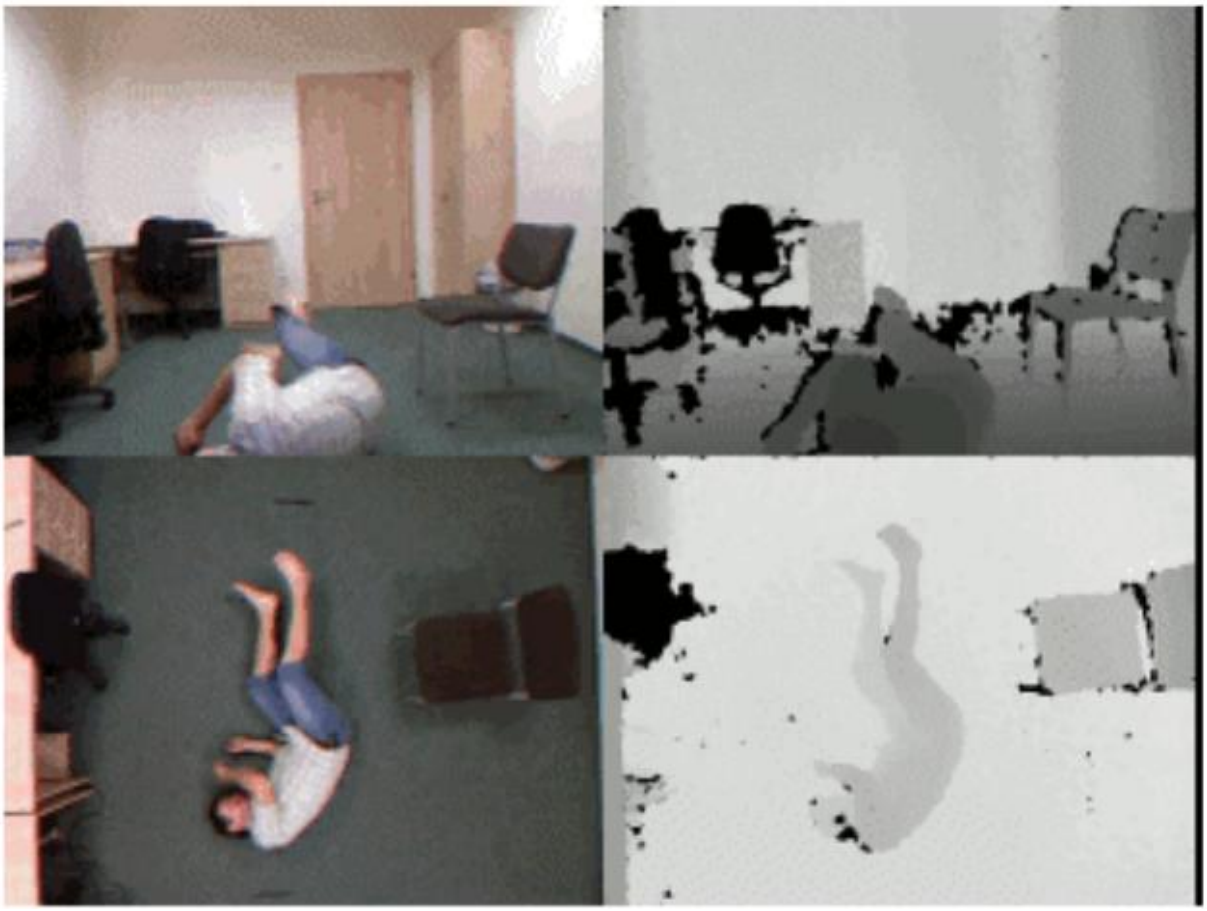

Figura 22. Base de datos UR Fall Detection.

Fuente: Tomado de [42]

3.2.2 Multicam Fall Detection

MCF [46], es un conjunto de videos obtenidos con 8 cámaras gadspot gs-4600 a una velocidad de 30 fps y resolución de 720 x 480 pixeles. En total el dataset contiene 24 acciones grabadas desde 8 ángulos diferentes, 22 acciones de caídas y 2 acciones de actividades diarias. Por su parte, los escenarios tienen en cuenta la incorporación de objetos típicamente encontrados en las viviendas y entornos hogareños como sillas, sofás, mesas, etc, sin embargo, el escenario no contiene cambios en cuanto a textura e iluminación, e incluso gran parte de las caídas ocurren sobre una colchoneta que sobresale en la textura del suelo (ver Figura 23).

Figura 23. Base de datos Multicam Fall Dataset.

Fuente: Tomado de [46]

3.2.3 LE2I

La presente base de datos [43] hace uso de una única cámara a una velocidad de 25 cuadros por segundo en resolución de 320 x 240 pixeles. Los videos fueron grabados en 4 entornos "office", "Caffe room", "home" y "lectura room" (ver Figura 24), obteniendo un total de 191 videos con secuencias que contienen iluminación variable y dificultades típicas como oclusiones.

Figura 24. Base de datos LE2I.

Fuente: Tomado de [43]

Los participantes realizaron actividades diarias como: caminar, sentarse, agacharse, recoger un objeto, acostarse y simularon caídas como: caer de una silla, caer al caminar, al tener un tropiezo. Sin embargo, a la fecha de febrero de 2022 los autores de la base de datos no tenían habilitado el acceso público al total de la base de datos.

3.2.4 UP-Fall

El conjunto de datos de detección de caídas UP-Fall [45] comprende conjuntos de 17 individuos jóvenes sanos sin ningún impedimento que realizaron 5 actividades diarias simples y 5 tipos diferentes de caídas, con tres intentos cada uno. La base de datos es multimodal es decir contiene información de sensores portátiles, sensores ambientales y dispositivos de visión (ver Figura 25). El conjunto de datos consolidado (812 GB), así como el conjunto de datos de características (171 GB) están disponibles públicamente.

El entorno de UP-Fall es sumamente controlado, no contiene cambios de iluminación, no contiene cambios de ángulos de caída de los participantes, ni distancia hasta la cámara, he incluso muchos de sus participantes usan chalecos que sobresalen en la escena visual.

Figura 25. Base de datos UP-Fall.

Fuente: Tomado de [45]

3.3 CAUCAFall

Los diferentes conjuntos de datos mostrados en esta investigación [42,43,44,45,46,143,144,145,146], presentan características que limitan el verdadero avance en el reconocimiento de caídas humanas, ya que son entornos sumamente controlados, en donde no existe información precisa de diferentes distancias desde la cámara hasta la caída, ni información de la variedad en los ángulos de caída, e incluso algunas no tienen cambios de iluminación, ni oclusiones. CAUCAFall se desarrolló con una sola cámara en un entorno doméstico, este entorno es intencionadamente realista e incluye características de los entornos no controlados, como oclusiones, cambios de iluminación (natural, artificial y nocturna), ropa diferente de los participantes, movimiento en el fondo, diferentes texturas en el suelo y en la habitación, y una variedad en los ángulos de caída y diferentes distancias de la cámara a la caída (ver Figura 26). Además, CAUCAFall es el único conjunto de datos que contiene etiquetas de caída y de no caída para ser utilizado en detectores de YOLO como método novedoso de detección y reconocimiento, además, es la única base de datos que detalla las distancias de la cámara a la caída humana y los ángulos de caída con referencia a la posición de la cámara, y también detalla los lux de iluminación de los diferentes entornos. CAUCAFall fue publicada en la revista Data In Brief [49]. La Tabla 5 compara las bases de datos más populares entre la comunidad científica.

Figura 26. CAUCAFall, ejemplo de entornos.

3.3.1 Protocolo de recopilación de datos

El proceso de recogida de datos se realizó en un entorno doméstico no controlado. Un kinesiólogo profesional instruyó a los participantes sobre la forma correcta de caerse, y se simularon las caídas más comunes en las personas mayores. Los elementos de protección seleccionados son coderas y rodilleras con gran capacidad de absorción de impactos y sin restricción de movimientos. Cada participante realizó 10 actividades, cinco actividades de la vida diaria (AVD) y cinco simulaciones de caídas. Los siguientes pasos detallan las fases esenciales para crear el conjunto de datos:

1. Se realizó una revisión bibliográfica para conocer los conjuntos de datos existentes y sus características, lo que permitió a los autores identificar sus deficiencias;

2. Se creó un entorno realista no controlado, incorporando elementos de distracción, oclusiones, condiciones ambientales e iluminación cambiante;

3. Se eligió una población variada de participantes en términos de edad, género, peso, altura y diferentes piernas dominantes para realizar las actividades y la simulación de caída;

4. Un ingeniero identificó los posibles ángulos de caída de los participantes y determinó diferentes distancias de caída frente a la cámara para garantizar un conjunto de datos variado;

5. Los datos se registraron y almacenaron para su posterior procesamiento.

3.3.2 Descripción de los datos

CAUCAFall [49] es un conjunto de datos que se adquirió utilizando una cámara RGB, el conjunto de datos contiene cinco tipos de caídas y cinco tipos de actividades de la vida diaria (AVD), en el cual se considera individuos de diferentes edades, pesos, alturas y piernas dominantes. Por otra parte, es la única base de datos que contiene detalles de la iluminación (lux de los escenarios), las distancias de la caída humana hasta la cámara y los ángulos de las diferentes caídas con referencia a la cámara. El conjunto de datos es también el único que contiene etiquetas para cada imagen. Los fotogramas que incluían caídas humanas registradas se etiquetaron como "caída", y las actividades AVD se marcaron como "no caída", útiles para el entrenamiento y el funcionamiento de los detectores de YOLO, además, la resolución (1920 x 1080 píxeles) y el ángulo en el que se grabó CAUCAFall permiten un alto rendimiento en los algoritmos modernos que detectan el mapa óseo humano y pueden utilizarse para el reconocimiento de caídas humanas, como OpenPose.

En total participaron diez sujetos (ver Tabla 6) quienes simularon las actividades de la vida diaria (AVD) y caídas hacia delante, caídas hacia atrás, caídas laterales hacia la izquierda, caídas laterales hacia la derecha y caídas que surgían al sentarse. Las AVD de los participantes eran caminar, saltar, recoger un objeto, sentarse y arrodillarse. Los fotogramas que registraron caídas humanas se etiquetaron como "caída", y las actividades AVD se etiquetaron como "no caída". Los fotogramas se etiquetaron como "caída" sólo cuando el cuerpo humano estaba en el suelo debido a una caída. En la simulación de caída, la posición inicial de los participantes es de pie, a excepción de "Caída sentada" cuya posición inicial es estar sentado.

Tabla 5. Comparación de conjuntos de datos para el reconocimiento de caídas humanas.

Base de datos	año	Cámara	Condiciones iluminación	Oclusión	Variedad en ángulos	Diferentes distancias	Formato archivos	Etiquetas YOLO	Lux	Especifica ángulos	Especifica distancias	Disponible (Ago,2022)
Multiple cameras fall dataset [46]	2010	RGB	artificial	✓	-	-	.avi	-	-	-	-	✓
Le2i [43]	2012	RGB	natural, artificial	✓	✓	-	.avi	-	-	-	-	-
SDUFall [143]	2014	Kinect	natural, artificial	-	-	-	Depth videos .avi	-	-	-	-	-
EDF&OCCU [144]	2014	Kinect	artificial	✓	-	✓	.txt	-	-	-	-	-
UR Fall Detection [42]	2014	Kinect	artificial	-	-	✓	.avi .csv	-	-	-	-	✓
FUKinect-Fall [145]	2016	Kinect	-	-	✓	✓	Depth videos .csv	-	-	-	-	✓
Fall Detection Dataset [146]	2017	RGB Kinect	natural, artificial	-	✓	-	.png .csv	-	-	-	-	✓
UPFall [45]	2019	RGB	natural, artificial	-	✓	-	.png .csv	-	-	-	-	✓
CAUCAFall [49] (Ours)	2022	RGB	natural, artificial, no light	✓	✓	✓	.jpeg .txt .avi	✓	✓	✓	✓	✓

Tabla 6. Características de los participantes.

Sujeto	Genero	Edad	Peso (Kg)	Estatura (Metros)	Condiciones de salud	Pierna dominante	Outfit
1	Female	27	56	1.65	Healthy	Right	Gray jacket, blue pants, black shoes, hair tied.
2	Male	34	70	1.73	Healthy	Left	Red jersey, blue pants, white shoes.
3	Female	31	58	1.60	Healthy	Left	Brown jacket, gray pants, blue shoes, loose hair.
4	Male	38	75	1.68	Healthy	Right	Black jacket, blue pants, gray shoes, cap.
5	Male	40	67	1.70	Healthy	Right	Black jacket, brown pants, black shoes.
6	Male	33	77	1.65	Healthy	Right	Black jacket, white pants, brown shoes.
7	Female	23	54	1.59	Healthy	Right	Gray jersey, black pants, blue shoes, hair tied.
8	Female	25	59	1.63	Healthy	Right	Blue jersey, gray pants, brown shoes, hair tied.
9	Male	37	79	1.74	Healthy	Left	Yellow jersey, brown pants, brown shoes.
10	Female	28	61	1.62	healthy	Right	Green shirt, purple pants, black shoes, loose hair.

Fuente: El autor

Por otra parte, la Figura 27 muestra un mapa con las dimensiones del escenario en el que se simularon las caídas humanas. En el escenario hay una ventana por la que entra luz natural. La cámara de grabación está situada a una altura de 2,15 m.

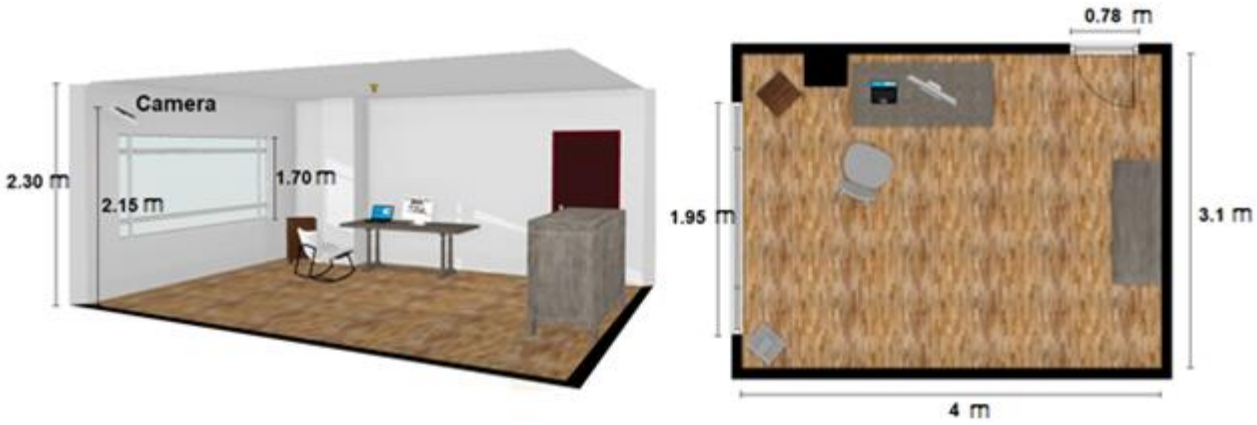

Figura 27. Dimensiones del escenario (en metros).

Los datos están organizados en 10 directorios principales correspondientes a los sujetos. Cada directorio contiene 10 carpetas con las diferentes actividades realizadas. Cada carpeta incluye un vídeo de la acción en formato .avi, imágenes de la acción en formato .png, y las etiquetas de segmentación de cada fotograma en formato .txt.

En la Figura 28 se detallan las carpetas de cada sujeto y las diferentes actividades. Se incluye un ejemplo para el sujeto 1, para la actividad "Caída hacia atrás". Esta actividad tiene el vídeo de la acción en formato .avi y cada una de las imágenes de la actividad en formato .png. Cada imagen tiene el mismo nombre base, y para cada imagen, hay una etiqueta respectiva. Por último, el archivo "classes.txt" especifica el nombre de las etiquetas utilizadas en las imágenes.

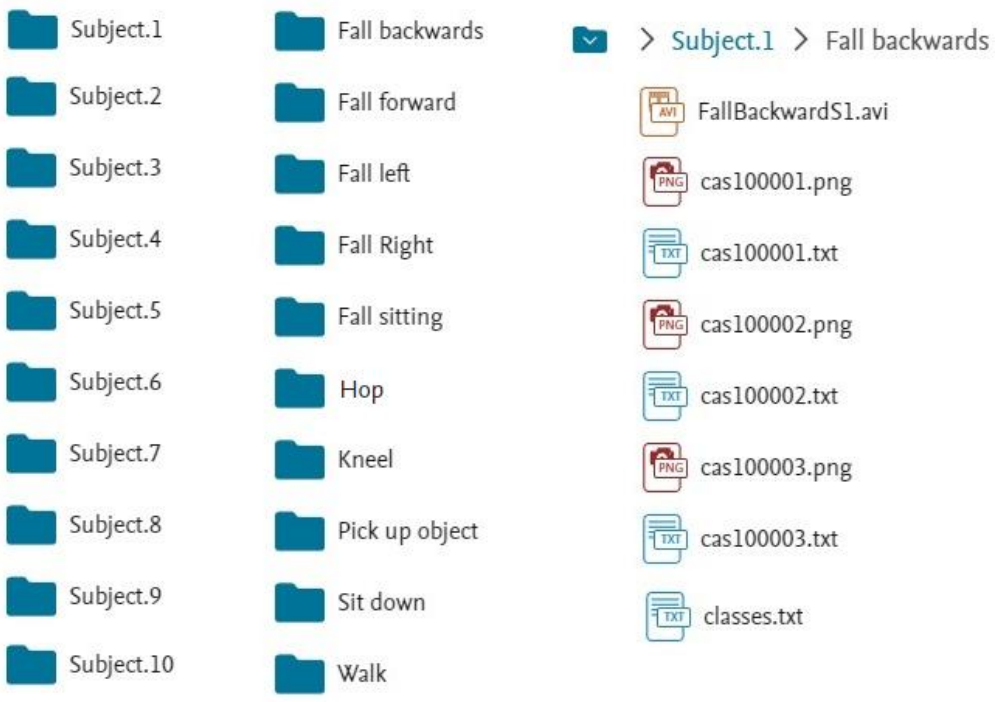

Figura 28. Carpetas para sujeto, con las diferentes actividades y archivos.

El contenido de los diferentes archivos .txt se muestra en la Figura 29. Los archivos contienen la información sobre el cuadro que delimita la silueta humana y el primer dígito (0 o 1) identifica la etiqueta de la acción que se realiza. El nombre de la etiqueta se define en el archivo "classes.txt": El 0 corresponde a "nofall" y el 1 a "fall".

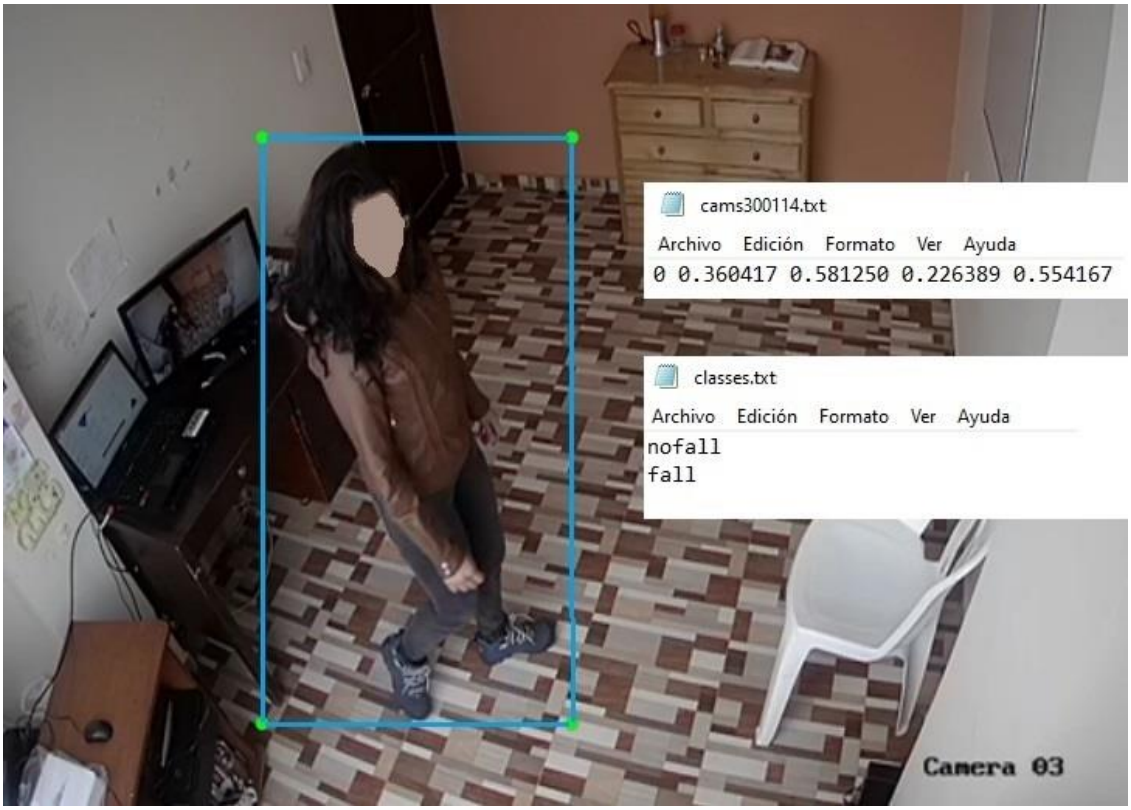

Figura 29. Contenido de los diferentes archivos .txt.

Los diferentes investigadores podrían cambiar las etiquetas del archivo "classes.txt" y utilizar los datos para realizar el reconocimiento de actividades humanas en lugar del reconocimiento de caídas.

CAUCAFall [49] también incorpora las diferentes distancias de las caídas humanas desde el centroide de la silueta humana hasta la cámara (ver Figura 30) y los ángulos de las caídas con referencia a la cámara (ver Figura 31). La Tablas (7,8,9,10,11,12,13,14,15,16) detallan distancias, ángulos, número de fotogramas, oclusiones y las condiciones de iluminación de los distintos escenarios, para cada uno de los sujetos, respectivamente.

Figura 30. Distancia Cámara - Caída.

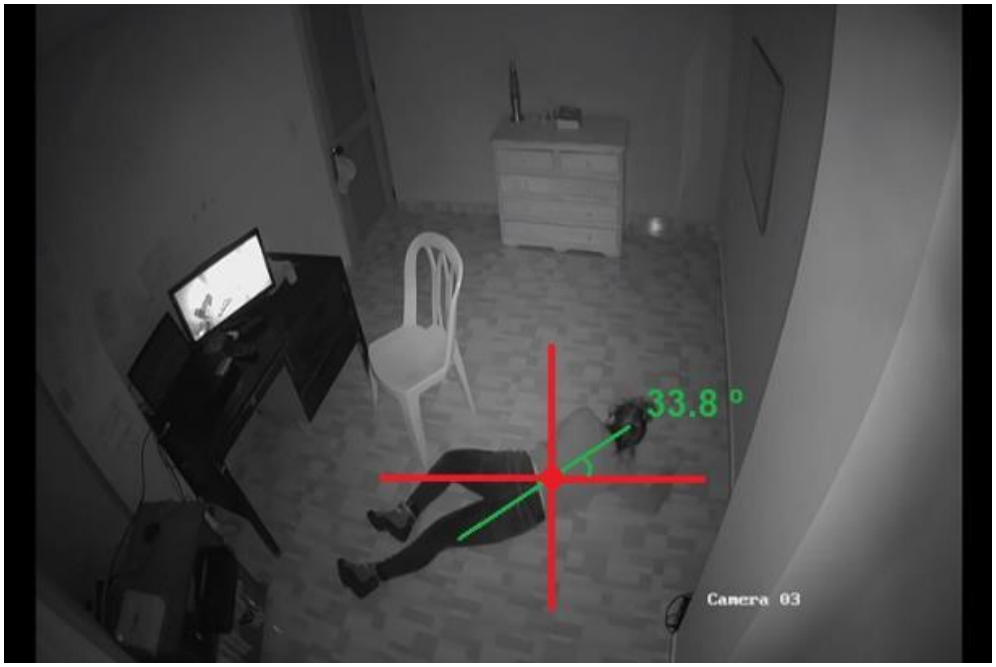

Figura 31. Ángulos de caída, referente a la cámara.

Tabla 7. Características sujeto1.

Sujeto1				
El escenario contiene luz natural, aproximadamente 210 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	183	-	-	-
Arrodillarse	220	-	-	-
Recoger Objeto	118	-	-	-
Sentarse	189	-	-	-
Caminar	243	-	-	-
Caída atrás	126	3.05	21.3º	-
Caída adelante	191	2.69	271.4º	-
Caída izquierda	87	2.85	263.7º	-
Caída derecha	115	3.41	27.3º	-
Caída sentado	183	2.6	7.3º	-

Tabla 8. Características sujeto2.

Sujeto2				
El escenario contiene luz natural, aproximadamente 203 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	172	-	-	-
Arrodillarse	131	-	-	-
Recoger Objeto	167	-	-	-
Sentarse	113	-	-	-
Caminar	245	-	-	-
Caída atrás	121	3.23	75.7º	-
Caída adelante	107	2.62	291.4º	-
Caída izquierda	110	2.94	13.8º	-
Caída derecha	147	2.97	21.3º	-
Caída sentado	158	2.63	83.8º	-

Tabla 9. Características sujeto3.

Sujeto3				
El escenario contiene luz natural, aproximadamente 218 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	163	-	-	-
Arrodillarse	214	-	-	-
Recoger Objeto	187	-	-	-
Sentarse	262	-	-	-

Caminar	177	-	-	-
Caída atrás	167	2.59	5.7º	-
Caída adelante	154	2.39	272.7º	-
Caída izquierda	172	2.42	273.8º	-
Caída derecha	221	2.81	47.2º	-
Caída sentado	203	3.34	63.8º	✓

Tabla 10. Características sujeto4.

Sujeto4				
El escenario contiene luz natural, aproximadamente 221 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	145	-	-	-
Arrodillarse	223	-	-	-
Recoger Objeto	180	-	-	-
Sentarse	200	-	-	-
Caminar	241	-	-	-
Caída atrás	193	2.67	283.5º	-
Caída adelante	169	2.58	275.7º	-
Caída izquierda	163	3.43	47.8º	-
Caída derecha	151	3.23	48.2º	-
Caída sentado	199	3.32	117.5º	-

Tabla 11. Características sujeto5.

Sujeto5				
El escenario contiene luz artificial, aproximadamente 127 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	123	-	-	-
Arrodillarse	209	-	-	-
Recoger Objeto	272	-	-	-
Sentarse	261	-	-	-
Caminar	227	-	-	-
Caída atrás	215	3.18	25.9º	-
Caída adelante	201	2.3	273.7º	-
Caída izquierda	248	2.96	350.8º	-
Caída derecha	220	3.08	214.2º	-
Caída sentado	230	2.45	346.1º	-

Tabla 12. Características sujeto6.

Sujeto6				
El escenario no contiene luz, 0 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	163	-	-	-
Arrodillarse	190	-	-	-
Recoger Objeto	203	-	-	-
Sentarse	190	-	-	-
Caminar	267	-	-	-
Caída atrás	257	2.75	47.9º	-
Caída adelante	203	2.42	283.7º	-
Caída izquierda	230	2.65	15.5º	-
Caída derecha	230	3.09	213.4º	-
Caída sentado	249	2.43	352.8º	-

Tabla 13. Características sujeto7.

Sujeto7				
El escenario contiene luz artificial, aproximadamente 125 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	199	-	-	-
Arrodillarse	203	-	-	-
Recoger Objeto	154	-	-	-
Sentarse	194	-	-	-
Caminar	194	-	-	-
Caída atrás	253	2.63	5.9º	-
Caída adelante	163	2.57	277.7º	-
Caída izquierda	213	2.98	5.8º	-
Caída derecha	167	2.53	333.9º	✓
Caída sentado	208	3.42	14.4º	✓

Tabla 14. Características sujeto8.

Sujeto8				
El escenario no contiene luz, 0 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	167	-	-	-
Arrodillarse	257	-	-	-
Recoger Objeto	267	-	-	-
Sentarse	213	-	-	-
Caminar	217	-	-	-

Caída atrás	213	2.84	357.7º	-
Caída adelante	123	2.47	285.3º	-
Caída izquierda	203	2.97	6.9º	-
Caída derecha	230	3.4	163.4º	-
Caída sentado	235	2.85	34.2º	-

Tabla 15. Características sujeto9.

Sujeto9				
El escenario contiene luz artificial, aproximadamente 128 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	213	-	-	-
Arrodillarse	217	-	-	-
Recoger Objeto	221	-	-	-
Sentarse	239	-	-	-
Caminar	235	-	-	-
Caída atrás	253	2.91	5.3º	-
Caída adelante	149	2.52	279.4º	-
Caída izquierda	208	2.59	325.6º	-
Caída derecha	190	2.72	194.3º	-
Caída sentado	217	2.53	349.5º	-

Tabla 16. Características sujeto10.

Sujeto10				
El escenario contiene luz artificial, aproximadamente 130 lux.				
Actividad	Frames	Distancia Cámara – Caída (metros)	Angulo de caída	Oclusiones
Saltar	249	-	-	-
Arrodillarse	262	-	-	-
Recoger Objeto	293	-	-	-
Sentarse	235	-	-	-
Caminar	226	-	-	-
Caída atrás	275	2.82	6.8º	-
Caída adelante	244	2.27	293.5º	-
Caída izquierda	213	3.12	177.7º	✓
Caída derecha	203	2.95	347.7º	✓
Caída sentado	289	3.34	19.6º	✓

La Tabla 17 resume la cantidad de videos e imágenes que contiene CAUCAFall para las diferentes actividades en las diversas condiciones de iluminación (la información está en videos-frames), mientras

que la Tabla 18 muestra la cantidad de videos e imágenes de cada una de las diferentes condiciones de caída que propone la base de datos.

Tabla 17. Cantidad de información por actividades y condiciones de iluminación, en (videos - frames).

Actividad	Luz Natural	Luz Artificial	Sin Luz	Total
Saltar	4 - 663	4 - 784	2 - 330	10 – 1.777
Arrodillarse	4 - 788	4 - 891	2 - 447	10 – 2.126
Recoger Objeto	4 - 652	4 - 940	2 - 470	10 – 2.062
Sentarse	4 - 764	4 - 929	2 - 403	10 – 2.096
Caminar	4 - 906	4 - 882	2 - 484	10 – 2.272
Caída atrás	4 - 607	4 - 996	2 - 470	10 – 2.073
Caída adelante	4 - 621	4 - 757	2 - 326	10 – 1.704
Caída izquierda	4 - 532	4 - 882	2 - 433	10 – 1.847
Caída derecha	4 - 634	4 - 780	2 - 460	10 – 1.874
Caída sentado	4 - 743	4 - 944	2 - 484	10 – 2.171
Total	40 – 6.910	40 – 8.785	20 – 4.307	100 – 20.002

Tabla 18. Cantidad de información por condiciones de caída.

Condiciones de Caída	Videos	Imágenes
Oclusiones	6	1.283
Distancia cámara-caída		
2.27 m – 2.66 m	20	3.747
2.67 m – 3.04 m	16	3.239
3.05 m – 3.43 m	14	2.683
Ángulos de Caída		
0º - 90º	23	4.496
91º - 180º	4	832
181º - 270º	10	1.736
271º - 360º	13	2.605

3.3.3 Materiales

El sistema óptico utilizado para capturar los vídeos de las acciones humanas está compuesto por una cámara IR HIKVISION [147], que se fijó en la esquina superior de la pared en los diferentes escenarios. Este sistema cubre un amplio campo de visión para monitorizar la actividad del usuario y se conectó a un DVR HIKVISION [148] con un disco duro de 1 TB incorporado para el almacenamiento y procesamiento de vídeo.

El DVR dispone de modos continuos, manuales y de detección de movimiento, por lo que la grabación comienza cuando el individuo entra en la escena. La cámara captura vídeo a una velocidad de 25 fps y una resolución de 1080 × 960 píxeles y admite cambios de iluminación (es decir, luz natural, poca luz o sin luz). El sensor de infrarrojos graba en color RGB durante la luz natural, mientras que en la oscuridad o sin luz, el sensor de infrarrojos proporciona haces de luz para grabar imágenes binarias.

3.3.4 Etiquetas

En general, los detectores YOLO se entrenan y trabajan con las etiquetas del conjunto de datos COCO [130], que contiene 80 clases, pero no incluye las caídas. CAUCAFall incluye etiquetas para que los detectores de YOLO también sean capaces de detectar caídas humanas, para lo cual los autores etiquetaron manualmente cada fotograma de cada actividad realizada, como "caída" o "no caída". Se etiquetaron un total de 20.002 fotogramas: 13.581 actividades AVD y 6.421 caídas. También se verificó la dimensionalidad de la imagen y el formato correcto, de modo que las imágenes contenían el tamaño y las dimensiones óptimas para enfatizar el análisis en el área de interés. Este paso ayuda significativamente a las técnicas de visión por ordenador y a las redes neuronales convolucionales, reduciendo el coste computacional.

3.4 Implementación de Algoritmos

En esta sección se explica la implementación de los diferentes algoritmos seleccionados y explicados en secciones anteriores, dando a conocer equipo, lenguaje, entornos y librerías utilizadas para su desarrollo.

Las arquitecturas se implementaron en lenguaje PYTHON, utilizando una computadora portátil ACER con CPU AMD Ryzen 5 3500U- 2,10 GHz y 8GB de RAM, que además cuenta con tarjeta gráfica Radeon Vega Mobile Gfx. Además, para entrenar los diferentes modelos implementados en esta investigación se utilizó la herramienta Google Colaboratory.

3.4.1 Algoritmo basado en extracción de características.

Para realizar la implementación del algoritmo que incorpora extracción de características ([123]) se utiliza el entorno de programación PyCharm Community Edition para extraer las diferentes características. Para lo cual se utilizó librerías como OpenCV, numpy, time, os y csv para el almacenamiento de la información.

Como se explicó anteriormente, uno de los pasos más importantes en la extracción de características es la detección de la silueta humana, un primer paso se realiza por medio del algoritmo MoG, seguido de la binarización de la imagen y operaciones morfológicas, para obtener la sustracción de fondo. Posteriormente se procede a la detección de la silueta humana, para lo cual se debe buscar los contornos que se encuentren en la imagen, en este caso el contorno que nos interesa es el de la silueta humana.

De esta forma se procede a realizar la extracción de características, calculando el área y el centroide de la silueta humana, además de delimitar su alto y ancho, para finalmente encontrar la distancia del centroide al suelo virtual y la relación de aspecto.

Para el almacenamiento de la información se utiliza un archivo .csv, en el cual se almacenan toda la información de los videos y la extracción de características de cada frame. Esta información es utilizada para el entrenamiento del modelo por medio de SVM.

Para la búsqueda de los paremetros SVM y su evaluación se hizo uso del entorno Google Colaboratory y de librerías como “matplotlib”, “pandas”, “pylab”, “numpy” y “sklearn”. Los datos se dividieron en entrenamiento, validación y prueba, dependiendo del experimento a realizar, y se entrenó el modelo usando “sklearn.svm” junto con “GridSearchCV”, para realizar una búsqueda por validación cruzada, además se utiliza una función Radial (RF) encontrando los mejores parámetros de gamma (g) y coste

(c), variables que permiten la mejor separación de las clases, para cada una de las bases de datos. La Figura 32 resume el diagrama de flujo.

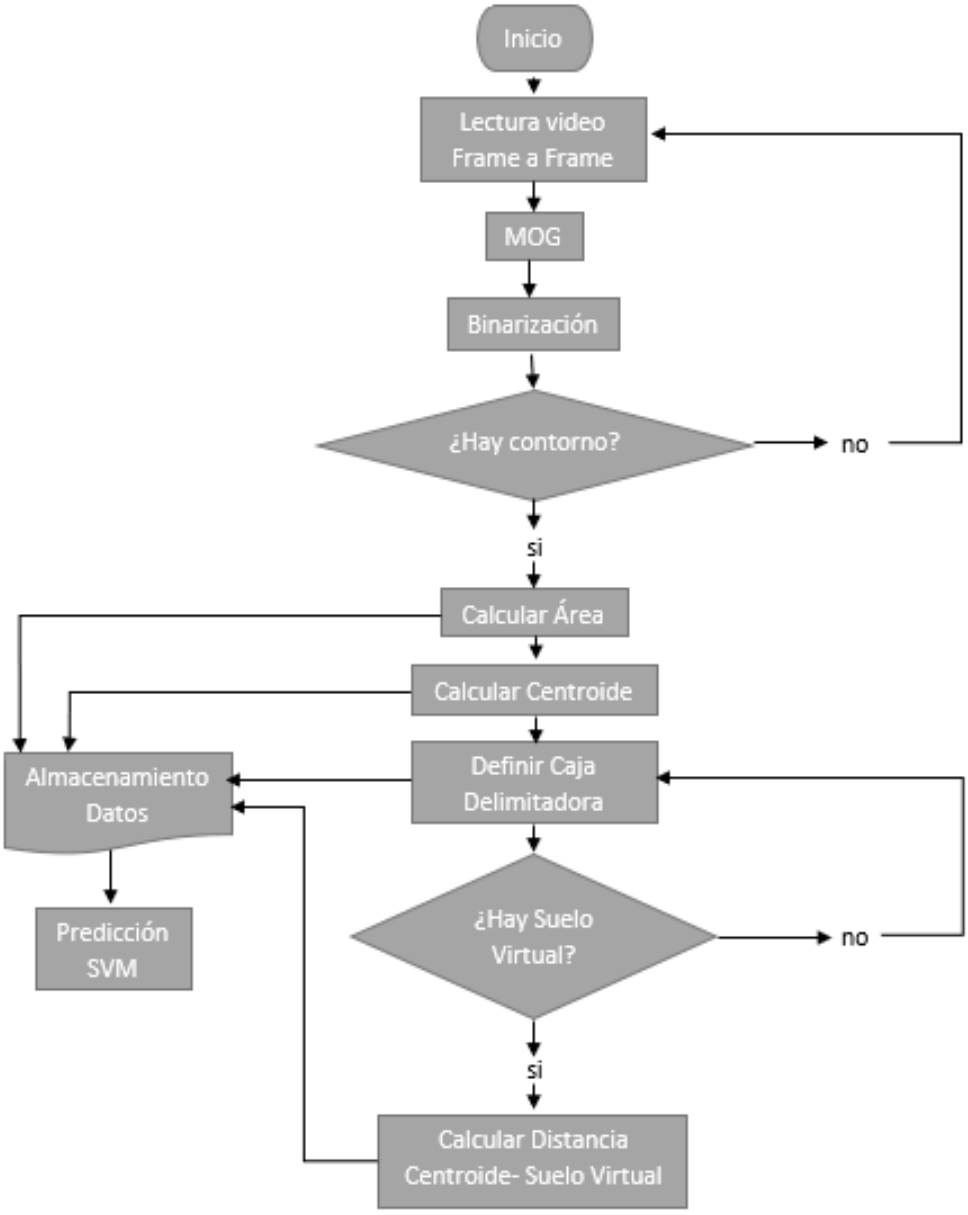

Figura 32. Diagrama de flujo del Algoritmo [123].

3.4.2 Algoritmo basado en YOLO/CNN

Para realizar la implementación del algoritmo que incorpora YOLO y redes neuronales convolucionales ([124]) se utiliza el entorno de programación Google Colab, utilizando librerías como “google.colab” para trabajar con información de las bases de datos en Drive, “tensorflow”, “keras”, “matplotlib”, “pandas” y “sklearn”.

En la primera fase del algoritmo se requiere el uso del detector YOLO, por lo que se requiere configurar parámetros de dicha red neuronal, para lo cual se descargan parámetros (darknet) desde el repositorio oficial de YOLO [149], se configuran los parámetros para que utilizando el pre-entrenamiento de YOLO permita detectar el objeto deseado, en este caso al individuo. De igual forma también se configuran parámetros para que el objeto detectado tenga un seguimiento continuo, activando Deep Sort.

Con el sujeto detectado y en continuo seguimiento, esta información debe ingresar a una red neuronal convolucional pre-entrenada cuyo modelo definido por los autores es la red VGG-16. El procedimiento es el siguiente (ver Figura 17):

- El primer paso es realizar un pre-procesamiento a los datos que ingresan a la CNN, para lo cual se utiliza "ImageDataGenerator" y "flow_from_directory", comandos que permiten normalizar los valores de pixeles de las imágenes, asignar el tamaño que requiere la red VGG-16 (224x224px), además, de realizar un aumento de datos.
- Después se debe cargar la red CNN pre-entrenada VGG-16, para lo cual utilizamos librerías como "tensorflow.keras.applications import vgg16", se configura la red y se modifican sus capas de salida, ya que VGG-16 por defecto puede clasificar 1000 clases, pero en la presente investigación solo se va a predecir "caída" y "no caída", por lo que se modifica su última capa densa a dos neuronas con activación softmax.
- Esta información se envía a LSTM, una red neuronal recurrente, para finalmente realizar la clasificación. La Figura 33 resume el proceso llevado a cabo.

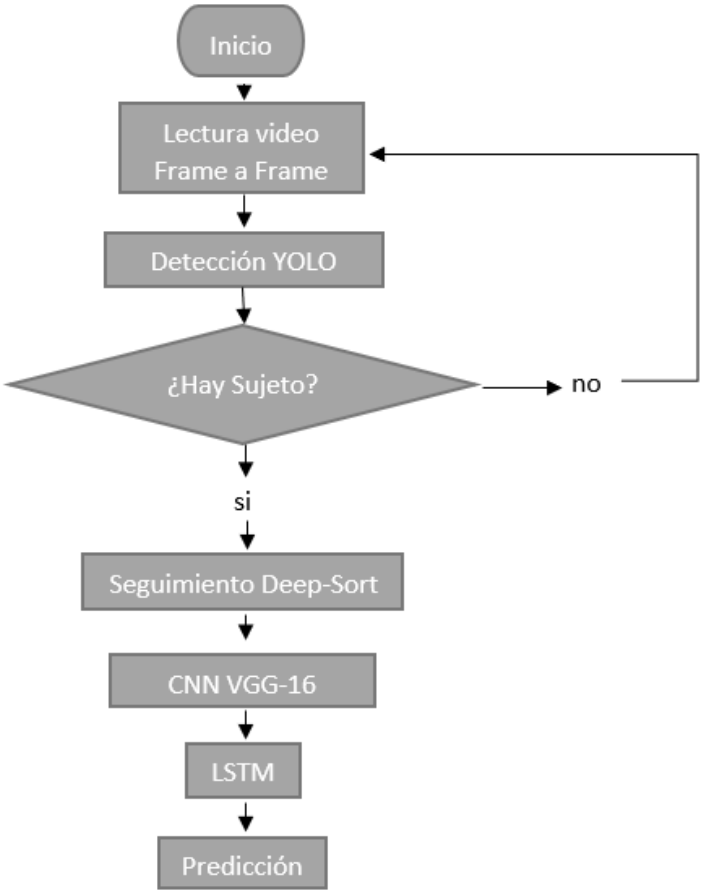

Figura 33.Diagrama de flujo del Algoritmo [124].

3.4.3 Algoritmo basado en OpenPose

Para realizar la implementación del algoritmo que incorpora YOLO y redes neuronales convolucionales ([125]) se utiliza el entorno de programación Google Colab, utilizando librerías como “google.colab”, “tensorflow”, “keras”, “matplotlib”, “pandas” y “sklearn”.

La Figura 19 representa la estructura del algoritmo, los datos de entrada deben ser ingresados a OpenPose, para lo cual se deben descargar e instalar configuraciones necesarias desde el sitio oficial de OpenPose [150], así mismo debe configurarse el uso de GPU facilitada por el entorno Google Colab, además, se realiza la programación necesaria para que en las imágenes de salida se obtenga únicamente los mapas óseos de la silueta humana, realizando sustracción de fondo.

Después, esta información es ingresada a una red neuronal convolucional pre-entrenada, específicamente el modelo Inception_ResNet_V2, el procedimiento es el siguiente:

- El primer paso es realizar un preprocesamiento a los datos que ingresan a la CNN, para lo cual se utiliza “ImageDataGenerator” y “flow_from_directory”, comandos que permiten normalizar los valores de pixeles de las imágenes, asignar el tamaño que requiere la red ResNet_V2 (299x299px), además, de realizar un aumento de datos.
- Después se debe cargar la red CNN pre-entrenada ResNet_V2, para lo cual utilizamos librerías como “tensorflow.keras.applications import InceptionResNetV2”, se configura la red y se modifican sus capas de salida, ya que ResNet_V2 por defecto puede clasificar 1000 clases, pero en la presente investigación solo se va a predecir “caída” y “no caída”, por lo que se modifica su última capa densa a dos neuronas con activación softmax.
- Finalmente se realiza la clasificación.

El entrenamiento de la red, se llevó a cabo usando “model.compile” y “model.fit”, asignando parámetros de entrenamiento definidos por los autores. La Figura 34 resume el proceso llevado a cabo.

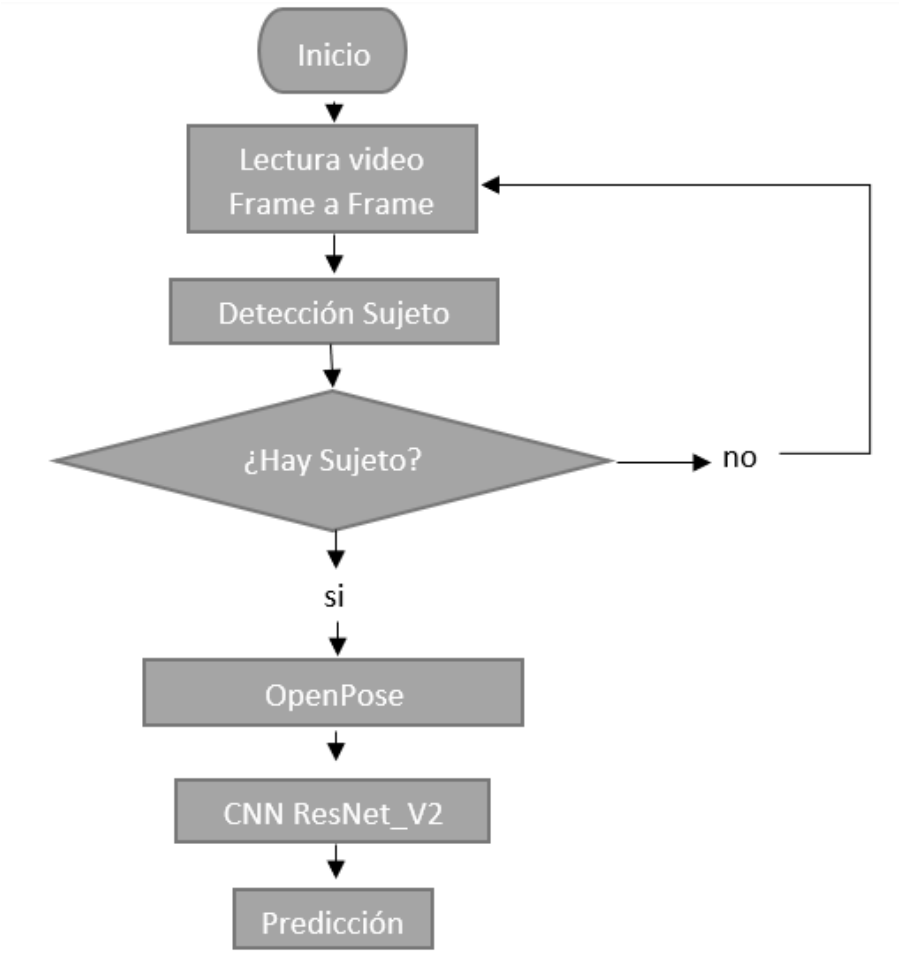

Figura 34. Diagrama de flujo del Algoritmo [125].

3.5 Experimentos Propuestos

Para comparar el desempeño de los tres algoritmos ([123,124,125]) para la detección de caídas humanas, se utilizan las bases de datos públicas más populares en la comunidad científica para la detección de caídas (UR Fall Detection [42], Multicam Fall Dataset [46], LE2I [43], UP-Fall [45]) además de utilizar CAUCAFall [49], para analizar su rendimiento en un ambiente no controlado. Para lo cual se proponen los siguientes experimentos:

a) Experimento 1: Cada uno de los algoritmos será evaluado con cada una de las bases de datos mencionadas, por separado. Tomando el 50% de los datos para entrenamiento, 25% para validación y 25% para prueba.

b) Experimento 2: Como se requiere analizar si las investigaciones que utilizan bases de datos de caídas humanas en entornos controlados, son capaces de generalizar predicciones de caídas en entornos similares a la realidad, en este experimento se realiza el entrenamiento del modelo con cada uno de los algoritmos usando las diferentes bases de datos ([42][43][45][46]), y se prueba el modelo en cada una de las condiciones que propone CAUCAFall, por separado como muestra la Figura 35.

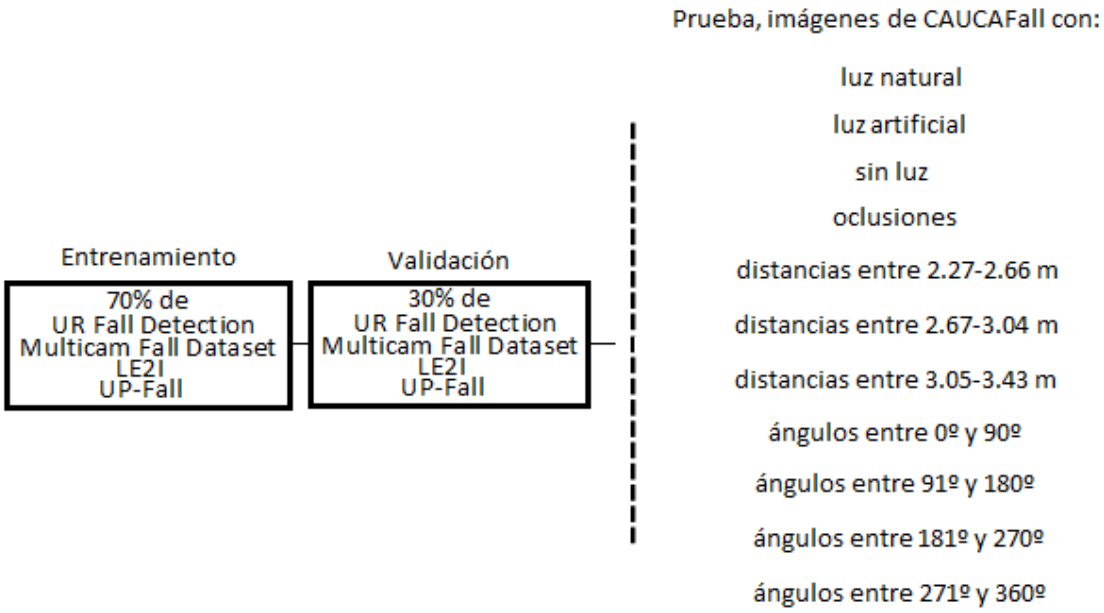

Figura 35. Experimento2.

c) Experimento 3: El mismo modelo entrenado en el experimento 2, se evalúa en la totalidad de CAUCAFall, como muestra la Figura 36.

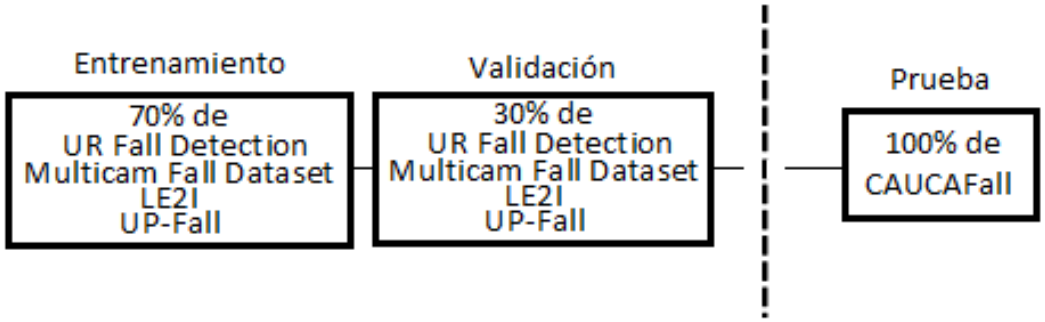

Figura 36. Experimento3.

d) Experimento 4: Se entrena el modelo con la totalidad de CAUCAFall, y se evalúa su rendimiento con las diferentes bases de datos propuestas ([42][43][45][46]), como muestra la Figura 37.

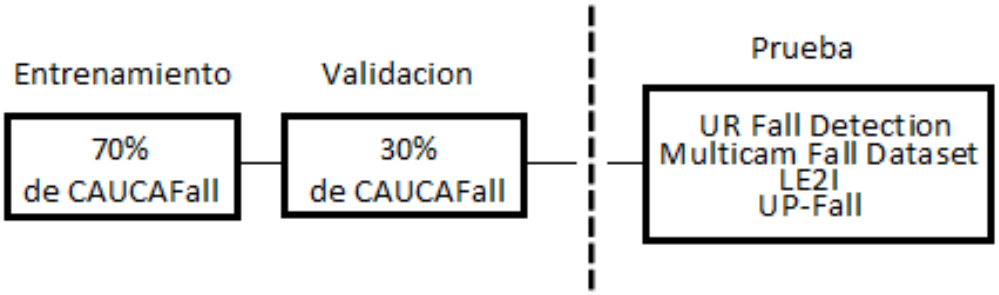

Figura 37. Experimento4.

3.6 Índices de desempeño

En esta sección se definen las métricas con las cuales se determina el desempeño de los diferentes algoritmos a comparar. Como se describió en la anterior sección las pruebas se realizan con los diferentes algoritmos usando las mencionadas bases de datos. Por lo que, en general, los índices de desempeño de los algoritmos dependen de la cantidad de aciertos y errores de clasificación, en la presente investigación se tiene en cuenta el cálculo de los indicadores de matrices de confusión, Precision, Recall y F-Score como base para presentar la eficiencia de los algoritmos.

Entendiendo por eficiencia, al proceso en el cual se abarca una necesidad de un problema determinado en función de la cantidad de recursos utilizados (costo computacional) y el desempeño del algoritmo. Es decir, un algoritmo es eficiente cuando cumple con una determinada tarea utilizando la menor cantidad de recursos posibles, minimizando el uso de memoria, de cálculos, procesos e incluso esfuerzo humano. Según Murillo [151] la eficiencia se mide en aspectos como el volumen de los datos utilizados, la calidad del código generado por el compilador y la rapidez en la ejecución del algoritmo.

Los índices de desempeño para evaluar los métodos propuestos usando las diferentes bases de datos se basan en los valores de: Verdadero Positivo (TP), Falso Positivo (FP), Verdadero Negativo (TN) y Falso Negativo (FN), para el caso de estudio de esta investigación se tiene la definición presentada en la Tabla 19.

Tabla 19. Variables para calcular índices de desempeño.

Valor	Definición	Significado
TP	Verdadero Positivo	Número de caídas que se clasifican como caídas
TN	Verdadero Negativo	Número de no caídas que se clasifican como no caídas
FP	Falso Positivo	Número de no caídas que se clasifican como caídas
FN	Falso Negativo	Número de caídas que se clasifican como no caídas

3.6.1 Matriz de confusión

La matriz de confusión contiene en sus filas las clases predichas, mientras que en sus columnas contienen las clases reales, la matriz se compone de los valores de las variables TP (verdadero positivo), FP (falso positivo), FN (falso negativo) y TN (verdadero negativo) (ver Figura 38), convirtiéndose en un instrumento para medir y visualizar el desempeño de un clasificador.

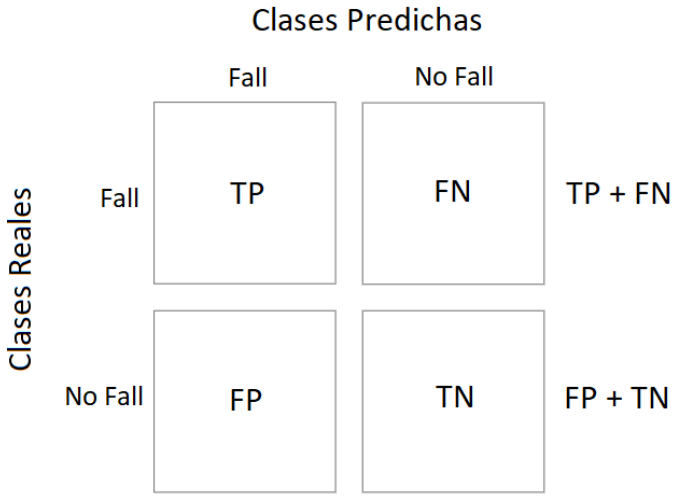

Figura 38. Matriz de Confusión.

3.6.2 Precisión

Métrica que está definida por la ecuación 22, en donde se divide el número de caídas correctamente clasificadas (TP), entre todas las imágenes clasificadas como caídas (TP+FP).

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad (22)$$

3.6.3 Recall

Métrica también llamada sensibilidad, que está definida por la ecuación 23, en donde se divide el número de caídas correctamente clasificadas (TP), entre el total de imágenes de caída (TP+FN). Mostrando la capacidad de clasificar caídas como caídas.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \qquad (23)$$

3.6.4 FScore

Métrica que está definida por la ecuación 24, y que relaciona precisión con sensibilidad.

$$\text{FScore} = \frac{2*\text{Precision}*\text{Recall}}{\text{Precision+Recall}} \qquad (24)$$

4 Presentación de Resultados

Los resultados de los experimentos realizados, son el aporte principal en la síntesis de la presente investigación, con el objetivo de encontrar las ventajas y desventajas que presentan las redes convolucionales con respecto a algoritmos de extracción de características basada en visión computacional en cuanto a precisión, costo computacional y eficiencia en el reconocimiento de caídas.

En esta sección se presentan los principales resultados de implementación de los diferentes algoritmos y su desempeño al ser evaluados en las diferentes bases de datos, principalmente en CAUCAFall.

La Figura 39, muestra en funcionamiento el algoritmo de extracción de características [123], al momento de evaluarlo en las diferentes bases de datos. Mientras que la Figura 40 muestra un ejemplo del almacenamiento de las características en un archivo (.csv).

Figura 39. (a) caída de CAUCAFall, (b) no caída de CAUCAFall, (c) caída de URFD, (d) no caída de URFD, (e) caída de le2i, (f) no caída de le2i, (g) caída de MCF, (h) no caída de MCF.

NFrame,Area,AspectoRelacional,DistanciaSuelo,Accion,AccNum			
2,6062.5,0,66,NoFall,0			
3,6062.5,0,66,NoFall,0			
36,7299.0,0,145,NoFall,0			
37,6844.5,0,141,NoFall,0			
38,5135.5,0,138,NoFall,0			
39,6012.0,0,141,NoFall,0			
40,5723.0,0,146,NoFall,0			
41,5723.0,0,146,NoFall,0			
43,7271.0,0,124,NoFall,0			
44,6561.0,0,114,NoFall,0			
45,5350.5,0,119,NoFall,0			
46,5068.0,0,118,NoFall,0			
106,8973.0,1,58,Fall,1			
107,9081.0,1,58,Fall,1			
108,8947.0,1,56,Fall,1			
109,8853.5,1,55,Fall,1			
110,9227.0,1,53,Fall,1			
111,9912.5,1,56,Fall,1			
112,10874.5,1,58,Fall,1			
113,11152.5,1,59,Fall,1			
114,11497.0,1,64,Fall,1			

Figura 40. Ejemplo de almacenamiento de características (.csv).

Por su parte la Figura 41, muestra en funcionamiento el algoritmo que incorpora YOLO/CNN [124], al momento de evaluarlo en las diferentes bases de datos, en la figura se detectan caídas marcadas de color verde y no caídas marcadas de color rosado.

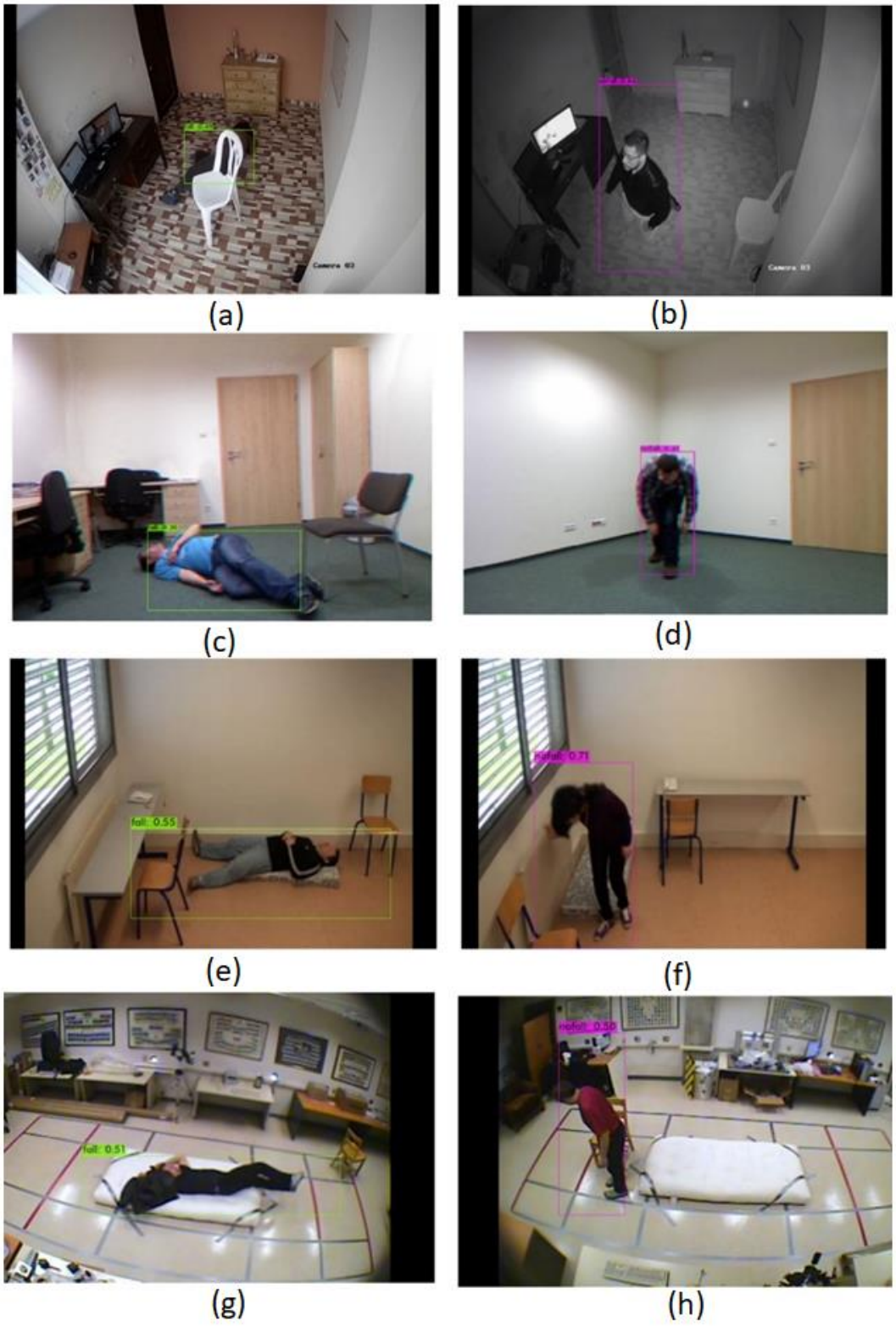

Figura 41. (a) caída de CAUCAFall, (b) no caída de CAUCAFall, (c) caída de URFD, (d) no caída de URFD, (e) caída de le2i, (f) no caída de le2i, (g) caída de MCF, (h) no caída de MCF.

Finalmente, la Figura 42 muestra en funcionamiento el algoritmo que incorpora OpenPose [125], al momento de evaluarlo en las diferentes bases de datos.

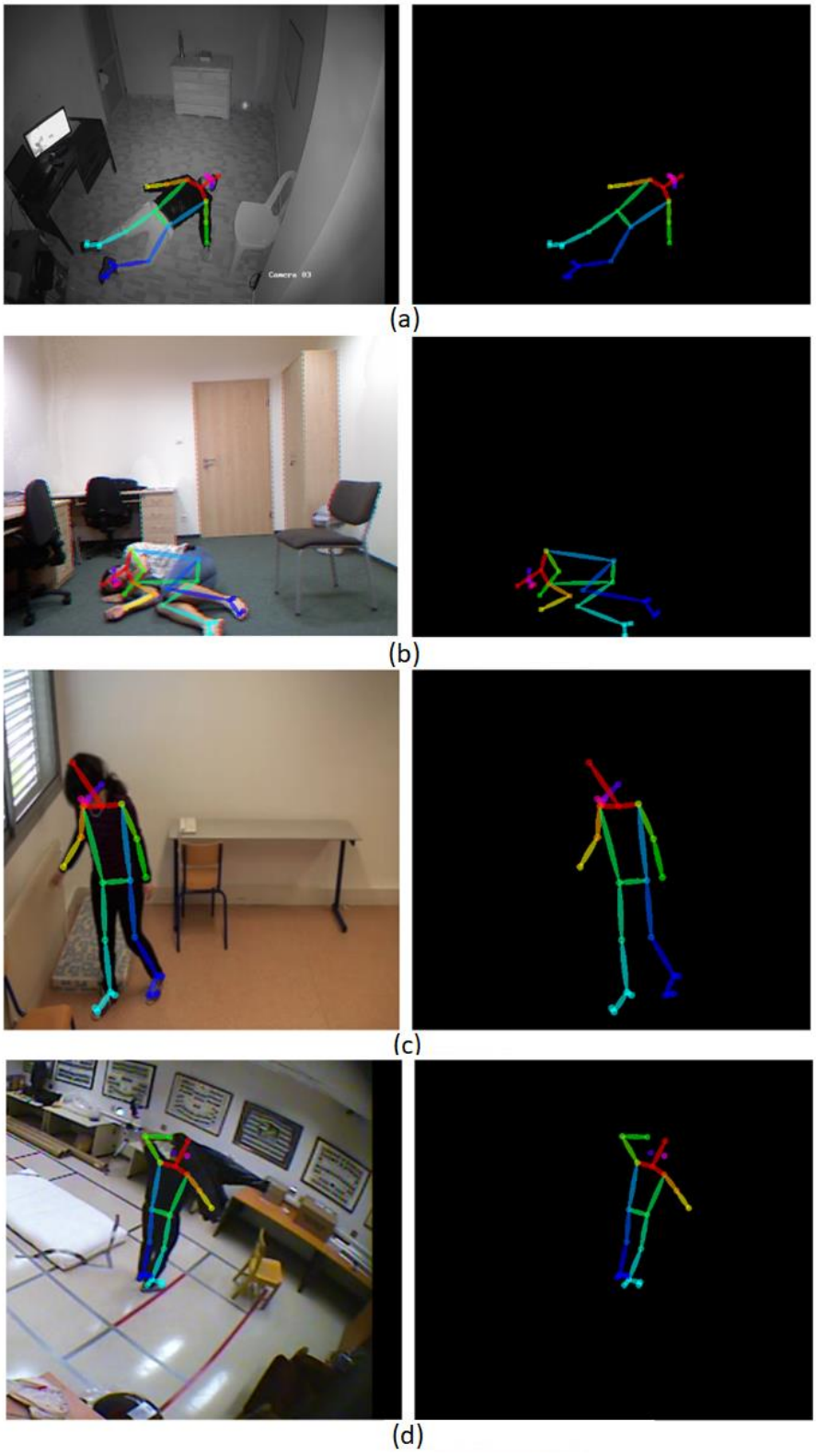

Figura 42. (a) caída de CAUCAFall, (b) caída de URFD, (c) no caída de le2i, (d) no caída de MCF.

Además, la Figura 43 detalla la estructura de red neuronal convolucional pre-entrenada VGG-16 implementada por los autores en el entorno Google Colab para el entrenamiento de los algoritmos CNN.

```
Number of layers in the base model:  19
Model: "model"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_1 (InputLayer)        [(None, 224, 224, 3)]     0

 block1_conv1 (Conv2D)       (None, 224, 224, 64)      1792

 block1_conv2 (Conv2D)       (None, 224, 224, 64)      36928

 block1_pool (MaxPooling2D)  (None, 112, 112, 64)      0

 block2_conv1 (Conv2D)       (None, 112, 112, 128)     73856

 block2_conv2 (Conv2D)       (None, 112, 112, 128)     147584

 block2_pool (MaxPooling2D)  (None, 56, 56, 128)       0

 block3_conv1 (Conv2D)       (None, 56, 56, 256)       295168

 block3_conv2 (Conv2D)       (None, 56, 56, 256)       590080

 block3_conv3 (Conv2D)       (None, 56, 56, 256)       590080

 block3_pool (MaxPooling2D)  (None, 28, 28, 256)       0
 block4_conv1 (Conv2D)       (None, 28, 28, 512)       1180160

 block4_conv2 (Conv2D)       (None, 28, 28, 512)       2359808

 block4_conv3 (Conv2D)       (None, 28, 28, 512)       2359808

 block4_pool (MaxPooling2D)  (None, 14, 14, 512)       0

 block5_conv1 (Conv2D)       (None, 14, 14, 512)       2359808

 block5_conv2 (Conv2D)       (None, 14, 14, 512)       2359808

 block5_conv3 (Conv2D)       (None, 14, 14, 512)       2359808

 block5_pool (MaxPooling2D)  (None, 7, 7, 512)         0

 flatten (Flatten)           (None, 25088)             0

 dense (Dense)               (None, 2)                 50178

=================================================================
Total params: 14,764,866
Trainable params: 50,178
Non-trainable params: 14,714,688
_________________________________________________________________
```

Figura 43. Implementación VGG-16 en entorno Colab.

4.1 Resultados Experimento1

Como se aprecia en la Figura 39, Figura 41 y Figura 42 los algoritmos fueron implementados en las diferentes bases de datos seleccionadas para esta investigación. Los resultados obtenidos se muestran en la Tabla 20.

Tabla 20. Resultados experimento 1.

Algoritmo	Database	Precision	Recall	F-Score
Extracción de características	CAUCAFall	87.02%	86.26%	85.15%
	UR Fall Detection	93.61%	93.70%	93.61%
	Multicam Fall Dataset	96.82%	96.70%	96.61%
	LE2I	97.43%	96.95%	97.07%
	UP-Fall	99.23%	99.23%	99.22%
YOLO/CNN	CAUCAFall	91.93%	91.76%	91.78%
	UR Fall Detection	96.20%	96%	96.04%
	Multicam Fall Dataset	92.59%	92.25%	92.20%
	LE2I	96.75%	96.75%	96.74%
	UP-Fall	100%	100%	100%
OpenPose/CNN	CAUCAFall	97.25%	97.25%	97.25%
	UR Fall Detection	87.88%	87.75%	85.40%
	Multicam Fall Dataset	90.05%	88.5%	88.46%
	LE2I	96.52%	96.25%	96.25%
	UP-Fall	100%	100%	100%

4.2 Resultados Experimento 2

La Tabla 21 detalla los resultados del experimento 2, utilizando los diferentes algoritmos (Alg1 [123], Alg2 [124], Alg3 [125]).

Tabla 21. Resultados experimento 2

Condiciones CAUCAFall	Precisión			Recall			F-Score		
	Alg 1	**Alg 2**	**Alg 3**	**Alg 1**	**Alg 2**	**Alg 3**	**Alg 1**	**Alg 2**	**Alg 3**
Luz Natural	84.46%	92.41%	90.39%	79.53%	91.75%	90%	76.79%	91.72%	89.97%
Luz Artificial	91.69%	89.9%	87.95%	90.79%	89.5%	87.75%	89.97%	89.47%	87.73%
Sin Luz	88.57%	91.68%	94.72%	86.46%	92%	94.5%	84.71%	91.64%	94.49%
Oclusiones	86.91%	71.12%	74.57%	65.92%	69.5%	73.87%	68.80%	70.35%	74.21%
Distancia cámara-caída									
2.27 m – 2.66 m	97.70%	86.31%	90.56%	42.38%	85.25%	85.25%	56.92%	85.70%	86.65%
2.67 m – 3.04 m	96.64%	91%	97.47%	68.91%	92%	97.5%	78.43%	91.53%	97.48%
3.05 m – 3.43 m	89.43%	83.80%	91.08%	60.19%	79.5%	86.75%	65.71%	81.07%	87.90%
Ángulos de Caída									
0º - 90º	96.39%	90.52%	96.77%	70.12%	91%	96.75%	79.09%	90.62%	96.75%
91º - 180º	82.41%	67.94%	85.86%	64.83%	62.25%	80.25%	64.15%	63.60%	80.96%
181º - 270º	95.78%	82.76%	97.59%	94.73%	83%	97.5%	94.93%	82.82%	97.4%
271º - 360º	95.84%	85.92%	88.77%	29.83%	84.5%	78.75%	40.64%	85.08%	81.27%

4.3 Resultados Experimento 3

La Tabla 22 detalla los resultados del experimento 3, utilizando los diferentes algoritmos.

Tabla 22. Resultados Experimento 3.

Algoritmo	Precisión	Recall	F-Score
Extracción de Características [123]	88.57%	86.52%	84.96%
YOLO/CNN [124]	88.03%	87%	87.19%
OpenPose [125]	87.63%	82.75%	83.14%

4.4 Resultados Experimento 4

La Tabla 23 detalla los resultados del experimento 4, utilizando los diferentes algoritmos.

Tabla 23. Resultados Experimento 4.

Algoritmo	Precisión	Recall	F-Score
Extracción de Características [123]	90.21%	91.37%	90.83%
YOLO/CNN [124]	93.58%	92.87%	93.32%
OpenPose [125]	94.58%	93.5%	94.07%

4.5 Costo Computacional

Para determinar la eficiencia de los algoritmos, aparte de los índices de desempeño presentados en la sección 3.6 se debe analizar el costo computacional, es decir la cantidad de recursos utilizados, el volumen de los datos utilizados, procesos realizados y la rapidez en la ejecución del algoritmo. La Tabla 24 presenta un análisis comparativo de los principales recursos utilizados por los diferentes algoritmos.

Tabla 24. Costo Computacional de los algoritmos.

Índice de Costo Computacional	Extracción de Características [123]	YOLO/CNN [124]	OpenPose [125]
Volumen de datos utilizados para el entrenamiento de los modelos [bytes].	Los archivos que contienen la extracción de características de las bases de datos y que son utilizados para el entrenamiento de los modelos tienen los siguientes pesos: CAUCAFall.csv (254Kb), URFD.csv (274Kb), LE2i.csv (270 Kb), Fall_UP.csv (240 Kb), MCF.csv (383Kb).	Las imágenes que se usan para los entrenamientos de YOLO/CNN, son imágenes RGB contienen los siguientes pesos: CAUCAFall (7.24 Gb), URFD (6.5Gb), LE2i (3.5 Gb), Fall_UP (8.3 Gb), MCF (6.7Gb).	Las imágenes que se usan para Ingresar a OpenPose son las mismas usadas en el algoritmo [124] (CAUCAFall (7.24 Gb), URFD (6.5Gb), LE2i (3.5 Gb), Fall_UP (8.3 Gb), MCF (6.7Gb)). Sin embargo, al salir de OpenPose las imágenes tienen menos peso (CAUCAFall (2.4 Gb), URFD (1.53Gb), LE2i (0.8 Gb), Fall_UP (3.23 Gb), MCF (1.87Gb)) y son las usadas para entrenar la red ResNet_v2.
Peso de los modelos Entrenados [bytes].	**Experimento 1**. Modelo con: CAUCAFall (174Kb), URFD (179Kb), LE2i (177 Kb), Fall_UP (174 Kb), MCF (176Kb). **Experimento 2 y 3:** 190Kb **Experimento 4:** 204 Kb	**Experimento 1**. Modelo con: CAUCAFall (213Mb), URFD (213Mb), LE2i (213Mb), Fall_UP (213Mb), MCF (213Mb). **Experimento 2 y 3:** 230Mb **Experimento 4:** 285Mb También se tiene que tener en cuenta el modelo pre-entrenado de YOLO (240Mb).	**Experimento 1**. Modelo con: CAUCAFall (213Mb), URFD (213Mb), LE2i (213Mb), Fall_UP (213Mb), MCF (213Mb). **Experimento 2 y 3:** 213Mb **Experimento 4:** 213Mb
Procesos realizados	1. Ingresan datos. 2. Sustracción de fondo. 3. Detección de objeto. 4. Extracción de características. 5. Clasificador SVM. 6. Evaluación del algoritmo.	1. Ingresan datos. 2. Preprosesamiento de datos. 3. Detector YOLO. 4. Procesamiento de datos. 5. Red VGG-16. 6. Red LSTM. 7. Evaluación del algoritmo.	1 Ingresan datos. 2 OpenPose. 3 Procesamiento de datos. 4 Red ResNet-v2. 5 Evaluación del algoritmo.
Ejecución	En el computador que se llevó a cabo las implementaciones de los algoritmos y que se describió en la sección 3.3, el algoritmo de extracción de características tuvo un buen desempeño e incluso se pudo ejecutar en tiempo real, obteniendo las predicciones mostradas en los resultados.	Con los recursos del computador utilizado, no se pudo ejecutar el algoritmo, ya que YOLO necesita de una GPU dedicada, es por eso que se utilizó la herramienta Google Colab, en donde si fue posible entrenar los modelos y clasificar las diferentes imágenes.	El algoritmo no pudo ser utilizado con los recursos propios del computador, por lo que se ejecutó en Colab para su entrenamiento y evaluación. Para que OpenPose pueda trabajar en tiempo real, se deben tener mayores requisitos de hardware. Según su autor [125] se necesita la aceleración de cuatro GPU y si se usa una computadora portátil normal, cada fotograma del video tendrá un retraso de 1 s a 1,5 s y además el efecto en tiempo real es deficiente.
Tiempo de ejecución	0.7 segundos por imagen	2.4 segundos por imagen	4.5 segundos por imagen

5 Discusión de los resultados

Como se puede apreciar en la sección de resultados, los tres algoritmos de reconocimiento de actividades funcionan correctamente en términos generales. El algoritmo de extracción de características extrae el área de la silueta, distancia del centroide al suelo y espectro radial, con lo cual utiliza un clasificador SVM para reconocer caídas humanas. Por su parte el algoritmo que implementa YOLO, reconoce y encuadra la silueta humana, reconociendo las caídas del sujeto, y finalmente OPENPOSE obtiene la imagen del esqueleto en 2D, efecto que para imágenes de alta definición es muy bueno, por lo que es necesario que la resolución de la imagen sea óptima. En esta sección se discute cuál de los algoritmos tuvo mejores resultados reconociendo caídas humanas en las diferentes bases de datos, incluyendo CAUCAFall, encontrando las ventajas y desventajas que presentan las redes convolucionales con respecto a algoritmos de extracción de características basados en visión computacional en cuanto a eficiencia, precisión y coste computacional en el reconocimiento de caídas.

5.1 Discusión experimento 1:

Este experimento se realiza para ver el correcto funcionamiento de los algoritmos, ya que predicen caídas humanas en imágenes de las mismas bases de datos con la que se entrenó el modelo. La Tabla 20 muestra los resultados del experimento 1 utilizando el algoritmo de extracción de características, se puede observar que el algoritmo obtiene mayor precisión con la base de datos UP-Fall [45], base de datos que intencionalmente fue escogida por los autores para evaluar los algoritmos, por ser una base de datos con un ambiente sumamente controlado, su enfoque es principalmente la recolección de datos de sensores de ambiente, sin embargo también trabaja con 2 cámaras de video, pero no tiene cambios de escenario, no tiene oclusiones, no tiene cambios de iluminación, únicamente tiene 2 ángulos de caída, los cuales se graban a la misma distancia desde la cámara e incluso algunos de los participantes que simulan las caídas utilizan un chaleco reflector, lo cual deriva en un alto rendimiento en el reconocimiento de las caídas.

El menor rendimiento se obtuvo con CAUCAFall, base de datos propuesta por los autores, debido a que la extracción de características y sustracción de fondo se dificultó en aquellos escenarios donde se contaba con oclusiones y existía movimiento en el fondo, además que al existir diferentes distancias de las caídas hasta la cámara, causa que la silueta humana de la acción de caída tenga diversidad en características como, área, distancia del centroide al suelo y relación de aspecto, lo que dificulta el reconocimiento de las caídas, sin embargo el rendimiento es aceptable.

Cuando se utilizó el algoritmo basado en detectores YOLO y redes neuronales convolucionales, se encuentra que el mejor rendimiento se obtiene nuevamente con UP-Fall [45] (ver Tabla 20), por sus facilidades de reconocimiento, seguido de LE2I [43], ya que es un escenario con una iluminación controlada y sin cambios de entorno, posición, ni distancia de las caídas a la cámara. Por su parte CAUCAFall [49], UR Fall Detection [42] y Multicam Fall Dataset [46] poseen un rendimiento similar. Sin embargo, el algoritmo evaluado con CAUCAFall [49] sigue teniendo el rendimiento más bajo, La diferencia radica en que CAUCAFall tiene en cuenta oclusiones parciales (ver Figura 44), diferentes iluminaciones (ver Figura 45), diferentes ángulos de caída (ver Figura 46) y distancias ante la cámara (ver Figura 47).

Figura 44. Predicción del algoritmo, teniendo en cuenta oclusiones parciales.

(a) (b) (c)

Figura 45. (a) Iluminación natural, (b) Iluminación artificial, (c) Sin iluminación.

(a) (b) (c)

Figura 46. (a) caída con 47. 8º de ángulo, (b) caída con 182.3º de ángulo, (c) caída con 277.5º de ángulo.

Figura 47. (a) distancia de caida:2,7m, (b) distancia de caida:3.12m, (c) distancia de caida:3.4m.

Además, la Tabla 20 muestra los resultados de la evaluación de las bases de datos en el algoritmo basado en OpenPose, el funcionamiento de este algoritmo depende de la buena calidad de los videos y frames, que para tener buenos resultados deben ser de alta definición, además, OpenPose tiene dificultades para trabajar con oclusiones como se puede observar en la Figura 48, y también se necesita de una buena iluminación o segmentación de la silueta humana, es decir que si la imagen es oscura no va a encontrar el mapa óseo del sujeto. CAUCAFall es una base de datos creada con cámara de alta resolución, lo que ayuda a que se pueda detectar el esqueleto humano de mejor forma, además, trabaja con una cámara que incorpora visión nocturna lo cual ayuda a que siempre cuente con imágenes en las que se puede reconocer el cuerpo humano y esto ayudó a realizar reconocimiento de caídas en los escenarios de poca o nula iluminación.

Figura 48. OpenPose con oclusiones.

Sin embargo, el mejor resultado se presenta cuando se utiliza UP-Fall [45], ya que también es creada con una cámara de alta resolución, además de que únicamente considera dos ángulos de caída, lo cual facilita el reconocimiento de la misma. A pesar de que ninguno de los dos ángulos es óptimo para OpenPose, ya que se pierden muchos puntos óseos de la silueta humana, se obtiene una excelente precisión por que se entrena, evalúa y prueba en las mismas imágenes, como lo muestra la Figura 49.

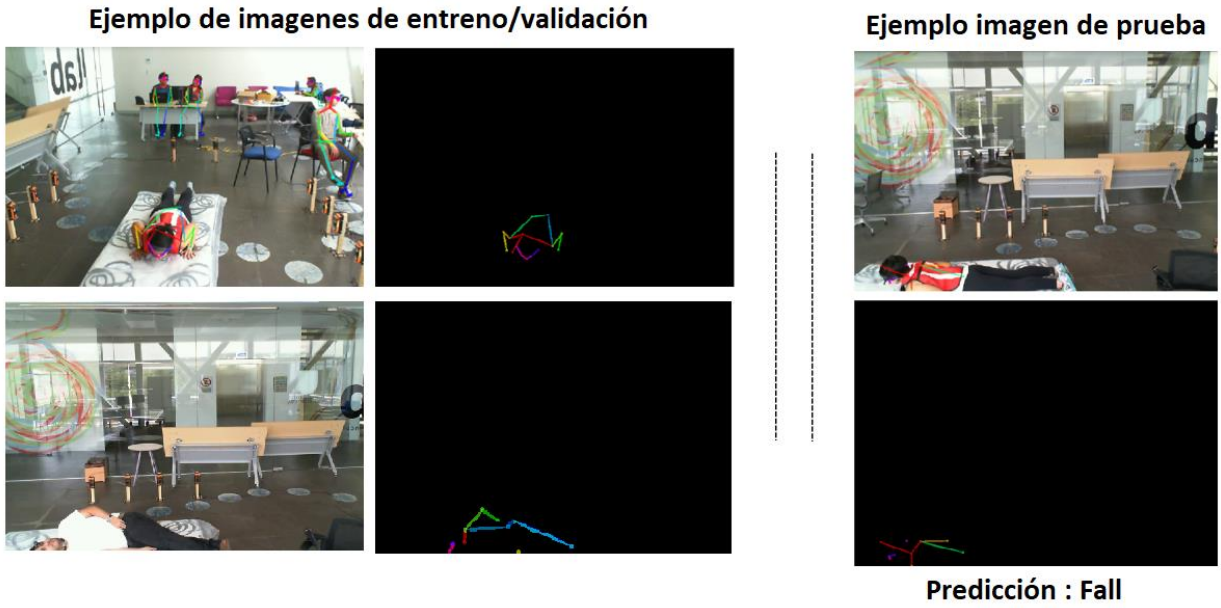

Figura 49. Ejemplo de reconocimiento de caída en UP-Fall.

5.2 Discusión experimento 2:

El experimento 2, entrena el modelo con las diferentes bases de datos y evalúa el rendimiento en las diferentes condiciones de CAUCAFall, ninguna otra investigación tiene resultados de reconocimiento de caídas, especificando las condiciones como se presentan a continuación:

La Tabla 21, detalla los resultados del experimento 2 utilizando extracción de características, al entrenar el modelo con las bases de datos en ambientes controlados y evaluar su rendimiento en CAUCAFall, los resultados con las diferentes condiciones varían, se tiene un buen reconocimiento de caídas en entornos con Luz artificial y un buen resultado sin luz, esto se debe a que la cámara de CAUCAFall tiene visión nocturna y se puede hacer una buena extracción de características. Sin embargo, el rendimiento es inferior cuando existen oclusiones y cuando las caídas se realizan muy cerca de la cámara o muy alejadas de ella. Además, se tiene un rendimiento inferior cuando las caídas tienen ángulos entre los rangos de 91º - 180º y 271º - 360º. Esto se debe a que las bases de datos que sirvieron para el entrenamiento del modelo, no cuentan con gran cantidad de estas características.

Al comparar el algoritmo de extracción de características y el algoritmo que utiliza YOLO, redes neuronales convolucionales (ver Tabla 21), se puede apreciar que el reconocimiento de caídas, en acciones que tienen oclusión bajó su rendimiento, sin embargo, la precisión aumentó en distancias lejanas y cercanas a la cámara, al igual que en los diferentes ángulos de caída. En cuanto a condiciones de luz, se tiene un rendimiento similar, esto se debe a que la cámara tiene sensores de visión nocturna, lo que permite captar imágenes en buena definición sin luz.

Por otra parte, el rendimiento de OpenPose, evaluado en CAUCAFall aumentó el rendimiento, ya que el modelo se encarga de aprender la forma de los esqueletos humanos, sin importar las distracciones del entorno, OpenPose solo necesita tener imágenes claras con buena resolución para obtener el mapa ósea del sujeto, lo que permite tener un buen rendimiento, a pesar de su alto costo computacional, sin embargo, los problemas que causa la oclusión se mantienen.

5.3 Discusión experimento 3:

El tercer experimento utiliza el modelo entrenado por las bases de datos comúnmente usadas y evaluado en la totalidad de CAUCAFall. Al comparar los índices de desempeño de la Tabla 22, se observa un rendimiento muy similar, sin embargo, el mejor rendimiento se consigue con el algoritmo que propone redes neuronales convolucionales con YOLO, seguido de la extracción de características y por último el algoritmo basado en OpenPose. Esto se debe a que la mayoría de las imágenes de las bases de datos ([42][43][45][46]), no son en alta resolución, lo que impide extraer correctamente el mapa óseo de la silueta humana, dificultando el entrenamiento de la red.

5.4 Discusión experimento 4:

En el último experimento se entrena y valida los diferentes modelos usando la totalidad de CAUCAFall y se evalúa con las bases de datos propuestas ([42][43][45][46]), la Tabla 23 muestra claramente que el rendimiento aumentó notablemente, demostrando que CAUCAFall cumple con condiciones que al utilizarlas como datos de entrenamiento son capaces de generalizar el reconocimiento de caídas humanas. Los tres algoritmos son capaces de extraer las características más importantes de las imágenes y aprenderlas, caídas en diferentes ángulos, distancias y condiciones de luz, ayuda a evaluar su rendimiento satisfactoriamente.

En esta ocasión el algoritmo con mejor rendimiento es el que combina OpenPose con CNN, gran porcentaje de este éxito se debe a la calidad de las imágenes, además, los mapas óseos extraídos de CAUCAFall, al tener variedad de tamaño, ángulos y distancias son buenos para generalizar y predecir caídas en las otras bases de datos.

Finalmente es importante mencionar que para que OPENPOSE y YOLOV3 puedan trabajar en tiempo real reconociendo caídas por medio de visión computacional, se deben tener mayores requisitos de rendimiento de hardware. Según el autor [125] OpenPose necesita la aceleración de cuatro GPU y si se usa una computadora portátil normal, cada fotograma del video tendrá un retraso de 1 s a 1,5 s y además el efecto en tiempo real es deficiente.

La Tabla 25 resume la comparación en cuanto a ventajas y desventajas que presentan las redes convolucionales utilizadas en los algoritmos [124] y [125] con respecto al algoritmo de extracción de características [123] en cuanto a eficiencia, precisión y coste computacional en el reconocimiento de caídas humanas.

Tabla 25. Comparación de Eficiencia de Algoritmos.

	Extracción de Características [123]	**YOLO/CNN [124]**	**OpenPose [125]**
Ventajas	• El efecto de sustracción de fondo que es importante para la extracción de características, tiene buenos resultados en entornos sin oclusión, buena iluminación y sin movimiento en el fondo. • Su rendimiento es alto en bases de datos que no contienen variedad en distancias entre la cámara y la caída, ni variedad de ángulos de caída. • Alto rendimiento en entornos con luz artificial y con cámara nocturna.	• Su rendimiento en cuando a precisión, es sobresaliente en entornos con diferente iluminación y en entornos con poca saturación de elementos distractorios. • No se ve afectado por la diversidad de distancias entre la cámara y la caída, ni tampoco por la diversidad de ángulos de caída. • El algoritmo es capaz de realizar detecciones precisas ante oclusiones parciales.	• Si se logra una excelente extracción del mapa óseo, su rendimiento es sobresaliente. • Elimina la saturación del entorno del sujeto, concentrándose en el mapa óseo humano, lo que ayuda al reconocimiento de caídas. • No se ve afectado por la variedad de ángulos de caída.
Desventajas	• Su rendimiento es menor con bases de datos que contienen oclusión, mala iluminación y movimientos en el fondo. • Poco rendimiento cuando las caídas se realizan muy cerca de la cámara o muy alejadas de ella. • El clasificador SVM usado en el algoritmo presenta poco rendimiento en bases de datos que contienen caídas a diferentes distancias ante la cámara y a diferentes ángulos, ya que esto ocasiona dispersión en características como área de la silueta humana, distancia del centroide al suelo y relación de aspecto.	• El algoritmo presenta dificultades ante oclusiones totales. • Necesita gran cantidad de datos para su entrenamiento.	• Su rendimiento depende de que los videos e imágenes tengan alta definición. • Si el ángulo de la cámara no enfoca la totalidad del cuerpo del sujeto, se pierden puntos clave del mapa óseo humano. • Para realizar una extracción de la totalidad del cuerpo humano, depende de que los movimientos del sujeto sean comunes, es decir que no sean antinaturales. • El efecto de OpenPose es nulo ante imágenes con iluminación nula. • El algoritmo presenta dificultades ante oclusiones totales.
Costo computacional	El volumen de datos utilizado es el inferior de los tres algoritmos, al igual que el tiempo de entrenamiento y peso de los modelos, he incluso el tiempo de ejecución del algoritmo por imagen es de 0.7 segundos.	El volumen de datos utilizado es el mayor de los tres algoritmos, al igual que el tiempo de entrenamiento y peso de los modelos, sin embargo, YOLO y el algoritmo se pudieron ejecutar con los recursos del computador en un tiempo de respuesta de 2.4 segundos.	El volumen de datos utilizado es menor en comparación con el algoritmo YOLO, al igual que el tiempo de entrenamiento y peso de los modelos, sin embargo, fue imposible ejecutar el algoritmo en tiempo real. Al ejecutar el algoritmo con imágenes pregrabadas el algoritmo tiene un tiempo de respuesta de 4.5 segundos.
Eficiencia	La eficiencia del algoritmo es sobresaliente, destacando su bajo costo computacional y su ejecución en tiempo real, sin embargo, su rendimiento se ve opacado por factores de iluminación, variedad en distancias y ángulos, lo que afecta directamente al reconocimiento de caídas humanas en entornos no controlados.	A pesar de que el entrenamiento del modelo y los datos utilizados son los de mayor peso, su rendimiento en cuanto a precisión es el mejor de los algoritmos planteados, además de que es posible su ejecución en tiempo real, por lo que se considera que el algoritmo es el de mayor eficiencia.	A pesar de que el algoritmo tiene buenos resultados de reconocimiento, depende estrictamente de la resolución de las imágenes, además su alto costo computacional no le permite ejecutar en tiempo real, en una computadora de características comunes, por lo que el algoritmo no el de menor eficiencia.

6 Conclusiones y recomendaciones

En el presente trabajo se realizó una revisión sobre la actualidad de las caídas de personas de tercera edad y los avances en su reconocimiento, dando prioridad a aquellas investigaciones que trabajan con visión computacional, para poder evaluar y evidenciar el rendimiento de algunas propuestas novedosas de reconocimiento de caídas en una base de datos con entornos realistas y poco controlados.

Para lo cual se diseñó y realizo CAUCAFall, base de datos que agrupa las cualidades de las diferentes bases de datos públicas más populares en la comunidad científica, con la intención de crear dichos entornos realistas y con características no existentes. El conjunto de datos considera individuos de diferentes edades, pesos, alturas y piernas dominantes. Los datos se adquirieron utilizando una cámara RGB en un entorno doméstico. Este entorno era intencionadamente realista e incluía características no controladas, como oclusiones, cambios de iluminación (natural, artificial y nocturna), ropa diferente de los participantes, movimiento en el fondo, diferentes texturas en el suelo y en la habitación, y una variedad en los ángulos de caída y diferentes distancias de la cámara a la caída. CAUCAFall es la única base de datos que contiene detalles de la iluminación lux de los escenarios, las distancias de la caída humana a la cámara y los ángulos de las diferentes caídas con referencia a la cámara. El conjunto de datos es también el único que contiene etiquetas para cada imagen. Los fotogramas que incluían caídas humanas registradas se etiquetaron como "caída", y las actividades AVD se marcaron como "no caída", siendo útil para algoritmos que incorporan detectores YOLO. Además, su alta resolución permite que novedosos algoritmos como OpenPose puedan extraer más información de las imágenes.

Usando diferentes bases de datos, se evaluó el desempeño de tres propuestas de reconocimiento de caídas, que basan su funcionamiento en extracción de características, YOLOV3 en unión con redes neuronales convolucionales y OpenPose con redes neuronales convolucionales, llegando a la conclusión de que en la propuesta que implementa extracción de características, el desempeño es alto en aquellas bases de datos con entornos sumamente controlados; sin cambios de iluminación, sin diferentes ángulos de caída, sin oclusiones, sin cambios de escenario, sin movimiento en el fondo, sin variedad en la distancia de las caídas ante la cámara e incluso sin cambios en las texturas tanto de los entornos como en la ropa de los participantes, concluyendo que estos métodos presentan gran sensibilidad al ruido, oclusiones y cambios en el punto de vista. Por otra parte, las propuestas que basan su trabajo en Redes Neuronales Convolucionales como base de sus investigaciones, tienen un rendimiento mayor en entornos no controlados, en comparación con la extracción de características. Sin embargo, su costo computacional es mucho mayor, siendo casi imposible utilizarlo en detección de caídas en tiempo real, además de que se necesitan grandes cantidades de conjuntos de datos, con diferentes posturas, distancias, ángulos, oclusiones con diferentes objetos, diferentes entornos e iluminaciones para que los modelos puedan tener un buen rendimiento.

Con la ventaja de que CAUCAFall fue creada con cámara de alta resolución se pudo utilizar modernos algoritmos para el reconocimiento del mapa óseo humano a partir de imágenes en 2D, lo cual es una gran ventaja en comparación con el resto de bases de datos que contienen caídas humanas.

Además, al proponer una base de datos que contenga eventos de caída en escenas complejas y cercanas a la realidad, y al evaluar su rendimiento con 3 métodos de reconocimiento de caídas, se demuestra que aún hay mucho trabajo y aspectos por mejorar e investigar en el área de reconocimiento de caídas, puesto que aún no existe un algoritmo que reconozca caídas humanas con tan alta precisión y rapidez.

Finalmente se ha mostrado la importancia de un eficiente reconocimiento de caídas y el gran potencial a futuro de la investigación en esta área, ya que si bien es cierto se obtienen buenos resultados con las diferentes técnicas planteadas por los autores en este artículo, los entornos de CAUCAFall aún pueden ser mejorados, es decir, no se cuenta con una base de datos que contenga caídas reales, lo cual no contribuye al verdadero avance de las investigaciones. Sin embargo, se considera sería de gran impacto, que la comunidad científica presente resultados de estas investigaciones en entornos semejantes a la realidad o a CAUCAFall. Además, es primordial investigaciones que centren su trabajo en la elaboración de bases de datos con caídas reales de personas adultas y en entornos no controlados. Como trabajo futuro, planeamos evaluar los métodos propuestos utilizando al menos dos cámaras, agregando técnicas como el reconocimiento de objetos en los escenarios y el análisis de velocidad de las diferentes siluetas humanas, además de evaluar novedosos métodos de detección como YOLOV4 y YOLOV5 en CAUCAFall.

Por otra parte, también es primordial avanzar en la investigación de algoritmos que puedan realizar las mismas o similares tareas, que OpenPose y YOLO, pero con un costo computacional menor, de tal forma que algoritmos como estos, puedan ser utilizados en tiempo real, para aplicaciones de la vida diaria.

También seria de gran avance para la investigación, agregar técnicas multimodales e incorporar cámaras de profundidad y sensores portables no invasivos como por ejemplo relojes smartwatch, para el reconocimiento de caídas humanas.

7 Bibliografía

[1] R. Saini, P. Kumar, P. Pratim, and D. Prosad, “Neurocomputing A novel framework of continuous human-activity recognition using Kinect,” *Neurocomputing*, vol. 311, pp. 99–111, 2018. doi: 10.1016/j.neucom.2018.05.042.

[2] T. Uzunovic, E. Golubovic, Z. Tucakovic, Y. Acikmese, and A. Sabanovic, “Task-Based Control and Human Activity Recognition for Human-Robot Collaboration,” *IECON 2018 - 44th Annu. Conf. IEEE Ind. Electron. Soc.*, vol. 1, pp. 5110–5115.

[3] E. Brophy, Z. Wang, and T. E. Ward, “A Machine Vision Approach to Human Activity Recognition using Photoplethysmograph Sensor Data,” 2018.

[4] M. Vrigkas, C. Nikou, and I. A. Kakadiaris, “A Review of Human Activity Recognition Methods,” *Front. Robot. AI*, vol. 2, Nov. 2015, doi: 10.3389/frobt.2015.00028.

[5] S. Lohit, A. Bansal, N. Shroff, J. Pillai, P. Turaga, and R. Chellappa, “Predicting Dynamical Evolution of Human Activities from a Single Image,” Jun. 2018. doi: 10.1109/CVPRW.2018.00079.

[6] E. Lawrence, C. Sax, K. F. Navarro, and M. Qiao, “Interactive Games to Improve Quality of Life for the Elderly: Towards Integration into a WSN Monitoring System,” Feb. 2010. doi: 10.1109/eTELEMED.2010.21.

[7] J. Han, L. Shao, D. Xu, and J. Shotton, “Enhanced Computer Vision With Microsoft Kinect Sensor: A Review,” *IEEE Trans. Cybern.*, vol. 43, no. 5, Oct. 2013. doi: 10.1109/TCYB.2013.2265378.

[8] R. Akhavian and A. H. Behzadan, “Smartphone-based construction workers activity recognition and classification,” *Autom. Constr.*, vol. 71, Nov. 2016. doi: 10.1016/j.autcon.2016.08.015.

[9] M. Ryoo, “Human Activity Prediction : Early Recognition of Ongoing Activities from Streaming Videos*,” IEEE International Conference on Computer Vision*. Barcelona, España. 2011.

[10] J. Preis, M. Kessel, M. Werner, and C. Linnhoff-popien, “Gait Recognition with Kinect.” ResearchGate, Conference paper. doi: https://www.researchgate.net/publication/239862819

[11] Y. Liu, X. Li, and L. Jia, “Abnormal crowd behavior detection based on optical flow and dynamic threshold,” Jun. 2014. doi: 10.1109/WCICA.2014.7053189.

[12] A. Ben Mabrouk and E. Zagrouba, “Abnormal behavior recognition for intelligent video surveillance systems: A review,” *Expert Syst. Appl.*, vol. 91, Jan. 2018. doi: 10.1016/j.eswa.2017.09.029.

[13] S. Coşar, G. Donatiello, V. Bogorny, C. Gárate, and L. O. Alvares, “Towards Abnormal Trajectory and Event Detection in Video Surveillance,” vol. 8215, no. c, pp. 1–14, 2016. doi: 10.1109/TCSVT.2016.2589859.

[14] J.-W. Hsieh, C.-H. Chuang, S. Alghyaline, H.-F. Chiang, and C.-H. Chiang, “Abnormal Scene Change Detection from a Moving Camera Using Bags of Patches and Spider-Web Map,” *IEEE Sens. J.*, 2014. doi: 10.1109/JSEN.2014.2381257.

[15] S. Wang, S. Zabir and B.Leibe, “Lying Pose Recognition for Elderly Fall Detection,” Conference: Robotics: Science and Systems VII, University of Southern California, Los Angeles, CA, USA, June 27-30, 2011. Doi: http://www.roboticsproceedings.org/rss07/p44.pdf

[16] O. Banos, M. Damas, H. Pomares, A. Prieto, and I. Rojas, “Expert Systems with Applications Daily living activity recognition based on statistical feature quality group selection,” *Expert Syst. Appl.*, vol. 39, no. 9, pp. 8013–8021, 2012, doi: 10.1016/j.eswa.2012.01.164.

[17] L. Chen, C. D. Nugent, and H. Wang, “A Knowledge-Driven Approach to Activity Recognition in Smart Homes,” *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 6, Jun. 2012, doi: 10.1109/TKDE.2011.51.

[18] A. Avci, S. Bosch, M. Marin-Perianu, R. Marin-Perianu, and P. Havinga, “Activity Recognition Using Inertial Sensing for Healthcare, Wellbeing and Sports Applications : A Survey.” 23th International Conference on Architecture of Computing Systems 2010.

[19] E. Kim, S. Helal, and D. Cook, “Human Activity Recognition and Pattern Discovery,” *IEEE Pervasive Comput.*, vol. 9, no. 1, Jan. 2010, doi: 10.1109/MPRV.2010.7.

[20] E. Sazonov, K. Metcalfe, P. Lopez-Meyer, and S. Tiffany, “RF hand gesture sensor for monitoring of cigarette smoking,” Nov. 2011, doi: 10.1109/ICSensT.2011.6137014.

[21] S. J. Ismail, M. A. A. Rahman, S. A. Mazlan, and H. Zamzuri, “Human gesture recognition using a low cost stereo vision in rehab activities,” Oct. 2015, doi: 10.1109/IRIS.2015.7451615.

[22] J. Rafferty, C. D. Nugent, J. Liu, and L. Chen, “From Activity Recognition to Intention Recognition for Assisted Living Within Smart Homes,” *IEEE Trans. Human-Machine Syst.*, vol. 47, no. 3, Jun. 2017, doi: 10.1109/THMS.2016.2641388.

[23] J. Usharani and U. Sakthivel, “Human Activity Recognition using Android Smartphone,” 1st International Conference on Innovations in Computing & Networking. pp. 191–197. Doi: https://www.ijana.in/Special%20Issue/S41.pdf

[24] E. B and W. Gomaa, "A Survey on Human Activity Recognition Based on Temporal Signals of Portable Inertial Sensors ", The International Conference on Advanced Machine Learning Technologies and Applications. Vol.(921),2019.

[25] “OMS | Datos interesantes acerca del envejecimiento,” *WHO*, 2015. http://www.who.int/ageing/about/facts/es/ (accessed Dec. 19, 2021).

[26] N. Thakur and C. Han, “A Study of Fall Detection in Assisted Living: Identifying and Improving the Optimal Machine Learning Method,” J. Sens. Actuator Netw. 2021, 10, 39. https://doi.org/10.3390/jsan10030039

[27] F. Shu and J. Shu “An eight-camera fall detection system using human fall pattern recognition via machine Learning by a low-cost android box,” Scientific Reports | (2021) 11:2471. https://doi.org/10.1038/s41598-021-81115-9

[28] Li. H, H. Shrestha, F. Fioranelli, L. Kernec and H. Heidari, “Hierarchical Classification on Multimodal Sensing for Human Activity Recogintion and Fall Detection,” *2018 IEEE SENSORS*, pp. 1–4. 2018. Doi: 10.1109/ICSENS.2018.8589797

[29] Z. Khan and W. Sohn, “A hierarchical abnormal human activity recognition system based on R-transform and kernel discriminant analysis for elderly health care,” Computing. Vol. 95, no. 2, pp. 109–127, 2013.

[30] S. Amiri, M. Pourazad, P. Nasiopoulos and V. Leung, “Improved human action recognition in a smart home environment setting,” IRBM. Vol. (35), no. 6, pp. 321–328, 2014.

[31] M. Yu, Y. Yu, Y. Rhuma, S. Naqvi, L. Wang and J. Chambers, “An online one class support vector machine-based person-specific fall detection system for monitoring an elderly individual in a

room environment," *IEEE J. Biomed. Heal. Informatics*. Vol. (17), no. 6, pp. 1002–1014, 2013.

[32] F. Concone, G. Re and M. Morana, "A Fog-Based Application for Human Activity Recognition Using Personal Smart Devices," *ACM Transactions on Internet Technology*. Vol. (19), no. 2, 2019.

[33] A. Sucerquia, J. López and J.Vargas, "SisFall: A Fall and Movement Dataset," *Sensors 2017*, vol. 17, no.198, 2017. doi:10.3390/s17010198.

[34] Y. Yang, C. Hou, Y. Lang, D. Guan, D. Huang, and J. Xu, "Open-set human activity recognition based on micro-Doppler signatures*," Pattern Recognit*, 2019.

[35] R. Kahani, A. Talebpour, and A. Mahmoudi-aznaveh, "A correlation based feature representation for first-person activity recognition," *Multimedia Tools and Applications*, mar, 2019.

[36] X. Ma, H. Wang, B. Xue, M. Zhou, B. Ji, and Y. Li, "Depth-based human fall detection via shape features and improved extreme learning machine," *IEEE J. Biomed. Heal. Informatics*, vol. 18, no. 6, pp. 1915–1922, 2014.

[37] B. Ameur, M. Belahcene, S. Masmoudi, A. Derbel, A. Hamida, "A new GLBSIF Descriptor for Face Recognition in the Uncontrolled Environments," *3rd International Conference on Advanced Technologies*, may, 2017.

[38] A. Herrera, "Reconocimiento de marca y modelo de vehículos en entornos no controlados," *Instituto Nacional de Astrofísica, Óptica y Electrónica*, tesis de maestro en ciencias en la especialidad de electrónica, 2013.

[39] D. Liu, Y. Yan, M. Shyu, G. Shao, "Spatio-Temporal Analysis for Human Action Detection and Recognition in Uncontrolled Environments*," International Journal of Multimedia Data Engineering and Management*, vol. 17, no. 6, jan, 2015.

[40] E. Vazquez, "Técnicas De Visión Artificial Robustas En Entornos No Controlados," *IEEE J. Biomed. Heal. Informatics*, Tesis Doctoral, Departamento de Teoría de la Señal y Comunicaciones, Universidad de Vigo, 2015

[41] H. Ben, S. Bouguezzi and C. Souani, "Face recognition in unconstrained environment with CNN," *Vis Comput*. 2020. https://doi.org/10.1007/s00371-020-01794-9

[42] B. Kwolek and M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Comput. Methods Programs Biomed.*, vol. 117, no. 3, pp. 489–501, 2014, doi: 10.1016/j.cmpb.2014.09.005.

[43] I. Charfi, J. Miteran, J. Dubois, M. Atri, and R. Tourki, "Optimized spatio-temporal descriptors for real-time fall detection: comparison of support vector machine and Adaboost-based classification," *J. Electron. Imaging*, vol. 22, no. 4, Jul. 2013, doi: 10.1117/1.JEI.22.4.041106.

[44] Computer Vision Department (COMVIS) of MICA International Research Institute (MICA) and Posts & Telecommunications Institute of Technology (PTIT), "Continuous Multimodal Multi-view Dataset of Human Fall (CMDFALL)." https://www.mica.edu.vn/perso/Tran-Thi-Thanh-Hai/CMDFALL.html (accessed Dec. 20, 2020).

[45] L. Martínez-Villaseñor, H. Ponce, J. Brieva, E. Moya-Albor, J. Núñez-Martínez, and C. Peñafort-Asturiano, "UP-Fall Detection Dataset: A Multimodal Approach," *Sensors*, vol. 19, no. 9, Apr. 2019, doi: 10.3390/s19091988.

[46] E. Auvinet, C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Multiple cameras fall dataset," *Université de Montréal*. http://www.iro.umontreal.ca/~labimage/Dataset/ (accessed

Dec. 20, 2020).

[47] K. Soomro, A. Roshan, and M. Shah, “UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild,” *Comput. Vis. Pattern Recognition.*, 2012.

[48] J. Barbosa, C. Barbosa, and M. Rodríguez, “Revision y análisis documental para estado del arte: una propuesta metodológica desde el contexto de la sistematización de experiencias educativas.” Investigación Bibliotecológica, 27(61), 83−105. 2013.

[49] J.C. Eraso, E. Muñoz, M. Muñoz, J. Pinto, “Dataset for human fall recognition in an uncontrolled environment”, *Data in Brief*, Available online 17 September 2022. doi: https://doi.org/10.1016/j.dib.2022.108610

[50] M. Mehedi, Z. Uddin, A. Mohamed and A. Almogren, “A robust human activity recognition system using smartphone sensors and deep learning,” Future Generation Computer Systems., vol. (81), pp. 307–313. 2018.

[51] Z. A. Khan and W. Sohn, “Abnormal human activity recognition system based on R-transform and kernel discriminant technique for elderly home care,” *IEEE Trans. Consum. Electron.*, vol. 57, no. 4, pp. 1843–1850, 2011, doi: 10.1109/TCE.2011.6131162.

[52] X. Ma, H. Wang, B. Xue, M. Zhou, B. Ji, and Y. Li, “Depth-based human fall detection via shape features and improved extreme learning machine,” *IEEE J. Biomed. Heal. Informatics*, vol. 18, no. 6, pp. 1915–1922, 2014, doi: 10.1109/JBHI.2014.2304357.

[53] L. Yang, Y. Ren, and W. Zhang, “3D depth image analysis for indoor fall detection of elderly people,” *Digit. Commun. Networks*, vol. 2, no. 1, pp. 24–34, 2016, doi: 10.1016/j.dcan.2015.12.001.

[54] S. Rosati, G. Balestra, and M. Knaflitz, “Comparison of Different Sets of Features for Human Activity Recognition by Wearable Sensors,” *Sensors*, vol. 18, no. 12, Nov. 2018, doi: 10.3390/s18124189.

[55] L. Panahi and V. Ghods, “Human fall detection using machine vision techniques on RGB–D images,” *Biomed. Signal Process. Control*, vol. 44, pp. 146–153, 2018, doi: 10.1016/j.bspc.2018.04.014.

[56] G. Goudelis, G. Tsatiris, K. Karpouzis, and S. Kollias, “Fall detection using history triple features,” *8th ACM Int. Conf. PErvasive Technol. Relat. to Assist. Environ. PETRA 2015 - Proc.*, 2015, doi: 10.1145/2769493.2769562.

[57] S. Zhang, Z. Wei, J. Nie, L. Huang, S. Wang, and Z. Li, “A Review on Human Activity Recognition Using Vision-Based Method,” *J. Healthc. Eng.*, vol. 2017, 2017, doi: 10.1155/2017/3090343.

[58] D. Das Dawn and S. H. Shaikh, “A comprehensive survey of human action recognition with spatio-temporal interest point (STIP) detector,” *Vis. Comput.*, vol. 32, no. 3, pp. 289–306, 2016, doi: 10.1007/s00371-015-1066-2.

[59] M. Yu, S. M. Naqvi, A. Rhuma, and J. Chambers, “One class boundary method classifiers for application in a video-based fall detection system,” *IET Comput. Vis.*, vol. 6, no. 2, pp. 90–100, 2012, doi: 10.1049/iet-cvi.2011.0046.

[60] H. Foroughi, B. S. Aski, and H. Pourreza, “Intelligent video surveillance for monitoring fall detection of elderly in home environments,” *Proc. 11th Int. Conf. Comput. Inf. Technol. ICCIT 2008*, no. Iccit, pp. 219–224, 2008, doi: 10.1109/ICCITECHN.2008.4803020.

[61] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, “Fall detection from human shape and

motion history using video surveillance," *Proc. - 21st Int. Conf. Adv. Inf. Netw. Appl. Work. AINAW'07*, vol. 1, pp. 875–880, 2007, doi: 10.1109/AINAW.2007.181.

[62] E. Auvinet, F. Multon, A. Saint-Arnaud, J. Rousseau, and J. Meunier, "Fall detection with multiple cameras: An occlusion-resistant method based on 3-D silhouette vertical distribution," *IEEE Trans. Inf. Technol. Biomed.*, vol. 15, no. 2, pp. 290–300, 2011, doi: 10.1109/TITB.2010.2087385.

[63] V. A. Nguyen, T. H. Le, and T. T. Nguyen, "Single camera based fall detection using motion and human shape features," *ACM Int. Conf. Proceeding Ser.*, vol. 08-09-Dece, pp. 339–344, 2016, doi: 10.1145/3011077.3011103.

[64] A. A. Efros, A. C. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2, pp. 726–733, 2003.

[65] C. Harris and M. Stephens, "A combined edge and corner detector," *Proc 4th Alvey Vis. Conf.*, pp. 147–152, 1988.

[66] I. Laptev and T. Lindeberg, "Space-time Interest Points," *Comput. Vis. Act. Percept. Lab.*, pp. 1–7, 2003.

[67] B. Chakraborty, M. B. Holte, T. B. Moeslund, and J. Gonzàlez, "Selective spatio-temporal interest points," *Comput. Vis. Image Underst.*, vol. 116, no. 3, pp. 396–410, 2012, doi: 10.1016/j.cviu.2011.09.010.

[68] I. Laptev, "On space-time interest points," *Int. J. Comput. Vis.*, vol. 64, no. 2–3, pp. 107–123, 2005, doi: 10.1007/s11263-005-1838-7.

[69] T. V. Nguyen, Z. Song, and S. Yan, "STAP: Spatial-temporal attention-aware pooling for action recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 1, pp. 77–86, 2015, doi: 10.1109/TCSVT.2014.2333151.

[70] S. J. Berlin and M. John, "Human interaction recognition through deep learning network," *Proc. - Int. Carnahan Conf. Secur. Technol.*, pp. 1–4, 2017, doi: 10.1109/CCST.2016.7815695.

[71] S. Venkatesha and M. Turk, "Human activity recognition using local shape descriptors," *Proc. - Int. Conf. Pattern Recognit.*, pp. 3704–3707, 2010, doi: 10.1109/ICPR.2010.902.

[72] X. Peng, L. Wang, X. Wang, and Y. Qiao, "Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice," *Comput. Vis. Image Underst.*, vol. 150, pp. 109–125, 2015, doi: 10.1016/j.cviu.2016.03.013.

[73] Y. Zhu, X. Zhao, Y. Fu, and Y. Liu, "Sparse coding on local spatial-temporal volumes for human action recognition," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6493 LNCS, no. PART 2, pp. 660–671, 2011, doi: 10.1007/978-3-642-19309-5_51.

[74] H. B. Zhang *et al.*, "A comprehensive survey of vision-based human action recognition methods," *Sensors (Switzerland)*, vol. 19, no. 5, pp. 1–20, 2019, doi: 10.3390/s19051005.

[75] J. Shotton *et al.*, "Real-time human pose recognition in parts from single depth images," Jun. 2011, doi: 10.1109/CVPR.2011.5995316.

[76] R. Planinc and M. Kampel, "Introducing the use of depth data for fall detection," *Pers. Ubiquitous Comput.*, vol. 17, no. 6, pp. 1063–1072, 2013, doi: 10.1007/s00779-012-0552-z.

[77] W. Yang, L., Ren, Y. and Zhang, "A hierarchical abnormal human door fall detection of elderly people," *Digit. Commun. Networks.*, vol. 2, no. 1, pp. 24–34, 2016.

[78] Y. Nizam, M. N. H. Mohd, and M. M. A. Jamil, “Human Fall Detection from Depth Images using Position and Velocity of Subject,” *Procedia Comput. Sci.*, vol. 105, no. Iris 2016, pp. 131–137, 2017, doi: 10.1016/j.procs.2017.01.191.

[79] G. Mastorakis and D. Makris, “Fall detection system using Kinect’s infrared sensor,” *J. Real-Time Image Process.*, vol. 9, no. 4, pp. 635–646, 2014, doi: 10.1007/s11554-012-0246-9.

[80] L. Yao, W. Min, and K. Lu, “A new approach to fall detection based on the human torso motion model,” *Appl. Sci.*, vol. 7, no. 10, 2017, doi: 10.3390/app7100993.

[81] O. Paper, A. Jalal, J. T. Kim, and T. Kim, “Indoor and B uilt Recognition of Human Home Activities via Depth Silhouettes and R Transformation for Smart,” pp. 184–190, 2012, doi: 10.1177/1420326X11423163.

[82] Y. Du, W. Wang, and L. Wang, “Hierarchical recurrent neural network for skeleton based action recognition,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 07-12-June, pp. 1110–1118, 2015, doi: 10.1109/CVPR.2015.7298714.

[83] A. S. Keçeli and A. B. Can, “Recognition of basic human actions using depth information,” *Int. J. Pattern Recognit. Artif. Intell.*, vol. 28, no. 2, pp. 1–21, 2014, doi: 10.1142/S0218001414500049.

[84] H. Pazhoumand-Dar, C. P. Lam, and M. Masek, “Joint movement similarities for robust 3D action recognition using skeletal data,” *J. Vis. Commun. Image Represent.*, vol. 30, pp. 10–21, 2015, doi: 10.1016/j.jvcir.2015.03.002.

[85] X. Yang and Y. Tian, “Effective 3D action recognition using EigenJoints,” *J. Vis. Commun. Image Represent.*, vol. 25, no. 1, pp. 2–11, 2014, doi: 10.1016/j.jvcir.2013.03.001.

[86] Y. Hbali, S. Hbali, L. Ballihi, and M. Sadgal, “Skeleton-based human activity recognition for elderly monitoring systems,” *IET Comput. Vis.*, vol. 12, no. 1, Feb. 2018, doi: 10.1049/iet-cvi.2017.0062.

[87] B. Ni, Y. Pei, P. Moulin, and S. Yan, “Multilevel depth and image fusion for human activity detection,” *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1382–1394, 2013, doi: 10.1109/TCYB.2013.2276433.

[88] A. Jalal, Y. Kim, Y. Kim, S. Kamal, and D. Kim, “Robust human activity recognition from depth video using spatiotemporal multi-fused features,” *Pattern Recognit.*, vol. 61, pp. 295–308, 2017, doi: 10.1016/j.patcog.2016.08.003.

[89] L. Liu and L. Shao, “Learning discriminative representations from RGB-D video data,” *IJCAI Int. Jt. Conf. Artif. Intell.*, pp. 1493–1500, 2013.

[90] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), pp. 2278–2324, 1998

[91] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, C. Berg and L. Fei-Fei."ImageNet Large Scale Visual Recognition Challenge". IJCV, 2015.

[92] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database. IEEE Computer Vision and Pattern Recognition (CVPR), 2009.

[93] A. Verma, P. Singh, y J. S. Rani Alex, «Modified Convolutional Neural Network Architecture Analysis for Facial Emotion Recognition», en 2019 International Conference on Systems, Signals and Image Processing (IWSSIP), jun. 2019, pp. 169-173, doi: 10.1109/IWSSIP.2019.8787215.

[94] Y.-Z. Hsieh and Y.-L. Jeng, “Development of Home Intelligent Fall Detection IoT System Based on Feedback Optical Flow Convolutional Neural Network,” *IEEE Access*, vol. 6, 2018, doi: 10.1109/ACCESS.2017.2771389.

[95] S. Yan, Y. Xiong, and D. Lin, “Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition,” *Comput. Vis. Pattern Recognit.*, 2018.

[96] Q. Xu, G. Huang, M. Yu, and Y. Guo, “Fall prediction based on key points of human bones,” *Phys. A Stat. Mech. its Appl.*, vol. 540, Feb. 2020, doi: 10.1016/j.physa.2019.123205.

[97] L. Wang, Y. Qiao, and X. Tang, “Action Recognition with Trajectory-Pooled Deep-Convolutional Descriptors,” 2015.

[98] A. Núñez-Marcos, G. Azkune, and I. Arganda-Carreras, “Vision-Based Fall Detection with Convolutional Neural Networks,” *Wirel. Commun. Mob. Comput.*, vol. 2017, 2017, doi: 10.1155/2017/9474806.

[99] R. Espinosa, H. Ponce, S. Gutiérrez, L. Martínez-Villaseñor, J. Brieva, and E. Moya-Albor, “A vision-based approach for fall detection using multiple cameras and convolutional neural networks: A case study using the UP-Fall detection dataset,” *Comput. Biol. Med.*, vol. 115, Dec. 2019, doi: 10.1016/j.compbiomed.2019.103520.

[100] M. Rahnemoonfar and H. Alkittawi, “SPATIO-TEMPORAL CONVOLUTIONAL NEURAL NETWORK FOR ELDERLY FALL DETECTION IN DEPTH VIDEO CAMERAS,” Dec. 2018, doi: 10.1109/BigData.2018.8622342.

[101] K. Adhikari, H. Bouchachia, and H. Nait-Charif, “Activity recognition for indoor fall detection using convolutional neural network,” May 2017, doi: 10.23919/MVA.2017.7986795.

[102] N. Lu, X. Ren, J. Song, and Y. Wu, “Visual guided deep learning scheme for fall detection,” Aug. 2017, doi: 10.1109/COASE.2017.8256202.

[103] C. Ma, A. Shimada, H. Uchiyama, H. Nagahara, and R. Taniguchi, “Fall detection using optical level anonymous image sensing system,” *Opt. Laser Technol.*, vol. 110, Feb. 2019, doi: 10.1016/j.optlastec.2018.07.013.

[104] C. Khraief, F. Benzarti, and H. Amiri, “Elderly fall detection based on multi-stream deep convolutional networks,” *Multimed. Tools Appl.*, vol. 79, no. 27–28, Jul. 2020, doi: 10.1007/s11042-020-08812-x.

[105] I. Sreenidhi, “Real-Time Human Fall Detection and Emotion Recognition using Embedded Device and Deep Learning,” *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 3, Mar. 2020, doi: 10.30534/ijeter/2020/28832020.

[106] X. Cai, X. Liu, S. Li, and G. Han, “Fall Detection Based on Colorization Coded MHI Combining with Convolutional Neural Network,” Oct. 2019, doi: 10.1109/ICCT46805.2019.8947223.

[107] C. Khraief, F. Benzarti, and H. Amiri, “Convolutional Neural Network Based on Dynamic Motion and Shape Variations for Elderly Fall Detection,” *Int. J. Mach. Learn. Comput.*, vol. 9, no. 6, Dec. 2019, doi: 10.18178/ijmlc.2019.9.6.878.

[108] X. Li, T. Pang, W. Liu, and T. Wang, “Fall detection for elderly person care using convolutional neural networks,” Oct. 2017, doi: 10.1109/CISP-BMEI.2017.8302004.

[109] A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional

neural networks," *Proc. 26th Annu. Conf. Neural Inf. Process. Syst. Lake Tahoe, Nev, USA, 2012.*, 2012.

[110] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Jun. 2016, doi: 10.1109/CVPR.2016.90.

[111] A. El Kaid, K. Baïna, and J. Baïna, "Reduce False Positive Alerts for Elderly Person Fall Video-Detection Algorithm by convolutional neural network model," *Procedia Comput. Sci.*, vol. 148, pp. 2–11, 2019, doi: 10.1016/j.procs.2019.01.004.

[112] G. Debard *et al.*, "Camera-based fall detection using real-world versus simulated data: How far are we from the solution?," *J. Ambient Intell. Smart Environ.*, vol. 8, no. 2, Mar. 2016, doi: 10.3233/AIS-160369.

[113] Y. Fan, M. D. Levine, G. Wen, and S. Qiu, "A deep neural network for real-time detection of falling humans in naturally occurring scenes," *Neurocomputing*, vol. 260, Oct. 2017, doi: 10.1016/j.neucom.2017.02.082.

[114] J. Redmon, S. Divvala, R. Girshick y A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", Computer Vision and Pattern Recognition, 2015.

[115] J. Redmon, A. Farhadi, " YOLO9000: Better, Faster, Stronger", Computer Vision and Pattern Recognition, 2016.

[116] J. Redmon, A. Farhadi, "YOLOv3: An Incremental Improvement", Computer Vision and Pattern Recognition, 2018.

[117] A. Bochkovskiy, C.Yao, H.Yuan, " YOLOv4: Optimal Speed and Accuracy of Object Detection", Computer Vision and Pattern Recognition, 2020.

[118] B. Cruz Angel, "Seguimiento Y Evaluación De Personas En Ambientes Cerrados / Abiertos", Tesis de grado, Universidad del Rosario, 2021.

[119] N. Wojke, A. Bewley, and D. Paulus, " Simple Online and Realtime Tracking with a Deep Association Metric," Computer Vision and Pattern Recognition, 21 Mar 2017. https://doi.org/10.48550/arXiv.1703.07402

[120] A. Regal, J. Morzan, C. Fabbri, G. Herrera, "Proyección del precio de criptomonedas basado en Tweets empleando LSTM," Revista chilena de ingeniería, vol. 27 Nº 4, pp. 696-706, 2019.

[121] Z. Cao, G. Hidalgo, T. Simon, S.Wei, and Y.Sheikh, " OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields", IEEE Transactions On Pattern Analysis And Machine Intelligence, 2019.

[122] Google Colaboratory, "bienvenida a Google Colaboratory ", Entorno de ejecucion, acceso noviembre 2022. https://colab.research.google.com/?utm_source=scs-index

[123] S. Nwe, T. Zin and P. Tin, " Image Processing Technique and Hidden Markov Model for an Elderly Care Monitoring System*,"J. Imaging 2020*, 6, 49, 2020. doi:10.3390/jimaging6060049

[124] Q. Feng, C. Gao, L. Wang, Y. Zhao, T. Song, Q. Li, "Spatio-temporal fall event detection in complex scenes using attention guided LSTM" , *Pattern Recognition Letters (2020*), 2020. doi: https://doi.org/10.1016/j.patrec.2018.08.031

[125] Q. Xu, G. Huang, M. Yu, Y. Guo" Fall prediction based on key points of human bones," *Physica A,* 2020. https://doi.org/10.1016/j.physa.2019.123205

[126] J. Gutiérrez, V. Rodríguez and S. Martin, "Comprehensive Review of Vision-Based Fall Detection

Systems," *Sensors 2021*, 21, 947. 2011. https://doi.org/10.3390/s21030947

[127] K. Adhikari, H. Bouchachia, H. Nait-Charif, "Activity recognition for indoor fall detection using convolutional neural network," *In Proceedings of the 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA),* Nagoya, Japan, 8–12 May 2017.

[128] CENTRE FOR DIGITAL HOME—MMU. Available online: http://foe.mmu.edu.my/digitalhome/FallVideo.zip (accessed on 27 January 2021).

[129] MOT Dataset. Available online: https://motchallenge.net/ (accessed on 27 January 2021).

[130] T. Yi, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. Lawrence, P. Dollár, "Microsoft COCO: Common Objects in Context", *Computer Vision and Pattern Recognition*, 2015. https://doi.org/10.48550/arXiv.1405.0312

[131] A. Shahroudy, J. Liu, T. Ng, G. Wang, "NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis," *In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 27–30 June 2016; pp. 1010–1019, 2016.

[132] F. Harrou, N. Zerrouki, Y. Sun, A. Houacine, "Vision-based fall detection system for improving safety of elderly people," IEEE Instrum. Meas. Mag. 2017, 20, 49–55. 2017.

[133] S. Ali, R. Khan, A. Mahmood, M. Hassan, M. Jeon, "Using Temporal Covariance of Motion and Geometric Features via Boosting for Human Fall Detection," *Sensors 2018*, 18, 1918. 2018.

[134] D. Kumar, A. Ravikumar, V. Dharmalingam, V. Kafle,"Elderly Health Monitoring System with Fall Detection Using Multi-Feature Based Person Tracking," *In Proceedings of the 2019 ITU Kaleidoscope*: ICT for Health: Networks, Standards and Innovation (ITU K), Atlanta, GA, USA, 4–6 December 2019.

[135] F. Harrou, N. Zerrouki, Y. Sun, A. Houacine, "An Integrated Vision-Based Approach for Efficient Human Fall Detection in a Home Environment," IEEE Access 2019, 7, 114966–114974. 2019.

[136] Y. Fan, M. Levine, G. Wen, S. Qiu, "A deep neural network for real-time detection of falling humans in naturally occurring scenes," Neurocomputing 2017, 260, 43–58.2017.

[137] W. Min, H. Cui, H. Rao, Z. Li, L. Yao, "Detection of Human Falls on Furniture Using Scene Analysis Based on Deep Learning and Activity Characteristics," IEEE Access 2018, 6, 9324–9335.

[138] C. Ma, A. Shimada, H. Uchiyama, H. Nagahara, R. Taniguchi, "Fall detection using optical level anonymous image sensing system," *Opt. Laser Technol*, 110, 44–61. 2019.

[139] R. Espinosa, H. Ponce, S. Gutiérrez, L. Martínez-Villaseñor, J. Brieva, E. Moya-Albor, "A vision-based approach for fall detection using multiple cameras and convolutional neural networks: A case study using the UP-Fall detection dataset," *Comput. Biol. Med*. 2019, 115, 103520. 2020.

[140] B. Wang, J. B, J. Yu, K. Wang, Bao,K. Mao, "Fall Detection Based on Dual-Channel Feature Integration," IEEE Access 2020, 8, 103443–103453. 2020.

[141] S. Nwe, T. Zin and H. Hama, "Virtual Grounding Point Concept for Detecting Abnormal and Normal Events in Home Care Monitoring Systems," *Appl. Sci. 2020*, 10, 3005, 2020. doi:10.3390/app10093005

[142] P. Mehta, A. Lee, C. Lee, M. Balazinska, A. Rokem, "Multilabel multiclass classification of OCT images augmented with age, gender and visual acuity data," *bioRxiv.* 2018. doi:10.1101/316349

[143] X. Ma, H. Wang, B. Xue, M. Zhou, B. Ji, Y. Li Y, "Depth-based human fall detection via shape features and improved extreme learning machine*," IEEE J Biomed Health Inform,* 18(6):1915–

1922. 2014. https://doi.org/10.1109/JBHI.2014.2304357.

[144] Z. Zhang, C. Conly, and V. Athitsos, "Evaluating depth-based computer vision methods for fall detection under occlusions*," In International Symposium on Visual Computing*, pages 196–207. Springer, 2014.

[145] M. Aslan, Y. Akbulut, A.Sengur, "Skeleton based efficient fall detection," *Journal of the Faculty of Engineering and Architecture of Gazi University*, 32(4):1025-1034, 2016.

[146] K. Adhikari, B. Hamid and N. Hammadi, "Activity recognition for indoor fall detection using convolutional neural network," *Machine Vision Applications (MVA),* 2017 Fifteenth IAPR International Conference on. IEEE, 2017.

[147] HIKVISION, DS-2CE16D0T-IRP, https://www.hikvision.com/en/products/Turbo-HD-Products/Turbo-HD-Cameras/Value-Series/ds-2ce16d0t-irp-c-/ (accessed 11 March 2022).

[148] HIKVISION, DS-7104HGHI-F1, https://hikvisioncolombia.com/producto/mini-dvr-4-canales-hikvision-turbo-hd-720p-ds7104hghif1/ (accessed 11 March 2022).

[149] A. Bochkovskiy, "darknet", G*ithub, available on 23 oct 2022.* https://github.com/AlexeyAB/darknet

[150] CMU-Perceptual-Computing-Lab, "OpenPose", Github, *available on 23 oct 2022. https://github.com/CMU-Perceptual-Computing-Lab/openpose*

[151] J. Murillo, "Comparison between recursive and Iterative algorithms, and its measurement in terms of efficiency", *UNICIENCIA,* Vol. 27, No. 1, [341-350]. 2013.