

CENTRO DE INVESTIGACIÓN EN INTELIGENCIA ARTIFICIAL

**Mejorar el reconocimiento de acciones y caídas  
utilizando algoritmos de selección de características.**

T E S I S

Que para obtener el Grado de:

Doctora en Inteligencia Artificial

Presenta

Carolina Gabriela Maldonado Méndez

Maestra en Inteligencia Artificial

Director de tesis:

Dr. Homero Vladimir Ríos Figueroa

Xalapa-Enríquez, Veracruz, México, 2018

# Contenido

Resumen.....	4
Abstract .....	5
Acrónimos .....	9
Capítulo 1: Introducción.....	10
1.1. Planteamiento de Problema .....	11
1.2. Hipótesis.....	11
1.3. Objetivos .....	12
1.3.1. Objetivo General .....	12
1.3.2. Objetivos Particulares .....	12
1.4. Contribución.....	12
1.5. Publicaciones.....	14
1.6. Estructura de la tesis .....	14
Capítulo 2: Trabajos relacionados .....	15
2.1. Reconocimiento de acciones.....	15
2.2. Reconocimiento de caídas .....	22
2.3. Clasificadores .....	31
2.4. Selección de características.....	33
2.5. Resumen.....	38
Capítulo 3: Reconocimiento de acciones y caídas .....	39
3.1. Captura de información Visual.....	40
3.2. Selección de características.....	42
3.2.1. Conjunto de características.....	43
3.2.2. Algoritmo Genético .....	46

3.2.3.	Algoritmo RFD-SD.....	50
3.2.4.	PCA .....	54
3.3.	Reconocimiento de caída y acciones .....	55
3.3.1.	Reconocimiento de caída .....	55
3.3.2.	Reconocimiento de acciones.....	59
3.4.	Resumen.....	62
Capítulo 4: Experimentos y Resultados.....		63
4.1.	Reconocimiento de Caídas .....	63
4.1.1.	Experimento 1 .....	65
4.1.2.	Experimento 2 .....	73
4.1.3.	Experimento 3 .....	78
4.1.4.	Experimento 4 .....	84
4.1.5.	Discusión de los Resultados .....	93
4.2.	Experimentos de Reconocimiento de Acciones.....	94
4.2.1.	Experimento 1 .....	95
4.2.2.	Experimento 2 .....	102
4.2.3.	Discusión de los Resultados .....	107
Capítulo 5: Conclusiones y Trabajo Futuro .....		109
Bibliografía .....		111

## Resumen

Durante los últimos años se ha desarrollado investigación en reconocimiento automático de acciones. En los trabajos revisados, los conjuntos de características son definidos mediante estrategias diferentes. En esta tesis se seleccionaron características con un algoritmo genético, el algoritmo RFD-SD propuesto en esta tesis y el análisis de componentes principales, se comparó y evaluó cuáles conjuntos de características seleccionadas incrementaron la exactitud del reconocimiento de acciones y caídas con diferentes clasificadores y datos de pruebas.

A continuación se mencionan las principales contribuciones de esta tesis:

- a) Se trabajó con un conjunto de atributos que contiene características utilizadas en los trabajos revisados, así como los invariantes 3D a traslación, rotación y escala (TRE). Las características fueron extraídas a partir de información visual adquirida mediante cámaras de profundidad.
- b) Se propuso el algoritmo *Reduction of Feature Dimensions based on Standard Deviation* (RFD-SD) para seleccionar características en series de tiempo multi-características. La clasificación se realizó con el algoritmo *K* vecinos más cercanos (KNN) y con *Dynamic Time Warping* (DTW) se midió la similitud entre series de tiempo. Las bases de acciones públicas UTD-MHAD y KARD fueron utilizadas en los experimentos, la exactitud del reconocimiento obtenida fue de 76.6% y 94.2%, respectivamente. Los resultados reportados en el estado del arte son 66.1% (Chen, Jafari, & Kehtarnavaz, 2015) con la base UTD-MHAD y 88.3% (Gaglio, Re, & Morana, 2015) con la base KARD.
- c) Se seleccionaron características con un algoritmo genético (AG) para reconocer posturas de caídas. Las características seleccionadas se utilizaron con los clasificadores: K-medias, Modelo Generativo Gaussiano (MGG) y Máquina de Soporte Vectorial (SVM). Las bases *UR Fall Detection Dataset* y *TST Fall detection dataset v2* fueron utilizadas en los experimentos, los resultados sugieren que independientemente del clasificador y bases de datos utilizados; las características seleccionadas con el AG incrementaron la exactitud del reconocimiento, en comparación con los resultados obtenidos cuando se utilizaron las características seleccionadas con PCA.
- d) Se evaluaron los filtros de Kalman y Savitsky-Golay con el propósito de minimizar el ruido en las articulaciones del esqueleto y se determinó que el filtro de Kalman incrementa la exactitud de reconocimiento. Con la base de acciones MSR-Action3D se obtuvo una exactitud del reconocimiento de 83% cuando se utilizó el filtro de Kalman y se seleccionaron características con el algoritmo RFD-SD, este resultado supera al reportado en (Climent-Pérez, Chaaraoui, Padilla-López, & Flórez-Revuelta, 2012) para un experimento realizado en condiciones similares.

## Abstract

During the last years several research works have been developed about the automatic recognition of actions. In the reviewed works, the features are selected by different strategies. In this thesis the characteristics were selected with a genetic algorithm, the RFD-SD algorithm proposed in this thesis and the principal components analysis. The sets of selected characteristics were compared and evaluated in order to discover which of them increased the accuracy of the recognition of actions and falls with different classifiers and data sets.

The contributions of this work are the following:

- a) A set of characteristics that contains characteristics used in the reviewed works, as well as the 3D invariants in translation, rotation and scale (TRE) was defined. The characteristics were extracted from visual information, which was acquired by depth cameras.
- b) To recognize actions, the Reduction of Feature Dimensions based on Standard Deviation (RFD-SD) algorithm was proposed to select characteristics in time series with more than one feature. The K nearest neighbors (KNN) algorithm and Dynamic Time Warping (DTW) were used to classify and measure the similarity. The experiments were carried out with the public actions datasets UTD-MHAD and KARD, the accuracy of recognition were 76.6% and 94.2% respectively. The results reported in the state of the art were 66.1% (Chen, Jafari, & Kehtarnavaz, 2015) for the dataset UTD-MHAD and 88.3% (Gaglio, Re, & Morana, 2015) for the dataset KARD.
- c) To recognize falls, the characteristics were selected by means of a genetic algorithm (AG). The selected characteristics were used with the classifiers: K-means, Gaussian Generative Model (MGG) and Support Vector Machine (SVM). The experiments were carried out with the datasets UR Fall Detection Dataset and TST Fall detection dataset v2. The results of the experiments suggest that regardless of the classifier and datasets used; the characteristics selected by the AG increased the accuracy of the recognition, compared to the results of using the characteristics selected by PCA.
- d) The Kalman and Savitsky-Golay filters were evaluated to eliminate noise in the joints of the skeleton. Subsequently, it is proved Kalman filter contributed in improving the accuracy of the classification. A recognition accuracy of 83% was obtained using the MSR-Action3D action base, eliminating noise with the Kalman filter and selecting the characteristics with the RFD-SD algorithm; this result improves the one reported by (Climent-Pérez, Chaaoui, Padilla-López, & Flórez-Revuelta, 2012) in an experiment carried out in a similar way.

## Índice de Figuras

FIGURA 1. FASES DE ENTRENAMIENTO Y PRUEBA PARA RECONOCER ACCIONES Y CAÍDAS.....	39
FIGURA 2. A) CÁMARA ESTÉREO, B) TIEMPO DE VUELO (ToF) Y C) DISPOSITIVOS DE LUZ ESTRUCTURADA.....	41
FIGURA 3 LAS IMÁGENES MOSTRADAS EN LA PARTE SUPERIOR FORMAN PARTE DE LA BASE TST FALL DETECTION DATASET V2, LOS PUNTOS ROJOS CORRESPONDEN A LAS POSICIONES DE LAS ARTICULACIONES DEL ESQUELETO. EN LA PARTE INFERIOR SE MUESTRAN LAS NUBES DE PUNTOS DE LA PERSONA DETECTADA. ....	41
FIGURA 4. LA SEGMENTACIÓN DE LA NUBE DE PUNTOS DE UNA PERSONA ES EL RESULTADO DE UNA DIFERENCIA DE IMÁGENES. .	42
FIGURA 5. EN LA IMAGEN SE MUESTRA EL <i>BOUNDING BOX</i> DE LA NUBE DE PUNTOS DE UNA PERSONA ASÍ COMO LOS PUNTOS P1, P2 Y P3 UTILIZADOS EN EL CÁLCULO DE DIECISIETE CARACTERÍSTICAS.....	43
FIGURA 6. ARTICULACIONES DEL ESQUELETO.....	45
FIGURA 7 EL MATERIAL GENÉTICO LOCALIZADOS ENTRE LOS PUNTOS DE CRUZA P1 Y P2 DE LOS PADRES SE INTERCAMBIA PARA GENERAR DOS HIJOS. ....	48
FIGURA 8. EN LA IMAGEN SE MUESTRAN TRES TIPOS DE MUTACIÓN: UN PUNTO, VARIOS PUNTOS Y GLOBAL.....	49
FIGURA 9. EN LAS IMÁGENES DE PROFUNDIDAD UNA PERSONA CAE AL PISO. ....	55
FIGURA 10. EN LAS IMÁGENES DE PROFUNDIDAD LA PERSONA CAMINA. ....	56
FIGURA 11. LA GRÁFICA SUPERIOR PRESENTA LA MISMA SEÑAL EN DIFERENTES VELOCIDADES. EN LA GRÁFICA DE ABAJO SE MUESTRAN LAS SEÑALES ALIENADAS. ....	60
FIGURA 12. EN LA GRÁFICA SUPERIOR SE MUESTRA EL MOVIMIENTO DEL ÁNGULO ENTRE LA ARTICULACIÓN DE LA CABEZA Y DEL CUELLO PARA LA ACCIÓN DE ONDULAR EL BRAZO EN EL EJE HORIZONTAL, LA ACCIÓN ES REALIZADA POR DOS PERSONAS S1 Y S2. EN LA GRÁFICA DE ABAJO SE MUESTRAN LAS SEÑALES ALINEADAS, LA SIMILITUD OBTENIDA CON DTW ES DE 80.91....	60
FIGURA 13. EN LA GRÁFICA SUPERIOR SE MUESTRA EL MOVIMIENTO DEL ÁNGULO ENTRE LA ARTICULACIÓN DE LA CABEZA Y DEL CUELLO, UNA PERSONA REALIZA LA ACCIÓN DE ONDULAR EL BRAZO EN EL EJE HORIZONTAL Y LA SEGUNDA PERSONA REALIZA LA ACCIÓN DE SENTARSE. EN LA GRÁFICA DE ABAJO SE MUESTRAN LAS SEÑALES ALINEADAS, LA SIMILITUD OBTENIDA CON DTW ES DE 174.21 .....	61
FIGURA 14. PARA CADA CLASE SE ALMACENAN M SECUENCIAS PATRONES. SE CREA UN REPOSITORIO R PARA CADA CLASE. LA SECUENCIA A RECONOCER SE COMPARA CON CADA UNA DE LAS SECUENCIAS EN LOS REPOSITORIOS Y SE CALCULA LA PROBABILIDAD DE PERTENENCIA A CADA CLASE. ....	62
FIGURA 15. CON UNA CRUZ '+' SE REPRESENTA UNA POSTURA DE CAÍDA, SU NORMA DEL VECTOR DE CARACTERÍSTICAS SELECCIONADAS CON EL AG TIENE UN VALOR MENOR DE 2.5. CON UN UMBRAL DE 1.7 SE DISTINGUE ENTRE UNA POSTURA DE CAÍDA Y UNA DE NO CAÍDA.....	68
FIGURA 16. CON UNA CRUZ '+' SE REPRESENTA UNA POSTURA DE CAÍDA. EN LA GRÁFICA SE OBSERVA QUE ES IMPOSIBLE ESTABLECER UN UMBRAL PARA DIFERENCIAR ENTRE UNA POSTURA DE CAÍDA Y UNA DE NO CAÍDA. ....	68
FIGURA 17. LA PERSONA CAMINA Y CAE AL PISO. EL VALOR DE LA NORMA DEL VECTOR DE CARACTERÍSTICAS SELECCIONADAS POR EL AG EN LAS IMÁGENES CON POSTURA DE CAÍDA ES MENOR QUE 1.7, POR LO TANTO LA CAÍDA SE DETECTA CORRECTAMENTE. ....	69
FIGURA 18. LA PERSONA CAMINA Y CAE AL SUELO. EN LA GRÁFICA SE OBSERVA QUE ES IMPOSIBLE DEFINIR UN UMBRAL PARA DIFERENCIAR ENTRE LAS POSTURAS DE CAÍDA Y LAS POSTURAS DE NO CAÍDA.....	70

FIGURA 19. LA PERSONA CAMINA Y SE PONE EN CUCLILLAS. LA NORMA DEL VECTOR DE CARACTERÍSTICAS SELECCIONADAS CON EL AG EN TODAS LAS IMÁGENES ES MAYOR QUE 1.7, POR TANTO NO SE DETECTA UNA POSTURA DE CAÍDA, LO CUAL ES CORRECTO. ....	71
FIGURA 20. LA PERSONA CAMINA Y SE PONE EN CUCLILLAS. LA NORMA DEL VECTOR DE CARACTERÍSTICAS SELECCIONADAS CON PCA CUANDO LA PERSONA SE PONE EN CUCLILLAS (A PARTIR DE LA IMAGEN 70) ES MENOR DE QUINCE.....	71
FIGURA 21. UNA PERSONA CAMINA Y SE ACUESTA EN EL PISO. LA NORMA DEL VECTOR DE CARACTERÍSTICAS SELECCIONADAS CON EL AG ES MENOR DE 1.7 A PARTIR DE LA IMAGEN 83, POR TANTO CUANDO LA PERSONA SE ACUESTA EN EL SUELO SE DETECTAN POSTURAS DE CAÍDA.....	72
FIGURA 22. EN ROJO SE MUESTRA LA NORMA DEL VECTOR DE CARACTERÍSTICAS SELECCIONADAS CON EL AG EN SECUENCIAS CON TREINTA IMÁGENES CUANDO LA PERSONA CAE Y EN AZUL PARA SECUENCIAS DE TREINTA IMÁGENES CUANDO LA PERSONA SE ACUESTA EN EL PISO. EL CAMBIO DE LA NORMA DEL VECTOR ES MAYOR CUANDO LA PERSONA CAE AL PISO COMPARADO CON EL CAMBIO DE LA NORMA DEL VECTOR CUANDO LA PERSONA SE ACUESTA EN EL PISO. ....	73
FIGURA 23. EL ROBOT SE ENCUENTRA EN EL CENTRO DE INVESTIGACIONES DE INTELIGENCIA ARTIFICIAL. EN LA PARTE SUPERIOR DEL ROBOT SE OBSERVA EL KINECT MONTADO SOBRE UNA PTU. ....	74
FIGURA 24. LA ARQUITECTURA TIENE TRES CAPAS: 1) LA CAPA DE DISPOSITIVOS DE ENTRADA Y SALIDA; 2) LA CAPA DE SEGUIMIENTO DE PERSONA Y RECONOCIMIENTO DE CAÍDAS; 3) LA CAPA DE DETECCIÓN DE ROSTROS.....	74
FIGURA 25. LA SEGUNDA CAPA CUENTA CON DOS MÓDULOS: 1) RECONOCIMIENTO DE PERSONAS Y CAÍDAS Y 2) SEGUIMIENTO DE PERSONAS. ....	75
FIGURA 26. LA PERSONA CAMINA Y CAE AL PISO. EN TODAS LAS SECUENCIAS DETECTAN POSTURAS CUYA NORMA DEL VECTOR DE CARACTERÍSTICAS ES MENOR QUE 1.7, POR TANTO LA CAÍDA SE DETECTA CORRECTAMENTE. ....	76
FIGURA 27. LA PERSONA CAMINA DE UN LADO A OTRO. LA NORMA DEL VECTOR DE CARACTERÍSTICAS EN TODAS LAS IMÁGENES ES MAYOR QUE 1.7. ....	76
FIGURA 28. LA PERSONA CAMINA Y SE SIENTA. LA NORMA DEL VECTOR DE CARACTERÍSTICAS EN TODAS LAS IMÁGENES ES MAYOR QUE 1.7.....	77
FIGURA 29. EN LA IMAGEN DE LA ESQUINA SUPERIOR IZQUIERDA, LA CAPA TRES DETECTA EL ROSTRO DE UNA PERSONA, EN LA IMAGEN SE OBSERVA COMO LA PERSONA ESTÁ FUERA DEL FOCO DE LA IMAGEN. LOS MÓDULOS DE LA CAPA DOS SE ACTIVAN Y SE OBSERVA COMO LA PERSONA ES CENTRADA EN EL PLANO DE LA IMAGEN Y POSTERIORMENTE SE DETECTA LA CAÍDA. ....	77
FIGURA 30. EN LA SECUENCIA LA PERSONA CAMINA Y CAE DE ESPALDAS. EN COLOR ROJO (CLASE UNO) SE MUESTRAN LAS IMÁGENES CON POSTURAS DE UNA PERSONA APROXIMÁNDOSE AL PISO RÁPIDAMENTE Y EN COLOR AZUL (CLASE TRES) CUANDO LA POSTURA CORRESPONDE A UNA PERSONA DE PIE O SENTADA. ....	82
FIGURA 31. EN LA SECUENCIA LA PERSONA CAMINA Y CAE DE FRENTE. EN COLOR ROJO (CLASE UNO) SE MUESTRAN LAS IMÁGENES CON POSTURAS DE UNA PERSONA APROXIMÁNDOSE AL PISO RÁPIDAMENTE, EN COLOR VERDE (CLASE DOS) CUANDO LA POSTURA DE LA PERSONA SE APROXIMA AL PISO LENTAMENTE Y EN COLOR AZUL (CLASE TRES) CUANDO LA POSTURA CORRESPONDE A UNA PERSONA DE PIE O SENTADA.....	82
FIGURA 32. EN LA SECUENCIA LA PERSONA CAMINA Y SE ACUESTA EN EL PISO. EN COLOR ROJO (CLASE UNO) SE MUESTRAN POSTURAS DE UNA PERSONA APROXIMÁNDOSE AL PISO RÁPIDAMENTE, EN COLOR VERDE (CLASE DOS) CUANDO LA POSTURA DE LA PERSONA SE APROXIMA AL PISO LENTAMENTE Y EN COLOR AZUL (CLASE TRES) CUANDO LA POSTURA CORRESPONDE A UNA PERSONA DE PIE O SENTADA. SE CONFIRMA UN EVENTO DE CAÍDA CUANDO LA MAYORÍA DE LAS POSTURAS EN UNA VENTANA DE TREINTA IMÁGENES SON CLASIFICADAS EN LA CLASE UNO.....	83
FIGURA 33. EN LA SECUENCIA LA PERSONA SE SIENTA, TODAS LAS POSTURAS EN LAS IMÁGENES SON CLASIFICADAS EN LA CLASE 3 (COLOR AZUL). ....	83
FIGURA 34. EN LA SECUENCIA LA PERSONA CAMINA DE UN LADO A OTRO, CASI TODAS LAS POSTURAS SON CLASIFICADAS EN LA CLASE 3 (COLOR AZUL) Y SÓLO UNA POSTURA EN LA CLASE 1 (COLOR ROJO), LA CUAL PODRÍA SER RUIDO. ....	84

FIGURA 35. EN LA SECUENCIA UNA PERSONA CAMINA. TODAS LAS POSTURAS SON CLASIFICADAS EN LA CLASE 1 (COLOR ROJO) POR TANTO NO SE DETECTARON POSTURAS DE CAÍDA. ....	87
FIGURA 36. EN LA SECUENCIA LA PERSONA SE SIENTA. TODAS LAS POSTURAS SON CLASIFICADAS EN LA CLASE 1 (COLOR ROJO) POR LO TANTO NO SE DETECTARON POSTURAS DE CAÍDA. ....	87
FIGURA 37. EN LA SECUENCIA LA PERSONA CAE AL PISO. AL PRINCIPIO DE LA SECUENCIA LA PERSONA ESTÁ DE PIE Y SE CLASIFICAN EN LA CLASE 1(COLOR ROJO), A PARTIR DE LA IMAGEN 95, SE DETECTARON POSTURAS DE CAÍDA Y SON CLASIFICADAS EN LA CLASE -1 (COLOR AZUL). EN LA SECUENCIA SE RECONOCE UNA POSIBLE CAÍDA, PERO SE DEBE CONFIRMAR SI LA PERSONA ESTÁ CAYENDO O SE ACUESTA EN EL PISO. ....	88
FIGURA 38. AL COMIENZO DE LA SECUENCIA UNA PERSONA ESTÁ DE PIE Y LAS IMÁGENES SON CLASIFICADAS EN LA CLASE 1 (COLOR ROJO), A PARTIR DE LA IMAGEN 119 SE DETECTARON POSTURAS EN DONDE LA PERSONA SE APROXIMA AL PISO (COLOR AZUL). EN LA SECUENCIA SE RECONOCE UNA POSIBLE CAÍDA, PERO SE DEBE CONFIRMAR SI LA PERSONA ESTÁ CAYENDO O SE ACUESTA EN EL PISO. ....	88
FIGURA 39. EN LA SECUENCIA LA PERSONA CAE AL PISO. EN AZUL SE MUESTRA LA MEDICIÓN DEL CENTROIDE, EN ROJO LA MUÑECA IZQUIERDA, EN MORADO LA MUÑECA DERECHA Y EN VERDE EL UMBRAL. LA GRÁFICA SUPERIOR CORRESPONDE A LA VELOCIDAD Y LA DE ABAJO A LA ACELERACIÓN. ....	90
FIGURA 40. EN LA SECUENCIA LA PERSONA SE ACUESTA EN EL PISO. EN AZUL SE MUESTRA LA MEDICIÓN DEL CENTROIDE, EN ROJO LA MUÑECA IZQUIERDA, EN MORADO LA MUÑECA DERECHA Y EN VERDE EL UMBRAL. LA GRÁFICA SUPERIOR CORRESPONDE A LA VELOCIDAD Y LA DE ABAJO A LA ACELERACIÓN. ....	91
FIGURA 41. EN LAS GRÁFICAS SE MUESTRAN LOS VALORES DE VELOCIDAD Y ACELERACIONES PARA EL CENTROIDE (COLOR AZUL), MUÑECA DERECHA (COLOR ROJO) Y MUÑECA IZQUIERDA (COLOR MORADO) PARA LAS IMÁGENES CLASIFICADAS CON POSTURAS APROXIMÁNDOSE AL SUELO. AL MENOS DOS DE LAS TRES MEDICIONES EN CADA GRÁFICA SON MAYORES AL UMBRAL POR TANTO SE CONFIRMA QUE LA PERSONA CAYÓ AL PISO. ....	92
FIGURA 42. EN LAS GRÁFICAS SE MUESTRAN LOS VALORES DE VELOCIDAD Y ACELERACIONES PARA EL CENTROIDE (COLOR AZUL), MUÑECA DERECHA (COLOR ROJO) Y MUÑECA IZQUIERDA (COLOR MORADO) PARA LAS IMÁGENES CLASIFICADAS CON POSTURAS APROXIMÁNDOSE AL SUELO. SÓLO UNA DE LAS MEDICIONES EN CADA GRÁFICA ES MAYOR AL UMBRAL POR ESTA RAZÓN SE DETERMINA QUE LA PERSONA SE ACOSTÓ EN EL PISO. ....	93
FIGURA 43. EN LA GRÁFICA SE MUESTRA LA EXACTITUD DE RECONOCIMIENTO EN CADA CLASE Y LA EXACTITUD PROMEDIO PARA LAS VEINTISIETE ACCIONES CUANDO SE USAN TODAS LAS CARACTERÍSTICAS, LAS CARACTERÍSTICAS SELECCIONADAS CON EL ALGORITMO RFD-SD Y LAS CARACTERÍSTICAS SELECCIONADAS CON EL AG Y EL ALGORITMO RFD-SD. ....	98
FIGURA 44. EN LA GRÁFICA SE MUESTRA LA EXACTITUD DE CLASIFICACIÓN EN LAS CLASES Y LA EXACTITUD PROMEDIO DEL RECONOCIMIENTO DE LAS DIECIOCHO ACCIONES DE LA BASE KARD, CUANDO SE UTILIZANDO TODAS LAS CARACTERÍSTICAS Y CUANDO SE UTILIZARON LAS CARACTERÍSTICAS SELECCIONADAS POR EL ALGORITMO RFD-SD. ....	101
FIGURA 45. RECONOCIMIENTO DE LAS ACCIONES: REPOSITORIO DE CARACTERÍSTICAS SELECCIONADAS CON LAS SECUENCIAS PATRONES, SECUENCIA A RECONOCER, FASE DE FILTRADO PARA CORREGIR LAS POSICIONES DE LAS ARTICULACIONES DEL ESQUELETO, CALCULO DE CARACTERÍSTICAS, CÁLCULO DE SIMILITUD ENTRE LA SECUENCIA A RECONOCER Y LAS SECUENCIAS PATRONES Y POR ÚLTIMO RECONOCIMIENTO DE LA ACCIÓN. ....	103
FIGURA 46. SE MUESTRAN SEIS IMÁGENES DEL ESQUELETO DE UNA PERSONA FLEXIONÁNDOSE AL PISO, EN COLOR ROJO SE MUESTRAN LOS ESQUELETOS ADQUIRIDOS DIRECTAMENTE DEL DISPOSITIVO DE PROFUNDIDAD, EN LOS COLOR AZUL LOS ESQUELETOS CORREGIDOS CON EL FILTRO DE KALMAN Y EN COLOR VERDE LOS ESQUELETOS SUAVIZADOS CON EL FILTRO DE SAVITSKY-GOLAY. ....	104
FIGURA 47. SE MUESTRA LA EXACTITUD DE RECONOCIMIENTO EN CADA CLASE Y LA EXACTITUD PROMEDIO CON LAS VEINTE ACCIONES DE LA BASE MSR-ACTION3D, CUANDO SE UTILIZARON TODAS LAS CARACTERÍSTICAS, CARACTERÍSTICAS SELECCIONADAS Y LA FASE DE FILTRADO. ....	107



## Acrónimos

<b>ADL</b>	Activities of daily living
<b>AG</b>	Algoritmo Genético
<b>ANOVA</b>	Analysis of Variance
<b>DTW</b>	Dynamic Time Warping
<b>EM</b>	Expectation Maximization
<b>HMM</b>	Hidden Markov Model
<b>KNN</b>	k nearest neighbor
<b>KMSDA</b>	Kernel Mixture subclass discriminant analysis
<b>KSDA</b>	Kernel Subclass discriminant analysis
<b>KDA</b>	Kernel discriminant analysis
<b>LDA</b>	Linear discriminant analysis
<b>LFS</b>	Linear Forward Selection
<b>PTU</b>	Pan-Tilt Unit
<b>PCD</b>	Point Cloud Data
<b>PCL</b>	Point Cloud Library
<b>PCA</b>	Principal component analysis
<b>RGB</b>	Red-Green-Blue
<b>RGBD</b>	Red-Green-Blue-Depth
<b>RFD-SD</b>	Reduction of Feature Dimensions based on Standard Deviation
<b>RVM</b>	Relevance Vector Machine
<b>ROS</b>	Robot Operating System
<b>SBFS</b>	Sequential Backward Feature Selection
<b>SFFS</b>	Sequential Forward Feature Selection
<b>SDK</b>	Software Development Kit
<b>SVM</b>	Support Vector Machine
<b>ToF</b>	Time of flight
<b>TRE</b>	Traslación, Rotación y Escala
<b>3D</b>	Tres dimensiones

## Capítulo 1: Introducción

En la actualidad los adultos mayores tienden a vivir solos y en caso de sufrir una caída pueden permanecer horas sin recibir atención, además olvidan realizar actividades cotidianas como comer, lavarse las manos, entre otras. Con los resultados obtenidos en los trabajos de reconocimiento de acciones se mejora la calidad de vida de las personas. Por ejemplo con el reconocimiento de acciones se detecta cuando un adulto mayor come, descansa, hace ejercicio, entre otras actividades; si en un lapso de tiempo no se detecta que ha comido, mediante interacción humano-robot se alerta a la persona que debe comer; otro ejemplo es detectar cuando una persona cae al suelo y activar una alarma sonora, llamar por celular, enviar un correo electrónico o utilizar otro medio de comunicación y atender a la persona inmediatamente.

Los sistemas de reconocimiento acciones y caídas utilizan algunos de los siguientes elementos(Chen, Wei, & Ferryman, 2013):

- **Cámaras:** los descriptores de forma y orientación de una persona se extraen a partir de información visual adquirida con cámaras RGB ó RGBD. Esta solución no es intrusiva.
- **Sensores corporales:** acelerómetros o giroscopios son colocados en varias partes del cuerpo para detectar caídas, los sensores se colocan en la cintura y/o muñeca de la mano. Esta solución es intrusiva, las personas pueden olvidar o rechazar colocarse los sensores. Actualmente las personas cargamos el celular casi todo el tiempo, por lo anterior, en algunos trabajos se utiliza el acelerómetro del celular para detectar caídas.
- **Sensores en casas:** en los pisos de las casas se colocan sensores que detectan vibraciones. Las vibraciones se presentan cuando la persona cae al piso, pero también cuando una persona corre, pasa un camión entre otros, entonces para confirmar la caída se utiliza información de sensores corporales o visuales.

En esta tesis se utiliza sólo información visual para reconocer acciones y caídas. La selección de características se hace con el algoritmo RFD-SD, un algoritmo genético y con análisis de componentes principales. Los resultados obtenidos en los experimentos sugieren que las características seleccionadas con el algoritmo RFD-SD y con un algoritmo genético incrementan la exactitud del reconocimiento. En los experimentos se utilizan diferentes clasificadores y bases públicas de acciones y caídas.

## **1.1. Planteamiento de Problema**

En el estado del arte se reporta una exactitud de reconocimiento de caídas del 100% (Gasparrini, y otros, 2016) cuando utilizan información de cámaras y dispositivos inerciales, los últimos son colocados en el cuerpo de la persona. La desventaja de este tipo de sistemas se presenta cuando la persona olvida colocarse los sensores o se niega a usarlos por ser incómodos. Para evitar esta situación, en esta tesis se reconocen acciones y caídas basado sólo en información visual. En el reconocimiento de acciones se utilizan las posiciones de las articulaciones, sin embargo presentan deformaciones cuando la persona se mueve rápido o cuando se aproxima al suelo, por esta razón se evalúan los filtros de Kalman y Savitsky-Golay con el propósito de minimizar las deformaciones e incrementar la exactitud del reconocimiento de las acciones.

En los trabajos revisados se proponen conjuntos de características definidos de manera experimental y en pocos casos mediante un algoritmo de selección de características. La desventaja de seleccionar características experimentalmente se presenta cuando el conjunto de acciones aumenta y requiere otro análisis experimental de características para definir si se utilizan las mismas o deben incluirse algunas otras. En esta tesis se seleccionan características con un algoritmo genético, con el algoritmo RFD-SD y con análisis de componentes principales con el propósito de mostrar cuáles de los algoritmos seleccionan características que incrementan la exactitud del reconocimiento en comparación con utilizar el conjunto completo de características propuesto.

## **1.2. Hipótesis**

La exactitud de reconocimiento de acciones y caídas mejora cuando se utilizan características seleccionadas con un algoritmo genético o con el algoritmo RFD-SD. La propuesta de esta tesis para reconocer acciones y caídas consiste en:

- Definir un conjunto con características utilizadas en los trabajos relacionados y los invariantes 3D-TRE.
- Seleccionar características con el algoritmo RFD-SD para reconocer acciones (series de tiempo) y con un algoritmo genético para reconocer posturas.
- Incrementar la exactitud del reconocimiento cuando se utilizan las características seleccionadas con el algoritmo RFD-SD y con el AG, en comparación con utilizar el conjunto de características completo o el subconjunto seleccionado por PCA.

## **1.3. Objetivos**

### **1.3.1. Objetivo General**

Incrementar la exactitud del reconocimiento de acciones y caídas cuando se utilizan las características seleccionadas con el algoritmo RFD-SD y con un algoritmo genético, cuando se compara con utilizar el conjunto completo de características o las características seleccionadas con PCA.

### **1.3.2. Objetivos Particulares**

- Detectar las posiciones  $(x, y, z)$  de las articulaciones del esqueleto y la nube de puntos de una persona.
- Proponer un conjunto de características que describan la forma y orientación de la nube de puntos de una persona para detectar caídas.
- Proponer un conjunto de ángulos calculados con las posiciones de las articulaciones del esqueleto para reconocer las acciones. El esqueleto no se normaliza en escala u orientación con las características propuestas, ya que los ángulos son invariantes a rotación y escala.
- Seleccionar características con un algoritmo genético, un algoritmo RFD-SD y PCA y evaluar el impacto de los conjuntos de características seleccionadas en la exactitud del reconocimiento de caídas y acciones.
- Analizar los resultados de los experimentos y obtener conclusiones.

## **1.4. Contribución**

Los invariantes 3D-TRE se incluyen en el conjunto de descriptores de posturas, estas características no han sido utilizadas en los trabajos revisados. Cuando se seleccionan características con el AG y PCA al menos uno de los tres momentos invariantes es seleccionado, lo anterior sugiere que estas características son importantes para distinguir posturas. Las bases *UR Fall Detection Dataset*<sup>1</sup> y *TST Fall detection dataset v2*<sup>2</sup> son utilizadas en los experimentos.

---

<sup>1</sup> <http://fenix.univ.rzeszow.pl/~mkepski/ds/uf.html>

En un ambiente controlado se detecta cuando una persona cae al suelo, la información visual se adquiere con un Kinect y una unidad *pant-tilt* (PTU por sus siglas en inglés) montados en un robot móvil. Las características seleccionadas con el AG con los datos de la base *UR Fall Detection Dataset* son utilizadas para detectar la caída. Los resultados obtenidos demuestran que las características seleccionadas y los umbrales establecidos experimentalmente son suficientes para reconocer la caída en un 100%. En este experimento se realizan tres acciones: la persona camina de un lado a otro, la persona se sienta y por último la persona camina y cae al piso.

La exactitud de reconocimiento de posturas de caída se incrementa cuando se utilizan las características seleccionadas con el AG. Cuando se utiliza la base *TST Fall detection dataset v2*, la exactitud del reconocimiento es de 90%. En (Gasparrini, y otros, 2016) detectan las caídas utilizando la base de caídas mencionada, las características son calculadas a partir de información visual y dos sensores inerciales, se realizan tres experimentos y se obtiene una exactitud del reconocimiento de la caída del 79%, 90% y 99%. En (Gasparrini, y otros, 2016) mejoran la exactitud del reconocimiento cambiando la ubicación de los sensores inerciales colocados sobre el cuerpo de la persona. La exactitud del reconocimiento obtenida en esta tesis utilizando sólo información visual es comparable con lo reportado en el estado del arte.

La exactitud del reconocimiento de posturas de caída se incrementa cuando se utilizan las características seleccionadas con un AG, lo anterior se descubre utilizando diferentes clasificadores y bases de datos de caídas.

La exactitud del reconocimiento de las acciones se incrementa cuando se utilizan las características seleccionadas por el algoritmo RFD-SD. En los experimentos se utilizaron las bases de acciones públicas KARD<sup>3</sup> y UTD Multimodal Human Action Dataset (UTD-MHAD<sup>4</sup>), en los experimentos se obtiene una exactitud promedio de 94.2% y de 83.8%, respectivamente. La exactitud promedio reportada en el estado del arte (Gaglio, Re, & Morana, 2015) es 88.3% para la base KARD y de (Chen, Jafari, & Kehtarnavaz, 2015) 66.1% para la base UTD-MHAD utilizando sólo información visual.

---

<sup>2</sup> <https://iee-dataport.org/documents/tst-fall-detection-dataset-v2>

<sup>3</sup> <https://data.mendeley.com/datasets/k28dtm7tr6/1>

<sup>4</sup> <https://www.utdallas.edu/~kehtar/UTD-MHAD.html>

## 1.5. Publicaciones

Maldonado-Mendez, C., Rios-Figueroa, H., Mezura-Montes, E., & Marin-Hernandez, A. (Feb de 2016). Feature selection to detect fallen pose using depth images. *2016 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, 94-100.

Maldonado-Mendez, C., Rios-Figueroa, H., & Marin-Hernandez, A. (Nov de 2016). Improving action recognition by selection of features. *2016 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, 1-6.

Hernandez-Mendez, S., Maldonado-Mendez, C., Marin-Hernandez, A., & Rios-Figueroa, H. (Feb de 2017). Detecting falling people by autonomous service robots: A ROS module integration approach. *2017 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, 1-7.

Maldonado-Mendez, C., Solis, A. L., Rios-Figueroa, H., & Marin-Hernandez, A. (Nov de 2017). Human fallen pose detection by using feature selection and a generative model. *2017 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, 1-6.

Maldonado-Mendez, C., Hernandez-Mendez, S., Solis, A. L., Rios-Figueroa, H., & Marin-Hernandez, A. (Nov de 2017). The Effects of Using a Noise Filter and Feature Selection in Action Recognition: An Empirical Study. *2017 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)*, 43-48.

## 1.6. Estructura de la tesis

Esta tesis consta de cinco capítulos, los cuales están organizados de la siguiente manera:

- En el **capítulo 2**, se hace una revisión de trabajos relacionados con selección de características, clasificadores, reconocimiento de acciones y caídas.
- En el **capítulo 3**, se describe la solución propuesta para reconocer acciones y caídas, los conjuntos de características utilizados, los algoritmos de selección de características y los clasificadores evaluados en los experimentos.
- En el **capítulo 4**, se muestran los experimentos realizados, se analizan y discuten los resultados obtenidos.
- Para concluir, en el **capítulo 5**, se resumen los hallazgos de este trabajo de investigación y se mencionan posibles trabajos futuros.

## Capítulo 2: Trabajos relacionados

En este capítulo se presentan trabajos relacionados con selección de características, clasificadores, reconocimiento de acciones y caídas, por consiguiente, el lector se familiarizará con la terminología y el estado del arte.

### *2.1. Reconocimiento de acciones*

El trabajo desarrollado en reconocimiento de acciones tiene diversas aplicaciones, por citar algunas en: video vigilancia, interacción humano computadora, análisis y mejora del desempeño de atletas y robots de asistencia.

Las acciones realizadas por una persona cuando son adquiridas mediante una cámara se almacenan como secuencias de imágenes. Una misma acción puede ser ejecutada en diferentes velocidades, entonces secuencias de una misma acción contienen diferente cantidad de imágenes, esto último es uno de los problemas a resolver cuando se hace el reconocimiento de acciones.

Para reconocer el movimiento de las personas, en (Liang & Zheng, 2015) proponen los siguientes niveles de abstracción:

- **Gestos:** la persona mueve sólo una parte del cuerpo, por mencionar algunos: un brazo, una pierna y la cabeza.
- **Acción:** es la ejecución de varios gestos, los cuales siguen un orden temporal. Un ejemplo se presenta cuando una persona se agacha a recoger un objeto.
- **Actividad:** es la ejecución de varias acciones, las cuales son realizadas por una o más personas.

El objetivo de esta tesis es mejorar la exactitud del reconocimiento de acciones, se asume que es dependiente del conjunto de características utilizadas. De acuerdo con los resultados de los experimentos realizados en esta tesis, el subconjunto de características seleccionadas por el algoritmo RFD-SD incrementa el reconocimiento de acciones con todos los clasificadores y datos utilizados. Como trabajo futuro se podrá reconocer actividades, utilizando las características seleccionadas para detectar acciones.

Algunos trabajos en reconocimiento de acciones se presentan a continuación:

En (Li, Zhang, & Liu, 2010) reconocen acciones utilizando secuencias de imágenes en profundidad. A partir de una bolsa de posturas características modelan la dinámica de las acciones en un grafo. Para cada postura calculan tres proyecciones: la primera en el plano  $xy$ , la segunda en el plano  $xz$

y la tercera en el plano  $yz$ . En sus experimentos con la base MSR-Action3D reportan una exactitud promedio de reconocimiento de 90%, las acciones son agrupadas en tres categorías.

En (Sung, Ponce, Selman, & Saxena, 2011) capturan las secuencias de acciones con el Kinect, el clasificador está basado en modelos de Markov. En su propuesta reconocen doce acciones realizadas por cuatro personas en diferentes ambientes. Los autores reportan una exactitud de reconocimiento de 84.3% cuando la acción la realiza una persona cuyas secuencias son utilizadas en el entrenamiento y de 64.2% cuando la acción la realiza una persona que no forma parte del conjunto de entrenamiento. Las características utilizadas son 47 ángulos calculados mediante las posiciones de las articulaciones, la posición de la mano e información del movimiento. El conjunto completo contiene 459 características.

En (Sempena, Maulidevi, & Aryan, 2011) miden la similitud de dos acciones con DTW. El conjunto de características incluye la orientación de las articulaciones del esqueleto, las cuales son invariantes al tamaño del cuerpo, por lo tanto no normalizan el esqueleto. En las pruebas utilizan una base de seis acciones, en donde el actor se coloca de frente a la cámara, se reconocen las acciones, pero no indican el porcentaje de reconocimiento.

En (Zhang & Parker, 2011) utilizan características espacio-temporales locales, las cuales combinan información de intensidad y profundidad capturada con el Kinect. La asignación latente de Dirichlet y el muestreo de Gibbs son utilizadas para reconocer las acciones. En las pruebas reconocen seis acciones: levantarse, moverse, mover la mano, caminar, señalar, empujar. Los autores reportan una exactitud de reconocimiento de 77.67% cuando utilizan la información de intensidad, del 85.80% cuando utilizan la información de profundidad y de 91.5% cuando combinan las dos informaciones.

En (Reyes, Dominguez, & Escalera, 2011) representan las acciones como secuencias de imágenes de profundidad. Para comparar dos acciones utilizan DTW, además definen un esquema de características con asignación de pesos. Los pesos dependen de la variabilidad de los gestos entre las clases. Las características son las posiciones  $(x, y, z)$  de las articulaciones normalizadas. En los experimentos reconocen cinco acciones: saltar, doblarse, aplaudir, saludar, anotar con la mano y reportan una exactitud de reconocimiento del 68% cuando utilizan pesos y del 61.12% cuando utilizan DTW sin pesos.

En (Yang & Tian, 2012) calculan las características con las posiciones de las articulaciones del esqueleto y clasifican las acciones con un Bayes ingenuo. En los experimentos con la base MSR-Action3D reportan una exactitud de reconocimiento de 95.8% cuando cada acción contiene entre quince y veinte imágenes.

En (Zhang & Tian, 2012) reconocen trece actividades de la vida diaria, por mencionar algunas: hablar por teléfono, escribir en el pizarrón, beber agua, lavarse los dientes y descansar en un sillón, además de cinco tipos de caídas. Las secuencias son adquiridas utilizando el Kinect. En el



reconocimiento de las acciones de la vida diaria utilizan dos tipos de características: las primeras describen el movimiento de las partes del cuerpo y las segundas describen la postura. En los experimentos clasifican con SVM las acciones de la vida diaria de la base Cornell 3D y reportan una exactitud de reconocimiento de una acción realizada por una persona que no forma parte del conjunto de entrenamiento de 81.79%. Por otra parte, en el reconocimiento de la caída utilizan ocho articulaciones ubicadas entre la cabeza y el torso, los autores mencionan que dichas articulaciones presentan menos ruido, en los experimentos participan cinco personas en ambientes con suficiente iluminación y reportan una exactitud de reconocimiento de la caída del 100%.

En (Zhao, Liu, Yang, & Cheng, 2012) extraen el histograma de gradiente, el histograma de flujo en el canal RGB y el patrón de profundidad local en el canal de profundidad adquirido con un Kinect. Con las características mencionadas clasifican con SVM las acciones de la base RGBD-HuDaAct y reportan una exactitud de reconocimiento de 89.1%.

En (Xia, Chen, & Aggarwal, 2012) representan las posturas con histogramas de las coordenadas esféricas en 3D de las articulaciones del esqueleto. Los histogramas son re-proyectados utilizando LDA, posteriormente se crean posturas características de acciones. Las acciones son clasificadas con modelos ocultos de Markov. Los autores reportan una exactitud en el reconocimiento de 96.2% cuando utilizan la base MSR-Action3D, este resultado supera el reportado por (Li, Zhang, & Liu, 2010).

En (Miranda, Vieira, Martinez, Lewiner, Vieira, & Campos, 2012) representan las posturas con ángulos esféricos calculados a partir de las articulaciones del esqueleto adquiridas con el Kinect. El método de reconocimiento propuesto es invariante al ruido de la información visual. En una fase de entrenamiento con SVM se seleccionan posturas claves, las cuales se ocupan para reemplazar las posturas contenidas en las acciones que son clasificadas con un árbol de decisión. Las pruebas realizadas con la base MSR-Action3D alcanzan una exactitud de reconocimiento de 80.3%.

En (Yang, Zhang, & Tian, 2012) las acciones son representadas por un histograma de gradiente orientado calculado a partir del mapa de movimientos de las imágenes en profundidad. En los experimentos con la base MSR-Action3D reportan una exactitud promedio de reconocimiento de 95.8% cuando cada acción contienen entre 30 y 35 imágenes.

En (Charaoui, Climent-Pérez, & Flórez-Revuelta, 2012) reconocen acciones con un sistema de múltiples vistas, para esto, seleccionan con el algoritmo K-medias una bolsa de posturas claves a partir de las siluetas de las personas y miden la similitud entre las acciones con DTW. En las pruebas con la base MuHAVi reportan una exactitud de reconocimiento de 94.1%.

En (Ni, Wang, & Moulin, 2013) crean la base de acciones pública RGBD-HuDaAct y reconocen acciones con puntos de interés espacio-temporales e imágenes con el contenido histórico del

movimiento en las secuencias. Los autores mencionan que cuando utilizan la información de profundidad y de color obtienen un mejor reconocimiento comparado con sólo utilizar el color.

En (Xia & Aggarwal, 2013) utilizan puntos de interés espacio-temporal y de profundidad para crear una característica de similitud cuboidal de profundidad, con la cual reconocen acciones sin depender de la información del esqueleto. Los autores reportan una exactitud de reconocimiento de 89.3% y de 88.2% con las bases de acciones MSR-Action3D y MSRDailyActivity3D, respectivamente.

En (Zhu, Chen, & Guo, 2013) utilizan características espacio-temporales y las articulaciones del esqueleto para clasificar acciones con SVM. En los experimentos con las bases: MSR-Action3D, UTKinect-Action y CAD60 reportan una exactitud de reconocimiento de 94.3%, 91.9% y 87.5%, respectivamente.

En la propuesta de (Fanello, Gori, Metta, & Odone, 2013) reconocen acciones en tiempo real mediante una representación escasa. Con las imágenes RGBD calculan histogramas 3D de flujo de la escena e histogramas globales de gradiente orientados, posteriormente clasifican con SVM las acciones. En los experimentos utilizan tres bases de acciones, además los autores implementan su propuesta en una aplicación de interacción humano-robot.

En (Imtiaz, Mahbub, Schaefer, & Ahad, 2013) utilizan la transformada wavelet discreta bidimensional con el objetivo de extraer características de múltiple resolución. Los autores proponen un algoritmo para reducir la dimensión del conjunto de características. El conjunto de características extraídas muestra alta compactibilidad dentro de la clase y separabilidad entre clases. Al trabajar en el dominio de la frecuencia, el sistema propuesto es robusto a los movimientos de la cámara y a la distancia entre la persona y la cámara. Las acciones son clasificadas con SVM. En las pruebas con la base de acciones de Weizmann reportan una exactitud de reconocimiento de 100%.

En (Charaoui, Climent-Pérez, & Flórez-Revuelta, 2013) utilizan varias vistas de contorno de la silueta de la persona, de esta forma alguna de las cámaras captura la vista completa o se reconstruye la vista completa con las diferentes vistas capturadas. Los autores utilizan posturas claves, con el objetivo de eliminar ruido y valores atípicos. En los experimentos reportan una exactitud en reconocimiento de 92.77%, 91.18% y 86.85% con las Weismann Dataset, MuHAVi e IXMAS, respectivamente.

En (Charaoui & Flórez-Revuelta, 2014) seleccionan características con un algoritmo evolutivo cooperativo, trabajan con múltiples vistas de la silueta, la cual dividen en dieciocho regiones de manera radial, cada región es una característica. En los experimentos reportan una exactitud de reconocimiento de 100% con las bases Weismann Dataset, MuHAVi e IXMAS.

En (Kim, Kim, Sandhan, & Choi, 2014) crean imágenes con la historia del movimiento e imágenes con la historia de lo que se mantiene estático, utilizan características espacio-temporales 4D, con las cuales generalizan información a partir de un número finito de vistas. Las características son extraídas utilizando información en 3D. En la fase de entrenamiento los autores utilizan la base IXMAS y en la fase de pruebas la base SNU y reportan una exactitud de reconocimiento de 77.2% y 73%, respectivamente. Los resultados confirman que las características propuestas generalizan la información de cada acción.

En (Chen & Koskela, 2015) normalizan las posiciones  $(x, y, z)$  de las articulaciones del esqueleto, con el propósito de utilizar características invariantes a escala y rotación. Las acciones son clasificadas con máquinas de aprendizaje extremo. En las pruebas con las bases de acciones: HDM05, MSRC-12 y MHAD reportan una exactitud de reconocimiento de 95%, 99.3%, y 99.5%, respectivamente.

En (Sheng, Yang, & Sun, 2015) utilizan pares de características dependientes de la dirección, así como un modelo escaso de baja densidad con el cual codifican las características. En los experimentos con las bases KTH, Weizmann y UCF Sports reportan una exactitud de reconocimiento de 94.99%, 96.77% y 94.99%, respectivamente.

En (Liu, Shao, Li, & Lu, 2016) las características son aprendidas con el movimiento presente en las acciones en el espacio y el tiempo, es decir, las características no son definidas previamente. Los autores utilizan programación genética con operadores Gabor y wavelet para aprender característica invariantes a escala y corrimiento. En las pruebas con las bases KTH, HMDB51, UCF YouTube y Hollywood2 reportan un reconocimiento de 95%, 48.4%, 82.3% y 46.8%, respectivamente. Los resultados obtenidos con las bases HMDB51, UCF YouTube superan los resultados reportados en el estado del arte cuando se utilizó un conjunto de características diferente.

En (Ding & Chang, 2016) utilizan las posiciones  $(x, y, z)$  y ángulos entre las articulaciones del esqueleto. En los experimentos reconocen catorce clases de acciones, reportan una exactitud de reconocimiento de 87.5% cuando utilizan todo el conjunto de características y de 82.14% cuando utilizan las posiciones de las articulaciones.

En (Jiang, Zhong, Peng, & Qin, 2016) reconocen acciones con las posiciones de las articulaciones del esqueleto. El método propuesto trabaja efectivamente con el ruido y desajustes presentes en las coordenadas de las articulaciones. Las acciones son clasificadas con un grafo con pesos, el cual codifica la transición de las características en el espacio y tiempo. En las pruebas con las bases MSRC-12 y UCFKinect reportan una exactitud de reconocimiento de 95.12% y 99.69%, respectivamente.

En (Li, Hou, Wang, & Li, 2017) utilizan *Deep Learning* para explotar las características discriminantes propuestas, las cuales son referidas como Mapas de distancias de articulaciones.

Las características codifican información espacio temporal de la secuencia de esqueletos. En los experimentos con las bases NTU RGB+D y en UTD-MHAD reportan una exactitud de reconocimiento de 82.3% y 88.10%, respectivamente.

En esta tesis se revisan trabajos publicados a partir del año 2010, en estos trabajos reconocen acciones con información visual RGBD o RGB. En la Tabla 1 se indica con una X el tipo de información utilizada en cada trabajo. En la columna SC se muestra una X cuando en el trabajo seleccionan características, solamente en tres trabajos lo hacen: en uno utilizan PCA, en el segundo un AG y en el tercero utilizan programación genética (PG). En la columna RA se muestra la técnica utilizada para reconocer acciones. La columna Características presenta cuáles características son utilizadas en cada trabajo y finalmente en la columna Bases de Pruebas se menciona el nombre de la base de datos pública utilizada en los experimentos. La mayoría de los trabajos evalúan sus conjuntos de características y algoritmos de clasificación con bases públicas. La base MSR-Action3D es la más utilizada porque presenta ruido en las posiciones de las articulaciones y contiene acciones con diferentes niveles de dificultad. Estos aspectos son los retos a superar cuando se reconocen acciones en tiempo real. En la Tabla 1 se observa que en la mayor parte de los trabajos revisados extraen las características a partir de las posiciones de las articulaciones.

**Tabla 1 Trabajos relacionados con el reconocimiento de acciones**

Reconocimiento de Acciones						
	Imagen RGBD	Imagen RGB	SC	RA	Características	Bases de Pruebas
(Li, Zhang, & Liu, 2010)	X			Grafo	Proyecciones de las imágenes.	MSR-Action3D
(Sung, Ponce, Selman, & Saxena, 2011)	X			Markov	Ángulos, posición de la mano. Total 459 Características.	
(Sempena, Maulidevi, & Aryan, 2011)	X			DTW	Orientación de las articulaciones.	
(Zhang & Parker, 2011)	X			Asignación latente de Dirichlet con el muestreo de Gibbs	Espacio-temporales (combinan información de intensidad y de profundidad.	
(Reyes, Dominguez, & Escalera, 2011)	X			DTW	Posiciones de las articulaciones.	
(Yang & Tian, 2012)	X			Bayes	Información de la postura estática, movimiento y desplazamiento de articulaciones.	MSR-Action3D

(Zhang & Tian, 2012)	X			SVM	Postura y movimiento a partir de ocho articulaciones.	Cornell 3D
(Zhao, Liu, Yang, & Cheng, 2012)	X			SVM	Histograma de gradiente y el histograma de flujo en el canal RGB y el patrón de profundidad local.	RGBD-HuDaAct
(Xia, Chen, & Aggarwal, 2012)	X			Markov	Histogramas de ubicaciones de las articulaciones en 3D	MSR-Action3D
(Bloom, Makris, & Argyriou, 2012)	X			AdaBoost	Diferencia de posición, velocidad y ángulos de las articulaciones. En total son 170 características.	G3D
(Miranda, Vieira, Martinez, Lewiner, Vieira, & Campos, 2012)	X			SVM	Ángulos esféricos.	MSR-Action3D
(Yang, Zhang, & Tian, 2012)	X				HOG en tres planos ortogonales.	MSR-Action3D
(Chaaroui, Climent-Pérez, & Flórez-Revuelta, 2012)		X		DTW	Siluetas.	MuHAVi
(Ni, Wang, & Moulin, 2013)	X			SVM	Histórico de movimiento en tres dimensiones.	RGBD-HuDaAct
(Xia & Aggarwal, 2013)	X				Proponen una característica de similitud cuboidal de profundidad.	MSR-Action3D, MSRDailyActivity3D
(Zhu, Chen, & Guo, 2013)	X			SVM	Espacio-temporales y articulaciones del esqueleto.	MSR-Action3D, UTKinect-Action y CAD60
(Fanello, Gori, Metta, & Odone, 2013)	X			SVM	Histogramas 3D de flujo de la escena e histogramas globales de gradiente orientados.	
(Imtiaz, Mahbub, Schaefer, & Ahad, 2013)		X	X PCA	SVM	Dominio de la frecuencia.	Weizmann
(Chaaroui, Climent-Pérez, & Flórez-Revuelta, 2013)		X		DTW	Silueta de la persona.	Weismann, MuHAVi y IXMAS
(Chaaroui & Flórez-Revuelta, 2014)		X	X AG	DTW	Silueta de la persona.	Weismann, MuHAVi y IXMAS
(Kim, Kim, Sandhan, & Choi, 2014)	X				Espacio-temporales 4D.	IXMAS, SNU

(Chen & Koskela, 2015)	X			Máquina de aprendizaje	Posiciones de las articulaciones.	HDM05, MSRC-12 y MHAD
(Sheng, Yang, & Sun, 2015)		X		Proponen un algoritmo basado en el uso de un grafo	Características dependientes de la dirección.	KTH, Weizmann y UCF Sports
(Liu, Shao, Li, & Lu, 2016)		X	X PG	SVM	Movimiento de las acciones en el espacio y tiempo.	KTH, HMDB51, UCF YouTube y Hollywood2
(Ding & Chang, 2016)	X			DTW	Posiciones $(x, y, z)$ y ángulos de las articulaciones.	
(Jiang, Zhong, Peng, & Qin, 2016)	X			Grafo con pesos	Posiciones de las articulaciones del esqueleto.	MSRC-12, UCFKinect
(Li, Hou, Wang, & Li, 2017)	X			Deep Learning	Mapas de distancias de articulaciones.	NTU RGB+D y en UTD-MHAD

## 2.2. Reconocimiento de caídas

La Organización Mundial de la Salud (Organization, 2015) informa que entre el 28% y 35% de la población mayor de 65 años sufren caídas cada año y el número se incrementa conforme la gente envejece. Con el propósito de mejorar la calidad de vida de los adultos mayores, el reconocimiento de caídas continua siendo un área de investigación activa.

En (Yu, 2008) identifican diferentes tipos de caídas, las cuales suceden cuando la persona está: dormida, sentada en una silla, caminando y utilizando un soporte, además proporcionan características de movimiento, posturas y duraciones de cada tipo de caída.

En esta tesis se detecta la caída cuando la persona camina y está sentada, de acuerdo con (Yu, 2008) la primera caída dura de uno a dos segundos y la segunda hasta tres segundos. En ambas caídas la altura de la cabeza con respecto al piso disminuye, la persona cae bruscamente en una dirección, la cabeza y el centro de masa de la persona se mueven en un plano durante la caída, finalmente la persona está tendida sobre el suelo. Las características mencionadas están presentes en otras actividades, por ejemplo acostarse en el piso y ponerse en cuclillas, por esta razón es difícil reconocer las caídas.

A continuación se presentan trabajos relacionados con el reconocimiento de caídas:

En (Brulin, Benezeth, & Courtial, 2012) con cuatro características describen la forma de la silueta de una persona. Las características son calculadas con el centro de gravedad de la persona, dimensiones de la caja envolvente de la persona y distancia del piso al centro de gravedad. En las

pruebas reconocen cuatro posturas: acostado, sentado, parado y en cuclillas, el método propuesto es invariante a la distancia entre la persona y la cámara, así como a la morfología de la persona. Los autores reportan una exactitud de reconocimiento de 74.29% cuando utilizan 62 secuencias con imágenes capturadas en diferentes ambientes.

En (Kawatsu, Li, & Chung, 2013) calculan la distancia al piso de cada articulación y la velocidad promedio de las articulaciones en varias imágenes. El sistema propuesto asume una caída cuando el valor más grande de las distancias y la velocidad promedio son menores que dos umbrales. El sistema envía un correo electrónico con imágenes de la caída. Los experimentos se realizan con caídas simuladas. Los umbrales deben modificarse para detectar caídas reales debido a que la velocidad en las caídas simuladas es diferente en las caídas reales.

En (Kwolek & Kepski, 2014) detectan una potencial caída con la información de un acelerómetro y con cinco características extraídas de la nube de puntos en 3D de una persona detectada en una imagen y con un SVM se confirma la caída. En las pruebas con la base UR Fall Detection reportan una exactitud de reconocimiento de 98.33% cuando utilizan información visual y el acelerómetro y de 90% cuando utilizan información visual.

En (Ma, Wang, Xue, Zhou, Ji, & Li, 2014) las posturas son representadas por un conjunto de curvaturas, las cuales son calculadas con información visual en 3D. Los autores reportan una exactitud de reconocimiento de 86.83% cuando utilizan la base SDUFall. En otro experimento con la base MultiCam, la cual contiene eventos cuyas características se confunden con la caída, por mencionar alguno: cuando la persona se agacha, reportan una sensibilidad de 99.93%. El resultado en el estado del arte utilizando cuatro cámaras es de 99.7% y usando tres cámaras es de 80.6%, por consiguiente el método propuesto con una cámara demuestra un desempeño similar al obtenido cuando utilizan múltiples cámaras.

En (Nguyen, Le, Do, Duong, Thai, & Tran, 2014) detectan cuando una persona cae al piso o realiza otras actividades como estar parado o sentarse, calculan la razón de aspecto, ángulo del rectángulo y elipse envolvente a la persona. El sistema propuesto reconoce la caída al comparar los valores de las características con umbrales. En las pruebas utilizan videos grabados en exteriores e interiores en diferentes condiciones de luz, en total son 3996 imágenes: en 2873 la persona cae y en las restantes contienen acciones con persona sentada o de pie. Los autores reportan una exactitud de 92.5%.

En (Soewitoea, Irwana, Antonyová, & Gunawana, 2015) detectan la caída comparando la información del acelerómetro de un teléfono inteligente con dos umbrales definidos experimentalmente. En las pruebas utilizan una base datos con los siguientes eventos: el celular cae de una altura de veinte cm y de cincuenta cm; una persona coloca el celular en la bolsa del su pantalón y se acuesta en el suelo; una persona coloca el celular en la bolsa de su camisa y se

acuesta en el suelo; cada actividad se repite treinta veces. Los autores reportan una detección de falsos positivos del 30% cuando el celular está en la bolsa de la persona y de 1.5% cuando se deja caer el celular.

En (Bian, Hou, Chau, & Magnenat-Thalmann, 2015) analizan los movimientos de las articulaciones del esqueleto de una persona y detectan las regiones asociadas con las articulaciones seleccionadas (cabeza, cadera entre otras) con un árbol de decisión aleatorio. En las pruebas reconocen caídas de espaldas, hacia el frente, lateral a la derecha, lateral a la izquierda y acciones de la vida diaria como sentarse, acostarse en la cama, caminar, entre otras, reportan una exactitud de reconocimiento de 97.9%, adicionalmente realizan experimentos con métodos utilizados en otros dos trabajos y obtienen una exactitud del 85.8% y del 76.8%.

En (Wang, Zhang, Dong, Lu, Zhang, & He, 2015) calculan cuatro grupos de características: en el primero calculan el rectángulo envolvente de la silueta, con éste calculan el área, perímetro, altura, anchura y razón de altura/anchura. En el segundo calculan los momentos invariantes de Hu. En el tercer tipo calculan el ángulo entre los vectores que pasan por el centro del rectángulo y los vértices superiores. Por último en el cuarto calculan la velocidad de las coordenadas baricéntricas del cuerpo humano. Las características mencionadas son las entradas del clasificador. En los experimentos utilizan una base con acciones (caer, sentarse, hacer sentadillas, pasear en ambientes interiores) creada por una Universidad en Malasia, reportan una sensibilidad de 93.3% y una especificidad del 96.2%.

En (Liu, Wang, Lv, & Cui, 2015) detectan la caída en imágenes de profundidad adquiridas con el Kinect, distinguen la acción de caer de la acción de agacharse. Primero elimina el ruido de la imagen de profundidad, posteriormente detectan la silueta humana y reparan sus bordes mediante el operador de corrosión. De la silueta extraen tres características: razón de aspecto del rectángulo envolvente a la persona, inclinación y centro de gravedad de la silueta. Las características son comparadas con un umbral para determinar si la persona cae al piso. Las pruebas son realizadas con una base de datos propia, pero no se indica la exactitud del reconocimiento.

En (Stone & Skubic, 2015) calculan el grado de certidumbre de una caída con árboles de decisión, cuyas entradas son la altura de la persona, posición del centroide, así como la velocidad y aceleración de las dos características mencionadas. En las pruebas utilizan videos de grabaciones realizadas por nueve años en casas de trece adultos mayores. La base contiene 454 caídas, de las cuales 445 son caídas simuladas y nueve son reales, reportan una exactitud de reconociendo del 98% cuando la persona cae a partir de una posición de pie, del 70% cuando la persona cae a partir de estar sentada y de 71% cuando la persona cae a partir de estar acostada. Además los tres eventos de caída mencionados son evaluados con oclusiones y reportan una exactitud de reconocimiento de 79%, 58% y 5%, respectivamente.



En (Kaneko & Cao, 2016) calculan 251 características de auto-correlación local cúbica de orden superior, las cuales son obtenidas con las secuencias de esqueletos adquiridas con un Kinect. El Kinect se encuentra sobre un robot móvil. El conjunto de 251 características es reducido con PCA. Las secuencias utilizadas en los experimentos contienen las siguientes acciones: caminar, acostarse en el suelo y caer. Los autores clasifican con KNN y evalúan el desempeño de su propuesta con la métrica F y obtiene un valor de 0.87.

En (Baptista-Ríos, Martínez-García, Losada-Gutiérrez, & Marrión-Romera, 2016) detectan y siguen a más de una persona en una escena. En las pruebas utilizaron las acciones: caminar, correr, sentarse y caer; en promedio son setenta secuencias por acción, reportan una exactitud de reconocimiento de 97.82%. En otro experimento con las bases CAVIAR y KTH evalúan el seguimiento de personas.

En (Gasparrini, y otros, 2016) detectan la caída con información de acelerómetros colocados en la muñeca de la mano y en la cintura de una persona, además utilizan las posiciones de las articulaciones del esqueleto. En los experimentos utilizan una base con cuatro acciones de la vida diaria (sentarse, caminar, tomar un objeto del piso y acostarse) y cuatro tipos de caídas. Los autores realizan tres experimentos: en el primero utilizan la información del acelerómetro de la muñeca y el cambio de la articulación ubicada en la base de la columna vertebral, reportan una exactitud promedio de reconocimiento de 79%. En el segundo utilizan la información del acelerómetro colocado en la cintura y los cambios en la articulación situada en la base de la columna vertebral, reportan una exactitud promedio de reconocimiento de 90%. En el tercero utilizan información del acelerómetro colocado en la cintura, los cambios en la articulación ubicada en la base de la columna vertebral, así como la distancia dicha articulación al piso, reportan una exactitud promedio de reconocimiento de 99%.

En (Mejia, Montalvo, Mena, & Ramirez, 2016) implementan a PIANI, una plataforma inteligente a través de la cual monitorean las actividades de las personas en exteriores, adicionalmente calculan el nivel de riesgo de dichas actividades. Cuando el nivel de riesgo calculado es mayor que cierto umbral una alerta se activa. PIANI se ejecuta en teléfonos inteligentes y no se considera intrusiva porque las personas tenemos apego al celular. Cuando se instala la aplicación, durante un mes recopila información de las actividades cotidianas de la persona y calcula la probabilidad de riesgo de las mismas. Los autores desarrollaron PIANI con la finalidad de mejorar la calidad de vida a los adultos mayores.

En (Valcourt, De La Hoz, & Labrador, 2016) detectan la caída con el acelerómetro y giroscopio del celular, realizan cuatro experimentos: cuando la persona está escribiendo y cae reportan una exactitud de reconocimiento de 95.8%; cuando el celular está en la bolsa pantalón y la persona cae, reportan una exactitud de reconocimiento de 87.5%; cuando el celular está en la bolsa de la

camisa y la persona cae reportan una exactitud de reconocimiento de 83.3% y cuando el celular está en la cintura de la persona y cae, reportan una exactitud de reconocimiento de 100%. En cada experimento definen experimentalmente los umbrales.

En (Merrouche & Baha, 2016) detectan la caída con la forma del cuerpo humano y siguiendo la cabeza y el centro de masa de una persona. En los experimentos utilizan la base SDU Fall y reportan una exactitud de reconocimiento de 92.98%.

En (Verma, Merchant, Seetharaman, & Yu, 2016) colocan acelerómetros en el pecho y el muslo de una persona, en la fase de aprendizaje obtienen los valores de aceleración presentes en los eventos en donde la persona está: parada, sentada, acostada, camina y se sienta, camina y cae de frente. Los autores analizan la entropía a múltiples escalas y predicen la postura de una persona con un algoritmo de lógica difusa. Los experimentos son realizados con cinco personas y reportan una exactitud de reconocimiento de 92%.

En (Wang, Liu, & Liu, 2016) detectan una posible caída con los cambios de la altura y ancho de una persona, adicionalmente con un método de flujo óptico calculan la rapidez del movimiento de un cuerpo, por último confirman el evento de caída comparando la postura del cuerpo con una base de posturas, en las cuales la persona está tirada en el piso. En las pruebas utilizan una base de datos pública, la cual contiene las actividades: caminar, sentarse, pararse, caer, entre otras, realizadas en diferentes lugares, con diferentes condiciones de iluminación y textura. Los autores reportan una exactitud de reconocimiento de 81.3%.

En (Kostopoulos, Kyritsis, Deriaz, & Konstantas, 2016) reconocen la caída comparando la información del rebote presente cuando la persona cae y su movimiento residual con umbrales definidos experimentalmente. En las pruebas utilizan 384 caídas simuladas y 412 secuencias de acciones cotidianas (caminar, subir escaleras, bajar escaleras, levantarse y sentarse en una silla), reportan una sensibilidad del 93.48%, una especificidad del 98.54% y una exactitud del 96.01%.

En (Kepski & Kwolek, 2017) detectan una posible caída con la información obtenida con un acelerómetro y la confirman con información visual. Cuando detectan una aceleración mayor que un umbral, con un detector de posturas confirman la caída. En las pruebas con la base de datos URDF reportan una exactitud de reconocimiento de 99.55%.

En (Baldewijns, Debar, Mertes, Croonenborghs, & Vanrumste, 2017) utilizan un sistema con cinco cámaras RGB y fusionan la información visual de las cámaras. En las pruebas utilizan una base con caídas reales obtenidas al monitorear por varios meses a seis adultos mayores en sus casas. Dicha base contiene veintiséis caídas, once de las cuales son captadas sin oclusiones. Los autores calculan cinco características a partir de la caja y la elipse envolvente de la persona, entrenan su algoritmo con las once caídas reales y 268.7 horas de acciones de la vida diaria, por último en sus experimentos reportan una precisión del 70%.

En (Otanasp & Boonbrahm, 2017) detectan la pre-caída (evento previo a la caída) con la información visual adquirida con tres cámaras Kinect, para lo anterior comparan la aceleración de la cabeza con un umbral ajustado dinámicamente y el rango del centro de gravedad con el área de soporte de la persona. En las pruebas utilizan las acciones: caerse, sentarse, doblarse y acostarse, el umbral se re-calcula con cada persona tomando en cuenta su altura y la ubicación de su centroide. Los autores reportan una exactitud de reconocimiento de 96.2%.

En (Alaoui, El Hassouny, Thami, & Tairi, 2017) detectan la caída calculando el flujo óptico y la distribución de Von Mises. En los experimentos con la base DataBase\_Charfi2012, la cual contiene las acciones: dormir, sentarse, caídas, entre otras, reportan una precisión de reconocimiento de caídas de 91%.

En (Sannino, De Falco, & De Pietro, 2017) detectan una caída en una ventana de tiempo con la información de un sensor de posición colocado en el pecho de una persona y un conjunto de reglas si-entonces. Con los datos colectados por el sensor calculan en una ventana de tiempo: el promedio de la posición, desviación estándar, valor máximo, valor mínimo, diferencia entre el valor máximo y el mínimo, con esta información un algoritmo de evolución diferencial optimiza las reglas y los umbrales. En las pruebas con una base creada en el Departamento de Sistemas Inteligentes de Eslovenia, la cual contiene eventos de caídas y de la vida diaria, reportan una exactitud de reconocimiento del 80.5%.

En (Kao, Hung, & Huang, 2017) detectan la caída con la información adquirida con un reloj inteligente y seleccionan características con un algoritmo genético, con su propuesta evitan el sobre ajuste y reducen la complejidad de la clasificación. En las pruebas utilizan las acciones: caminar, sentarse, estar parado, saltar, caer de frente, caer de espaldas, caer lateralmente a la derecha y a la izquierda. Los autores reportan una exactitud de reconocimiento de 94%.

En (Adhikari, Bouchachia, & Nait-Charif, 2017) detectan posturas con la información adquirida con un Kinect y seleccionan dieciséis características con una red neuronal convolucional. En los experimentos, con una base de acciones creada por los autores, la cual contiene las acciones: estar parado, sentarse, acostarse, doblarse entre otras, reportan una exactitud de reconocimiento de 74%.

En (Mousse, Motamed, & Ezin, 2017) detectan la caída con la información visual obtenida por dos cámaras RGB, distinguen cuando la persona está acostada en el piso calculando el porcentaje de área del cuerpo sobre el piso y comparan con un umbral el tiempo transcurrido entre la postura en donde la persona no está sobre el piso y cuando está sobre el piso. El sistema asume que la persona cayó cuando el tiempo de cambio de postura es menor o igual a 1.5 segundos y no hay movimiento en al menos tres segundos. En las pruebas con la base pública Multiview fall dataset reportan una sensibilidad de 95.8% y una especificidad de 100%.

En (Pattamaset, Charoenpong, Charoenpong, & Chianrabutra, 2017) utilizan imágenes RGB, colocan marcadores en dieciséis articulaciones del cuerpo de la persona y calculan las posiciones 3D de las articulaciones con técnicas de visión estéreo, finalmente se calculan vectores corporales. Los autores calculan el ángulo entre el vector del cuerpo y el eje vertical, así como la velocidad del centroide de la persona. En las pruebas veinte personas caminan y caen en dirección de la cámara: de frente, de lado derecho y de lado izquierdo, los resultados indican una exactitud de 100%.

La Tabla 2 resume los trabajos presentados, se indica con una X el tipo de información utilizada en cada trabajo, la cual puede ser imagen RGB, RGBD o la obtenida por sensores inerciales (SI). En la columna SC se muestra una X cuando en el trabajo seleccionan características, sólo en dos trabajos lo hacen: en uno utilizan AG y en el segundo una Red Neuronal Convolucional. En la columna Reconocimiento se menciona la técnica de reconocimiento de caída utilizada. En la columna Características se mencionan las características utilizadas en cada trabajo. Por último en la columna Bases de Pruebas menciona el nombre de la base de datos pública utilizada en los experimentos. En la Tabla 2 se observa que la mayoría de los trabajos evalúan sus propuestas con bases propias.

**Tabla 2 Trabajos relacionados con detección de caídas**

Detección de caídas							
	RGBD	RGB	SI	SC	Reconocimiento	Características	Bases de Pruebas
(Brulin, Benezeth, & Courtial, 2012)		X			Lógica Difusa	Centro de gravedad, dimensiones de la caja envolvente a la persona y la distancia del piso al centro de gravedad.	
(Kawatsu, Li, & Chung, 2013)	X				Comparación con umbrales	Se calcula la distancia al piso de cada articulación y cuando la máxima distancia es menor a un umbral se determina que la persona cae al piso.	
(Kwolek & Kepski, 2014)	X		X		SVM	Cinco relaciones que describen la forma de la nube de puntos de una persona y lecturas de sensores inerciales.	UR Fall Detection
(Ma, Wang, Xue, Zhou, Ji, & Li, 2014)	X				Máquina de aprendizaje extremo	Curvatura de la silueta.	SDUFall y MultiCam
(Nguyen, Le, Do, Duong, Thai, & Tran, 2014)		X			Comparación con umbrales	A partir del rectángulo y elipse envolvente a la persona se calcula su razón de aspecto y ángulo.	
(Soewittoa, Irwana, Antonyová, & Gunawana, 2015)			X		Comparación con umbrales	Lectura se sensores inerciales.	

(Bian, Hou, Chau, & Magnenat-Thalman, 2015)	X				Árbol de decisión aleatorio y SVM	Movimiento de las articulaciones clave de la persona.	
(Wang, Zhang, Dong, Lu, Zhang, & He, 2015)		X			SVM	Calcula área, perímetro, altura, anchura y razón de altura/anchura del rectángulo envolvente a la persona; momentos invariantes de Hu; un ángulo formado entre los vectores que pasan por el centro del rectángulo envolvente y los vértices superiores, velocidad de las coordenadas baricéntricas del cuerpo humano.	
(Liu, Wang, Lv, & Cui, 2015)	X				Comparación con umbrales	Razón de aspecto del rectángulo envolvente a la persona, la inclinación de la misma y el centro de gravedad de la silueta.	
(Stone & Skubic, 2015)	X				Arboles de decisión	Altura de la persona, posición del centroide para determinar cuando la persona está parada, velocidad y aceleración de las características extraídas.	
(Kaneko & Cao, 2016)	X				KNN	251 características de auto-correlación local cúbica de orden superior.	Robot móvil
(Baptista-Ríos, Martínez-García, Losada-Gutiérrez, & Marrión-Romera, 2016)		X			SVM	Conjunto de descriptores.	CAVIAR y KTH
(Gasparrini, y otros, 2016)	X			X	Comparación con umbrales	Posición de articulaciones y lectura de sensores inerciales.	
(Valcourt, De La Hoz, & Labrador, 2016)				X	Comparación con umbrales	Posición de articulaciones y lectura de sensores inerciales.	
(Merrouche & Baha, 2016)	X				Comparación con umbrales	Forma del cuerpo humano, seguimiento de la cabeza y del centro de masa.	SDUFall
(Verma, Merchant, Seetharaman, & Yu, 2016)				X	Lógica Difusa	Lectura de sensores inerciales.	
(Wang, Liu, & Liu, 2016)		X			Comparación con umbrales	Ancho, altura de la persona y flujo óptico.	Fall detection

(Kostopoulos, Kyritsis, Deriaz, & Konstantas, 2016)			X		Comparación con umbrales	Lectura de sensores inerciales.	
(Kepski & Kwolek, 2017)	X		X		SVM y KNN	Relaciones de altura y ancho de la caja envolvente de la persona, distancia del centroide de la persona al piso, la relación de la altura de la caja envolvente a la persona, la altura física de la persona proyectada en el mapa de profundidad, desviación estándar del centroide, la relación del número de puntos que pertenecen a la persona contenidos en un cuboide circundante de cuarenta cm de altura desde el suelo, con respecto a un número total de puntos que pertenecen a la persona y lecturas de sensores inerciales.	URDF
(Baldewijns, Debar, Mertes, Croonenborghs, & Vanrumste, 2017)		X			SVM	Cinco características calculadas a partir de la caja envolvente de la persona y de la elipse envolvente.	
(Otanasp & Boonbrahm, 2017),	X				Comparación con umbrales ajustados dinámicamente	Posición de la cabeza y centro de gravedad.	
(Alaoui, El Hassouny, Thami, & Tairi, 2017)		X				Movimiento utilizando flujo óptico y distribución de Von Mises.	DataBase_Charfi2012
(Sannino, De Falco, & De Pietro, 2017)			X		Reglas si-entonces optimizadas usando evolución diferencial	Lecturas de sensores.	
(Kao, Hung, & Huang, 2017)			X	AG	SVM	Lecturas de sensores.	
(Adhikari, Bouchachia, & Nait-Charif, 2017)	X			RN C	Red neuronal convolucional	Imagen RGBD de 156X128.	

(Mousse, Motamed, & Ezin, 2017)		X			Comparación con umbrales	Porcentaje de área del cuerpo humano sobre el piso.	Multiview fall dataset
(Pattamaset, Charoenpong, Charoenpong, & Chianrabutra, 2017)		X			Comparación con umbrales	Vectores corporales calculados a partir de dieciséis articulaciones del cuerpo.	

### 2.3. Clasificadores

En los trabajos de reconocimiento de acciones se utilizan diversos clasificadores, en (Jr & J, 2015) clasifican gestos en ambientes virtuales con DTW, SVM, HMM, entre otros. Las acciones son tratadas como secuencias de imágenes, a partir de las cuales calculan un conjunto de descriptores y crean series de tiempo.

A continuación se presentan trabajos relacionados con el tema de clasificación:

En (Pravdova, Walczak, & Massart, 2002) alinean señales analíticas con la deformación temporal dinámica (DTW) y la deformación correlativa optimizada (COW), después analizan las señales con PCA. En las pruebas utilizan perfiles cromatográficos y espectroscópicos con datos reales y simulados, cualquiera de los dos métodos DTW y COW impacta positivamente en la fase de clasificación.

En (Ratanamahatana & Keogh, 2004) mejoran la exactitud de similitud entre series de tiempo cuando utilizan DTW, agregan restricciones para generar el camino de comparación. Algunas de las restricciones son: el tamaño de la ventana de comparación, la pendiente del camino, selección de los siguientes vecinos, entre otros; con la información de las series de tiempo ajustan las restricciones. En los experimentos utilizaron tres bases de series de tiempo. La primera contiene las coordenadas  $(x, y)$  de la mano, las secuencias son clasificadas en la clase uno cuando hace el movimiento de apuntar con una pistola y en la clase dos cuando hace el movimiento de apuntar con el dedo índice de la mano. La segunda base contiene casos de fallas en los instrumentos de una planta nuclear, contiene dieciséis clases con cuatro características cada una y cincuenta ejemplos por cada clase. La tercera base contiene palabras escritas a mano, utilizan 272 series de tiempo las cuales corresponden a las palabras *the*, *to*, *be* y *that*, extraen cuatro características. Los autores reportan los siguientes resultados: con distancia euclidiana obtienen un error de 5.5%, 11% y 4.78%, con DTW tradicional el error es 1.0%, 0% y 1.1% y con el método propuesto (DTW con restricciones) es 0.5%, 0% y 0.37% para cada base de prueba.

En (Li & Liu, 2010) optimizan los parámetros del clasificador RVM con un algoritmo genético. En los experimentos comparan el desempeño del método propuesto, una red neuronal con retro-propagación y un SVM. La exactitud obtenida con RVM-AG es de 90%, con SVM es en promedio de 80% y con la red neuronal es menor de 75%.

En (Li & Prakash, 2011) clasifican secuencias de acciones, proponen un método capaz de aprender la dinámica de las articulaciones en las secuencias, minimizar los retardos de tiempo al ejecutar las acciones y producir características; con el método propuesto crean clústeres de series de tiempo. El método propuesto utiliza un nuevo algoritmo EM que aprende los parámetros al momento de crear los clústeres. En los experimentos utilizan las bases MOCAPPOS y MOCAPANG, las cuales contienen secuencias de acciones correr y caminar, las secuencias son ajustadas para que todas tengan la misma longitud. El desempeño del método propuesto para crear clústeres se compara con los siguientes: PCA con K-medias, DFT con K-medias, DTW con K-medias, KF con K-medias, con la entropía condicional miden la calidad de los clústeres, cuanto más baja es, la calidad es mayor. Los autores reportan una entropía condicional de 0.38 y de 0.10 en los clústeres creados con el algoritmo EM propuesto para cada base de prueba, e indican la entropía es mayor cuando se utilizan los otros métodos.

En (Carmona & Climent, 2012) comparan el desempeño de DTW y Modelos Ocultos de Markov (HMM por sus siglas en inglés) en el reconocimiento de gestos. En las pruebas reconocen el movimiento de la mano, utilizan características de posición, orientación y velocidad de la mano. En los experimentos con una base que contiene 750 muestras de diez diferentes tipos de gestos, adquiridos con el Kinect, evalúan la robustez de DTW y HMM a diferentes distancias y morfología de la persona. Los autores reportan una exactitud de reconocimiento de 98.84% cuando utilizan DTW con KNN y del 96.46% cuando utilizan HMM.

En (Nakamura, Taki, Nomiya, Seki, & Uehara, 2013) proponen la métrica angular para similitud de forma (AMSS por sus siglas en inglés) mediante la cual cada serie de tiempo se transforma en una secuencia de vectores. La métrica AMSS propuesta mide correctamente la similitud entre señales aun cuando éstas cambian en el tiempo, amplitud, corrimiento y escala. En las pruebas clasifican con 1NN series de tiempo de la base UCR con la métrica AMSS, la distancia euclidiana, DTW, entre otros. Los resultados cuando utilizan DTW presentan un error promedio de 0.16, también realizan otros experimentos en los cuales combinan AMSS con otras de las métricas evaluadas, los resultados sugieren que al combinar AMSS con DTW el error es reducido a 0.13.

En (Raheja, Minhas, Prashanth, Shah, & Chaudhary, 2015) comparan el desempeño de DTW y Modelo ocultos de Markov (HMM) cuando reconocen gestos de la mano adquiridos con el Kinect. La exactitud del reconocimiento con los dos métodos es del 90%, sin embargo cuando utilizan DTW observan las siguientes ventajas: compara acciones con diferentes duraciones y detecta la similitud cuando la misma acción es realizada a diferente velocidad.



En (Izakian, Pedrycz, & Jamal, 2015) crean clústeres de series de tiempo con DTW. Con DTW reconocen patrones de comportamientos similares en señales aun cuando éstas son realizadas en diferentes periodos de tiempo. El centro del clúster se calcula con K-Medias y Fuzzy C-Medias, en los experimentos utilizaron series de tiempo de UCR.

En (Ayumi & Fanany, 2016) comparan el desempeño de RVM y SVM cuando reconocen acciones. En los experimentos con la bases Microsoft Research Cambridge-12 reportan una exactitud de reconocimiento de 98.75 con RVM y de 94.6% con SVM, con la base Badminton Sports reportan una exactitud de reconocimiento de 97% con RVM y de 95.89% con SVM.

En la Tabla 3 se resumen los trabajos presentados, con una X se indica el clasificador utilizando en los trabajos revisados, en la misma tabla se observa que DTW con KNN es el clasificador más utilizado cuando se trabaja con series de tiempo.

**Tabla 3 Clasificadores utilizados con series de tiempo**

	Clasificador						
	DTW-KNN	COW	Red Neuronal	SVM	HMM	RVM	Crean Clúster
(Pravdova, Walczak, & Massart, 2002)	X	X					
(Ratanamahatana & Keogh, 2004)	X						
(Li & Liu, 2010)			X	X		X	
(Li & Prakash, 2011)							X
(Carmona & Climent, 2012)	X				X		
(Nakamura, Taki, Nomiya, Seki, & Uehara, 2013)	X						
(Raheja, Minhas, Prashanth, Shah, & Chaudhary, 2015)	X				X		
(Izakian, Pedrycz, & Jamal, 2015)	X						
(Ayumi & Fanany, 2016)				X		X	

## **2.4. Selección de características**

En la actualidad el mundo es captado con diversos sensores como son cámaras, acelerómetros y láser. Lo anterior hace posible caracterizarlo mediante un conjunto con diversos atributos a partir de los cuales se reconocen objetos, actividades, entre otros. Sin embargo utilizar todas las características podría afectar negativamente la exactitud del reconocimiento. Es por esta razón que con métodos de selección de características se remueve información redundante o irrelevante. A continuación se describen los métodos de selección de características (Andrew R. Webb, 2011):

1. **Método de Filtrado:** las características son seleccionadas por su valor de importancia, por ejemplo cuando se utiliza PCA para calcular el valor de importancia, las características con un porcentaje alto son seleccionadas porque representan mayor variabilidad de la información.
2. **Método de encapsulamiento:** se evalúa la importancia de las características cuando son utilizadas con el clasificador, este último es una caja negra. Un subconjunto de características es mejor en la medida que incrementa la exactitud de la clasificación.
3. **Método embebido:** las características son seleccionadas como se describe en el método de encapsulamiento, pero además se toman decisiones de diseño de clasificador para lograr la efectividad de clasificación.

A continuación presentamos trabajos relacionados con la selección de características:

En (Chandrashekar & Sahin, 2014) (Jovi'c, Brkic, & Bogunovic, 2015) (Tang, Alelyani, & Liu, 2014) realizan una revisión de varios métodos para seleccionar características cuyo objetivo es mejorar la exactitud de la clasificación. Los autores describen diferentes criterios de análisis de características como son correlación, información mutua, chi cuadrada, Fisher, entre otros, los cuales son utilizados en los métodos de filtrado. Además presentan dos tipos de métodos de encapsulamiento. Los del primer tipo seleccionan características secuencialmente comienzan con el conjunto de características completo, posteriormente remueven características, otra alternativa es comenzar con un conjunto vacío de características y posteriormente agregarlas, algunos ejemplos son algoritmos de búsqueda secuencial: hacia adelante, hacia atrás y flotante. El segundo tipo realiza una búsqueda heurística de las características, algunos ejemplos son algoritmos genéticos, programación genética, colonia de hormigas entre otros.

En (Siedlecki & Sklansky, 1993) seleccionan características con un AG, cuando reducen el conjunto de características también disminuyen el costo y el tiempo de clasificación, además aumenta la confiabilidad en la exactitud de clasificación. Los autores descubren que el AG es una herramienta poderosa para seleccionar características cuando el tamaño del conjunto inicial supera los veinte atributos.

En (Yusta, 2009) seleccionan características con máquinas de aprendizaje, proponen tres estrategias meta-heurísticas: GRASP, búsqueda Tabú y algoritmos meméticos, las cuales son comparadas con un algoritmo genético, selección secuencial flotante hacia adelante (SFFS) y hacia atrás (SBFS). Con las técnicas meta-heurísticas encuentran soluciones óptimas sin explorar todo el espacio de búsqueda. En las pruebas utilizan las bases de datos: Spambase con 57 atributos, 2 clases y 4601 casos; Waveform con 40 atributos, 2 clases y 3347 instancias; Ionosphere con 34 atributos, 2 clases y 351 casos; Vehicle con 18 variables, 4 clases y 435 instancias; Wisconsin-cáncer de pecho con 30 atributos, 2 clases y 569 casos; German-Data-Numeric con 24 atributos, 2

clases y 1000 instancias. Cuando utilizan meta-heurísticas mejoran la clasificación, aunque en general con todas las estrategias obtienen resultados similares.

En (Vieira, Sousa, & Kaymak, 2012) tratan el problema de seleccionar características como un problema multi-objetivo, en el cual se maximiza el poder de discriminación del conjunto de características seleccionadas, se minimiza el error de clasificación, entre otros objetivos. Los autores seleccionan características mediante un criterio difuso, en los experimentos utilizan ocho bases del repositorio UCI y comparan el desempeño del clasificador cuando se utilizan todas las características, las seleccionadas con un algoritmo de optimización de colonia de hormigas y las seleccionadas con el criterio difuso, reportan una exactitud de reconocimiento de 82.7%, 90.0% y 93.5%, respectivamente.

En (Shmilovici & Ben-Shimon, 2013) proponen una estrategia de ordenamiento de las características de acuerdo a su importancia, posteriormente las seleccionan conforme su influencia con un SVM. El objetivo del experimento es predecir la aceleración angular de uno de los enlaces de un brazo de robot, por esta razón utilizan una base con movimientos de un brazo de robot con 32 características, de las cuales cinco son seleccionadas y evaluadas en la predicción. La precisión es la misma cuando utilizan las 32 y las cinco características.

En (Li, Shang, Qu, & Liang, 2014) proponen un método de selección de características basado en aprendizaje Manifold. Con la métrica de distancia utilizada maximizan el margen de las clases en espacios de menor dimensión de los Manifold embebidos, además minimizan la norma L1. Finalmente optimizan con evolución diferencial con manejo de restricciones dinámicas. En los experimentos con la base UCI SPECTF Heart, la cual tiene 267 instancias, con 44 características, seleccionan veinticinco características con su método propuesto y obtienen un 93.7% de exactitud en el reconocimiento. En otro experimento con un algoritmo de selección secuencial hacia adelante seleccionan veintidós características y obtienen una exactitud del 90.2%.

En (Bennasar, Hicks, & Setchi, 2015) proponen dos métodos no lineales de selección de características: maximización de información mutua condicional y maximización de información mutua condicional normalizada. Cuando los autores utilizan información mutua y el criterio de maximización resuelve el problema de sobreestimar las características más significativas. En los experimentos utilizan once bases y comparan el método propuesto con cuatro métodos de selección de características. De acuerdo con los resultados reportados con el método de maximización de información mutua condicional reducen el error promedio de la clasificación en un 6%.

En (Zou, Ni, Zhang, & Wang, 2015) proponen un método de selección de características basado en aprendizaje profundo, tratan el problema de selección de características como un problema de reconstrucción de las mismas. En las pruebas utilizan 2800 imágenes capturadas remotamente y clasificadas en siete categorías. Por ejemplo pasto, bosque, región industrial, región residencial

entre otras. Los autores reportan una exactitud de reconocimiento del 75% cuando utilizaron el 90% de las imágenes para el entrenamiento. La exactitud reportada es superior a la obtenida cuando se utilizan otros métodos.

En (Yan & Yang, 2015) proponen un método de selección escasa de característica discriminativo con el objetivo de minimizar la reconstrucción inter-clase y maximizar la reconstrucción entre clases con el subconjunto de características seleccionadas. En la búsqueda de características utilizan un algoritmo glotón y uno de selección conjunta, seleccionan las características con mayor correlación y con mayor poder de discriminación dentro de todo el espacio de búsqueda. En los experimentos utilizan cinco bases de datos, tres de ellas pertenecen al repositorio UCI, la cuarta es una base de dígitos escritos a mano, la quinta es una base con 16,128 caras de 38 personas en nueve diferentes poses y 64 condiciones de iluminación diferentes. En la prueba con la base de rostros, la cual contiene más de novecientas características, reportan un reconocimiento de 0.59 cuando utilizan menos de cincuenta características seleccionadas con el método propuesto. Por otra parte con otros métodos seleccionan más de cien características y reportan un reconocimiento menor a 0.5. En las bases restantes la tasa de reconocimiento con el método propuesto es mayor a la obtenida con otros métodos de selección de características.

En (Nazarpour & Adibi, 2015) implementan un modelo supervisado de reducción de características. En el modelo supervisado con la información de las clases optimizan la exactitud de la clasificación, reducen las características utilizando múltiples núcleos en dos etapas. En los experimentos utilizan diez diferentes bases de datos del repositorio UCI: dígitos escritos a mano, imágenes de objetos, entre otras. Los resultados con el método propuesto son comparados con los resultados obtenidos con métodos tradicionales de selección de características lineales como PCA y LDA, y no lineales como KDA, KSDA, KMSDA, entre otros. Los resultados muestran una exactitud del reconocimiento superior en algunas bases de datos cuando utilizan el método propuesto y en las bases restantes es comparable con la exactitud obtenida con otros métodos no lineales.

En (Liu & Zhang, 2016) con las características seleccionadas mediante un método de filtrado se crean clústeres en aplicaciones de aprendizaje no supervisado, proponen y evalúan tres métricas: distancia efectiva basada en laplacianos y dos distancias basadas en puntajes de dispersión. Los autores compararon las métricas propuestas con la distancia euclidiana, en los experimentos utilizan bases del repositorio UCI, Yale Face y ORL Face con 1024 características. Los resultados de los experimentos superan los reportados previamente en el estado del arte, por ejemplo con la base de rostros Yale Face con la métrica F obtienen un 0.38 de reconocimiento y el resultado previamente reportado es menor al 0.35.

En (Zare & Niazi, 2016) seleccionan características con un método de filtrado, proponen maximizar la probabilidad de la estructura del aprendizaje entre las características. Las características son seleccionadas con una estructura de grafos dirigidos entre las características, basados en el

concepto de manto de Markov. En las pruebas utilizan nueve bases de datos con diferentes cantidades de características (desde ocho hasta 100,000). Los autores reportan una mejor exactitud de reconocimiento cuando utilizan las características seleccionadas con el método propuesto y varios clasificadores, lo anterior sugiere que el subconjunto de características seleccionadas es robusto.

En (Yang, Wang, Liu, Deng, & Hedrick, 2017) entrenan los controladores de un vehículo autónomo, seleccionan con una red neuronal convolucional las características con mayor influencia en el desempeño de los controladores. A partir de un conjunto de imágenes extraen tres tipos de características: las relacionadas con el camino que debe seguir el vehículo, las relacionadas con los laterales del camino y las relacionadas con el cielo. Los resultados obtenidos con el método propuesto sugieren: un funcionamiento correcto de los controladores cuando se utilizan las características relacionadas con el camino; el vehículo navega bien en diferentes escenario cuando además se usan las características relacionadas con los laterales del camino y las características relacionadas con el cielo no mejoran la navegación del vehículo por lo tanto son irrelevantes.

En (Uysal & Murphey, 2017) seleccionan características, la cuales son utilizadas con modelos de aprendizaje profundo en aplicaciones de sentimiento (clasificación de textos). En los experimentos utilizan cuatro bases de textos, cada una con diferentes características. Los autores seleccionan las características con dos métodos: el primero utiliza una bolsa de palabras, el segundo utiliza palabras semánticas embebidas. Los resultados de los experimentos muestran una mejor clasificación cuando utilizan las características seleccionadas, en contraste con utilizar el conjunto completo de características.

La Tabla 4 muestra resume los trabajos presentados. Con una X se indica la técnica de selección de características implementada en los trabajos revisados. En la Tabla 4 se observa que las técnicas de computación evolutiva y máquinas de aprendizaje son las más utilizadas.

**Tabla 4 Trabajos relacionado con seleccionar características**

Selección de Características									
	CE	RNC	LD	MA	MNL	Varios métodos de filtrado y envoltura	PCA	LDA	Comparación de métricas por filtrado
(Siedlecki & Sklansky, 1993)	X								
(Yusta, 2009)	X								
(Vieira, Sousa, & and Kaymak, 2012)	X		X						
(SHMILOVICI & BEN-SHIMON, 2013)				X					
(Li, Shang, Qu, & Liang, 2014)	X			X					
(Bennasar, Hicks, & Setchi, 2015)					X				
(Zou, Ni, Zhang, & Wang, 2015)				X					

(Yan & Yang, 2015)				X					
(Nazarpour & Adibi, 2015)				X	X		X	X	
(Liu & Zhang, 2016)									X
(Zare & Niazi, 2016)									X
(Yang, Wang, Liu, Deng, & Hedrick, 2017)		X							
(Uysal & Murphey, 2017)						X			

CE=Computación evolutiva

RNC=Red neuronal convolucional

LD=Lógica difusa

MA=Máquina de aprendizaje

MNL=Métodos no lineales

## 2.5. Resumen

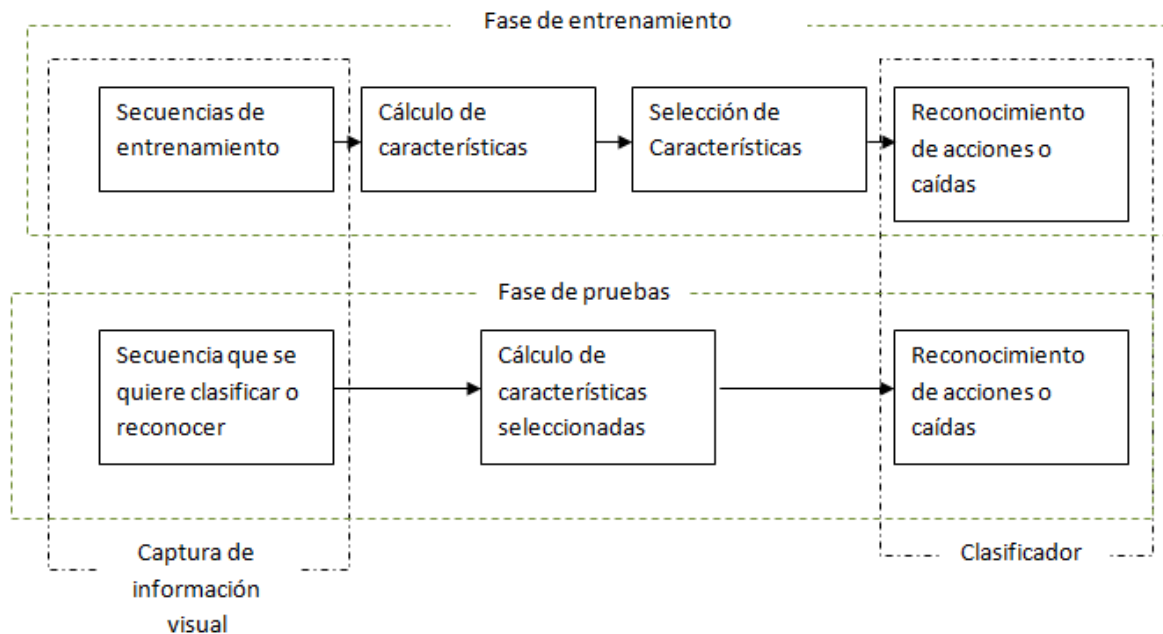
En los trabajos relacionados con el reconocimiento de acciones y de caídas mostrados en las secciones 2.1 y 2.2, respectivamente, se observa como la combinación de un conjunto de características y un clasificador cambia la exactitud de reconocimiento aun cuando utilizan la misma base de pruebas. En las Tablas 1 y 2 se presentan las características utilizadas, las cuales en la mayoría de los trabajos son definidas experimentalmente. En algunos casos los autores seleccionan características después de observar los cambios en las características propuestas. Adicionalmente los conjuntos de características utilizados comparten algunas, por ejemplo ángulos y posiciones de las articulaciones.

En la sección 2.3 se revisan los clasificadores utilizados en el estado del arte con el objetivo de utilizar en esta tesis los que han demostrado un mejor desempeño, como es el caso de DWT, finalmente en la sección 2.4 se observa como la exactitud del reconocimiento mejora cuando se utilizan características seleccionadas por algún algoritmo. En el siguiente capítulo se describe la propuesta de esta tesis para reconocer acciones y caídas.

## Capítulo 3: Reconocimiento de acciones y caídas

El sistema de reconocimiento de acciones y caídas propuesto en esta tesis consiste de las siguientes fases (Fig. 1):

- **Fase de entrenamiento:** en todas las acciones o caídas de entrenamiento se detectó la nube de puntos en 3D correspondiente a una persona o las posiciones  $(x, y, z)$  de las articulaciones de su esqueleto. Después se calculó el conjunto de características y finalmente se seleccionaron algunas características de manera iterativa, midiendo la exactitud de reconocimiento con las características seleccionadas. Cuando se utilizaron clasificadores parametrizables, también se ajustaron sus parámetros.
- **Fase de prueba:** en la acción o caída de prueba se detectó la nube de puntos en 3D correspondiente a una persona o las posiciones  $(x, y, z)$  de las articulaciones de su esqueleto. Se calculó el conjunto de características seleccionadas en la fase de entrenamiento y por último se clasificó la acción o caída.



**FIGURA 1.** Fases de entrenamiento y prueba para reconocer acciones y caídas.

En el resto del capítulo se describe cómo se adquirió la información visual, se calcularon y seleccionaron las características y los clasificadores utilizados para reconocer acciones y caídas.

### **3.1. Captura de información Visual**

De acuerdo con la información presentada en el capítulo dos en algunos trabajos utilizan cámaras RGB con las cuales capturan información de textura y color, sin embargo la información varía con los cambios de iluminación. Por otro lado, los trabajos desarrollados recientemente utilizan cámaras RGBD, las cuales proveen información de la distancia entre los objetos y la cámara, con la información mencionada se recupera la forma de los objetos. Las imágenes adquiridas con las cámaras RGBD reciben el nombre de imágenes en profundidad, la información capturada es menos sensible a los cambios de luz y facilita hacer la sustracción de imágenes o filtrar objetos. En (Chen, Wei, & Ferryman, 2013) presentan tres dispositivos para capturar imágenes en profundidad:

1. **Cámara estéreo** (Fig. 2-a): utiliza al menos dos cámaras, calculan la distancia entre los objetos y la cámara. Las cámaras utilizadas usualmente son RGB, por esta razón la información visual varía con los cambios de iluminación, lo cual afecta los cálculos de la profundidad de los objetos.
2. **Tiempo de Vuelo** (ToF, por sus siglas en inglés) (Fig. 2-b): emite pulsos de luz con una cámara, la distancia entre el objeto y la cámara es proporcional al tiempo de vuelo de la señal de luz multiplicado por la velocidad de la luz. Las imágenes adquiridas son de baja resolución de 64x48 hasta 200x200, el rango de alcance en profundidad es de 5 hasta 10 metros, la resolución en profundidad es de 5mm, su campo visual es de 43° en vertical y 69° horizontal, las imágenes no presentan huecos, son invariantes a los cambios de luz, sin embargo es un dispositivo de alto costo.
3. **De luz estructurada** (Fig. 2-c): utiliza una cámara RGB y un sensor de profundidad, una fuente de luz proyecta un patrón de luz infrarroja conocido, es decir una luz estructurada, tiene un sensor CMOS de infrarrojo y la profundidad de los objetos en una escena se calcula mediante triangulación. Con el sensor de profundidad capturan treinta imágenes de 640x480 por segundo, el rango de alcance es de 0.8 hasta 3.5m con una resolución de 1cm, su campo visual es del 43° en vertical y 57° horizontal, es invariante a los cambios de luz, es un dispositivo barato, aunque la información visual capturada presenta huecos. Ejemplos de estos dispositivos son Kinect v1 y Asus Xtion.

En esta tesis se utilizó el Kinect por las ventajas mencionadas. Además en (Chaquet, Carmona, & Fernández-Caballero, 2013) (Aggarwal & Xia, 2014) y otros trabajos revisados en el estado del arte se encuentran referencias a bases de datos públicas tanto de acciones como de caídas con información adquirida con dispositivos de luz estructurada. Algunas de las bases públicas se utilizaron en los experimentos de esta tesis para comparar la exactitud del reconocimiento con los resultados reportados en el estado del arte.



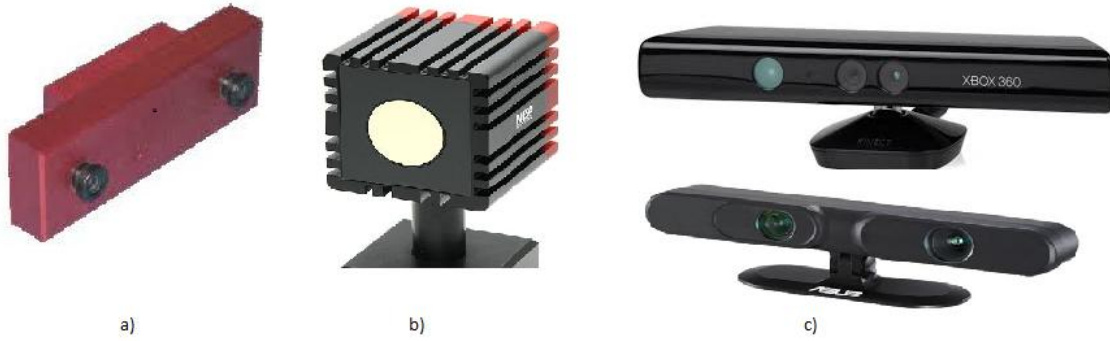


FIGURA 2. a) cámara estéreo, b) tiempo de vuelo (ToF) y c) dispositivos de luz estructurada.

En esta tesis se obtuvo la nube de puntos de una persona como se describe a continuación:

1. Con las posiciones de las articulaciones del esqueleto se calculó un *bounding box*, el cual contiene la nube de puntos de la persona detectada (Fig. 3). Las librerías SDK para Windows, PCL y OpenNI para Windows\Mac\Linux proveen las posiciones  $(x, y, z)$  de las articulaciones del esqueleto.
2. Con una imagen de la escena sin personas y una imagen con persona se calculó la diferencia de imágenes de profundidad (Fig. 4), esta última contiene la nube de puntos de la persona detectada.

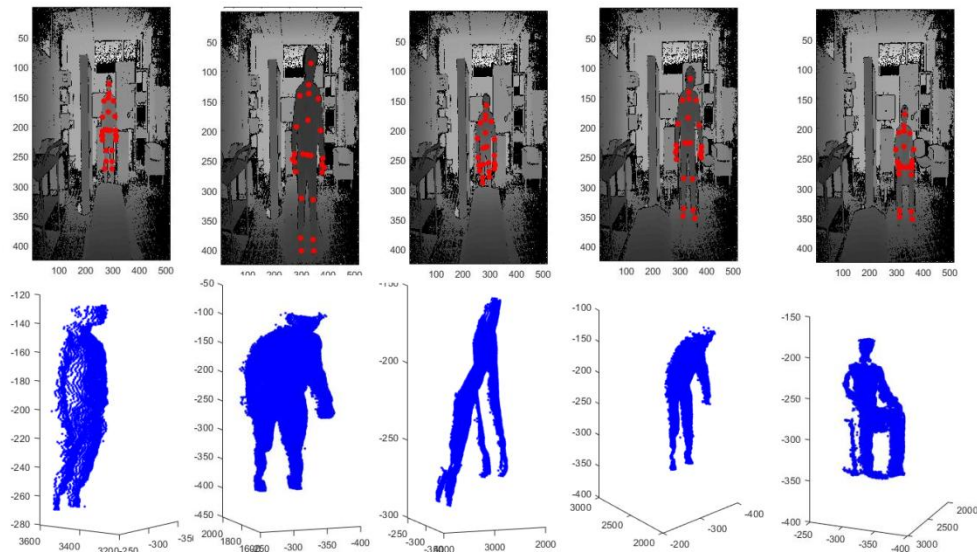
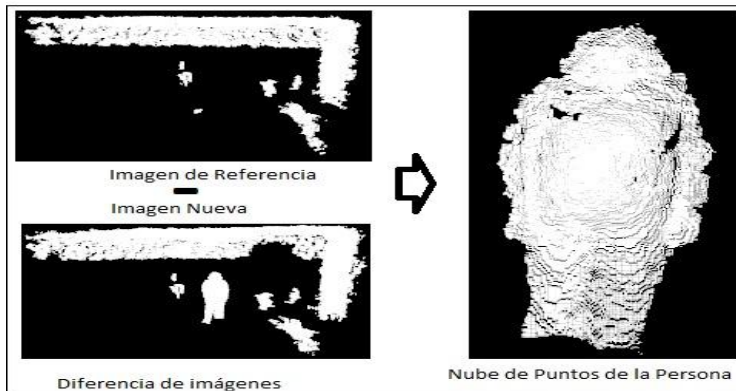


FIGURA 3 Las imágenes mostradas en la parte superior forman parte de la base TST Fall detection dataset v2, los puntos rojos corresponden a las posiciones de las articulaciones del esqueleto. En la parte inferior se muestran las nubes de puntos de la persona detectada.



**FIGURA 4.** La segmentación de la nube de puntos de una persona es el resultado de una diferencia de imágenes.

Con la nube de puntos de la persona detectada en una escena se calculó un conjunto de características, las cuales se describen en la siguiente sección.

### **3.2. Selección de características**

En las Tablas 1 y 2 se observa que en cada trabajo proponen un conjunto de características para reconocer acciones y caídas. Por ejemplo en (Papadopoulos, Axenopoulos, & Daras, 2014) y (Gaglio, Re, & Morana, 2015) analizan el movimiento de las articulaciones del esqueleto y seleccionan las articulaciones con mayor movimiento, con estas últimas calculan el conjunto de características formado con: ángulos, posiciones de articulaciones normalizadas entre otras.

Con las características propuestas en cada trabajo se reporta una exactitud de reconocimiento en varios casos superior al 90%. Por tal en esta tesis se asume que las características utilizadas son importantes para describir las acciones y caídas. Por lo antes expuesto, en esta tesis se propuso un conjunto con algunas características utilizadas en otros trabajos y se agregaron los invariantes 3D a TRE. De acuerdo con los trabajos revisados en el estado del arte no han sido utilizados para reconocer posturas. Después se seleccionaron características con un algoritmo genético, con el algoritmo RFD-SD y con PCA, con el objetivo de eliminar información redundante o irrelevante y maximizar la exactitud del reconocimiento. Las características propuestas son valores escalares invariantes a ciertas deformaciones (traslación, rotación y escala).

### 3.2.1. Conjunto de características

Un conjunto con veinte características (Tabla 5) fue propuesto para reconocer caídas. Las primeras diecisiete se calcularon con los vértices del *bounding box* de la nube de puntos de la persona (Fig. 5) así como las coordenadas de los siguientes puntos:

- P1: coordenadas  $(x, y, z)$  del centroide de la nube de puntos,
- P2: punto más alto en la nube de puntos y
- P3: punto más bajo de la nube de puntos.

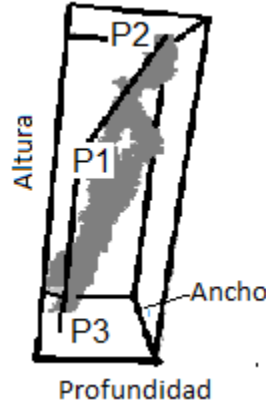


FIGURA 5. En la imagen se muestra el *bounding box* de la nube de puntos de una persona así como los puntos P1, P2 y P3 utilizados en el cálculo de diecisiete características.

Las características C18, C19 y C20 corresponden a los invariantes 3D de segundo orden (Flusser, Zitova, & Suk, 2009) los cuales describen la distribución de la nube de puntos con respecto a los ejes de coordenadas. Estas tres características describen la distribución de la masa y forma de la persona. Los invariantes  $\psi_1$ ,  $\psi_2$  y  $\psi_3$  se calcularon con las siguientes ecuaciones:

$$\psi_1 = v_{200} + v_{020} + v_{002} \quad (1)$$

$$\psi_2 = v_{020}v_{002} + v_{200}v_{002} + v_{200}v_{020} - (v_{011}^2 + v_{101}^2 + v_{110}^2) \quad (2)$$

$$\psi_3 = v_{200}v_{020}v_{002} + 2v_{110}v_{101}v_{011} - (v_{200}v_{011}^2 + v_{020}v_{101}^2 + v_{002}v_{110}^2) \quad (3)$$

La característica de ser invariante a escala y traslación se logra con las siguientes ecuaciones:

$$v_{pqr} = \frac{\mu_{pqr}}{\mu_{000}^w} \quad (4)$$

En donde

$$w = \frac{p+q+r}{3} + 1 \quad (5)$$

Y

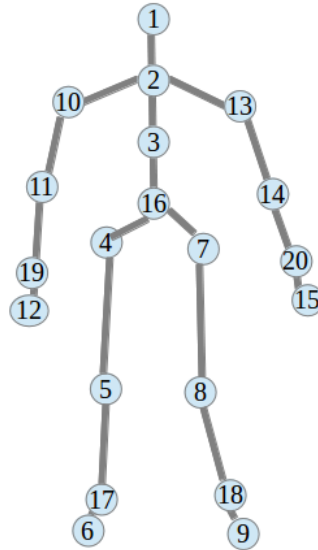
$$\mu_{pqr} = \iiint_{-\infty}^{\infty} (x - x_c)^p (y - y_c)^q (z - z_c)^r f(x, y, z) dx dy dz \quad (6)$$

$p, q$  y  $r$  son enteros positivos y su suma ( $p + q + r$ ) es igual al orden del momento invariante, además  $(x_c, y_c, z_c)$  son las coordenadas del centroide de la nube de puntos de la persona detectada en una imagen.

**Tabla 5. Características calculadas con la nube de puntos de una persona**

Característica	Descripción
C1	Ángulo entre P3 y P1 en el plano XY
C2	Ángulo entre P3 y P1 en el plano XZ
C3	Ángulo entre P1 y P2 en el plano XY
C4	Ángulo entre P1 y P2 en el plano XZ
C5	Ángulo de la persona respecto al piso, plano XY
C6	Ángulo de la persona respecto al piso, plano YZ
C7	Proporción de altura/ancho del <i>bounding box</i>
C8	Proporción de altura/profundidad del <i>bounding box</i>
C9	Proporción de ancho/profundidad del <i>bounding box</i>
C10	Distancia entre P3 y P1
C11	Distancia entre P2 y P1
C12	Distancia del centroide de la persona al piso
C13	Distancia del punto más alto de la persona al piso
C14	Longitud del ancho del <i>bounding box</i>
C15	Longitud de la altura del <i>bounding box</i>
C16	Longitud de la profundidad del <i>bounding box</i>
C17	Cambio de la posición del centroide con respecto a su posición en la primera imagen de la secuencia
C18	$\psi_1$ , Invariante 1 3D a traslación, rotación y escala
C19	$\psi_2$ , Invariante 2 3D a traslación, rotación y escala
C20	$\psi_3$ , Invariante 3 3D a traslación, rotación y escala

Las características utilizadas para reconocer acciones se calcularon con las posiciones  $(x, y, z)$  de las articulaciones del esqueleto. Con la librería SDK de Microsoft se capturan veinte articulaciones (Fig. 6) y con la librería OpenNI se capturan las articulaciones numeradas del uno al quince en la Fig. 6.



**FIGURA 6. Articulaciones del esqueleto.**

Con las posiciones de las articulaciones se calcularon los ángulos  $\alpha, \beta$  y  $\gamma$  entre pares de articulaciones adyacentes. A partir de las quince articulaciones capturadas con OpenNI se formaron catorce pares:  $(1,2), (2,3), (3,4), (3,7), (4,5), (5,6), (7,8), (8,9), (2,10), (10,11), (11,12), (2,13), (13,14), (14,15)$ , y con el SDK Microsoft se formaron diecinueve pares:  $(1,2), (2,3), (3,16), (16,4), (16,7), (4,5), (5,17), (17,6), (7,8), (8,18), (18,9), (2,10), (10,11), (11,19), (19,12), (2,13), (13,14), (14,20), (20,15)$ .

Con cada par de articulaciones adyacentes  $(J_u, J_v)$ , en donde  $(x_u, y_u, z_u)$  son las coordenadas de  $J_u$  y  $(x_v, y_v, z_v)$  son las coordenadas de  $J_v$ . Se calcularon los ángulos coseno  $\alpha, \beta$  y  $\gamma$  de la siguiente forma:

$$\alpha = \cos^{-1} \left( \frac{x_v - x_u}{\|J_v - J_u\|} \right) \quad (7)$$

$$\beta = \cos^{-1} \left( \frac{y_v - y_u}{\|J_v - J_u\|} \right) \quad (8)$$

$$\gamma = \cos^{-1} \left( \frac{z_v - z_u}{\|J_v - J_u\|} \right) \quad (9)$$

También se calculó el ángulo  $\theta$  formado entre tres articulaciones conectadas  $J_u, J_v$  y  $J_r$  :

$$\theta = \cos^{-1} \left( \frac{(J_p - J_q) \cdot (J_r - J_q)}{\|J_p - J_q\| \|J_r - J_q\|} \right) \quad (10)$$

Con las quince articulaciones capturadas con OpenNI se formaron catorce triadas de articulaciones conectadas: (1,2,3), (1,2,10), (1,2,13), (2,10,11), (10,11,12), (2,13,14), (13,14,15), (2,3,4), (2,3,7), (3,4,5), (3,7,8), (4,5,6), (7,8,9), (3,4,7) y con el SDK de Microsoft se formaron dieciocho triadas: (1,2,3), (1,2,10), (1,2,13), (2,10,11), (10,11,19), (11,19,12), (2,13,14), (13,14,20), (14,20,15), (2,3,16), (3,16,4), (3,16,7), (16,4,5), (16,7,8), (4,5,17), (5,17,6), (7,8,18), (8,18,9).

Las características utilizadas son ángulos porque son invariantes a la escala, rotación y traslación, por esta razón los esqueletos no se normalizaron. El conjunto contiene 56 características cuando se trabajó con acciones capturadas con OpenNI y de 75 cuando se trabajó con acciones capturadas con el SDK de Microsoft. De acuerdo con los trabajos mostrados en el estado del arte, los ángulos son comúnmente usados para reconocer acciones, pero sólo utilizan los que se forman entre algunas articulaciones seleccionadas experimentalmente. En otros trabajos las posiciones de las articulaciones son seleccionadas, sin embargo estas últimas deben normalizarse. Por esta razón no se utilizaron en esta tesis.

En las siguientes secciones se presentan los algoritmos utilizados para seleccionar características.

### 3.2.2. Algoritmo Genético

Un algoritmo genético (Coello, 2004) (Goldberg, 2013) es un algoritmo heurístico de búsqueda adaptativo, está inspirado en el proceso de selección natural, donde los individuos compiten entre ellos para ganar recursos de la naturaleza. En general los individuos mejor calificados sobreviven y los más débiles tienden a desaparecer, por esta razón conforme evolucionamos la población está mejor calificada. En un algoritmo genético los individuos representan soluciones del problema y se mide la aptitud de cada individuo. El AG utiliza mecanismos de selección, recombinación y mutación de individuos con el propósito de crear la nueva población de hijos y al evolucionar la población de individuos las soluciones están mejorando. Se recomienda utilizar el AG en espacios de búsquedas grandes en donde los métodos tradicionales no encuentran soluciones óptimas.

En esta tesis se utilizan individuos binarios con longitud igual al número total de características. Por ejemplo para reconocer la caída se propuso un conjunto con veinte características (Tabla 5), por lo tanto cada individuo tiene veinte componentes (Tabla 6).

**Tabla 6. Ejemplo de un individuo binario con veinte características**

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
1	0	0	1	0	0	0	1	1	1	0	0	0	1	0	1	0	0	0	1

Un individuo binario tiene valor de cero o de uno en cada característica, cuando la característica tiene valor de uno es seleccionada, en caso contrario cuando tiene valor de cero la característica no es seleccionada. En la Tabla 6 se muestra un ejemplo de individuo, el cual selecciona las características C1, C4, C8, C9, C10, C14, C16, C20.

La ejecución del algoritmo genético se describe a continuación:

- **Paso 1:** Crear la población inicial con N individuos
- **Paso 2:** Calcular la aptitud de los individuos. Con el subconjunto de características seleccionadas por el individuo, un conjunto de entrenamiento se utiliza para ajustar el modelo de los clasificadores parametrizables y con un conjunto de validación se evalúa la capacidad de predicción del modelo. Los ejemplos del conjunto de validación son clasificados y la aptitud del individuo es el porcentaje de ejemplos clasificados correctamente, conforme más grande es el valor mayor es la aptitud del individuo.
- **Paso 3:** Seleccionar el mejor individuo y transferirlo a la siguiente generación, entonces el mejor individuo siempre sobrevive.
- **Paso 4:** Seleccionar aleatoriamente pares de individuos, los cuales son padres de los individuos de la siguiente generación. El material genético de los padres se combina con el operador de cruce de dos puntos (Fig. 7) a un porcentaje de los padres seleccionados.
- **Paso 5:** Seleccionar aleatoriamente un porcentaje de individuos, los cuales son mutados (Fig.8).
- **Paso 6:** Aplicar Elitismo. El mejor individuo seleccionado en el paso tres reemplaza al individuo de la población resultante del paso cinco.
- **Paso 7:** Repetir el paso dos G generaciones.

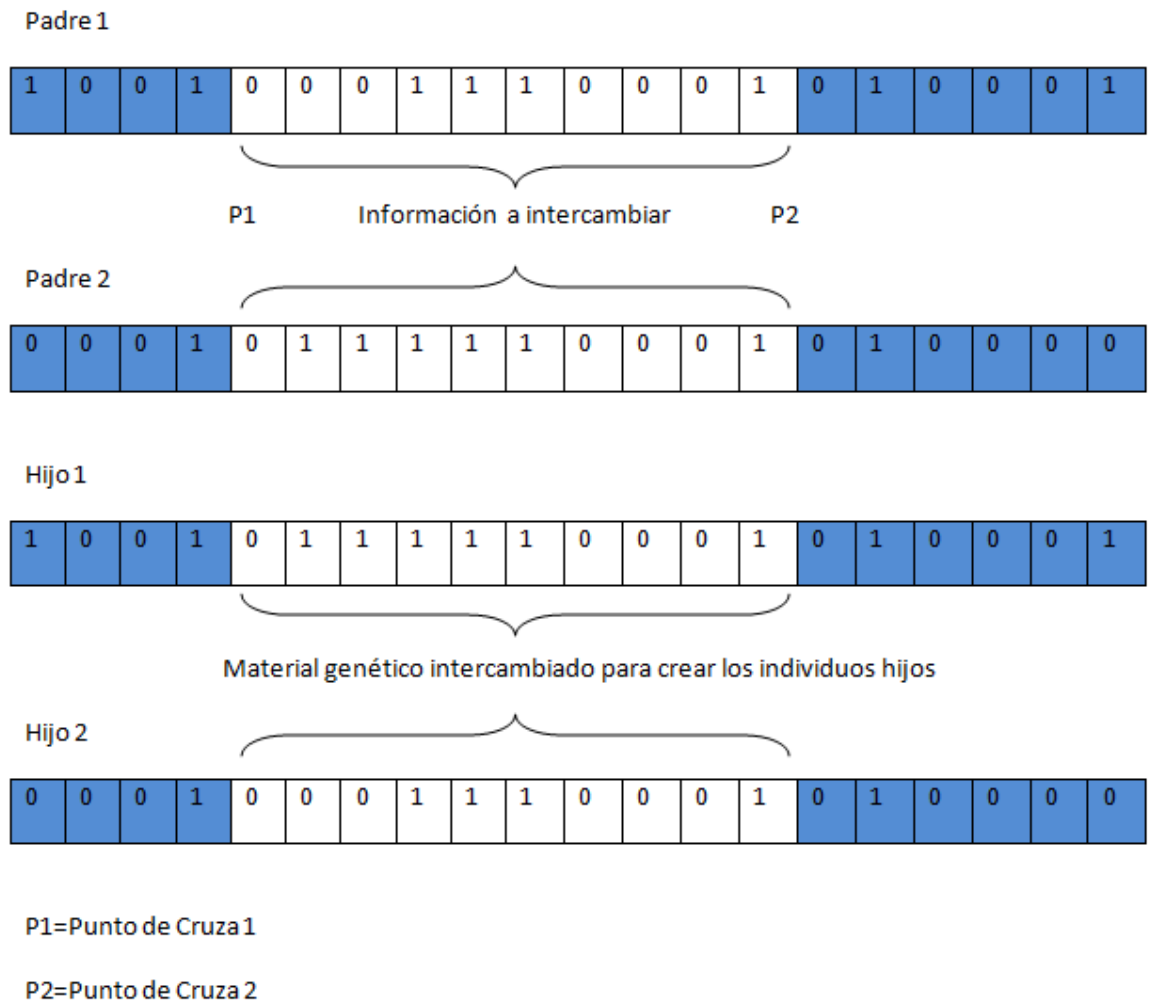


FIGURA 7 El material genético localizados entre los puntos de cruce P1 y P2 de los padres se intercambia para generar dos hijos.



Individuo seleccionado

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

a) Mutación de un Punto: selecciona una posición para cambiarla (PM) de valor, si tenía un 1 cambia a 0 y si tenía un 0 cambia a 1.

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

PM

Nuevo individuo:

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

b) Mutación de varios puntos: seleccionar varias posiciones para cambiarlas (PM) de valor, si tenía un 1 cambia a 0 y si tenía un 0 cambia a 1.

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

PM

PM

PM

Nuevo individuo:

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

c) Mutación global: todos los elementos del individuo cambian su valor.

Nuevo individuo

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

FIGURA 8. En la imagen se muestran tres tipos de mutación: un punto, varios puntos y global.

El pseudocódigo del algoritmo genético se muestra a continuación:

La matriz  $S_m$  contiene las  $T$  características calculadas a partir de las imágenes de entrenamiento, en total son  $M$  imágenes, las cuales son clasificadas previamente.

---

**Algoritmo AG:** Crea el vector  $C$  con las características seleccionadas para clasificar

---

**Requiere:** Matriz  $S_m = [S_{e_k}]$ ;  $m = 1, \dots, M$ ;  $k = 1, \dots, T$

GMAX número de iteraciones,  $P_c$  probabilidad de cruce,  $P_M$  probabilidad de Mutación

**Ensure:** Vector of características seleccionadas  $C$

1: **repeat**

2:     **Crear población con N individuos y T componentes**

3:         **for**  $i = 1$  to  $N$  **do**

4:             Clasificar  $S_m$

5:             Calcular la matriz de confusión y asignar la aptitud del individuo

6:             Seleccionar el mejor individuo

7:             Seleccionar aleatoriamente la población de padres

8:             Cruzar la población de padres conforma la probabilidad  $P_c$  y generar nuevos individuos

9:             Mutar la población resultante del paso anterior conforme la probabilidad  $P_M$  y generar nuevos individuos

10:            Aplicar elitismo

11:         **end for**

12: **until** GMAX iteraciones

En el Capítulo 4 se muestra el número de generaciones, tamaño de la población, la probabilidad de cruce y mutación, utilizados en los experimentos con el AG.

### 3.2.3. Algoritmo RFD-SD

En esta tesis se propuso el Algoritmo RFD-SD (*Reduction of Feature Dimensions based on Standard Deviation* por sus siglas en inglés), con el cual se seleccionaron descriptores de una acción. Las características seleccionadas en todas las secuencias de entrenamiento de una acción tienen una desviación estándar mayor a un umbral  $\varphi$ . El valor de la desviación estándar refleja cuanto movimiento existe en las características durante la ejecución de la acción.

A continuación se consideran cinco secuencias  $\{S_1, S_2, S_3, S_4, S_5\}$  con 75 características  $\{C_1, C_2, \dots, C_{75}\}$ , las secuencias son acciones de la clase  $X$ .

El valor de  $\varphi$  está dentro del rango de la desviación estándar máxima y mínima de las características en las cinco secuencias. Las características con desviación estándar mayor a  $\varphi$  en las secuencias son mostradas a continuación:

S1	S2	S3	S4	S5
C12	C12	C10	C12	C12
C21	C30	C12	C21	C30
C30	C33	C21	C30	C33
C33	C74	C30	C33	C55
	C75	C33	C55	
		C45		
		C74		

Después las características presentes en todas las secuencias {C12, C30, C33} son seleccionadas (color azul), las características restantes no son seleccionadas ya que probablemente representan ruido o deformaciones a causa de movimientos en donde la persona se aproxima al suelo, por ejemplo cuando la persona se dobla para tomar algo del piso.

Con el conjunto de características seleccionadas se clasifican las secuencias de entrenamiento para calcular el porcentaje de secuencias clasificadas correctamente (PC), el umbral  $\varphi$  se decrementa en un valor  $\alpha$  de la siguiente forma  $\varphi = \varphi - \alpha$  y se seleccionan otras características con el objetivo de mejorar el valor de PC. En los experimentos de esta tesis se utilizó  $\alpha=1$ . Con algunos de los siguientes criterios se continúa la ejecución del algoritmo RFD-SD:

- Mientras el valor de PC continúe incrementándose
- Mientras no se alcance un número específico de iteraciones

El pseudocódigo del algoritmo RFD-SD se muestra continuación:

El conjunto de entrenamiento SP contiene las secuencias de entrenamiento de cada clase de acción, en total son  $N_c \times M$ , en donde  $N_c$  es el número de clases y M la cantidad de secuencias utilizadas en cada clase. Cada secuencia contiene T características calculadas a partir de las I imágenes de la secuencia. Las secuencias pueden ser de diferentes longitudes.

La matriz  $S_m$  contiene M secuencias de la clase de acción X

---

**Algoritmo RFD-SD:** Calcula el vector  $C_X$  con las características seleccionadas para describir la clase X

---

**Requiere:** Matriz  $S_m = [Se_{i,k}]$ ;  $m = 1, \dots, M$ ;  $k = 1, \dots, T$ ,  $i = 1, \dots, l$

$S_m \subset SP$

Umbral  $\varphi$ ,  $\alpha$

NV vecinos más cercanos

**Ensure:** Vector de características seleccionadas  $C_X$

```
1: repeat
2:   for m = 1 to M do
3:     for k = 1 to T do
4:        $D_{m,k} \leftarrow$  Desviación Estándar( $S_m = [Se_{:,k}]$ );
5:     end for
6:   end for
7:   for m = 1 to M do
8:     n  $\leftarrow$  1
9:     for k = 1 to T do
10:      if  $D_{m,k} \geq \varphi$  then
11:         $TemC_{m,n} \leftarrow k$ 
12:        n  $\leftarrow$  n + 1
13:      end if
14:    end for
15:  end for
16: n  $\leftarrow$  número de elementos de  $TemC_1$ 
17:   for l = 1 to n do
18:     if  $TemC_{1,l} \in TemC_k$  para  $k = 2, \dots, M$  then
19:        $C_X \leftarrow [C_X; TemC_{1,l}]$ 
20:     end if
21:   end for
22: SPTem  $\leftarrow$  Selecciona en SP las características  $C_X$ 
22:   for l = 1 to TS do
23:     Sl del conjunto de entrenamiento SPTem
24:     con DTW compara Sl con las secuencias restantes del conjunto de entrenamiento
25:     Encuentra los NV vecinos más cercanos
26:     if la mayoría de los vecinos pertenecen a la clase X then
27:       Sl pertenece a la clase 1, reconoce la secuencia como acción X
28:     else
29:       Sl pertenece a la clase 2, reconoce la secuencia como otra acción diferente a X
30:     end if
31:   end for
32: PC  $\leftarrow$  Calcula el porcentaje de secuencias clasificadas correctamente
33:  $\varphi \leftarrow \varphi - \alpha$ 
34: until q iteraciones o PC incrementa respecto a su valor en la iteración anterior
```

### 3.2.3.1 Ejemplo de Selección de características con RFD-SD

En esta sección se explica cómo el algoritmo RFD-SD seleccionó los descriptores de una acción. Mediante las ecuaciones (7), (8), (9) y (10) se calcularon 56 características para las quince

secuencias de las acciones ondular el brazo horizontalmente, caminar y sentarse de la base KARD. El conjunto SP contiene las características calculadas con las 45 secuencias, quince para cada una de las tres clases mencionadas. Con el algoritmo RFD-SD se seleccionó un conjunto de características **C** con el objetivo de clasificar las secuencias de la acción ondular el brazo horizontalmente en la clase uno y las treinta secuencias restantes en la clase dos. El conjunto  $S_m$  contiene las quince secuencias de la acción ondular el brazo horizontalmente.

La ejecución del algoritmo comenzó calculando la desviación estándar de las características en las secuencias ( $S_m$ ) de la clase ondular el brazo horizontalmente y seleccionó las características cuya desviación estándar fue mayor que  $\varphi=20$ , las cuales son mostradas en la Tabla 7. Posteriormente en este subconjunto de características se seleccionaron las características presentes en todas las secuencias. Como se observa en la Tabla 7 sólo la característica quince cumple la condición mencionada.

**Tabla 7 Características seleccionadas en las secuencias de entrenamiento de la clase ondular el brazo horizontalmente**

S1		S2		S3		S4		S5		S6		S7		S8	
Car	DE	Car	DE	Car	DE	Car	DE	Car	DE	Car	DE	Car	DE	Car	DE
15	59	15	59	15	60	15	35	15	50	15	40	15	49	15	54
49	40	49	44	49	41	14	33	49	35	14	31	18	37	46	34
9	28	9	23	17	23			14	27	49	24	17	37	18	31
6	28	6	23	9	23			44	21			52	32	17	30
7	25	14	21	6	23							49	31	49	30
S9		S10		S11		S12		S13		S14		S15			
Car	DE	Car	DE	Car	DE	Car	DE	Car	DE	Car	DE	Car	DE		
15	57	15	56	15	57	15	57	15	39	15	58	15	65		
17	31	49	41	49	41	49	43	17	39	18	55	49	45		
49	31	13	28	9	33	46	25	14	38	9	39	46	38		
46	30	46	26	6	32	14	22	9	38	6	39	18	33		
9	25	3	22	7	26	13	21	6	38	17	36	17	23		

S=secuencia

Car=Indica el número de característica seleccionada

DE= indica la desviación estándar de la característica seleccionada dentro de la secuencia

Las características que no están presentes en las quince secuencias pueden ser ruido o algún movimiento realizado por la persona, pero no describe la acción a reconocer. Por ejemplo en la secuencia S1 la característica siete corresponde al ángulo formado entre el hombro izquierdo y el cuello. Esta característica no describe la acción ondular el brazo horizontalmente, porque la acción

se realizó con el brazo derecho. El conjunto de características seleccionadas es  $C = \{15\}$ , esta característica corresponde al ángulo formado entre las articulaciones del hombro derecho y el codo derecho.

En la siguiente parte del algoritmo se evaluó la capacidad de discriminación del conjunto de características  $C = \{15\}$  para reconocer las secuencias de la acción ondular el brazo horizontalmente. En el conjunto SP se seleccionó la característica quince y las secuencias con la característica seleccionada son almacenadas en SPTem. A continuación se clasificaron las secuencias de SPTem. Para esto, con DTW se midió la similitud de cada secuencia con las veintinueve secuencias restantes y con los  $NV=3$  vecinos más cercanos se clasificó cada secuencia. Las secuencias se clasificaron en la clase uno cuando la mayoría de los tres vecinos pertenecen a la acción ondular el brazo horizontalmente y a la clase dos cuando los vecinos más cercanos son secuencias de las acciones caminar o sentarse.

Después de clasificar las secuencias en SPTem, se obtuvo un porcentaje de secuencias clasificadas correctamente (PC) igual a 90%. Con la característica quince se obtuvo un buen reconocimiento porque en la clase caminar y sentarse está característica no cambia cuando se realizan las dos últimas acciones.

En la siguiente iteración del algoritmo se actualizó el valor de  $\varphi$  con la operación  $\varphi = \varphi - \alpha$ , considerando  $\alpha = 1$  entonces  $\varphi = 19$ , los pasos descritos en los párrafos anteriores se repitieron. Con el algoritmo se realizaron algunas iteraciones hasta que PC alcanzó un valor de 97.78%, se seleccionaron las características  $C = \{6, 15, 17\}$ , las cuales corresponden a ángulos formados entre las articulaciones del cuello, del hombro derecho y de la mano derecha; entre las articulaciones del hombro derecho y el codo derecho y por último entre las articulaciones del codo derecho y mano derecha.

Para cada una de las acciones restantes caminar y sentarse se debe ejecutar el algoritmo RFD-SD para obtener un conjunto de descriptores de cada clase de acción.

### **3.2.4. PCA**

El análisis de componentes principales (PCA por sus siglas en inglés) (Richards & Richards, 1999) (Jackson, 2005) (Jolliffe, 1986) es un método utilizado para reducir la dimensionalidad de un conjunto de características que describen un conjunto de datos, en esta tesis las características describen acciones.

Con PCA se determina cuáles características tienen mayor variación en el conjunto de datos y se genera un nuevo conjunto de características conocido como componentes principales. La primera componente principal describe la máxima variación de los datos y por lo general la suma de las

varianzas de unas cuantas primeras componentes concentran más del 80% de la varianza total del conjunto de datos.

En esta tesis se seleccionaron las primeras componentes cuya suma de varianzas fue mayor del 90%. El análisis de componentes principales se realizó con las funciones de PCA de MATLAB.

### **3.3. Reconocimiento de caída y acciones**

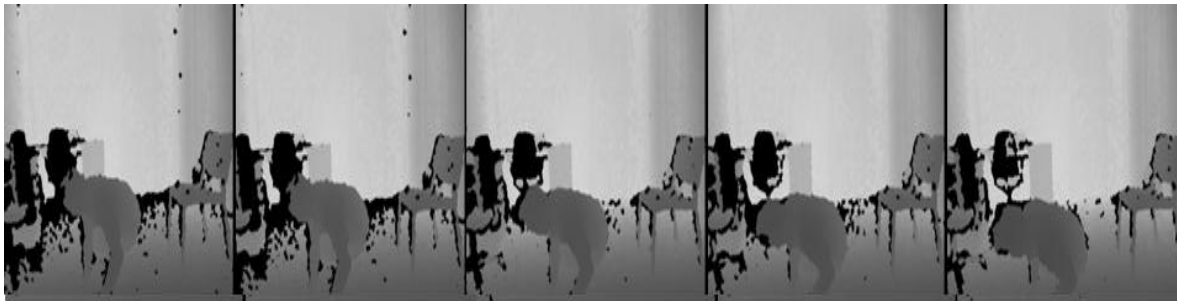
#### **3.3.1. Reconocimiento de caída**

Con los clasificadores K-medias, MGG y SVM se reconocieron las caídas. Cuando se seleccionaron las características con el algoritmo genético se utilizó el método de envoltura y con PCA el método de filtrado. En las siguientes secciones se explican los clasificadores utilizados.

##### **3.3.1.1. K-medias**

Con el algoritmo K-medias (Andrew R. Webb, 2011) se crearon dos clases:

- Clase 1: contiene las posturas de una persona aproximándose al suelo (Fig. 9).
- Clase 2: contiene las posturas de una persona de pie o sentada (Fig. 10)



**FIGURA 9.** En las imágenes de profundidad una persona cae al piso.



**FIGURA 10.** En las imágenes de profundidad la persona camina.

El conjunto de imágenes de entrenamiento se clasificó manualmente en una de las clases mencionadas (clase 1 o clase 2). Cuando se utilizó el AG cada individuo representó un conjunto de características seleccionadas, el algoritmo K-medias creó dos clases utilizando las características seleccionadas, se obtuvo la matriz de confusión con las clases creadas y las clasificadas manualmente y se calculó el porcentaje de imágenes correctamente clasificadas. Con este porcentaje se asignó la aptitud del individuo y se utilizó para determinar si un conjunto de características seleccionadas es mejor que otros.

Una vez seleccionado el conjunto de características con la aptitud más alta, se calculó la norma del vector de características seleccionadas, cuando su valor es menor que un umbral  $U_{pc}$  (establecido experimentalmente) entonces la imagen contiene una postura de caída la cual se presenta cuando una persona cae al piso y cuando una persona se acuesta en el suelo. Después se calculó el cambio de la norma del vector de características en treinta imágenes, se seleccionaron treinta porque conforme con (Yu, 2008) el evento de caída sucede en uno o dos segundos y con el Kinect se adquieren treinta imágenes por segundo. Experimentalmente se estableció el valor de un umbral  $U_v$  y cuando el cambio de la norma del vector de características cada treinta imágenes fue menor que  $U_v$  y la norma del vector de características fue menor de  $U_{pc}$  se reconoció un evento de caída.

### **3.3.1.2. Modelo Generativo Gaussiano**

Un modelo generativo (Prince, 2012) utiliza observaciones del mundo para construir un modelo probabilístico a partir del cual infiere la probabilidad con la que una nueva observación pertenece a una clase.

En esta tesis con un modelo generativo gaussiano (MGG) se reconocieron las siguientes situaciones:

- **Postura de caída:** el cuerpo de la persona se aproxima al piso, la postura se presenta cuando una persona se acuesta o se cae (Fig. 9)
- **Evento de caída:** la persona se aproxima al piso de manera rápida e involuntaria.



El modelo utilizó la distribución de probabilidad mostrada en la siguiente ecuación:

$$P_r(\mathbf{x}|w = k) = Norm_x[\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k] \quad (11)$$

Donde  $\mathbf{x}$  son los datos de entrenamiento y sus parámetros son contingentes a los estados del mundo  $w$ . Los estados del mundo son las clases.

En la fase de aprendizaje se optimizaron los parámetros  $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$ , con pares del conjunto de entrenamiento  $(\mathbf{x}_i, w_i)$ . Una nueva observación  $\mathbf{x}$  se clasificó mediante la siguiente ecuación:

$$P_r(w = k|\mathbf{x}) = P_r(\mathbf{x}|w = k) P_r(w = k) \quad (12)$$

Se calculó la probabilidad posterior de la observación  $\mathbf{x}$  sobre los posibles estados del mundo  $w$ .

Para trabajar con el MGG se propuso un conjunto de 37 características: veinte (Tabla 5) de ellas describen la forma y orientación de la persona y diecisiete (C21 hasta C37) se calcularon con las primeras diecisiete características mostradas en la Tabla 5. Las características C21 hasta C37 representan los cambios de forma y orientación en el tiempo (cada treinta imágenes), con estas características se evitó definir umbrales para diferenciar cuando una persona cae y cuando se acuesta en el piso.

La observación  $\mathbf{x}$  es un vector  $\mathbf{x}=[\mathbf{f}_s, \mathbf{f}_c]$ :

- $\mathbf{f}_s$  es el vector de características seleccionadas del conjunto de características C1 hasta la C20
- $\mathbf{f}_c$  es el vector de características seleccionadas del conjunto de características C21 hasta la C37

El clasificador se evaluó en dos modalidades:

- **Tipo 1:** con una observación  $\mathbf{x}$ , el algoritmo utilizó  $\mathbf{f}_s$  para inferir un estado en el mundo  $w$ ,  $w \in \{1 \text{ cuando detectó una postura de caída, } 2 \text{ cuando no detectó una postura de caída}\}$ . La observación se clasificó como postura de caída si cumplió lo siguiente:

$$P_r(w = 1|\mathbf{f}_s) > P_r(w = 2|\mathbf{f}_s)$$

Si  $\mathbf{x}$  se clasificó como postura de caída, entonces la persona se aproximó al piso. Después se utilizó  $\mathbf{f}_c$  para inferir el estado en el mundo  $w \in \{1 \text{ cuando detectó un evento de caída, } 2 \text{ cuando no se detectó un evento de caída}\}$ . De esta forma con el modelo se detectó cuando una persona se aproxima al piso rápidamente (cayó al piso) o cuando se aproxima lentamente (la persona se acostó en el piso). La observación se clasificó como evento de caída si cumplió lo siguiente:

$$P_r(w = 1|\mathbf{f}_c) > P_r(w = 2|\mathbf{f}_c)$$

- **Tipo 2:** con una observación  $\mathbf{x}$  el algoritmo reconoció un estado en el mundo  $w$ ,  $w \in \{1 \text{ cuando detectó una postura de caída y un evento de caída, } 2 \text{ cuando detectó una postura de caída y no detectó un evento de caída, } 3 \text{ con otra postura diferente a la de caída}\}$ . La observación se clasificó como una caída cuando cumplió con lo siguiente:

$$P_r(w = 1|\mathbf{x}) > P_r(w = 2|\mathbf{x}) \text{ y } P_r(w = 1|\mathbf{x}) > P_r(w = 3|\mathbf{x})$$

En la observación una persona se acostó en el piso cuando:

$$P_r(w = 2|\mathbf{x}) > P_r(w = 1|\mathbf{x}) \text{ y } P_r(w = 2|\mathbf{x}) > P_r(w = 3|\mathbf{x})$$

En la observación una persona caminó o se sentó cuando:

$$P_r(w = 3|\mathbf{x}) > P_r(w = 1|\mathbf{x}) \text{ y } P_r(w = 3|\mathbf{x}) > P_r(w = 2|\mathbf{x})$$

Con los dos clasificadores MGG tipo 1 y 2 se identificó sin utilizar umbrales cuando una persona se aproximó al suelo de manera rápida (la persona cayó), cuando se aproximó de manera lenta (se acostó en el piso) y cuando realizó alguna otra actividad como caminar o sentarse.

### 3.3.1.3. Máquinas de Soporte Vectorial

En esta tesis mediante una máquina de soporte vectorial se clasificaron las posturas en dos clases:

- **Postura de caída:** se presenta cuando una persona se aproxima al piso (Fig. 9)
- **Postura de no caída:** se presenta cuando la persona camina o se sienta (Fig.10)

Después se confirmó que una persona cayó al piso cuando la velocidad y aceleración de las articulaciones de las muñecas de la mano y del centroide de la nube de puntos son comparadas con umbrales definidos experimentalmente.

La máquina de soporte vectorial se ejecutó con las funciones de LS-SVMlab Toolbox<sup>5</sup> implementado en MATLAB.

### 3.3.2. Reconocimiento de acciones

Para reconocer las acciones se creó un conjunto de secuencias patrones de cada acción, el cual contiene  $m$  secuencias para cada acción. Una secuencia es un conjunto de imágenes con un orden temporal, para cada imagen se calculó un conjunto de características mediante las ecuaciones (7), (8), (9) y (10), de esta forma cada secuencia se representó como una serie de tiempo con múltiples características. La dimensión de una serie de tiempo es  $kxl$ , en donde  $k$  es la cantidad de imágenes y  $l$  el número de características.

Para clasificar una nueva observación de una acción se calculó su similitud con las secuencias patrones de cada clase de acción con DTW. En esta tesis se utilizó DTW porque de acuerdo con la revisión presentada en el estado del arte mide la similitud entre series de tiempo con diferentes longitudes. Las mismas acciones ejecutadas en diferentes velocidades son reconocidas como similares y al compararlo con HMM con DTW reportan mejores resultados.

En la Fig. 11 se muestra la misma señal en diferentes velocidades y amplitudes, cuando se utiliza DTW las dos señales son alineadas y el valor de similitud obtenido es casi cero, entonces las señales son casi idénticas. Cuando se mide la similitud con DTW entre más cercano a cero es el valor de similitud la series de tiempo son más parecidas.

En la Fig. 12 se muestra el valor del ángulo entre la articulación de la cabeza y del cuello, para la acción de ondular el brazo en el eje horizontal, la acción es realizada por dos personas, la similitud obtenida con DTW es de 80.91. En la Fig. 13 se muestra el mismo ángulo, en este caso una persona realiza la acción de ondular el brazo en el eje horizontal y la segunda persona realiza la acción de sentarse, la similitud obtenida es de 174.21.

---

<sup>5</sup> <https://www.esat.kuleuven.be/sista/lssvmlab/>

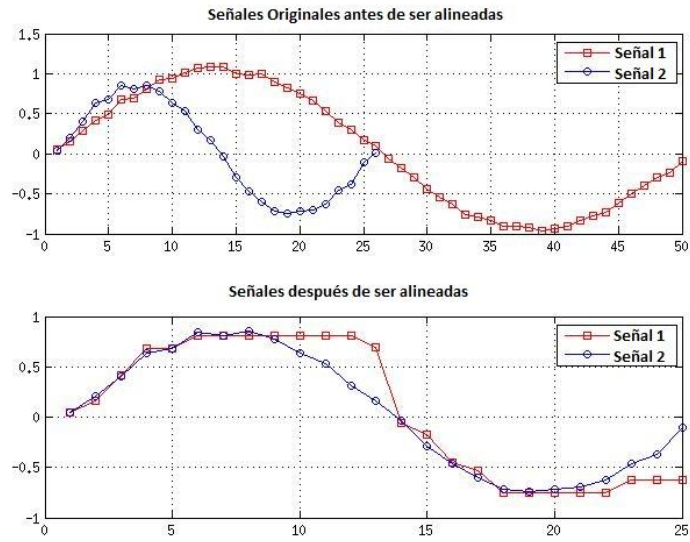


FIGURA 11. La gráfica superior presenta la misma señal en diferentes velocidades. En la gráfica inferior se muestran las señales alineadas.

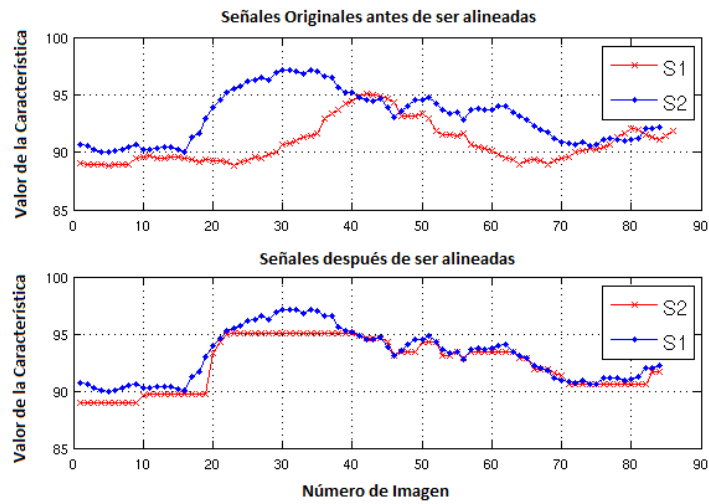
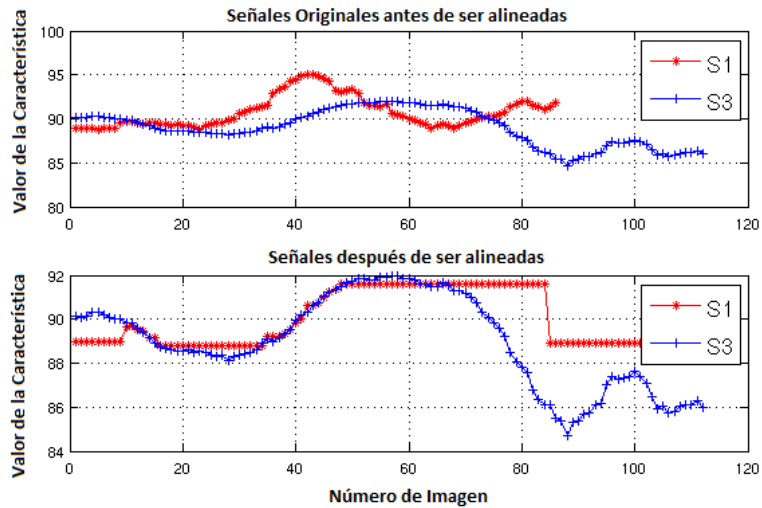


FIGURA 12. En la gráfica superior se muestra el movimiento del ángulo entre la articulación de la cabeza y del cuello para la acción de ondular el brazo en el eje horizontal, la acción es realizada por dos personas S1 y S2. En la gráfica inferior se muestran las señales alineadas, la similitud obtenida con DTW es de 80.91.



**FIGURA 13.** En la gráfica superior se muestra el movimiento del ángulo entre la articulación de la cabeza y del cuello, una persona realiza la acción de ondular el brazo en el eje horizontal y la segunda persona realiza la acción de sentarse. En la gráfica inferior se muestran las señales alineadas, la similitud obtenida con DTW es de 174.21

Con el algoritmo RFD-SD se selecciona un conjunto de características para cada clase de acción, cada conjunto puede contener diferente número de características. Como se mencionó antes, para cada clase de acción se almacenan  $m$  secuencias patrones, cuando se trabaja con  $M$  clases se almacenan en total  $M \times m$  secuencias patrones. Se crean  $M$  repositorios nombramos  $R_1, R_2,$  hasta  $R_M$ , cada repositorio contiene las  $M \times m$  secuencias patrones, pero sólo con las características seleccionadas para la clase  $M$ , por ejemplo en  $R_1$  se almacena todas las secuencias patrones con las características seleccionadas para reconocer la clase uno. Para clasificar (Fig. 14) una nueva observación de acción, se calculó su similitud con cada una de las secuencias patrones de cada repositorio mediante el conjunto de características seleccionadas en cada clase, se obtienen los  $N$  vecinos con menor valor de DTW. A continuación se calcula una probabilidad de pertenencia  $P_x$  a la clase de acción  $X$  con la siguiente ecuación:

$$P_x = \left( \frac{N_x}{N} \right)$$

$N$  es el número de vecinos más cercanos  
 $N_x$  es la cantidad de vecinos que pertenecen a la clase  $X$ .

Finalmente la secuencia observada es clasificada como una acción de la clase  $Y$  cuando  $P_y$  tiene el valor más alto (Fig. 14).

### 3.4. Resumen

En este capítulo se describen los componentes del sistema de reconocimiento de acciones y caídas utilizado en esta tesis:

- Conjunto de características propuestas para reconocer acciones y caídas.
- Algoritmos de selección de características.
- Clasificadores.

Estos elementos se utilizaron en los experimentos con el propósito de mejorar la exactitud del reconocimiento de acciones y caídas. En el siguiente capítulo se describen los experimentos y se discuten los resultados.

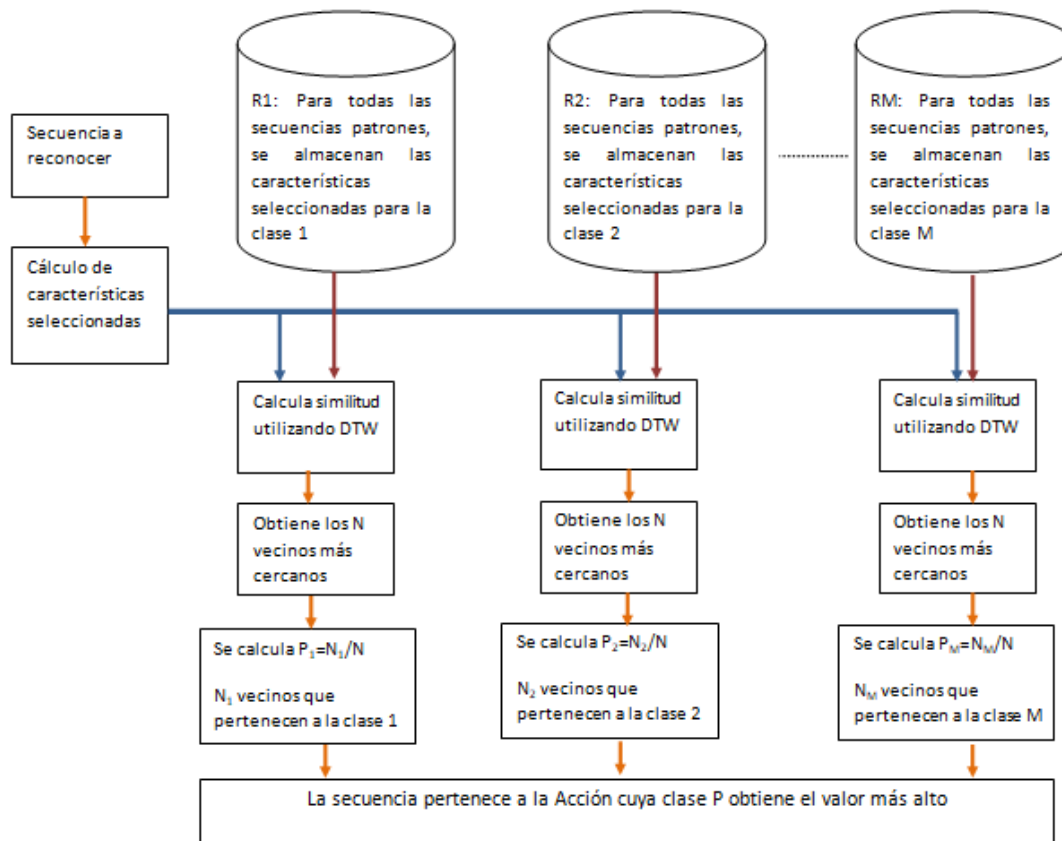


FIGURA 14. Para cada clase se almacenan m secuencias patrones. Se crea un repositorio R para cada clase. La secuencia a reconocer se compara con cada una de las secuencias en los repositorios y se calcula la probabilidad de pertenencia a cada clase.

## Capítulo 4: Experimentos y Resultados

En este capítulo se describen los experimentos realizados con el propósito de comparar el impacto en la exactitud del reconocimiento de acciones y caídas cuando se utilizaron los conjuntos de características seleccionadas por un AG, seleccionadas por el algoritmo RFD-SD, seleccionadas con PCA y todas las características.

En la Tabla 8 se muestran los principales elementos de las bases utilizadas en los experimentos.

**Tabla 8 Bases utilizadas en los experimentos**

Bases	Escenario	Clases	Tipo de datos	Sujetos	Muestras
UR Fall Detection Dataset	Caídas	2	Mapas de profundidad, RGB y lecturas de acelerómetro	5	30
	Actividades de la vida diaria	4	Mapas de profundidad, RGB y lecturas de acelerómetro	5	40
TST Fall detection dataset v2	Caídas	4	Mapas de profundidad, posiciones de las articulaciones del esqueleto y lecturas de acelerómetro	11	132
	Actividades de la vida diaria	4	Mapas de profundidad, posiciones de las articulaciones del esqueleto y lecturas de acelerómetro	11	132
MSRAction3D	Juegos	20	Mapas de profundidad y esqueleto	10	402
KARD	Actividades de la vida diaria	18	Mapas de profundidad y posiciones de articulaciones del esqueleto	10	540
UTD-MHAD	Actividades de la vida diaria	27	Mapas de profundidad, RGB, posiciones de articulaciones del esqueleto y lecturas de acelerómetro	8	864

### 4.1. Reconocimiento de Caídas

Con los siguientes cuatro experimentos se detectó cuando una persona cayó en el piso:

- **Experimento 1:** en cada mapa de profundidad de la base *UR Fall Detection Dataset* se detectó la nube de puntos de la persona, se calcularon las veinte características (Tabla 5),

se seleccionaron características con un AG y PCA y finalmente se comparó cuál de los dos conjuntos de características seleccionadas contribuyó en mejorar la clasificación. Con las características seleccionadas se detectaron posturas de caída (Fig. 9) en secuencias donde la persona cayó y en secuencias de actividades de la vida diaria. Se definieron dos umbrales: con el primero se detectó la postura de caída y con el segundo se distinguió cuando una persona cayó al piso o se acostó. La clasificación se hizo con KNN.

- **Experimento 2:** se realizó el seguimiento de una persona con una unidad *pant-tilt* y una cámara de profundidad. Primero se detectó la nube de puntos de la persona, segundo se calcularon las características seleccionadas en el experimento uno con el AG, con estas características se obtuvo un mejor reconocimiento de las caídas y de acciones de la vida diaria. En este experimento se evaluó la exactitud del reconocimiento de caídas en un ambiente real cuando una persona caminó de un lado a otro, caminó de un lado a otro y cayó al piso y caminó de un lado a otro y se sentó.
- **Experimento 3:** en este experimento se utilizó la base *TST Fall detection dataset v2*, las características se seleccionaron con un AG y PCA. La clasificación se hizo con modelos generativos gaussianos con los cuales se reconoció cuando una persona se aproximó al suelo rápidamente y cuando lo hace más lentamente. En el primer caso se clasificó como caída y en el segundo la persona se acostó en el piso.
- **Experimento 4:** en este experimento se utilizó la base pública *TST Fall detection dataset v2*, las características son seleccionadas con un AG y PCA. Con el objetivo de mejorar los resultados obtenidos en el experimento tres, se utilizaron características de velocidad y aceleración de las articulaciones de las manos y centroide de la persona y se clasificó con SVM.

El desempeño del reconocimiento de las caídas se midió con las siguientes métricas:

- **Verdadero positivo (VP):** una postura de caída se clasificó como caída.
- **Falso positivo (FP):** una postura de no caída se clasificó como caída.
- **Verdadero negativo (VN):** una postura de no caída se clasificó correctamente como postura de no caída.
- **Falso negativo (FN):** una postura de caída se clasificó como una postura de no caída
- **Sensibilidad (Se):** representa la capacidad de reconocer las caídas correctamente:

$$Se = \frac{VP}{(VP+FN)} \quad (13)$$

- **Especificidad (Sp):** indica la capacidad de reconocer correctamente las no caídas, cuando la persona caminó o se sentó:



$$Sp = \frac{VN}{(VN+FP)} \quad (14)$$

- **Exactitud (Acc):** representa la proporción de posturas clasificadas correctamente:

$$Acc = \frac{VP+VN}{(VP+VN+FP+FN)} \quad (15)$$

- **Error (Er):** representa la proporción de posturas clasificadas incorrectamente:

$$Er = \frac{FP+FN}{(VP+VN+FP+FN)} \quad (16)$$

#### 4.1.1. Experimento 1

En este experimento se distinguió entre una postura de caída (Fig. 9) y una postura de no caída (Fig. 10), se utilizaron las secuencias de la base *UR Fall Detection Dataset* grabadas con un Kinect colocado paralelo a la persona. Esta base contiene treinta secuencias de caídas y secuencias de acciones la vida diaria (ADL por sus siglas en inglés). Las secuencias están formadas por imágenes de profundidad en formato PNG16, las cuales en este trabajo de tesis se convirtieron a formato de nube de puntos (PCD por sus siglas en inglés) mediante librerías de PCL. La segmentación de la persona se hizo manualmente porque se contó con una imagen de la escena sin personas para hacer la diferencia de imágenes. El experimento se realizó en cuatro fases, las cuales se explican a continuación.

##### 4.1.1.1. Fase 1: Selección de Características

La selección de las características se hizo con la distribución de posturas mostradas en la Tabla 9.

**Tabla 9 Posturas de entrenamiento**

Postura	Porcentaje respecto al total de imágenes de entrenamiento	Número de imágenes
La persona está de pie	35%	95
La persona está	21%	55

sentada		
La persona se aproxima al piso	44%	118
TOTAL	100%	268

Las imágenes se clasificaron manualmente en una de las dos siguientes clases:

- La persona se aproximó al piso.
- La persona caminó o se sentó.

Las características se seleccionaron de un conjunto de veinte características (Tabla 5) con el método de encapsulamiento con un AG y el clasificador K-Medias (sección 3.2.2). Las imágenes de entrenamiento se clasificaron con K-Medias en dos grupos. La clasificación resultante se comparó con la clasificación manual, se calculó la matriz de confusión y el porcentaje de posturas clasificadas correctamente para obtener la aptitud de cada conjunto de características seleccionadas. La selección de características se realizó 35 veces, se utilizaron diez individuos, se aplicó un porcentaje de cruce igual a 80%, una mutación global y cada ejecución evolucionó cien generaciones. En la Tabla 10 se muestran los resultados estadísticos de las corridas.

**Tabla 10. Resultados estadísticos del algoritmo genético**

Métrica	Valor	Individuo
<b>Promedio</b>	98.91%	
<b>Mediana</b>	98.88%	
<b>Desviación estándar</b>	0.29	
<b>Mejor individuo</b>	99.62%	0 1 1 0 1 0 1 1 0 0 0 0 0 0 0 0 0 1 0
<b>Peor individuo</b>	98.13%	0 0 0 0 1 0 1 1 1 0 0 1 0 0 1 1 1 0 1 1

El mejor individuo clasificó correctamente el 99.62% de las posturas de entrenamiento y seleccionó seis características: C2, C3, C5, C7, C8 y C19 la última es uno de los invariantes 3D. Después se calculó un umbral para distinguir entre una postura de caída y una de no caída. Para esto se obtuvo la norma del vector de características seleccionadas con cada postura de entrenamiento (Fig. 15). La norma del vector de características seleccionadas en cada postura de caída resultó ser menor de 2.5. Después se calculó el promedio de la norma del vector de

características de todas las posturas de caídas del conjunto de entrenamiento y se obtuvo un valor de 1.7, el cual se utilizó como umbral  $U_{pc}$  para reconocer cuando una persona se aproximó al piso.

Adicionalmente con PCA se seleccionaron las características: C7, C8, C10, C11, C13, C15 y C18, las cuales describieron el 98% de la varianza de los datos. El clasificador K-medias utilizó las características seleccionadas con PCA y se obtuvo un 78.87% de posturas correctamente clasificadas. En la Fig. 16 se grafica la norma del vector de características seleccionadas en cada postura de entrenamiento, como se observa es imposible establecer un umbral para diferenciar entre las posturas de caída y las posturas de no caída.

#### **4.1.1.2. Fase 2: Evaluar el desempeño del clasificador en secuencias de caídas**

En esta fase se usaron ocho secuencias de caídas:

- En cuatro de ellas la persona caminó y posteriormente cayó.
- En las cuatro restantes la persona estuvo sentada y posteriormente cayó.

Los resultados obtenidos con las ecuaciones (13), (14), (15) y (16) se muestran en la Tabla 11:

**Tabla 11. Métricas de desempeño**

	<b>AG</b>	<b>PCA</b>
<b>Sensibilidad</b>	97.13%	95.69%
<b>Especificidad</b>	95.45%	84.98%
<b>Exactitud</b>	96.10%	89.14%
<b>Error</b>	3.90%	10.86%

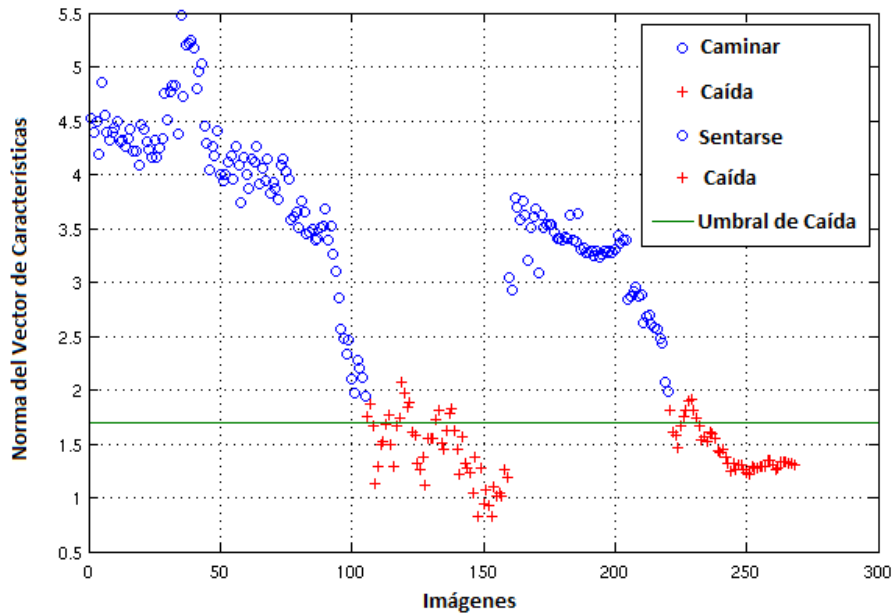


FIGURA 15. Con una cruz '+' se representa una postura de caída, su norma del vector de características seleccionadas con el AG tiene un valor menor de 2.5. Con un umbral de 1.7 se distingue entre una postura de caída y una de no caída.

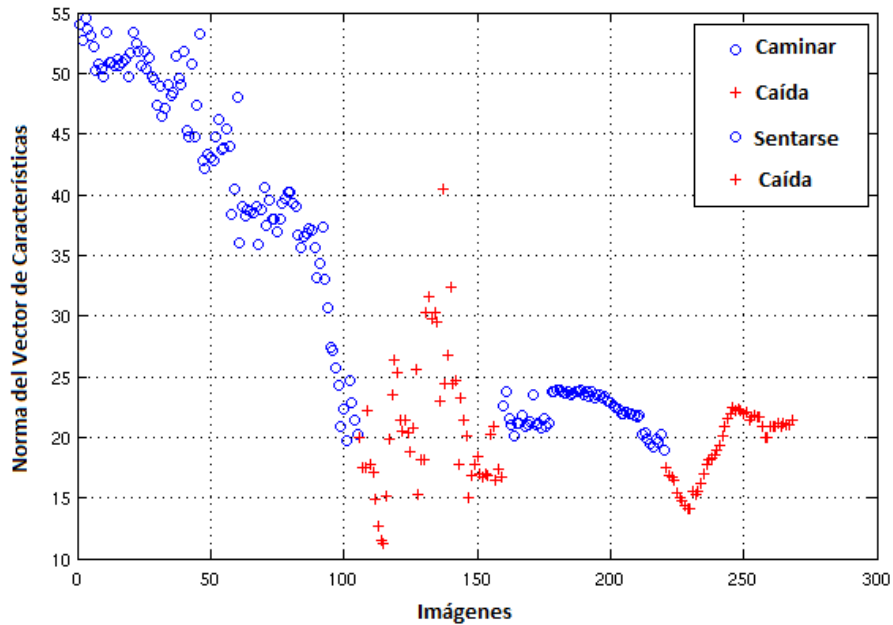


FIGURA 16. Con una cruz '+' se representa una postura de caída. En la gráfica se observa que es imposible establecer un umbral para diferenciar entre una postura de caída y una de no caída.

Los resultados obtenidos con las métricas sugieren que el subconjunto de características seleccionado por el AG contribuyó en mejorar todavía más la exactitud del reconocimiento en comparación con las características seleccionadas con PCA.

En las Fig. 17 y 18 se muestran las normas de los vectores de características seleccionadas por el AG y PCA, respectivamente para una secuencia en donde una persona estuvo de pie y cayó. En la Fig. 17 se aprecia que la magnitud de la norma del vector de características seleccionadas es menor de 1.7 en las posturas de caída. Por otra parte en la Fig. 18 es imposible establecer un umbral para diferenciar las posturas de caída y de no caída, cuando empezó la caída (alrededor de la imagen 180) la norma del vector tiene un valor aproximado de quince, pero en las siguientes imágenes el valor es mayor que veinte.

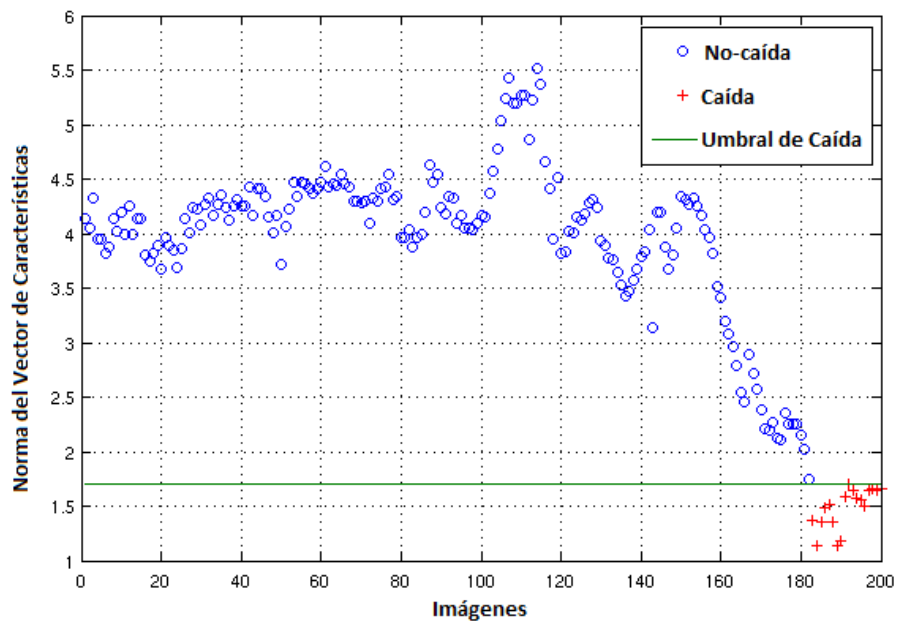
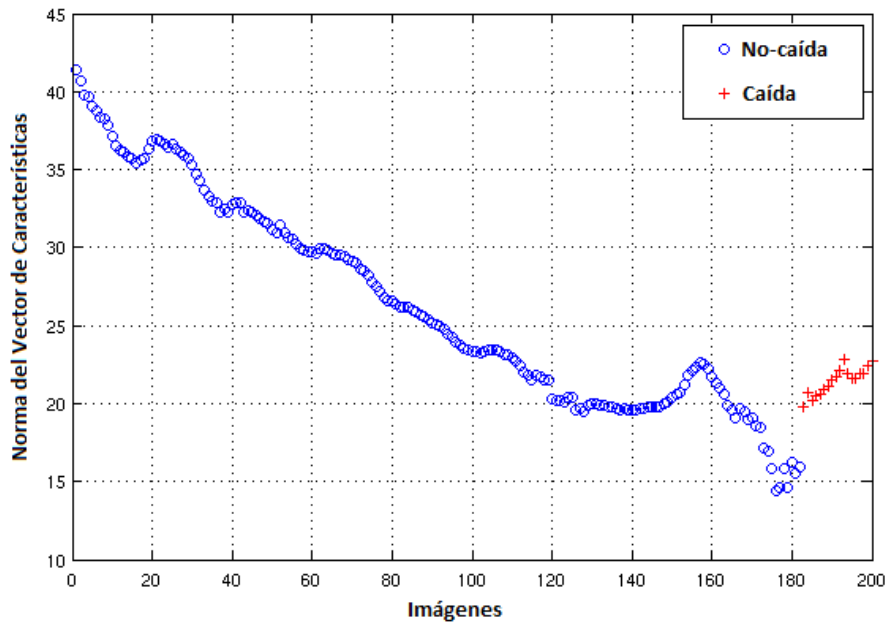


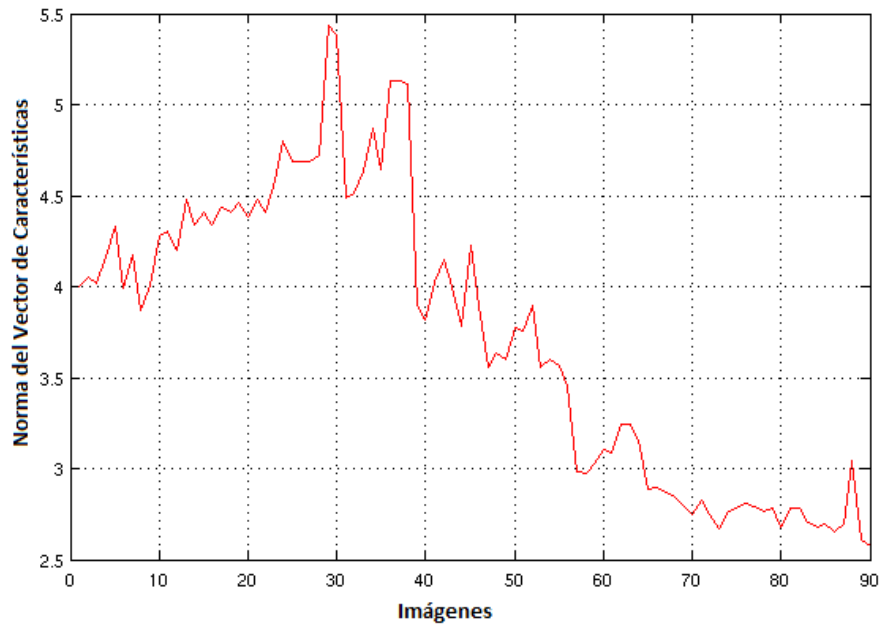
Figura 17. La persona camina y cae al piso. El valor de la norma del vector de características seleccionadas por el AG en las imágenes con postura de caída es menor que 1.7, por lo tanto la caída se detecta correctamente.



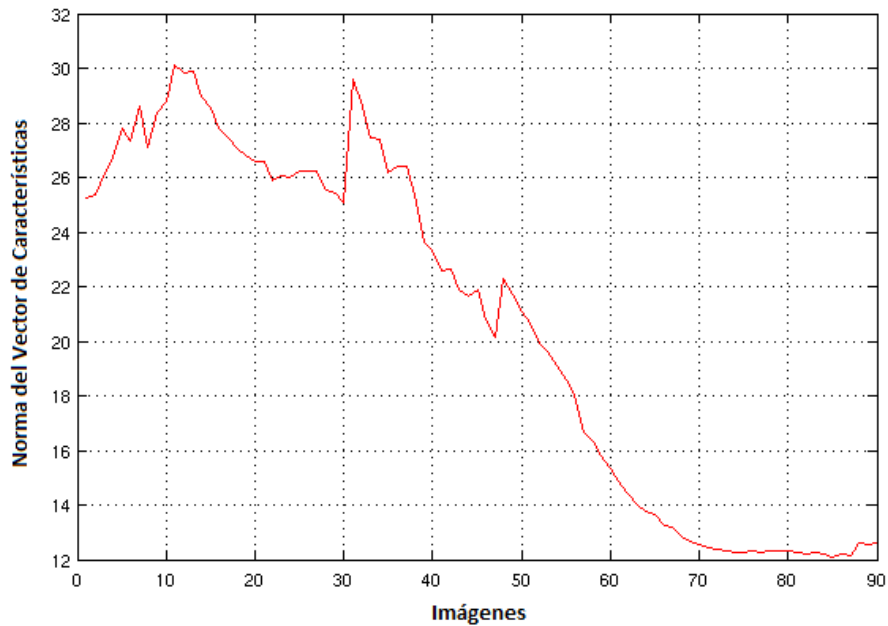
**FIGURA 18.** La persona camina y cae al suelo. En la gráfica se observa que es imposible definir un umbral para diferenciar entre las posturas de caída y las posturas de no caída.

#### 4.1.1.3. Fase 3: Evaluar el desempeño del clasificador con secuencias de acciones de la vida diaria

En esta fase se evaluó el desempeño del clasificador con secuencias ADL, por consiguiente se utilizaron cinco secuencias en las cuales una persona estuvo en cuclillas, levantó un objeto del piso y se acostó en el piso. En las Fig. 19 y 20 se grafica la norma del vector de características seleccionadas por el AG y PCA, respectivamente, para una secuencia en la cual una persona caminó y después se puso en cuclillas.



**FIGURA 19.** La persona camina y se pone en cuclillas. La norma del vector de características seleccionadas con el AG en todas las imágenes es mayor que 1.7, por tanto no se detecta una postura de caída, lo cual es correcto.



**FIGURA 20.** La persona camina y se pone en cuclillas. La norma del vector de características seleccionadas con PCA cuando la persona se pone en cuclillas (a partir de la imagen 70) es menor de quince.

La norma del vector de características seleccionadas con el AG en todas las imágenes (Fig. 19) de la secuencia es mayor que 1.7, por tanto no se detectó una caída, esto es correcto. Por otro lado en la Fig. 20 se grafica la norma del vector de características seleccionadas por PCA y se observa como la magnitud del vector desciende hasta un valor aproximado a doce. En ese momento la persona estuvo en cuclillas, si usamos un umbral con valor de quince como se mencionó en la Fig.18 se podría detectar una falsa caída.

#### 4.1.1.4. Fase 4: Determinar el umbral para confirmar una caída

Las acciones de caer y acostarse (Fig. 21) en el piso comparten las mismas posturas, pero en el primer caso el movimiento es rápido e involuntario y en el segundo la persona se mueve despacio y con cuidado. Se analizaron secuencias con personas cayendo al piso y acostándose para establecer el valor del umbral  $U_V$  con el cual se confirmó una caída. En la Fig. 22 se grafica la norma del vector de características seleccionadas con el AG en varias secuencias: en unas la persona cayó al piso y en otras se acostó en el piso.

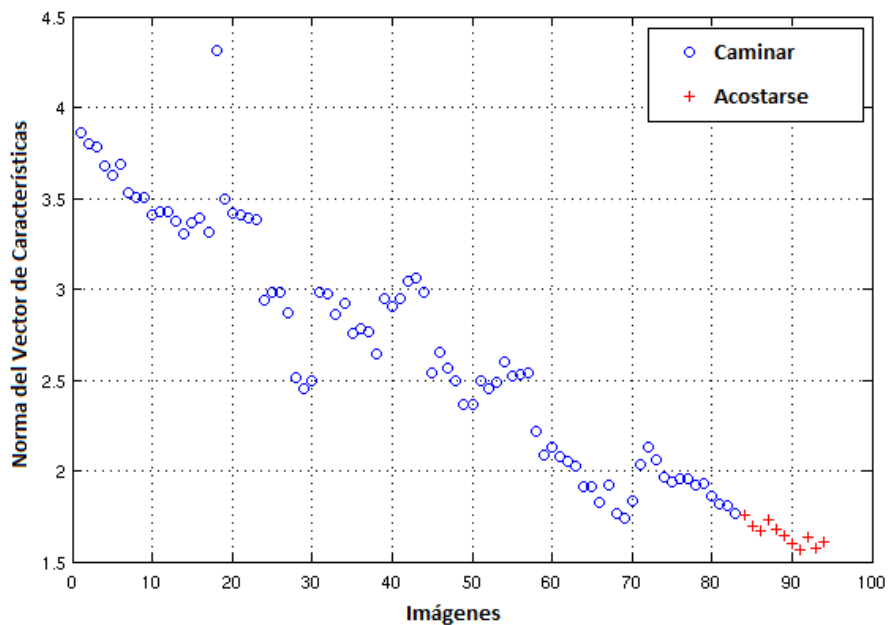
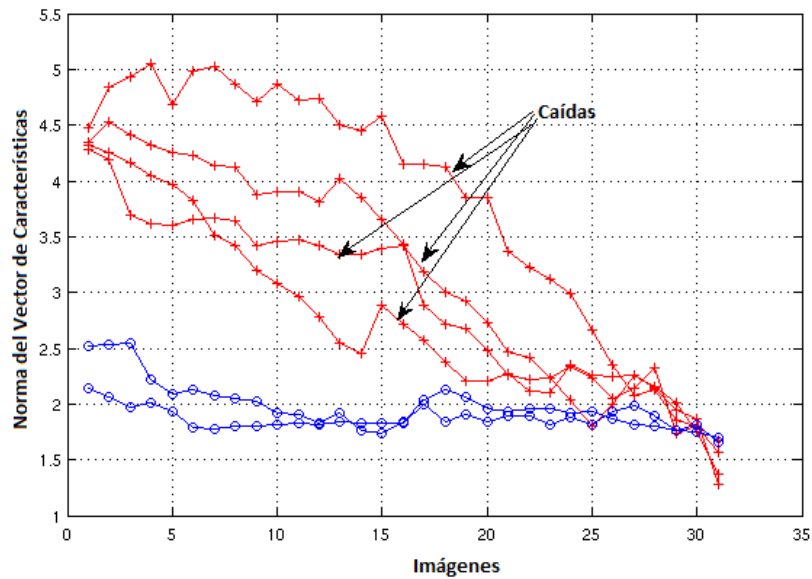


FIGURA 21. Una persona camina y se acuesta en el piso. La norma del vector de características seleccionadas con el AG es menor de 1.7 a partir de la imagen 83, por tanto cuando la persona se acuesta en el suelo se detectan posturas de caída.





**FIGURA 22.** En rojo se muestra la norma del vector de características seleccionadas con el AG en secuencias con treinta imágenes cuando la persona cae y en azul para secuencias de treinta imágenes cuando la persona se acuesta en el piso. El cambio de la norma del vector es mayor cuando la persona cae al piso comparado con el cambio de la norma del vector cuando la persona se acuesta en el piso.

Experimentalmente se estableció el valor de  $U_V$  en -1 cada treinta imágenes, ya que una caída dura de uno a dos segundos y con el Kinect se adquieren treinta imágenes por segundo. Para concluir se detectó una caída cuando la norma del vector de características fue menor de 1.7 y cambió en una magnitud menor de -1 cada treinta imágenes. Estos umbrales se utilizaron con las ocho secuencias de caídas utilizadas en la fase 2 y las cinco secuencias ADL utilizadas en la Fase 3. Se obtuvo una exactitud de reconocimiento de 100%

#### 4.1.2. Experimento 2

En este experimento se reconocieron caídas mediante el robot mostrado en la Fig. 23. El robot cuenta con un equipo Kinect montado sobre una unidad *pant-tilt* (PUT por sus siglas en inglés), se programó con ROS (*robotics operating system*). En la Fig. 24 se muestra la arquitectura modular del robot: en la primera capa se conectaron los dispositivos de entrada y salida, se obtuvieron las posiciones de la unidad *pant-tilt* y se capturó la información visual de la cámara RGBD; en la segunda capa se utilizaron los siguientes módulos:

- **Reconocimiento de personas y caídas:** con las coordenadas del centro del rostro se segmentó en profundidad la nube de puntos de la persona en la escena y se calculó su

centroide  $(C_x, C_y, C_z)$ , después se calcularon las características seleccionadas con el AG en el experimento uno, la norma del vector de características y el cambio de la norma del vector cada treinta imágenes, los dos últimos valores se compararon con los umbrales  $U_{PC}$  y  $U_V$  para reconocer cuando una persona cayó al piso.

- **Seguimiento de personas:** con un controlador PID la unidad *pant-tilt* siguió a la persona manteniendo las coordenadas  $(C_x, C_y, C_z)$  en el centro del campo visual de la cámara (Fig. 25).

La capa tres trabajó en modo exploratorio y cuando detectó el rostro de una persona envió a la capa dos las coordenadas del centro del rostro.



FIGURA 23. El robot se encuentra en el centro de investigaciones de inteligencia artificial. En la parte superior del robot se observa el Kinect montado sobre una PTU.

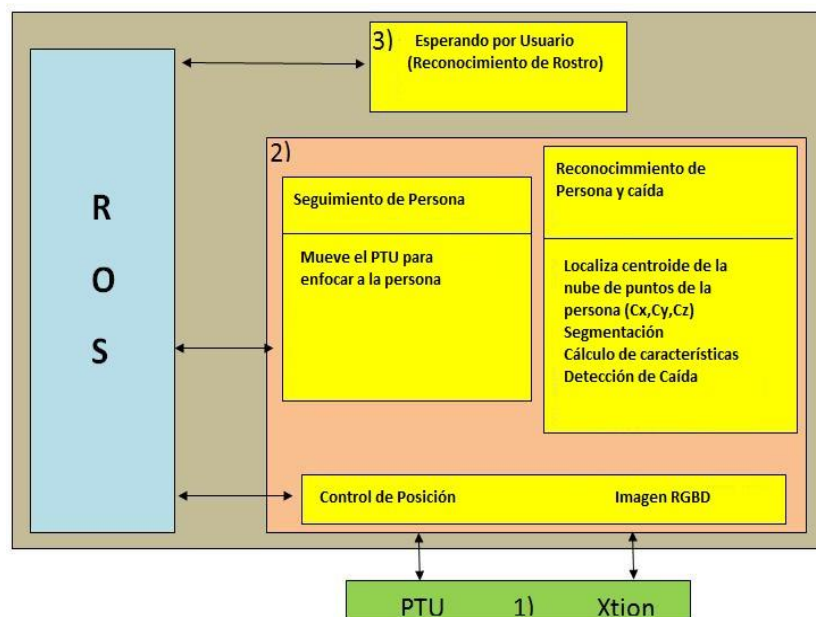
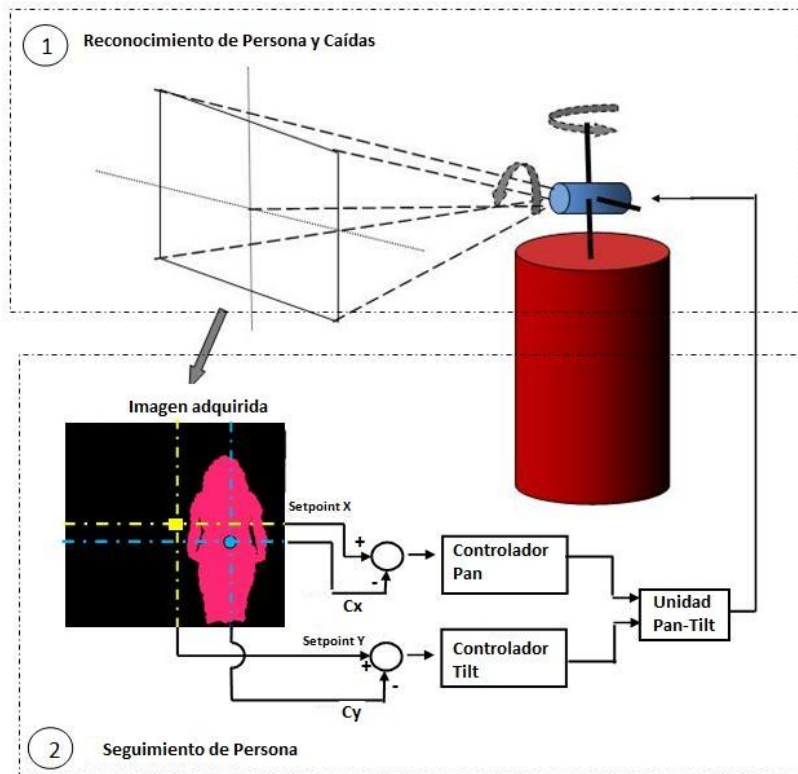


FIGURA 24. La arquitectura tiene tres capas: 1) la capa de dispositivos de entrada y salida; 2) la capa de seguimiento de persona y reconocimiento de caídas; 3) la capa de detección de rostros.

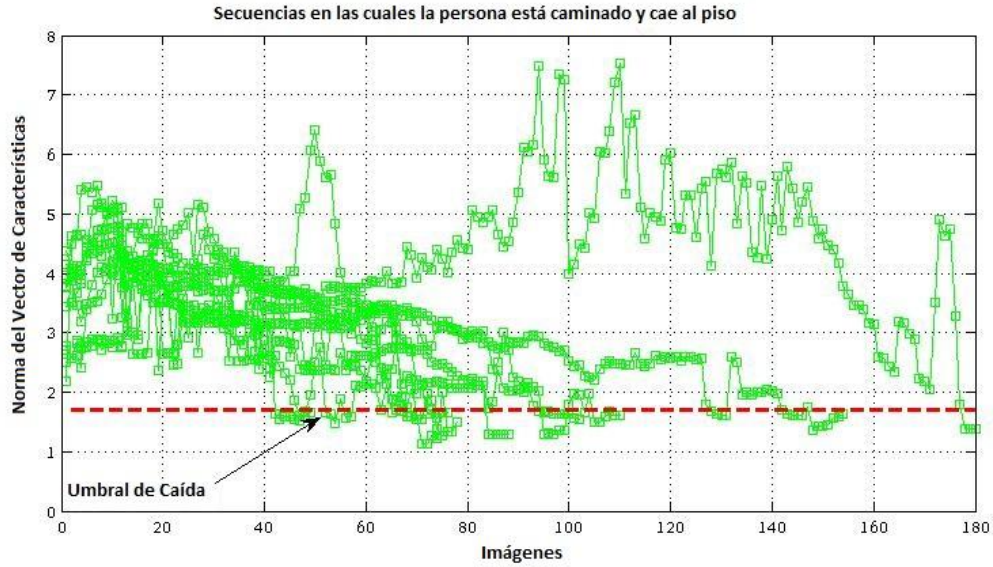


**FIGURA 25.** La segunda capa cuenta con dos módulos: 1) reconocimiento de personas y caídas y 2) seguimiento de personas.

En este experimento se realizaron tres acciones:

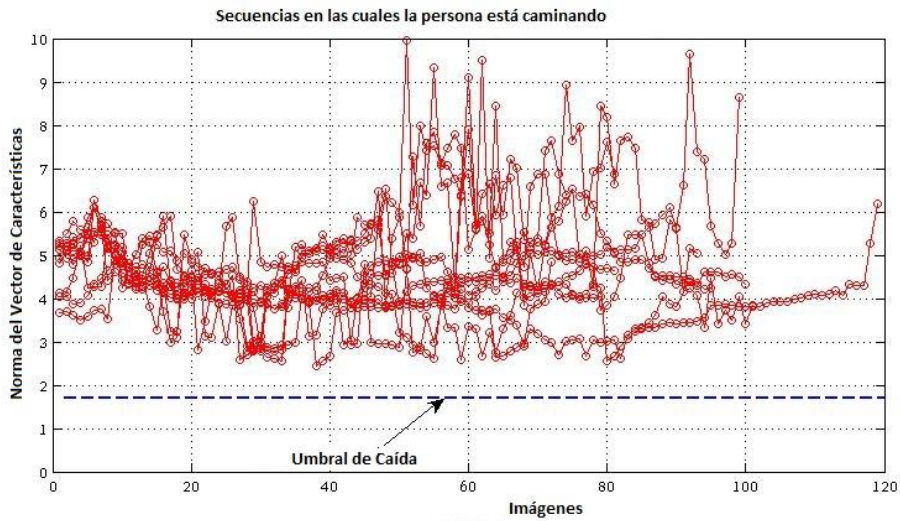
- Una persona caminó de un lado a otro y cayó al piso.
- Una persona caminó de un lado a otro.
- Una persona caminó de un lado a otro y se sentó.

Cada secuencia se realizó diez veces, en la Fig. 26 se grafica la magnitud de la norma del vector de características en las secuencias en donde la persona caminó y después cayó al piso y se obtuvo una exactitud de reconocimiento de la caída de 100%

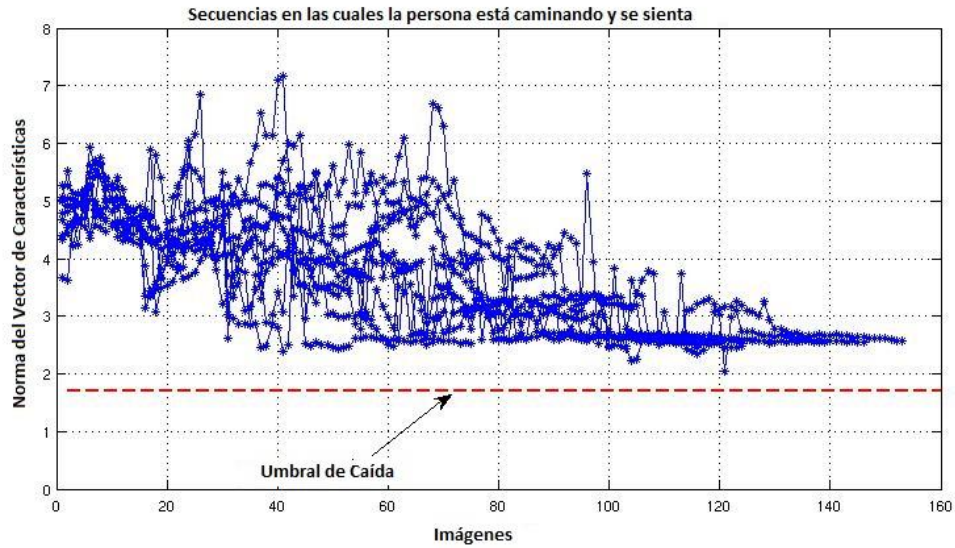


**FIGURA 26.** La persona camina y cae al piso. En todas las secuencias detectan posturas cuya norma del vector de características es menor que 1.7, por tanto la caída se detecta correctamente.

En la Fig. 27 se grafica la magnitud de la norma del vector de características en las secuencias donde la persona caminó de un lado a otro. En la Fig. 28 en las secuencias en donde la persona caminó y se sentó. En ambas figuras se observa que la magnitud de la norma del vector de características fue mayor de 1.7, por lo tanto no se detectó una postura de caída, lo cual fue correcto.



**FIGURA 27.** La persona camina de un lado a otro. La norma del vector de características en todas las imágenes es mayor que 1.7.



**FIGURA 28.** La persona camina y se sienta. La norma del vector de características en todas las imágenes es mayor que 1.7.

En la Fig. 29 se muestra una secuencia en donde la persona comenzó de pie, por tanto la capa tres reconoció un rostro y se activaron los módulos de seguimiento de personas y reconocimiento de personas y caídas. En la Fig. 29 se observa cómo la PTU siguió y centró a la persona en el plano de la imagen al mismo tiempo se calculó la norma del vector de características seleccionadas y se comparó con los umbrales para reconocer la caída.



**FIGURA 29.** En la imagen de la esquina superior izquierda, la capa tres detecta el rostro de una persona, en la imagen se observa como la persona está fuera del foco de la imagen. Los módulos de la capa dos se activan y se observa como la persona es centrada en el plano de la imagen y posteriormente se detecta la caída.

### 4.1.3. Experimento 3

En este experimento se evaluó la capacidad del MGG (sección 3.3.1.2) de distinguir cuando una persona se aproximó al suelo rápidamente (la persona cayó al piso) y lentamente (la persona se acostó en el piso). Las características se seleccionaron de un conjunto de 37 características, con veinte de ellas (Tabla 5) se describió la forma y orientación de la nube de puntos y con diecisiete se describió el cambio de la forma y orientación en treinta imágenes hacia atrás con respecto de la imagen en la que se calcularon las características. La base *TST Fall detection dataset v2* se utilizó en el experimento, la cual contiene imágenes de profundidad grabadas con el Kinect v2 y las coordenadas  $(x, y, z)$  de las articulaciones del esqueleto. En esta tesis con las coordenadas del esqueleto se segmentó la nube de puntos de una persona (Fig. 3) y se reconocieron las acciones caminar, sentarse, acostarse en el suelo, caer de frente, caer de espaldas, caer de lado y caer de espaldas e incorporarse. El experimento se realizó en dos fases, las cuales se explican en las siguientes secciones.

#### 4.1.3.1. Fase 1: Selección de Características

En esta fase se utilizaron dos secuencias de cada acción: caminar, sentarse, acostarse en el suelo, caer de frente, caer de espaldas, caer de lado y caer de espaldas e incorporarse. Con el conjunto de entrenamiento (segunda columna de la Tabla 12) se ajustaron los parámetros  $\mu_k, \Sigma_k$  del modelo generativo y con el conjunto de validación (cuarta columna de la Tabla 12) se evaluó la capacidad de predicción del modelo con las características seleccionadas. Cada imagen se clasificó manualmente (Tabla 12) en una de las siguientes clases:

- **Clase 1 postura de caída y evento de caída:** la postura de la persona se aproximó al piso en forma involuntaria y rápidamente.
- **Clase 2 postura de caída y no es evento de caída:** la postura de la persona se aproximó al suelo con movimientos lentos, por ejemplo cuando una persona se acostó en el piso.
- **Clase 3:** la persona caminó o se sentó.



**Tabla 12 Posturas de entrenamiento**

<b>Clase</b>	<b>Imágenes de entrenamiento</b>	<b>Porcentaje con respecto al conjunto de entrenamiento</b>	<b>Imágenes de validación</b>	<b>Porcentaje con respecto al conjunto de validación</b>
<b>1</b>	274	21%	252	15%
<b>2</b>	289	22%	373	22%
<b>3</b>	739	57%	1076	63%

Quando se seleccionaron las características con un AG y el MGG tipo uno y dos, se realizaron 35 ejecuciones, se utilizaron cien individuos, el porcentaje de cruce fue del 80%, se aplicó una mutación global y se evolucionó cada ejecución mil generaciones. En la Tabla 13 se muestran los resultados estadísticos de las ejecuciones:

**Tabla 13 Resultados estadísticos del algoritmo genético**

<b>Métrica</b>	<b>Clasificador Tipo 1</b>	<b>Clasificador Tipo 2</b>
<b>Promedio</b>	89.0%	86.51%
<b>Mediana</b>	90.1%	86.09%
<b>Desviación estándar</b>	0.29	0.29
<b>Mejor aptitud</b>	90.04%	87.65%
<b>Peor aptitud</b>	88.13%	84.56%

Con el AG y el clasificador MGG tipo uno, primero se seleccionaron las características  $f_s$ , las cuales describieron la forma y orientación de la nube de puntos de la persona, los individuos se definieron con veinte componentes, se utilizaron las imágenes de las tres clases (Tabla 12) para reconocer la postura de caída. Después se utilizó un segundo AG para seleccionar las características  $f_v$ , las cuales describieron el cambio de las características de la forma y orientación de la nube de puntos, sólo se entrenó con las imágenes de la clase uno y dos para reconocer un evento de caída. En la Tabla 14 en el segundo renglón se muestran las catorce características seleccionadas (tienen un valor de uno).

Con el AG y el clasificador MGG tipo 2, se utilizaron individuos de 37 componentes, con un AG se seleccionaron las características  $f_s$  y  $f_v$ , el entrenamiento se realizó con las imágenes de las tres clases. En la Tabla 14 en el tercer renglón se muestran las diecinueve características seleccionadas.

También se seleccionaron dieciocho características con PCA, las cuales describen el 99.46% del total de la información. Las características seleccionadas se muestran en la Tabla 14 en el cuarto renglón.

**Tabla 14 Características Seleccionadas**

Caract	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37		
AG-T1	1	0	0	1	0	0	1	1	1	0	0	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	1	0	0	0	1	0	1	0	1	1	0	0	
AG-T2	1	0	1	0	0	1	0	0	0	0	1	1	0	1	1	1	1	0	1	1	1	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0	1	1	1
PCA	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	0

#### 4.1.3.2. Fase 2: Detección de Postura de caída y Evento de caída

En este experimento se comparó el desempeño de los clasificadores (MGG tipo uno y dos) con los conjunto de características seleccionadas (Tabla 14). Para cuantificar el desempeño se utilizaron las ecuaciones (13), (14), (15) y (16). En esta fase se probaron 31 secuencias para cada una de las siete acciones seleccionadas en este experimento. El mejor desempeño (Tabla 15 y 16) se obtuvo con el clasificador Tipo 1 y las características seleccionadas con el AG. Se confirmó un evento de caída cuando en una ventana de treinta imágenes, la mayor cantidad contiene posturas de una persona aproximándose al suelo. La exactitud de reconocimiento de la postura de caída obtenida es comparable con los resultados reportados en el estado del arte y es aceptable considerando que sólo se utilizó información visual. Sin embargo cuando se detectó el evento de caída la exactitud del reconocimiento disminuyó a un 81.81%, posiblemente porque las caídas fueron simuladas y no se ejecutaron tan rápido como las caídas reales.



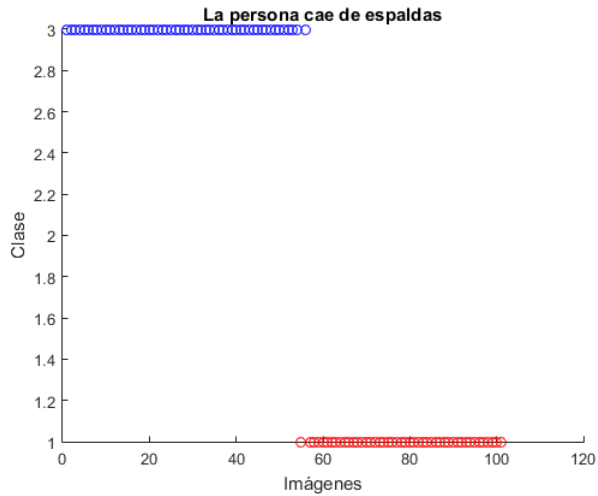
**Tabla 15 Métricas para reconocer postura de caída**

	<b>AG-Tipo 1</b>	<b>AG-Tipo 2</b>	<b>PCA-Tipo 1</b>	<b>PCA- Tipo 2</b>
<b>Sensibilidad</b>	97.63%	91.66%	85.49%	89.18%
<b>Especificidad</b>	100%	100%	100%	100%
<b>Exactitud</b>	98.26%	93.50%	87.87%	91.34%
<b>Error</b>	1.73%	6.49%	12.12%	8.65%

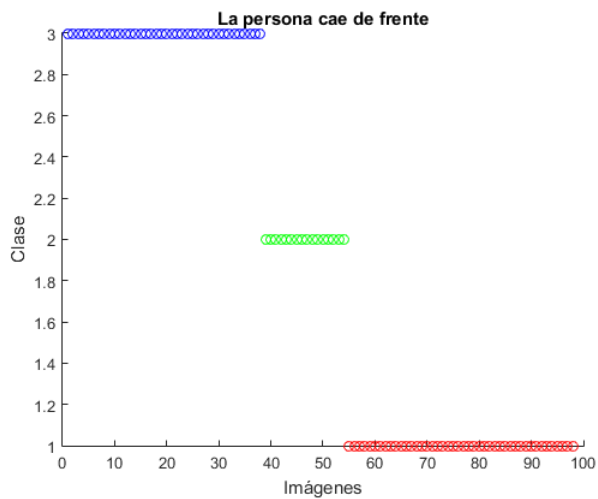
**Tabla 16 Métricas para reconocer el evento de caída**

	<b>AG-Tipo 1</b>	<b>AG-Tipo 2</b>	<b>PCA-Tipo 1</b>	<b>PCA- Tipo 2</b>
<b>Sensibilidad</b>	96.94%	89.58%	82.27%	86.84%
<b>Especificidad</b>	62%	58.62%	52.04%	58.22%
<b>Exactitud</b>	81.81%	77.92%	72.72%	77.05%
<b>Error</b>	18.18%	22.07%	27.27%	22.94%

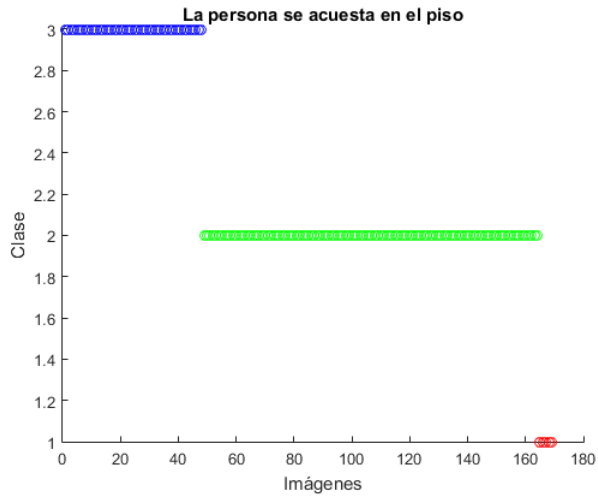
En las Fig. 30, 31, 32, 33 y 34 se muestran los resultados obtenidos cuando se utilizó el clasificador tipo uno con las características seleccionadas con el AG.



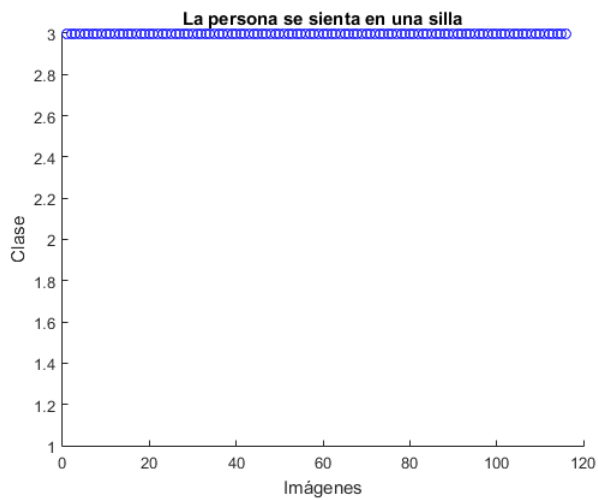
**FIGURA 30.** En la secuencia la persona camina y cae de espaldas. En color rojo (clase uno) se muestran las imágenes con posturas de una persona aproximándose al piso rápidamente y en color azul (clase tres) cuando la postura corresponde a una persona de pie o sentada.



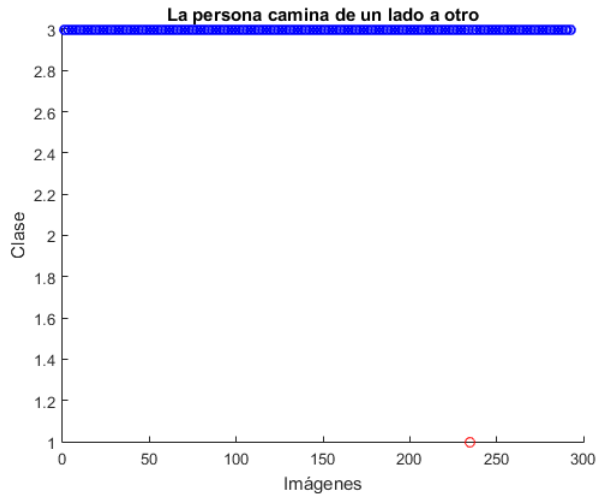
**FIGURA 31.** En la secuencia la persona camina y cae de frente. En color rojo (clase uno) se muestran las imágenes con posturas de una persona aproximándose al piso rápidamente, en color verde (clase dos) cuando la postura de la persona se aproxima al piso lentamente y en color azul (clase tres) cuando la postura corresponde a una persona de pie o sentada.



**FIGURA 32.** En la secuencia la persona camina y se acuesta en el piso. En color rojo (clase uno) se muestran posturas de una persona aproximándose al piso rápidamente, en color verde (clase dos) cuando la postura de la persona se aproxima al piso lentamente y en color azul (clase tres) cuando la postura corresponde a una persona de pie o sentada. Se confirma un evento de caída cuando la mayoría de las posturas en una ventana de treinta imágenes son clasificadas en la clase uno.



**FIGURA 33.** En la secuencia la persona se sienta, todas las posturas en las imágenes son clasificadas en la clase 3 (color azul).



**FIGURA 34.** En la secuencia la persona camina de un lado a otro, casi todas las posturas son clasificadas en la clase 3 (color azul) y sólo una postura en la clase 1 (color rojo), la cual podría ser ruido.

#### 4.1.4. Experimento 4

En este experimento se seleccionaron características del conjunto mostrado en la Tabla 5 con un AG y PCA, después se clasificaron las posturas de caída y posturas de no caída con un SVM. Las acciones caminar, sentarse, acostarse en el suelo, caer de frente, caer de espaldas, caer de lado y caer de espaldas e incorporarse de la base *TST Fall detection dataset v2* se utilizaron en el experimento.

El MGG propuesto en el experimento tres no fue capaz de abstraer la información para distinguir un evento de caída. Por esta razón en este cuarto experimento se utilizó la velocidad y aceleración de las articulaciones de ambas muñecas de la mano y del centroide de la nube de puntos de la persona, con el objetivo de mejorar el reconocimiento de un evento de caída.

El experimento se realizó en tres fases las cuales se presentan en las siguientes secciones.

##### 4.1.4.1. Fase 1: Selección de Características

En esta fase se utilizaron dos secuencias de cada acción: caminar, sentarse, acostarse en el suelo, caer de frente, caer de espaldas, caer de lado y caer de espaldas e incorporarse. Con el conjunto de entrenamiento (segunda columna de la Tabla 17) se creó el modelo SVM y con el conjunto de

validación (cuarta columna de la Tabla 17) se evaluó la capacidad de predicción del modelo con las características seleccionadas. Cada imagen se clasificó manualmente como se muestra en la siguiente tabla:

**Tabla 17 Posturas de entrenamiento**

<b>Clase</b>	<b>Imágenes de entrenamiento</b>	<b>Porcentaje con respecto al conjunto de entrenamiento</b>	<b>Imágenes de validación</b>	<b>Porcentaje con respecto al conjunto de validación</b>
<b>Posturas de caída</b>	148	10%	229	13%
<b>Posturas de no caídas</b>	1303	90%	1489	87%

El algoritmo genético maximizó el porcentaje de imágenes correctamente clasificadas. Se realizaron treinta ejecuciones, con quinientos individuos, el porcentaje de cruce fue del 80%, se utilizó mutación global y se evolucionó cada ejecución por mil generaciones. En la Tabla 18 se muestran los resultados estadísticos de las treinta ejecuciones:

**Tabla 18 Resultados estadísticos del algoritmo genético**

<b>Métrica</b>	<b>Clasificador SVM</b>
<b>Promedio</b>	94.2%
<b>Mediana</b>	93.8%
<b>Desviación estándar</b>	0.40
<b>Mejor aptitud</b>	94.87%

Las ocho características seleccionadas son C2, C4, C8, C9, C10, C15, C16, C17, dos de ellas corresponden a ángulos y las seis restantes a distancias. También se seleccionaron nueve características: C8, C9, C10, C11, C12, C13, C14, C15, C20 con PCA las cuales representan el 99% de la información total.

#### 4.1.4.2. Fase 2: Detección de Postura de Caída y Evento de Caída

En esta fase se clasificaron posturas de caída con SVM junto con los conjuntos de características seleccionados con el AG y PCA. En una ventana de treinta imágenes si la mayor cantidad de imágenes se clasificaron como postura de caída, se determinó que la persona cayó o se acostó en el piso, en caso contrario la persona caminó o se sentó.

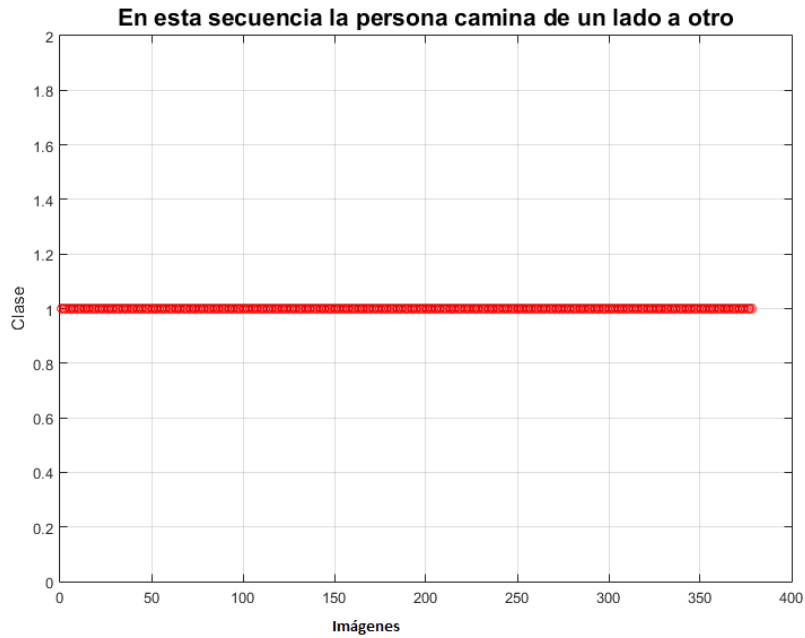
En la Tabla 19 en la segunda columna se muestran los valores obtenidos con las métricas cuando se utilizaron las características seleccionadas con el AG y en la tercera columna los resultados obtenidos con el conjunto de características seleccionadas con PCA.

**Tabla 19 Métricas para reconocer posturas de caída**

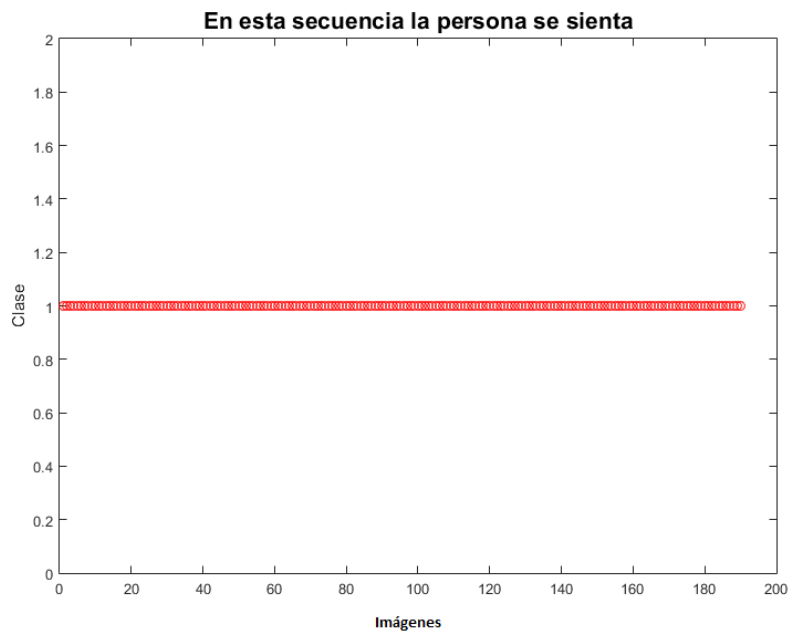
<b>Métrica</b>	<b>SVM-AG</b>	<b>SVM-PCA</b>
<b>Sensibilidad</b>	99%	93%
<b>Especificidad</b>	83%	67%
<b>Exactitud</b>	94%	84%
<b>Error</b>	6%	16%

En la Tabla 19 se observa una exactitud de reconocimiento del 94% cuando se utilizaron las características seleccionadas con el AG, la cual es superior en un 10% a la exactitud obtenida cuando se utilizaron las características seleccionadas con PCA.

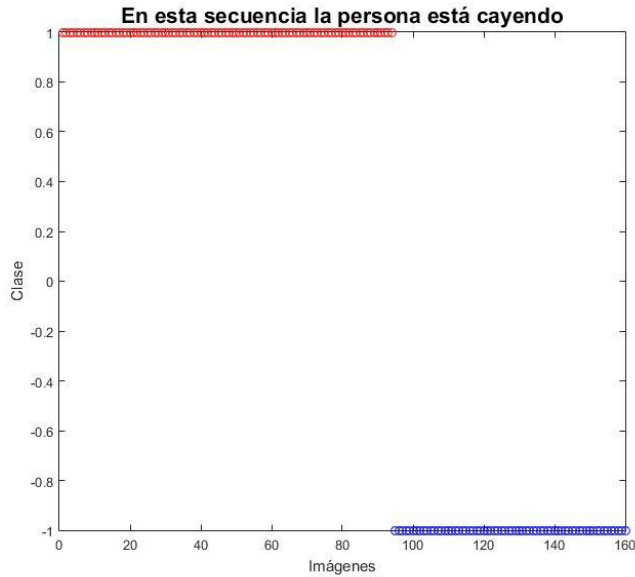
En las Fig. 35, 36, 37 y 38 se muestran los resultados obtenidos cuando se utilizaron las características seleccionadas con el AG para clasificar las acciones: caminar, estar sentado, caer de frente y acostarse en el piso, respectivamente. Cuando la persona se aproximó al suelo se clasificó en la clase con valor -1 y en caso contrario en la clase con valor de 1.



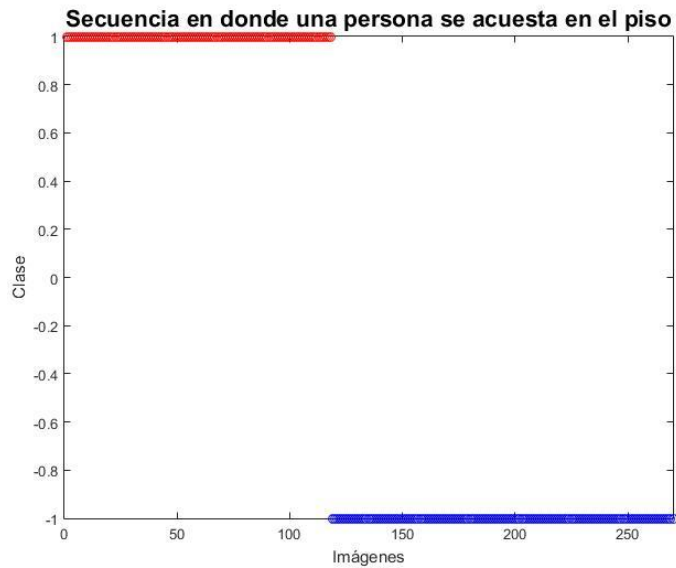
**FIGURA 35.** En la secuencia una persona camina. Todas las posturas son clasificadas en la clase 1 (color rojo) por tanto no se detectaron posturas de caída.



**FIGURA 36.** En la secuencia la persona se sienta. Todas las posturas son clasificadas en la clase 1 (color rojo) por lo tanto no se detectaron posturas de caída.



**FIGURA 37.** En la secuencia la persona cae al piso. Al principio de la secuencia la persona está de pie y se clasifican en la clase 1 (color rojo), a partir de la imagen 95, se detectaron posturas de caída y son clasificadas en la clase -1 (color azul). En la secuencia se reconoce una posible caída, pero se debe confirmar si la persona está cayendo o se acuesta en el piso.



**FIGURA 38.** Al comienzo de la secuencia una persona está de pie y las imágenes son clasificadas en la clase 1 (color rojo), a partir de la imagen 119 se detectaron posturas en donde la persona se aproxima al piso (color azul). En la secuencia se reconoce una posible caída, pero se debe confirmar si la persona está cayendo o se acuesta en el piso.



### 4.1.4.3. Fase 3: Reconocimiento de un evento de caída

En la fase dos se detectaron posturas con el cuerpo de la persona aproximándose al piso, adicionalmente en esta fase tres se calculó la velocidad ( $v_i$ ) y la aceleración ( $a_i$ ) de las articulaciones de las muñecas y del centroide de la nube de puntos de la postura de caída detectada en la imagen  $i$  mediante las siguientes ecuaciones:

$$v_i = \sqrt{v_{ix}^2 + v_{iy}^2 + v_{iz}^2} \qquad a_i = \frac{v_i - v_{i-n}}{t_i - t_{i-n}}$$

En donde

$$v_{ix} = \frac{x_i - x_{i-n}}{t_i - t_{i-n}}$$

$$v_{iy} = \frac{y_i - y_{i-n}}{t_i - t_{i-n}}$$

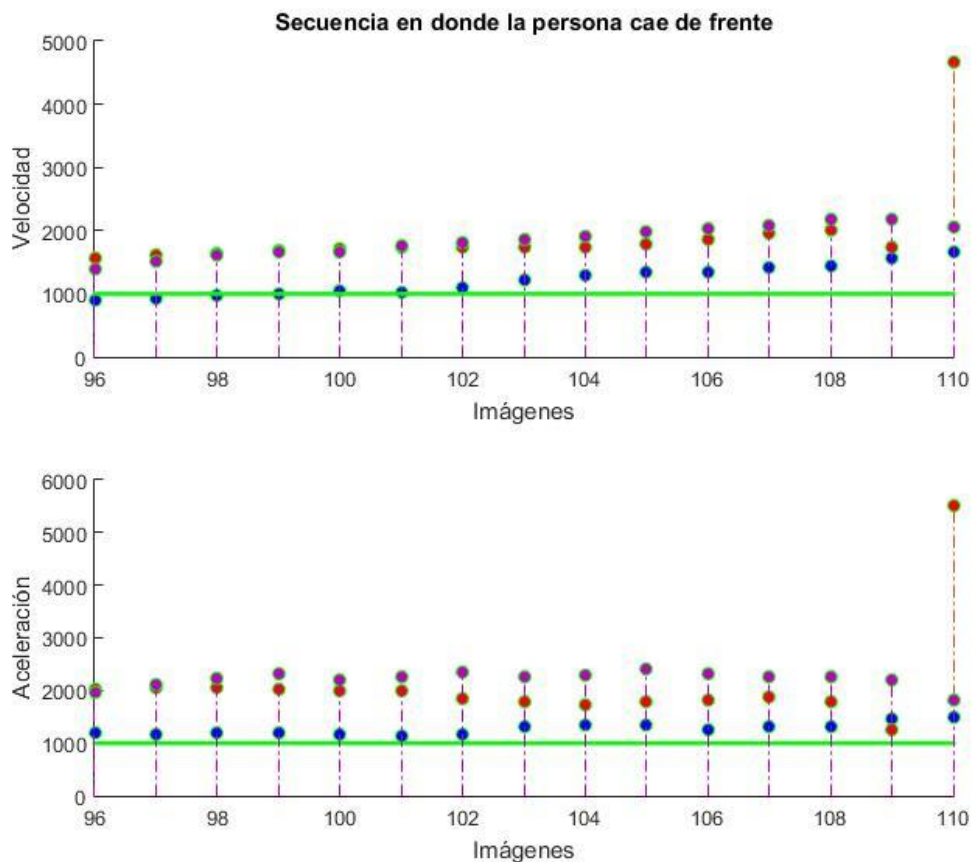
$$v_{iz} = \frac{z_i - z_{i-n}}{t_i - t_{i-n}}$$

$(x, y, z)$  representa la posición de una articulación,  $i$  es la imagen actual y  $n$  es la cantidad de imágenes hacia atrás con respecto a la imagen actual. La velocidad se midió en puntos/seg y la aceleración en puntos/seg<sup>2</sup>.

Con las secuencias de entrenamiento se determinó el valor de  $n$ . Cuando  $n = 1$  se compararon las posiciones de la imagen  $i$  con las posiciones de la imagen anterior, se observó un cambio pequeño en las velocidades y aceleraciones. Por esta razón no se distinguió cuando una persona cayó al piso y cuando se acostó en el piso. Posteriormente se varió el valor de  $n$  y finalmente con  $n=20$  se observó un cambio significativo en las velocidades y aceleraciones cuando la persona cayó al piso.

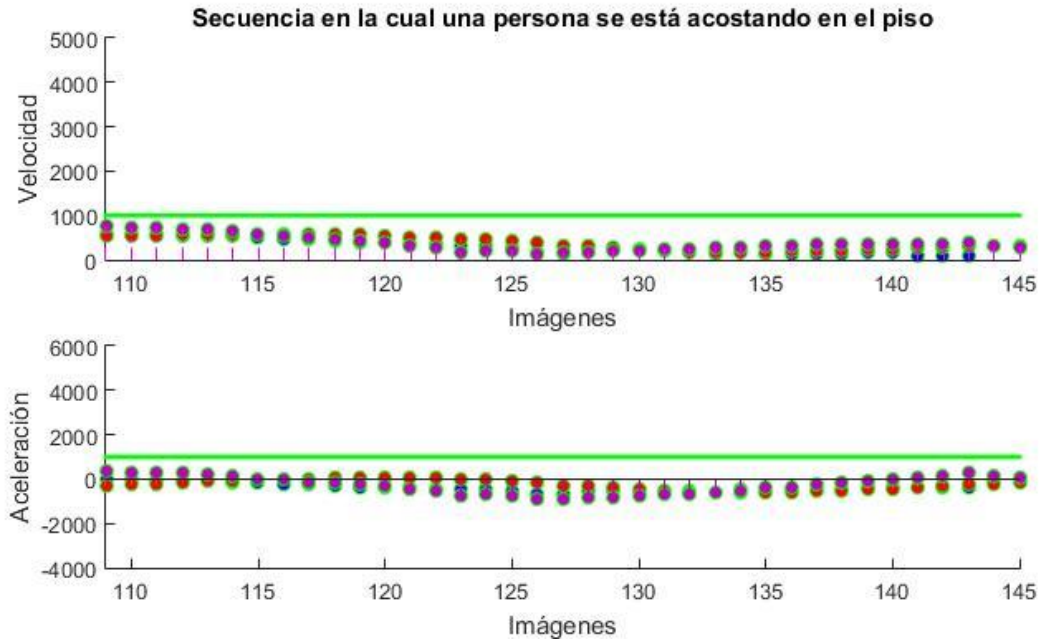
La base *TST Fall detection dataset v2* contiene las posiciones  $(x, y, z)$  de las articulaciones, los mapas de profundidad y el tiempo  $t_i$  en el que se capturó cada imagen  $i$  de la secuencia; las velocidades y aceleraciones se calcularon con las posiciones y los tiempos provistos en la base de pruebas. Con el Kinect se capturaron treinta imágenes por segundo, por tanto en aproximadamente cada 0.0333 segundos se capturó una imagen.

Cuando una persona cae las manos se mueven rápidamente para minimizar el impacto de la caída, y el centro de masa se mueve rápidamente en dirección al plano del piso, por esta razón sus posiciones se utilizaron para calcular velocidades y aceleraciones mediante las cuales se confirmó cuando una persona cayó al piso. En el conjunto de entrenamiento se observó que cuando las personas cayeron al piso la velocidad y aceleración de dos de los tres elementos (muñeca izquierda, muñeca derecha y centroide) fueron mayores al umbral establecido experimentalmente con valor de mil, en caso contrario la persona se acostó en el suelo. En la Fig. 39 se observa la velocidad y la aceleración de una secuencia con una persona cayendo al piso, las velocidades y aceleraciones de la muñeca izquierda, muñeca derecha y centroide fueron mayores al umbral, por esta razón el evento de caída se reconoció correctamente.



**FIGURA 39.** En la secuencia la persona cae al piso. En azul se muestra la medición del centroide, en rojo la muñeca izquierda, en morado la muñeca derecha y en verde el umbral. La gráfica superior corresponde a la velocidad y la de abajo a la aceleración.

En la Fig. 40 se muestra una secuencia con una persona acostándose en el piso, se detectaron posturas de caída, sin embargo las velocidades y aceleraciones de las muñecas y centroide fueron menores que los umbrales y por tanto no se confirmó como evento de caída.



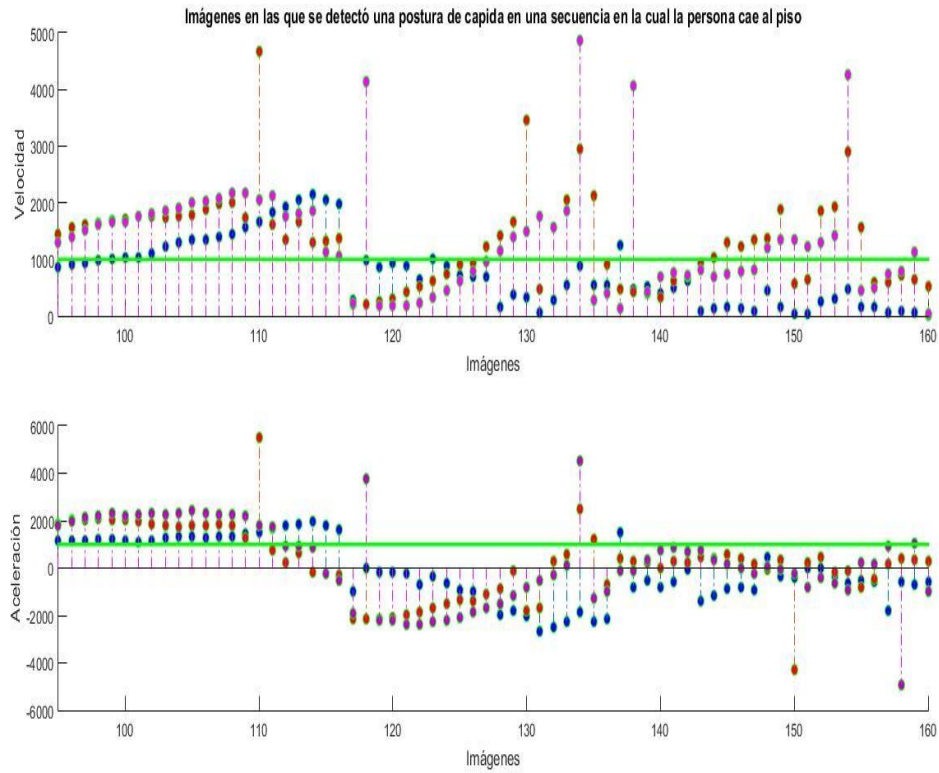
**FIGURA 40.** En la secuencia la persona se acuesta en el piso. En azul se muestra la medición del centroide, en rojo la muñeca izquierda, en morado la muñeca derecha y en verde el umbral. La gráfica superior corresponde a la velocidad y la de abajo a la aceleración.

En una ventana de treinta imágenes cuando la mayoría de ellas se clasificaron como posturas de caída entonces se compararon las velocidades y aceleraciones del centroide de la nube de puntos, la articulación de la muñeca derecha e izquierda con los umbrales y en su caso se confirmó el evento de caída. En la Tabla 20 se muestran los resultados de las métricas de desempeño para reconocer el evento de caída cuando se utilizó el conjunto de características seleccionadas con el AG y con PCA.

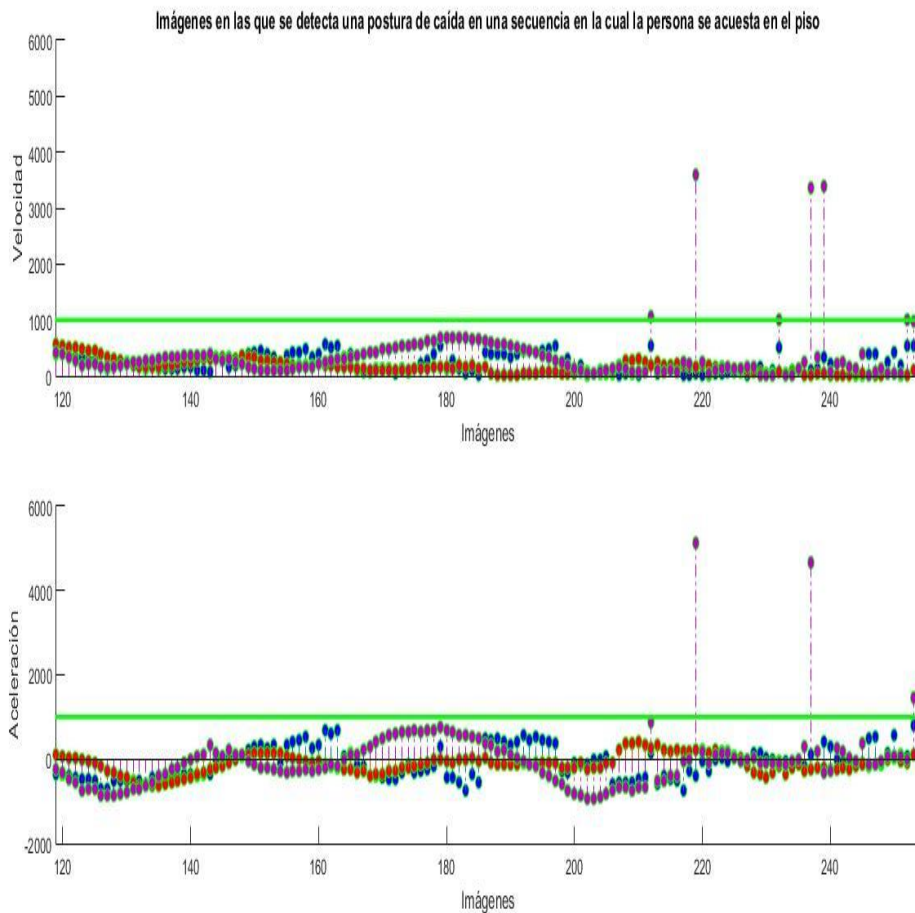
Tabla 20 Métricas para confirmar el evento de caída

Métrica	SVM-AG	SVM-PCA
Sensibilidad	98%	97%
Especificidad	69%	52%
Exactitud	90%	81%
Error	6%	13%

Con las características seleccionadas con el AG, las características de velocidad y aceleración se obtuvo una exactitud del reconocimiento de 90%, la cual es superior a la exactitud de reconocimiento obtenida en el experimento tres. En la Fig. 41 se muestran las velocidades y aceleraciones de una secuencia con una persona cayendo de frente. La Fig. 42 corresponde a una secuencia con una persona acostándose en el piso.



**FIGURA 41.** En las gráficas se muestran los valores de velocidad y aceleraciones para el centroide (color azul), muñeca derecha (color rojo) y muñeca izquierda (color morado) para las imágenes clasificadas con posturas aproximándose al suelo. Al menos dos de las tres mediciones en cada gráfica son mayores al umbral por tanto se confirma que la persona cayó al piso.



**FIGURA 42.** En las gráficas se muestran los valores de velocidad y aceleraciones para el centroide (color azul), muñeca derecha (color rojo) y muñeca izquierda (color morado) para las imágenes clasificadas con posturas aproximándose al suelo. Sólo una de las mediciones en cada gráfica es mayor al umbral por esta razón se determina que la persona se acostó en el piso.

#### 4.1.5. Discusión de los Resultados

En los experimentos uno, tres y cuatro se seleccionaron características con un algoritmo genético, la aptitud del individuo se calculó con el porcentaje de posturas de entrenamiento correctamente clasificadas, además se utilizó PCA para seleccionar características. En todos los experimentos realizados, los conjuntos de características seleccionados con el AG mejoraron la exactitud de reconocimiento cuando se comparó con el conjunto de características seleccionados por PCA. En la Tabla 21 se muestra un comparativo de los resultados obtenidos en los experimentos.

**Tabla 21 Resultados de los experimentos para detección de caídas**

	Porcentaje de exactitud con características seleccionadas por AG	Porcentaje de exactitud con características seleccionadas por PCA	Clasificador	Base de datos
Experimento 1	96.10%	89.14%	KNN	UR Fall Detection Dataset
Experimento 3	81.81%	77.05%	Modelo Generativo Gaussiano	TST Fall detection dataset v2
Experimento 4	90%	81%	SVM	TST Fall detection dataset v2

Los resultados mostrados en la Tabla 21 sugieren que el AG es un algoritmo de selección de características robusto. En el experimento dos se utilizaron las características seleccionadas por el AG en el experimento 1, en tiempo real y se obtuvo una exactitud de reconocimiento del 100%. Los resultados obtenidos en el experimento cuatro son comparables con los reportados en el estado del arte (Gasparrini, y otros, 2016), ellos reportan un 90% de exactitud con la misma base de datos y con características de cambios la posición de articulaciones del esqueleto e información capturada con dos sensores inerciales colocados en la muñeca de la mano de la persona y en su cintura.

## **4.2. Experimentos de Reconocimiento de Acciones**

Se realizaron dos experimentos para reconocer acciones, el conjunto de características se calculó con las ecuaciones (7), (8), (9) y (10), a continuación se describe brevemente cada experimento:

- **Experimento 1:** se calculó la exactitud del reconocimiento de acciones de las bases UTD-MHAD y KARD mediante DTW y KNN con las características seleccionadas con el algoritmo RFD-SD, además en algunas acciones se seleccionaron características con un AG con la finalidad de mejorar todavía más la exactitud del reconocimiento.
- **Experimento 2:** la base MSR-Action3D es una de las más utilizadas para evaluar los algoritmos de reconocimiento de acciones, por esta razón se utilizó en este experimento. Esta base contiene deformaciones en las posiciones de las articulaciones cuando una persona realizó acciones como agacharse, con el objetivo de minimizar las deformaciones se compararon los filtros de Savitsky-Golay y el de Kalman. Después de minimizar los

efectos del ruido se seleccionaron las características con el algoritmo RFD-SD y se clasificaron las acciones con DTW y KNN.

### 4.2.1. Experimento 1

Con el algoritmo propuesto RFD-SD se seleccionó un conjunto de características para cada acción, las características seleccionadas en cada conjunto presentan la mayor desviación estándar cuando se ejecuta la acción. En algunas clases se seleccionaron características con un AG para mejorar aún más la exactitud del reconocimiento. Por último se seleccionaron características con PCA con el objetivo de evaluar la exactitud del reconocimiento cuando se utilizaron los conjuntos de características seleccionados y el conjunto completo de características. En este experimento se clasificaron las acciones de las bases KARD y UTD Multimodal Human Action Dataset (UTD-MHAD) con DTW y KNN (se seleccionaron los cinco vecinos más cercanos).

#### 4.2.1.1. UTD-MHAD

La base UTD-MHAD contiene veintisiete acciones (Tabla 22), ocho personas realizaron cada acción con cuatro repeticiones.

**Tabla 22 Número de características seleccionadas en cada acción**

Acciones	RFD-SD	AG	PCA
Deslizar el brazo derecho a la izquierda	11	-	56
Deslizar el brazo derecho a la derecha	11	-	55
Ondular la mano derecha	13	51	61
Palmada frontal de dos manos	6	-	55
Lanzamiento del brazo derecho	10	64	59
Brazos cruzados en el pecho	6	53	56
Lanzamiento de baloncesto	8	57	51
Dibujar una x con la mano derecha	11	-	59
Dibujar un círculo con la mano derecha (en el sentido de las agujas del reloj)	8	-	61

Dibujar un círculo con la mano derecha (en sentido anti-horario)	10	51	57
Dibuja un triángulo	12	-	63
Lanzamiento de bolos (mano derecha)	6	48	69
Boxeo frontal	11	-	51
Columpio de béisbol desde la derecha	7	56	55
Golpe de Tenis con la mano derecha	8	-	65
Enrollamiento de brazos (dos brazos)	8	-	54
Servicio de tenis	7	49	58
Empuje de dos manos	13	-	50
Golpeteo de la mano derecha en la puerta	10	-	64
Mano derecha atrapa un objeto	8	-	64
Recoge y lanzamiento con la mano derecha	5	-	68
Correr en el lugar	9	-	59
Caminando en el lugar	6	-	63
Sentarse y pararse	4	-	68
Pararse para sentarse	4	-	69
Estocada delantera (pie izquierdo hacia adelante)	6	53	69
Sentadilla (dos brazos estiran)	10	-	63

Las acciones son secuencias de imágenes, pero no todas las secuencias tienen la misma cantidad de imágenes. Con las posiciones de las veinte articulaciones del esqueleto de cada imagen se calcularon 75 características con las ecuaciones (7), (8), (9) y (10).

Con el objetivo de comparar la exactitud del reconocimiento obtenida en este experimento con los resultados reportados en el estado del arte por (Chen, Jafari, & Kehtarnavaz, 2015), al igual que ellos, en este experimento se utilizaron las acciones realizadas por los sujetos uno, tres, cinco y



siete en la fase de entrenamiento y las realizadas por las personas dos, cuatro, seis y ocho en la fase de pruebas.

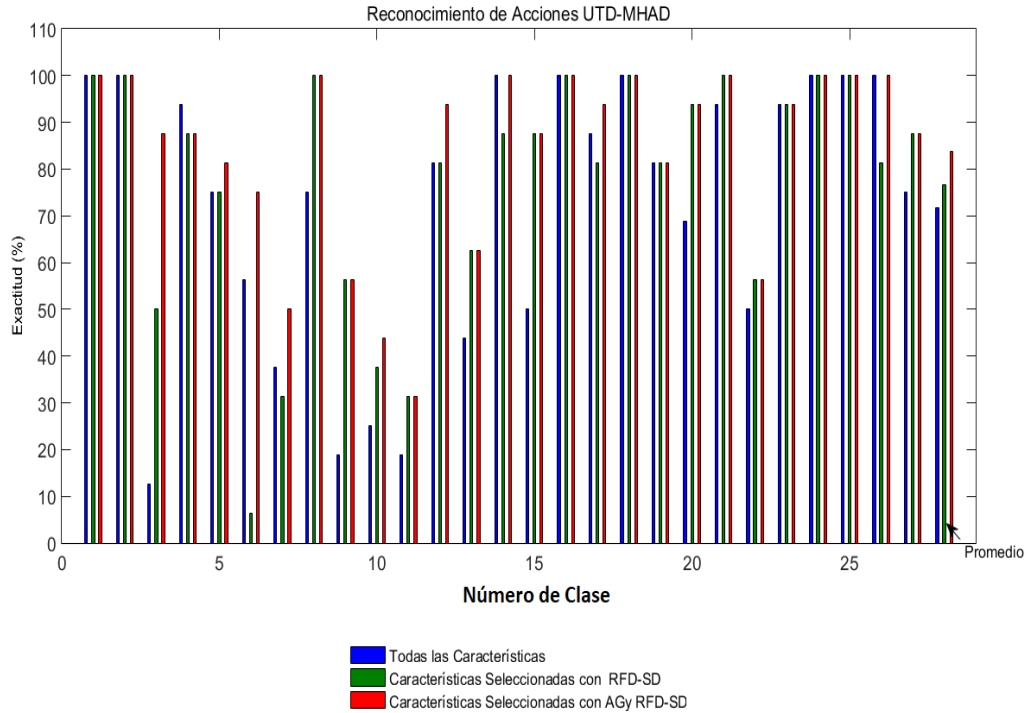
En la fase de entrenamiento con el algoritmo RFD-SD se redujo el tamaño del conjunto de características en cada clase. En la Tabla 22 en la segunda columna se muestra cuantas características se seleccionaron, la dimensión se redujo en promedio en un 88%.

En la fase de pruebas se reconocieron las acciones con las 75 características en cada clase y se obtuvo una exactitud de reconocimiento en promedio de 71.8%. Cuando se utilizaron las características seleccionadas con el algoritmo RDF-SD la exactitud en el reconocimiento se incrementó a un promedio de 76.6%.

En la Fig. 43 se muestra la exactitud del reconocimiento de cada clase, también se observa que en las acciones tres, cinco, seis, siete entre otras cuando se utilizaron las características seleccionadas la exactitud del reconocimiento es menor del 80%. Por esta razón en las clases mencionadas se seleccionaron características con el AG. En la tercera columna de la Tabla 22 se muestra la cantidad de características seleccionadas.

Cuando se utilizaron las características seleccionadas con el AG y con el algoritmo RFD-SD se redujo la dimensión de los conjuntos de características seleccionadas en un promedio de 28.6% y la exactitud en el reconocimiento se incrementó a un promedio de 83.8%. La exactitud en el reconocimiento reportado por (Chen, Jafari, & Kehtarnavaz, 2015) es del 66.1% cuando sólo utilizan el Kinect para reconocer acciones.

Adicionalmente con PCA se seleccionaron características (cuarta columna de la Tabla 22), el conjunto de características se redujo en un 20%. Esta reducción es menor a la obtenida con el algoritmo RFD-SD y el AG, entonces las características seleccionadas con PCA no se evaluaron con el clasificador.



**FIGURA 43.** En la gráfica se muestra la exactitud de reconocimiento en cada clase y la exactitud promedio para las veintisiete acciones cuando se usan todas las características, las características seleccionadas con el algoritmo RFD-SD y las características seleccionadas con el AG y el algoritmo RFD-SD.

#### 4.2.1.2. KARD

La base KARD contiene dieciocho acciones ejecutadas por diez personas con tres repeticiones. Las acciones se agruparon en tres conjuntos de actividades (Tabla 23). Cada grupo presentó diferentes niveles de dificultad durante el reconocimiento.

Las acciones son secuencias de imágenes, pero no todas las secuencias tienen la misma cantidad de imágenes. Con las posiciones de las quince de las articulaciones del esqueleto de cada imagen se calcularon 56 características con las ecuaciones (7), (8), (9) y (10).

**Tabla 23 Acciones de la base KARD organizadas en tres grupos**

<b>Grupo 1</b>	<b>Grupo 2</b>	<b>Grupo 3</b>
Ondular el brazo horizontalmente	Ondular el brazo en alto	Dibujar /
Ondular las dos manos	Patada lateral	Beber
Flexionarse	Capturar	Sentarse
Llamar por teléfono	Dibujar /	Llamar por teléfono
Ponerse de pie	Aplaudir	Tomar una sombrilla
Patear hacia adelante	Patear hacia adelante	Aventar un papel
Dibujar una x	Flexionarse	Golpe alto
Caminar	Sentarse	Ondular el brazo horizontalmente

Con el propósito de comparar los resultados obtenidos en este experimento con los reportados en la prueba C de (Gaglio, Re, & Morana, 2015), al igual que ellos la mitad de las secuencias se utilizaron en la fase de entrenamiento y la otra mitad en la fase de pruebas. Las acciones realizadas por los sujetos uno, tres, cinco, siete y nueve se utilizaron para seleccionar características y con las acciones realizadas por los sujetos dos, cuatro, seis, ocho y diez se evaluó y comparó la exactitud del reconocimiento con varios conjuntos de características.

En la fase de entrenamiento con el algoritmo RFD-SD se redujo el tamaño de los conjuntos de características en cada clase. En la Tabla 24 en la segunda columna se muestra cuantas características se seleccionaron, la dimensión se redujo en promedio en un 89%. Además cuando se seleccionaron características utilizando PCA (tercera columna de la Tabla 24) se redujo el conjunto de características en un 20%. Esta reducción es menor a la obtenida con el algoritmo RFD-SD, por esto las características seleccionadas con PCA no se evaluaron con el clasificador.

**Tabla 24 Número de características seleccionadas en cada acción**

<b>Acción</b>	<b>RFD-SD</b>	<b>PCA</b>
Ondular el brazo horizontalmente	3	47

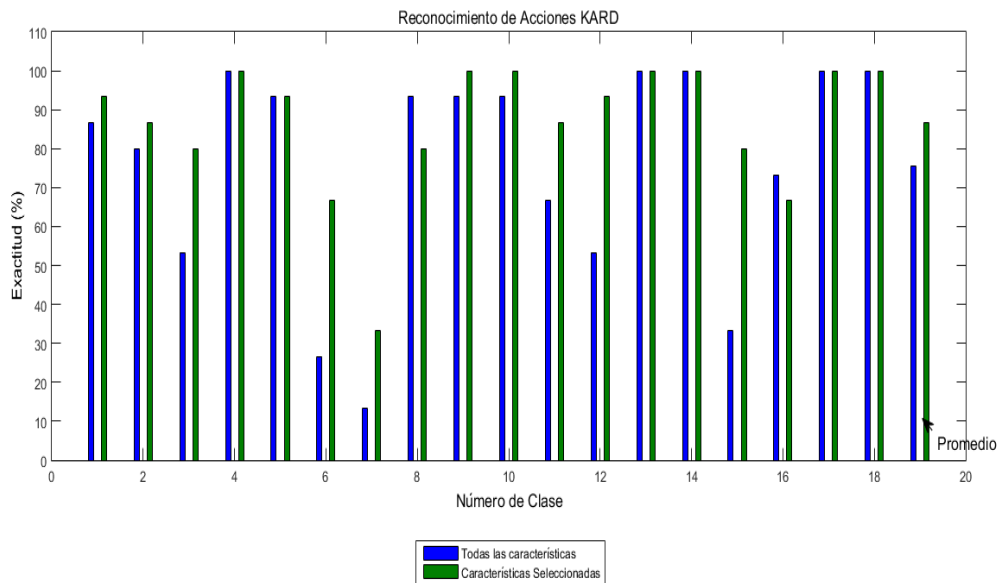
Ondular el brazo en alto	7	53
Ondular las dos manos	4	44
Capturar	5	52
Golpe alto	8	49
Dibujar una x	7	51
Dibujar /	6	53
Tirar un papel	6	56
Patear hacia adelante	3	55
Patada lateral	12	52
Tome el paraguas	7	54
Flexionarse	3	48
Aplaudir	10	48
Caminar	4	48
Llamar por teléfono	4	50
Beber	5	49
Sentarse	8	49
Ponerse de pie	12	49

En la fase de pruebas se clasificaron las acciones por grupos (Tabla 23), cuando las 56 características se utilizaron en cada clase se obtuvo una exactitud de reconocimiento en promedio de 79.2%. Cuando se utilizaron las características seleccionadas por el algoritmo RDF-SD la exactitud en el reconocimiento se incrementó a un promedio de 94.17% y en (Gaglio, Re, & Morana, 2015) en un experimento similar se reporta una exactitud en promedio de 88.3%. En la Tabla 25 se muestran los resultados.

**Tabla 25 Exactitud de reconocimiento de acciones para los grupos de acciones de la base KARD**

	<b>Grupo 1</b>	<b>Grupo 2</b>	<b>Grupo 3</b>
Todas las Características	83.8%	83.8%	70.0%
<b>Características seleccionadas con algoritmo RFD-SD</b>	<b>97.5%</b>	<b>95%</b>	<b>90%</b>
Resultados Reportados (Gaglio, Re, & Morana, 2015)	93%	90.1%	81.8%

En la fase de pruebas también se realizó el reconocimiento de una nueva acción comparándola con las dieciocho acciones (Tabla 24). Cuando se utilizaron las 56 características en cada clase se obtuvo una exactitud de reconocimiento de 75.6% en promedio. Cuando se utilizaron las características seleccionadas por el algoritmo RDF-SD se incrementó la exactitud en el reconocimiento a un 86.7% en promedio. Los resultados obtenidos se muestran en la Fig. 44.



**FIGURA 44.** En la gráfica se muestra la exactitud de clasificación en las clases y la exactitud promedio del reconocimiento de las dieciocho acciones de la base KARD, cuando se utilizando todas las características y cuando se utilizaron las características seleccionadas por el algoritmo RFD-SD.

En la Fig.44 se observa que la exactitud de reconocimiento en algunas clases, por ejemplo la seis, siete y dieciséis cuando se utilizaron las características seleccionadas fue menor al 70%. Entonces para las clases mencionadas se seleccionaron características con el AG, sin embargo no se mejoró la exactitud del reconocimiento. Por esta razón sólo se utilizaron las características seleccionadas por el algoritmo RFD-SD y todas las características en la fase de pruebas.

#### **4.2.2. Experimento 2**

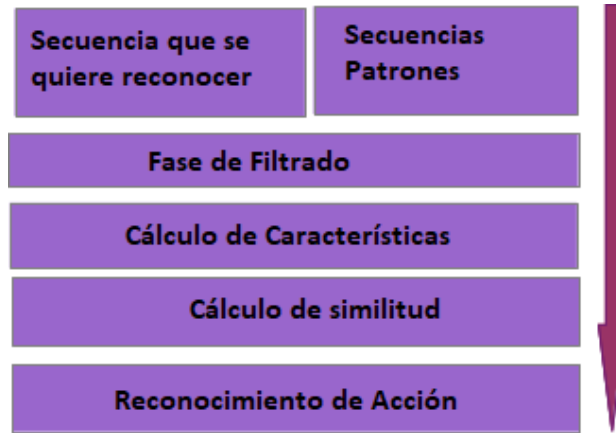
En (Padilla-López, Chaaraoui, & Flórez-Revuelta, 2014) se hace una revisión de 62 trabajos, cuyas propuestas de reconocimiento de acciones son evaluadas con la base MSR-Action3D<sup>6</sup>, lo cual significa que es una de las bases más utilizadas en el estado del arte, por esta razón en este experimento se utilizó dicha base.

La base MSR-Action3D contiene veinte acciones realizadas por diez personas con tres repeticiones en promedio para cada acción. En total son 567 secuencias. Esta base provee las posiciones  $(x, y, z)$  de veinte articulaciones del esqueleto para cada una de las imágenes que integran una secuencia de acción. Una característica importante de esta base son las deformaciones presentadas en las articulaciones cuando las personas se aproximan al piso para agacharse, recoger algo o acostarse en el piso, las cuales se presentan cuando se realizan acciones en ambientes reales.

En este experimento se corrigieron las posiciones de las articulaciones con los filtros de Kalman y el de Savitzky-Golay. El primero es un filtro ampliamente conocido, incluso se han implementado diferentes versiones del mismo (Wan & Van Der Merwe, 2000) (LaViola, 2003) (Loumponias, Vretos, Daras, & Tsaklidis, 2016) y el segundo se utilizó en (Du, Wang, & Wang, 2015) para corregir las posiciones de las articulaciones y reconocer acciones. En la Fig. 45 se muestra el sistema de reconocimiento de acciones propuesto en este experimento, se agregó la fase de filtrado y las fases restantes se describen en la sección 3.3.2.

---

<sup>6</sup> <https://www.uow.edu.au/~wanqing/#Datasets>



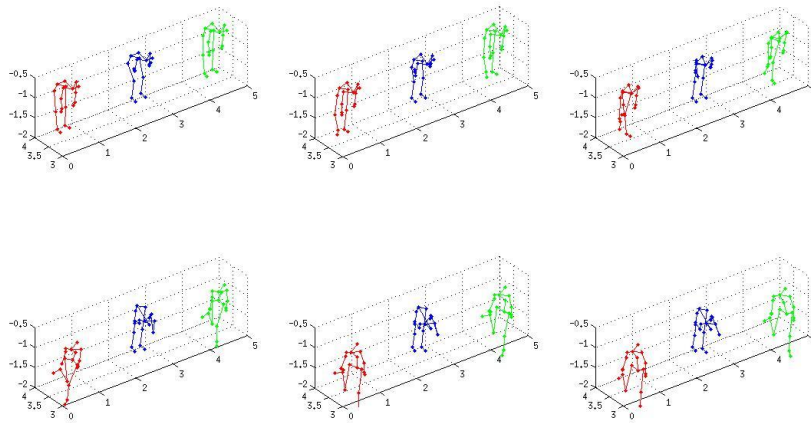
**FIGURA 45.** Reconocimiento de las acciones: repositorio de características seleccionadas con las secuencias patrones, secuencia a reconocer, fase de filtrado para corregir las posiciones de las articulaciones del esqueleto, calculo de características, cálculo de similitud entre la secuencia a reconocer y las secuencias patrones y por último reconocimiento de la acción.

#### **4.2.2.1. Fase 1: Filtrado de los esqueletos**

En esta fase se evaluaron los filtros de Kalman y de Savitsky-Golay, los detalles de implementación del primero se pueden revisar en (Maybeck, 1982) y (Loumponias, Vretos, Daras, & Tsaklidis, 2016), el segundo se utilizó en (Du, Wang, & Wang, 2015) para corregir las posiciones de las articulaciones y reconocer de acciones.

El Filtro de Savitsky-Golay es un filtro de suavizado, en esta tesis se utilizaron los coeficientes [-3 12 17 12 -3] para suavizar los cambios en las posiciones  $(x, y, z)$  de las articulaciones del esqueleto en la imagen actual. El nuevo valor de las articulaciones se obtuvo con las posiciones de las articulaciones en dos imágenes anteriores, en la imagen actual y en dos imágenes posteriores, los detalles de la implementación se pueden revisar en (Gorry, 1990).

En la Fig. 46 se muestran los resultados obtenidos con los filtros de Kalman y Savitsky-Golay con seis esqueletos, en los resultados se observa una mejor corrección de las posiciones de las articulaciones cuando se utilizó el filtro de Kalman, las deformaciones y por lo tanto el ruido fue minimizado en mayor grado cuando se compara con el resultado obtenido con el filtro de Savitsky-Golay. Sin embargo no se tuvo acceso a esqueletos de referencia sin deformaciones para calcular el ruido presente en los esqueletos, por tanto fue imposible determinar cuál de los dos filtros minimizó en mayor medida la señal de ruido, entonces se decidió evaluar los dos filtros junto con la selección de características y el clasificador.



**FIGURA 46.** Se muestran seis imágenes del esqueleto de una persona flexionándose al piso, en color rojo se muestran los esqueletos adquiridos directamente del dispositivo de profundidad, en los color azul los esqueletos corregidos con el filtro de Kalman y en color verde los esqueletos suavizados con el filtro de Savitsky-Golay.

#### 4.2.2.2. Fase 2: Selección de Características

Cada secuencia de acción contiene diferente número de imágenes y para cada imagen se tienen las posiciones  $(x, y, z)$  de las veinte articulaciones del esqueleto, con las ecuaciones (7), (8), (9) y (10) se obtuvo un conjunto de 75 características. Las acciones realizadas por los sujetos uno, tres, cinco, siete y nueve se utilizaron en la fase de entrenamiento y las realizadas por las personas dos, cuatro, seis, ocho y diez en la fase de pruebas.

Con el algoritmo RFD-SD y con PCA se seleccionaron características. En la Tabla 26 se listan las veinte acciones y la cantidad de características seleccionadas para cada acción. En las columnas dos, tres y cuatro son mostradas la cantidad de características seleccionadas por el algoritmo RFD-SD para los siguientes casos: sin filtro, con filtro de Kalman y con filtro de Savitsky-Golay, respectivamente. En las columnas cinco, seis y siete son mostradas la cantidad de características seleccionadas por PCA para los siguientes casos: sin filtro, con filtro de Kalman y con filtro de Savitsky-Golay, respectivamente.



**Tabla 26 Cantidad de características seleccionadas para cada acción**

Acciones	RFD-SD			PCA		
	Sin Filtro	Filtro Kalman	Filtro Savitsky-Golay	Sin Filtro	Filtro Kalman	Filtro Savitsky-Golay
Ondular el brazo en alto	6	6	6	68	68	68
Ondular el brazo horizontalmente	14	14	14	71	71	71
Golpear con un martillo	4	5	11	69	72	74
Atrapar con la mano	8	4	9	70	73	75
Golpe hacia adelante	10	10	10	71	71	71
Lanzar alto	25	25	25	70	70	70
Dibujar una cruz	15	15	15	70	70	70
Dibujar una /	11	11	11	69	69	69
Dibujar un círculo	10	10	10	68	68	68
Aplaudir	4	4	4	71	71	71
Ondular las dos manos	20	21	23	70	68	73
Boxear	26	26	26	70	70	70
Flexionarse hacia el piso	1	8	10	72	71	74
Patada hacia el frente	21	14	30	73	74	75
Patada lateral	11	11	11	71	71	71
Trotar	20	20	20	73	73	73
Columpio de tenis	9	9	9	72	72	72
Servicio de tenis	32	26	22	72	70	74
Columpio de golf	21	21	21	69	69	69

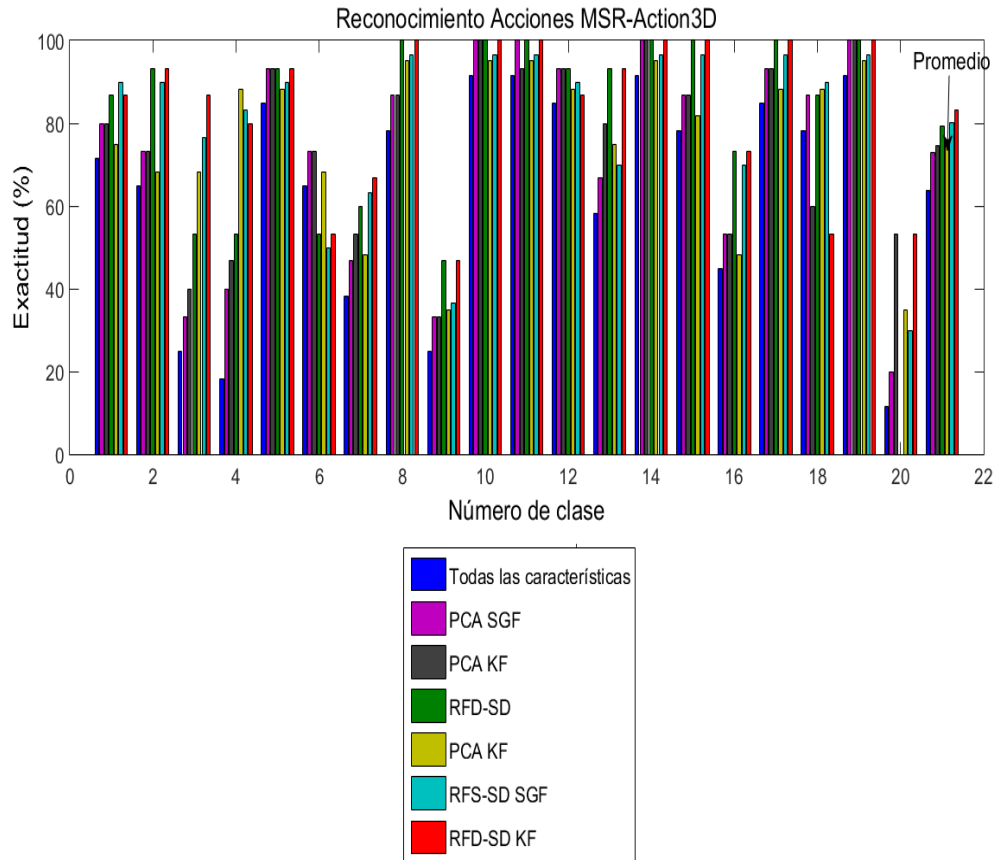
Recoger y tirar	5	7	7	71	61	75
-----------------	---	---	---	----	----	----

Se calculó la exactitud del reconocimiento de acciones cuando se utilizaron todas las características y los conjuntos de características seleccionados por el algoritmo RFD-SD y PCA sin la fase de filtrado. Los resultados obtenidos son 63.8%, 79.3% y 73%, respectivamente. Los resultados se muestran en la Fig. 47, los últimos dos superan el resultado reportado en (Climent-Pérez, Chaaaraoui, Padilla-López, & Flórez-Revuelta, 2012) con una exactitud de 71.1%.

Adicionalmente se calculó la exactitud del reconocimiento de acciones cuando se utilizaron los conjuntos de características seleccionados por el algoritmo RFD-SD y PCA después de realizar la fase de filtrado. En la Tabla 26 se observa que en la acción recoger y tirar, la cantidad de características seleccionadas con el algoritmo RDF-SD se incrementó a siete. Esto se puede explicar, porque inicialmente las secuencias presentaron deformaciones y cuando se minimizaron, entonces otras articulaciones que antes se interpretaban como ruido después del filtrado aportaron información para describir las acciones.

Cuando se seleccionaron características con el algoritmo RFD-SD después de usar el filtro de Kalman se incrementó la exactitud de reconocimiento a 83% y después de utilizar el filtro de Savitzky-Golay se incrementó a 80.2%. Por otro lado, cuando se hace la selección de características con PCA después de usar el filtro de Kalman incrementó la exactitud del reconocimiento a 76% y después de utilizar el filtro de Savitzky-Golay se incrementó a 74.7%. Los resultados se muestran gráficamente en la Fig. 47.

Los resultados de este experimento sugieren que cuando se seleccionaron características después de utilizar el filtro de Kalman la exactitud del reconocimiento se incrementó de un 63.8% hasta un 83%.



**FIGURA 47.** Se muestra la exactitud de reconocimiento en cada clase y la exactitud promedio con las veinte acciones de la base MSR-Action3D, cuando se utilizaron todas las características, características seleccionadas y la fase de filtrado.

### 4.2.3. Discusión de los Resultados

Las acciones son secuencias de imágenes, en los experimentos se calculó un conjunto de características para cada imagen, por esta razón en esta tesis las acciones se representaron como series de tiempo multi-característica.

En el experimento 1 se seleccionaron características con el algoritmo RFD-SD y PCA. Con PCA se obtuvo una reducción de características del 20% y con el algoritmo RFD-SD se obtuvo una reducción del 88%. Por esta razón sólo se evaluó el clasificador con las características seleccionadas con el algoritmo RFD-SD. En el experimento 2 además de seleccionar características se evaluaron dos filtros para minimizar las deformaciones en las posiciones de las articulaciones y se mejoró aún más la exactitud de la clasificación. En la Tabla 27 se muestran los resultados de los experimentos. Cuando se utilizaron las características seleccionadas con el algoritmo RFD-SD se obtuvieron mejores resultados comparado con los resultados reportados en el estado del arte.

**Tabla 27 Resultados de los experimentos de reconocimiento de acciones**

	<b>Base de Datos</b>	<b>Reconocimiento obtenido en esta tesis</b>	<b>Reconocimiento reportado en el estado el arte</b>
Experimento 1	UTD-MHAD	76.6% utilizando características seleccionadas con el algoritmo RFD-SD  83.8% utilizando características seleccionadas con el algoritmo RFD-SD y en algunas clases se hizo la selección de características utilizando el AG	66.1% utilizando sólo información visual (Chen, Jafari, & Kehtarnavaz, 2015)
Experimento 1	KARD	94.2% utilizando características seleccionadas con el algoritmo RFD-SD	88.3% (Gaglio, Re, & Morana, 2015)
Experimento 2	MSR-Action3D	83% utilizando el filtro de Kalman y características seleccionadas con el algoritmo RFD-SD	71.1% en (Climent-Pérez, Chaaraoui, Padilla-López, & Flórez-Revuelta, 2012)

## Capítulo 5: Conclusiones y Trabajo Futuro

En la mayoría de los trabajos revisados de reconocimiento de acciones y caídas cada autor propone un conjunto de características. Sólo en algunos utilizan algoritmos de selección de características. En (Charaoui, Padilla-López, Climent-Pérez, & Flórez-Revuelta, 2014) reconocen las acciones de la base MSR-Action3D mediante las veinte posiciones  $(x, y, z)$  de las articulaciones del esqueleto y con un algoritmo evolutivo seleccionan características, sin embargo las características propuestas son normalizadas para hacerlas invariantes a traslación y rotación. En (Ding & Chang, 2016) proponen un conjunto con 84 características: veinte son posiciones de articulaciones del esqueleto y 64 son ángulos formados entre las articulaciones de los brazos y de las piernas; además proponen un algoritmo de selección de características con el cual seleccionaron ochenta características para reconocer cada clase.

De acuerdo con los trabajos presentados en el capítulo dos y los mencionados en el párrafo anterior, en la minoría de ellos se evalúa si seleccionando características se mejora la exactitud del reconocimiento y en los trabajos que utilizan algoritmos de selección de características lo aplican al reconocimiento de acciones. En esta tesis se seleccionaron características para mejorar el reconocimiento de caídas y acciones.

En esta tesis se obtuvo la mejor exactitud de reconocimiento de las posturas cuando se seleccionaron las características con un AG y de las acciones (series de tiempos con múltiples características) cuando se seleccionaron las características con el algoritmo RFD-SD. Adicionalmente cuando se corrigieron las posiciones de las articulaciones mediante el filtro de Kalman se mejoró aún más la exactitud del reconocimiento de las acciones. En las secciones 4.1.5 y 4.2.3 se presenta una discusión de los resultados obtenidos en los experimentos, los principales hallazgos se resumen a continuación:

1. En todos los experimentos que utilizaron el conjunto de características seleccionadas con el AG se obtuvo una mejor exactitud de reconocimiento en comparación con la exactitud de reconocimiento obtenida cuando se utilizaron las características seleccionadas por PCA o cuando se utilizaron todas las características.
2. El algoritmo RFD-SD propuesto en esta tesis reduce la dimensión del conjunto de características seleccionadas en un mayor porcentaje al reportado en (Ding & Chang, 2016) mediante su algoritmo propuesto.
3. En todos los experimentos en los que se utilizaron las características seleccionadas con el algoritmo RFD-SD se obtuvo una mejor exactitud de reconocimiento en comparación con los resultados reportados en el estado del arte (Tabla 27).

Como trabajo futuro se propone reconocer caídas y de acciones de la vida diaria utilizando un robot móvil como el que se encuentra en el Centro de Investigación en Inteligencia Artificial de la Universidad Veracruzana, para continuar desarrollando trabajos de investigación en interacción humano-computadora.

## Bibliografía

Adhikari, K., Bouchachia, H., & Nait-Charif, H. (2017). Activity recognition for indoor fall detection using convolutional neural network. (IEEE, Ed.) *Machine Vision Applications (MVA), 2017 Fifteenth IAPR International Conference on* , 81-84.

Aggarwal, J. K., & Xia, L. (2014). Human activity recognition from 3d data: A review. (Elsevier, Ed.) *Pattern Recognition Letters* , 48, 70-80.

Alaoui, A. Y., El Hassouny, A., Thami, R. O., & Tairi, H. (2017). Video based human fall detection using von Mises distribution of motion vector. (IEEE, Ed.) *2017 Intelligent Systems and Computer Vision (ISCV)* , 1-5.

Andrew R. Webb, K. D. (2011). *Statistical Pattern Recognition*. John Wiley & Sons, Ltd.

Ayumi, V., & Fanany, M. I. (2016). A comparison of SVM and RVM for human action recognition. *Internetworking Indonesia Journal* , 8 (1), 29-33.

Baldewijns, G., Debar, G., Mertes, G., Croonenborghs, T., & Vanrumste, B. (2017). Improving the accuracy of existing camera based fall detection. (IEEE, Ed.) *Engineering in Medicine and Biology Society (EMBC), 2017 39th Annual International Conference of the IEEE* , 2667-2671.

Baptista-Ríos, M., Martínez-García, C., Losada-Gutiérrez, C., & Marrión-Romera, M. (2016). Human activity monitoring for falling detection. A realistic framework. (IEEE, Ed.) *Indoor Positioning and Indoor Navigation (IPIN), 2016 International Conference on* .

Bennasar, M., Hicks, Y., & Setchi, R. (2015). Feature selection using joint mutual information maximisation. *Expert Systems with Applications* , 42 (22), 8520-8532.

Bian, Z.-P., Hou, J., Chau, L.-P., & Magnenat-Thalmann, N. (2015). Fall detection based on body part tracking using a depth camera. (IEEE, Ed.) *IEEE journal of biomedical and health informatics* , 19 (2), 430-439.

Bloom, V., Makris, D., & Argyriou, V. (2012). G3d: A gaming action dataset and real time action recognition evaluation framework. (IEEE, Ed.) *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on* , 7-12.

Bouhamed, S. A., Kallel, I. K., Masmoudi, D. S., & Solaiman, B. (2015). Feature selection in possibilistic modeling. (Elsevier, Ed.) *Pattern Recognition* , 48 (11), 3627-3640.

- Brulin, D., Benezeth, Y., & Courtial, E. (2012). Posture recognition based on fuzzy logic for home monitoring of the elderly. (IEEE, Ed.) *IEEE transactions on information technology in biomedicin* , 16 (5), 974-982.
- Carmona, J. M., & Climent, J. (2012). A performance evaluation of HMM and DTW for gesture recognition. (Springer, Ed.) *Iberoamerican Congress on Pattern Recognition* , 236-243.
- Chaararoui, A. A., & Flórez-Revuelta, F. (2014). Optimizing human action recognition based on a cooperative coevolutionary algorithm. (Elsevier, Ed.) *Engineering Applications of Artificial Intelligence* , 116-125.
- Chaararoui, A. A., Climent-Pérez, P., & Flórez-Revuelta, F. (2012). An efficient approach for multi-view human action recognition based on bag-of-key-poses. (Springer, Ed.) *International Workshop on Human Behavior Understanding* , 29-40.
- Chaararoui, A. A., Climent-Pérez, P., & Flórez-Revuelta, F. (2013). Silhouette-based human action recognition using sequences of key poses. (Elsevier, Ed.) *Pattern Recognition Letters* , 1799-1807.
- Chaararoui, A. A., Padilla-López, J. R., Climent-Pérez, P., & Flórez-Revuelta, F. (2014). Evolutionary joint selection to improve human action recognition with RGB-D devices. (Elsevier, Ed.) *Expert systems with applications* , 41 (3), 786-794.
- Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. (Elsevier, Ed.) *Computers & Electrical Engineering* , 40 (1), 16-28.
- Chaquet, J. M., Carmona, E. J., & Fernández-Caballero, A. (2013). A survey of video datasets for human action and activity recognition. (Elsevier, Ed.) *Computer Vision and Image Understanding* , 117 (6), 633-659.
- Chen, C., Jafari, R., & Kehtarnavaz, N. (2015). Utd-mhad: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor. (IEEE, Ed.) *Image Processing (ICIP), 2015 IEEE International Conference on* , 168-172.
- Chen, L., Wei, H., & Ferryman, J. (2013). A survey of human motion analysis using depth imagery. (Elsevier, Ed.) *Pattern Recognition Letters* , 34 (15), 1995-2006.
- Chen, X., & Koskela, M. (2015). Skeleton-based action recognition with extreme learning machines. (Elsevier, Ed.) *Neurocomputing* , 149, 387-396.
- Christopher, M. B. (2016). *PATTERN RECOGNITION AND MACHINE LEARNING*. York, Springer-Verlag New.
- Climent-Pérez, P., Chaararoui, A. A., Padilla-López, J. R., & Flórez-Revuelta, F. (2012). Mexican International Conference on Artificial Intelligence. (Springer, Ed.) 163-174.



- Coello, C. A. (2004). Introducción a la computación evolutiva (Notas de curso). *CINVESTAV-IPN, Departamento de Ingeniería Eléctrica, Sección de Computación. México, DF* .
- Ding, J., & Chang, C.-W. (2016). Feature design scheme for Kinect-based DTW human gesture recognition. (Springer, Ed.) *Multimedia Tools and Applications* , 75 (16), 9669-9684.
- Du, Y., Wang, W., & Wang, L. (2015). Hierarchical recurrent neural network for skeleton based action recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition* , 1110-1118.
- Fanello, S. R., Gori, I., Metta, G., & Odone, F. (2013). Keep it simple and sparse: Real-time action recognition. (J. org, Ed.) *The Journal of Machine Learning Research* , 2617-2640.
- Flusser, J., Zitova, B., & Suk, T. (2009). *Moments and moment invariants in pattern recognition*. Sons, John Wiley \&.
- Gaglio, S., Re, G. L., & Morana, M. (2015). Human activity recognition process using 3-D posture data. (IEEE, Ed.) *IEEE Transactions on Human-Machine Systems* , 45 (5), 586-597.
- Gasparrini, S., Cippitelli, E., Gambi, E., Spinsante, S., Wahslén, J., Orhan, I., y otros. (2016). Proposal and experimental evaluation of fall detection solution based on wearable and depth data fusion. *ICT innovations 2015* , 99-108.
- Ghamisi, P., Couceiro, M. S., & Benediktsson, J. A. (2015). A novel feature selection approach based on FODPSO and SVM. *IEEE Transactions on Geoscience and Remote Sensing* , 53 (5), 2935-2947.
- Goldberg, D. E. (2013). *The design of innovation: Lessons from and for competent genetic algorithms* (Vol. 7). Springer Science \& Business Media.
- Gorry, P. A. (1990). General least-squares smoothing and differentiation by the convolution (Savitzky-Golay) method. (A. Publications, Ed.) *Analytical Chemistry* , 62 (6), 570-573.
- Hernandez-Mendez, S., Maldonado-Mendez, C., Marin-Hernandez, A., & Rios-Figueroa, H. V. (Feb de 2017). Detecting falling people by autonomous service robots: A ROS module integration approach. (IEEE, Ed.) *2017 International Conference on Electronics, Communications and Computers (CONIELECOMP)* , 1-7.
- Hira, Z. M., & Gillies, D. F. (2015). A review of feature selection and feature extraction methods applied on microarray data. (Hindawi, Ed.) *Advances in bioinformatics* .
- Huber, M., Seitz, A., Leiser, M., & Sternad, D. (2015). Validity and reliability of Kinect skeleton for measuring shoulder joint angles: a feasibility study. (Elsevier, Ed.) *Physiotherapy* , 101 (4), 389-393.

- Imtiaz, H., Mahbub, U., Schaefer, G., & Ahad, M. A. (2013). A Multi-resolution Action Recognition Algorithm Using Wavelet Domain Features. (IEEE, Ed.) *Pattern Recognition (ACPR), 2013 2nd IAPR Asian Conference on* , 537-541.
- Izakian, H., Pedrycz, W., & Jamal, I. (2015). Fuzzy clustering of time series data using dynamic time warping distance. (Elsevier, Ed.) *Engineering Applications of Artificial Intelligence* , 235-244.
- Jackson, J. E. (2005). A user's guide to principal components. (J. W. Sons, Ed.) 587.
- Jain, A., & Zongker, D. (1997). Feature selection: Evaluation, application, and small sample performance. (IEEE, Ed.) *IEEE transactions on pattern analysis and machine intelligence* , 19 (2), 153-158.
- Jiang, X., Zhong, F., Peng, Q., & Qin, X. (2016). Action recognition based on global optimal similarity measuring. (Springer, Ed.) *Multimedia Tools and Applications* , 75 (18), 11019-11036.
- Jolliffe, I. T. (1986). *Principal component analysis*. (Springer, Ed.)
- Jovi'c, A., Brkic, K., & Bogunovic, N. (2015). A review of feature selection methods with applications. (IEEE, Ed.) *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2015 38th International Convention on* , 1200-1205.
- Jr, L., & J, J. (2015). Context aware 3D gesture recognition for games and virtual reality. (ACM, Ed.) *ACM SIGGRAPH 2015 Courses* , 10.
- Kaneko, T., & Cao, M. (2016). Human fall detection using CHLAC features with skeletal image sequences. (IEEE, Ed.) *Robotics and Intelligent Sensors (IRIS), 2016 IEEE International Symposium on* , 163-167.
- Kao, H.-C., Hung, J.-C., & Huang, C.-P. (2017). GA-SVM applied to the fall detection system. (IEEE, Ed.) *Applied System Innovation (ICASI), 2017 International Conference on* , 436-439.
- Kawatsu, C., Li, J., & Chung, C.-J. (2013). Development of a fall detection system with Microsoft Kinect. (Springer, Ed.) *Robot Intelligence Technology and Applications 2012* , 623-630.
- Kepski, M., & Kwolek, B. (2017). Event-driven system for fall detection using body-worn accelerometer and depth sensor. (IET, Ed.) *IET Computer Vision* , 48-58.
- Kim, S. J., Kim, S. W., Sandhan, T., & Choi, J. Y. (2014). View invariant action recognition using generalized 4D features. (Elsevier, Ed.) *Pattern Recognition Letters* , 49, 40-47.
- Kostopoulos, P., Kyritsis, A. I., Deriaz, M., & Konstantas, D. (2016). F2D: a location aware fall detection system tested with real data from daily life of elderly people. (Elsevier, Ed.) *Procedia computer science* , 98, 212--219.

Kwolek, B., & Kepski, M. (2014). Human fall detection on embedded platform using depth maps and wireless accelerometer. (Elsevier, Ed.) *Computer methods and programs in biomedicine* , 117 (3), 489-501.

Kwolek, B., & Kepski, M. (2015). Improving fall detection by the use of depth sensor and accelerometer. *Neurocomputing* , 168, 637-645.

LaViola, J. J. (2003). A comparison of unscented and extended Kalman filtering for estimating quaternion motion. (IEEE, Ed.) *American Control Conference, 2003. Proceedings of the 2003* , 2435-2440.

Li, C., Hou, Y., Wang, P., & Li, W. (2017). Joint distance maps based action recognition with convolutional neural networks. (IEEE, Ed.) *IEEE Signal Processing Letters* , 24 (5), 624-628.

Li, L., & Prakash, B. A. (2011). Time series clustering: Complex is simpler! *ICML* , 185-192.

Li, W., Zhang, Z., & Liu, Z. (2010). Action recognition based on a bag of 3d points. (IEEE, Ed.) *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on* , 9-14.

Li, Y., & Liu, T. (2010). Study on classification model based on relevance vector machine with genetic algorithm. (IEEE, Ed.) *Information and Financial Engineering (ICIFE), 2010 2nd IEEE International Conference on* , 538-541.

Li, Z., Shang, Z., Qu, B.-Y., & Liang, J. J. (2014). Feature selection based on manifold-learning with dynamic constraint handling differential evolution. (IEEE, Ed.) 332-337.

Liang, B., & Zheng, L. (2015). A survey on human action recognition using depth sensors. (IEEE, Ed.) *Digital Image Computing: Techniques and Applications (DICTA), 2015 International Conference on* , 1-8.

Limón, M. G., Escalante, H. J., Morales, E., & Pineda, L. V. (2015). Class-specific feature generation for 1NN through genetic programming. (IEEE, Ed.) *Power, Electronics and Computing (ROPEC), 2015 IEEE International Autumn Meeting on* , 1-6.

Liu, L., Shao, L., Li, X., & Lu, K. (2016). Learning spatio-temporal representations for action recognition: A genetic programming approach. (IEEE, Ed.) *IEEE transactions on cybernetics* , 46 (1), 158-170.

Liu, M., & Zhang, D. (2016). Feature selection with effective distance. (Elsevier, Ed.) *Neurocomputing* , 100-109.

Liu, Y., Wang, N., Lv, C., & Cui, J. (2015). Human body fall detection based on the Kinect sensor. (IEEE, Ed.) *Image and Signal Processing (CISP), 2015 8th International Congress on* , 367-371.

- Loumpionias, K., Vretos, N., Daras, P., & Tsaklidis, G. (2016). USING KALMAN FILTER AND TOBIT KALMAN FILTER IN ORDER TO IMPROVE THE MOTION RECORDED BY KINECT SENSOR II. *Proceedings of the 29th Panhellenic Statistics Conference* .
- Ma, X., Wang, H., Xue, B., Zhou, M., Ji, B., & Li, Y. (2014). Depth-based human fall detection via shape features and improved extreme learning machine. (IEEE, Ed.) *IEEE journal of biomedical and health informatics* , 18 (6), 1915-1922.
- Maldonado, C., Ríos, H., Mezura-Montes, E., & Marin, A. (Feb de 2016). Feature selection to detect fallen pose using depth images. (IEEE, Ed.) *2016 International Conference on Electronics, Communications and Computers (CONIELECOMP)* , 94-100.
- Maldonado, C., Rios-Figueroa, H. V., & Marin-Hernandez, A. (Nov de 2016). Improving action recognition by selection of features. (IEEE, Ed.) *2016 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)* , 1-6.
- Maldonado-Mendez, {., Solis, A. L., Rios-Figueroa, H. V., & Marin-Hernandez, A. (Nov de 2017). Human fallen pose detection by using feature selection and a generative model. (IEEE, Ed.) *2017 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)* , 1-6.
- Maybeck, P. S. (1982). Stochastic models, estimation, and control. *Academic press* , 3.
- Mehmood, R. M., Du, R., & Lee, H. J. (2017). Optimal Feature Selection and Deep Learning Ensembles Method for Emotion Recognition From Human Brain EEG Sensors. (IEEE, Ed.) *IEEE Access* (5), 14797-14806.
- Mejia, M. E., Montalvo, J. R., Mena, F. M., & Ramirez, F. A. (2016). Intelligent Platform for Non Intrusive Assistance of Elderly People. (IEEE, Ed.) *IEEE Latin America Transactions* , 14 (5), 2433-2439.
- Mendez, C. G., Mendez, S. H., Solis, A. L., Figueroa, H. V., & Hernandez, A. M. (Nov de 2017). The Effects of Using a Noise Filter and Feature Selection in Action Recognition: An Empirical Study. (IEEE, Ed.) *2017 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)* , 43-48.
- Merrouche, F., & Baha, N. (2016). Depth camera based fall detection using human shape and movement. (IEEE, Ed.) *Signal and Image Processing (ICSIP), IEEE International Conference on* , 586-590.
- Miranda, L., Vieira, T., Martinez, D., Lewiner, T., Vieira, A. W., & Campos, M. F. (2012). Real-time gesture recognition from depth data through key poses learning and decision forests. (IEEE, Ed.) *Graphics, patterns and images (SIBGRAPI), 2012 25th SIBGRAPI conference on* , 268-275.

- Mousse, M. A., Motamed, C., & Ezin, E. C. (2017). Percentage of human-occupied areas for fall detection from two views. (Springer, Ed.) *The Visual Computer* , 33 (12), 1529-1540.
- Muller, M. (2007). Dynamic time warping. (Springer, Ed.) *Information retrieval for music and motion* , 69-84.
- Nakamura, T., Taki, K., Nomiya, H., Seki, K., & Uehara, K. (2013). A shape-based similarity measure for time series data with ensemble learning. (Springer, Ed.) *Pattern Analysis and Applications* , 16 (4), 535-548.
- Nazarpour, A., & Adibi, P. (2015). Two-stage multiple kernel learning for supervised dimensionality reduction. (Elsevier, Ed.) *Pattern Recognition* , 48 (5), 1854-1862.
- Nguyen, V. D., Le, M. T., Do, A. D., Duong, H. H., Thai, T. D., & Tran, D. H. (2014). An efficient camera-based surveillance for fall detection of elderly people. (IEEE, Ed.) *Industrial Electronics and Applications (ICIEA), 2014 IEEE 9th Conference on* , 994-997.
- Ni, B., Wang, G., & Moulin, P. (2013). Rgbd-hudaact: A color-depth video database for human daily activity recognition. (Springer, Ed.) *Consumer Depth Cameras for Computer Vision* , 193-208.
- Organization, W. H. (2015). *World report on ageing and health*.
- Otanasap, N., & Boonbrahm, P. (2017). Pre-impact fall detection approach using dynamic threshold based and center of gravity in multiple Kinect viewpoints. (IEEE, Ed.) *Computer Science and Software Engineering (JCSSE), 2017 14th International Joint Conference on* , 1-6.
- Padilla-López, J. R., Charaoui, A. A., & Flórez-Revuelta, F. (2014). A discussion on the validation tests employed to compare human action recognition methods using the MSR action3d dataset. *CoRR* , [abs/14.07.7390](https://arxiv.org/abs/14.07.7390).
- Papadopoulos, G. T., Axenopoulos, A., & Daras, P. (2014). Real-time skeleton-tracking-based human action recognition using kinect data. (Springer, Ed.) *International Conference on Multimedia Modeling* , 473-483.
- Pattamaset, S., Charoenpong, T., Charoenpong, P., & Chianrabutra, C. (2017). Human fall detection by using the body vector. (IEEE, Ed.) *Knowledge and Smart Technology (KST), 2017 9th International Conference on* , 162-165.
- Pravdova, V., Walczak, B., & Massart, D. L. (2002). A comparison of two algorithms for warping of analytical signals. (Elsevier, Ed.) *Analytica Chimica Acta* , 456, 77-92.
- Prince, S. J. (2012). *Computer vision: models, learning, and inference*. Cambridge University Press.

- Raheja, J., Minhas, M., Prashanth, D., Shah, T., & Chaudhary, A. (2015). Robust gesture recognition using Kinect: A comparison between DTW and HMM. (Elsevier, Ed.) *Optik-International Journal for Light and Electron Optics* , 126 (11-12), 1098-1104.
- Ratanamahatana, C. A., & Keogh, E. (2004). Making time-series classification more accurate using learned constraints. (SIAM, Ed.) *Proceedings of the 2004 SIAM International Conference on Data Mining* , 11-22.
- Reyes, M., Dominguez, G., & Escalera, S. (2011). Featureweighting in dynamic timewarping for gesture recognition in depth data. (IEEE, Ed.) *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on* , 1182-1188.
- Richards, J. A., & Richards, J. (1999). *Remote sensing digital image analysis* (Vol. 3). Springer.
- Sannino, G., De Falco, I., & De Pietro, G. (2017). Detection of falling events through windowing and automatic extraction of sets of rules: Preliminary results. (IEEE, Ed.) *Networking, Sensing and Control (ICNSC), 2017 IEEE 14th International Conference on* , 661-666.
- Sempena, S., Maulidevi, N. U., & Aryan, P. R. (2011). Human action recognition using dynamic time warping. (IEEE, Ed.) *Electrical Engineering and Informatics (ICEEI), 2011 International Conference on* , 1-5.
- Senin, P. (2008). Dynamic time warping algorithm review. (Citeseer, Ed.) *Information and Computer Science Department University of Hawaii at Manoa Honolulu, USA* , 855, 1-23.
- Sheng, B., Yang, W., & Sun, C. (2015). Action recognition using direction-dependent feature pairs and non-negative low rank sparse model. (Elsevier, Ed.) *Neurocomputing* , 158, 73-80.
- SHMILOVICI, A., & BEN-SHIMON, D. (2013). A feature selection strategy for the relevance vector machine. *Recent Advances in Knowledge Engineering and Systems Science* , 73-78.
- Siedlecki, W., & Sklansky, J. (1993). A note on genetic algorithms for large-scale feature selection. (W. Scientific, Ed.) *Handbook Of Pattern Recognition And Computer Vision* , 88-107.
- Soewittoa, B., Irwana, Antonyová, A., & Gunawana, F. E. (2015). Fall Detection Algorithm to Generate Security Alert. (Elsevier, Ed.) *Procedia Computer Science* , 59, 350-356.
- Stone, E. E., & Skubic, M. (2015). Fall detection in homes of older adults using the Microsoft Kinect. (IEEE, Ed.) *IEEE journal of biomedical and health informatics* , 19 (1), 290-301.
- Sung, J., Ponce, C., Selman, B., & Saxena, A. (2011). Human Activity Detection from RGBD Images. *Plan, activity, and intent recognition* , 64.

- Tang, J., Alelyani, S., & Liu, H. (2014). Feature selection for classification: A review. (C. Press, Ed.) *Data Classification: Algorithms and Applications* .
- Tipping, M. E. (2001). Sparse Bayesian learning and the relevance vector machine. *Journal of machine learning research* , 1 (Jun), 211-244.
- Uysal, A. K., & Murphey, Y. L. (2017). Sentiment classification: Feature selection based approaches versus deep learning. (IEEE, Ed.) *Computer and Information Technology (CIT), 2017 IEEE International Conference on* , 23-30.
- Valcourt, L., De La Hoz, Y., & Labrador, M. (2016). Smartphone-based human fall detection system. (IEEE, Ed.) *IEEE Latin America Transactions* , 14 (2), 1011-1017.
- Verma, A., Merchant, R. A., Seetharaman, S., & Yu, H. (2016). An intelligent technique for posture and fall detection using multiscale entropy analysis and fuzzy logic. (IEEE, Ed.) *Region 10 Conference (TENCON), 2016 IEEE* , 2479-2482.
- Vieira, S. M., Sousa, J. M., & and Kaymak, U. (2012). Fuzzy criteria for feature selection. *Fuzzy Sets and Systems* , 189 (1), 1-18.
- Wan, E. A., & Van Der Merwe, R. (2000). The unscented Kalman filter for nonlinear estimation. (IEEE, Ed.) *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000* , 153-158.
- Wang, R.-d., Zhang, Y.-l., Dong, L.-p., Lu, J.-w., Zhang, Z.-q., & He, X. (2015). Fall detection algorithm for the elderly based on human characteristic matrix and SVM. (IEEE, Ed.) *Control, Automation and Systems (ICCAS), 2015 15th International Conference on* , 1190-1195.
- Wang, X., Liu, H., & Liu, M. (2016). A novel multi-cue integration system for efficient human fall detection. (IEEE, Ed.) *Robotics and Biomimetics (ROBIO), 2016 IEEE International Conference on* , 1319-1324.
- Xia, L., & Aggarwal, J. (2013). Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. (IEEE, Ed.) *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on* , 2834-2841.
- Xia, L., Chen, C.-C., & Aggarwal, J. (2012). View invariant human action recognition using histograms of 3d joints. (IEEE, Ed.) *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on* , 20-27.
- Yan, H., & Yang, J. (2015). Sparse discriminative feature selection. (Elsevier, Ed.) *Pattern Recognition* , 48 (5), 1827-1835.

Yang, S., Wang, W., Liu, C., Deng, W., & Hedrick, J. K. (2017). Feature analysis and selection for training an end-to-end autonomous vehicle controller using deep learning approach. En *Intelligent Vehicles Symposium (IV), 2017 IEEE* (págs. 1033-1038). IEEE.

Yang, X., & Tian, Y. L. (2012). Eigenjoints-based action recognition using naive-bayes-nearest-neighbor. (IEEE, Ed.) *Computer vision and pattern recognition workshops (CVPRW), 2012 IEEE computer society conference on* , 14-19.

Yang, X., Zhang, C., & Tian, Y. (2012). Recognizing actions using depth motion maps-based histograms of oriented gradients. (ACM, Ed.) *Proceedings of the 20th ACM international conference on Multimedia* , 1057-1060.

Yu, X. (Julio de 2008). Approaches and principles of fall detection for elderly and patient. *HealthCom 2008 - 10th International Conference on e-health Networking, Applications and Services* , 42-47.

Yusta, S. C. (2009). Different metaheuristic strategies to solve the feature selection problem. (Elsevier, Ed.) *Pattern Recognition Letters* , 30 (5), 525-534.

Zare, H., & Niazi, M. (2016). Relevant based structure learning for feature selection. (Elsevier, Ed.) *Engineering Applications of Artificial Intelligence* , 55, 93-102.

Zhang, C., & Tian, Y. (2012). RGB-D camera-based daily living activity recognition. *Journal of Computer Vision and Image Processing* , 2 (4), 12.

Zhang, H., & Parker, L. E. (2011). 4-dimensional local spatio-temporal features for human activity recognition. (IEEE, Ed.) *Intelligent robots and systems (IROS), 2011 IEEE/RSJ international conference on* , 2044-2049.

Zhao, Y., Liu, Z., Yang, L., & Cheng, H. (2012). Combing rgb and depth map features for human activity recognition. (IEEE, Ed.) *Signal & Information Processing Association Annual Summit and Conference (APSIPA ASC), 2012 Asia-Pacific* , 1-4.

Zhu, Y., Chen, W., & Guo, G. (2013). Fusing spatiotemporal features and joints for 3d action recognition. (IEEE, Ed.) *Computer vision and pattern recognition workshops (CVPRW), 2013 IEEE conference on* , 486-491.

Zou, Q., Ni, L., Zhang, T., & Wang, Q. (2015). Deep learning based feature selection for remote sensing scene classification. (IEEE, Ed.) *IEEE Geoscience and Remote Sensing Letters* , 12 (11), 2321-2325.