



Tecnológico Nacional de México

Centro Nacional de Investigación y
Desarrollo Tecnológico

Tesis de Maestría

Detección automática de eventos de riesgo en el
manejo de un automóvil utilizando un teléfono
inteligente y la plataforma FIWARE

presentada por

Ing. León Alberne Torres Restrepo

como requisito para la obtención del grado de

**Maestro en Ciencias de
la Computación**

Director de tesis

Dra. Alicia Martínez Rebollar

Codirector de tesis

Dr. Fabio Alberto Vargas Agudelo

Cuernavaca, Morelos, México. Junio de 2018.



SEP
SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MÉXICO

Centro Nacional de Investigación y Desarrollo Tecnológico

"2019, Año del Caudillo del Sur, Emiliano Zapata"

Cuernavaca, Morelos a 25 de enero del 2019
OFICIO No. DCC/004/2019

Asunto: **Aceptación de documento de tesis**

DR. GERARDO V. GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO
PRESENTE

Por este conducto, los integrantes de Comité Tutorial del **Ing. León Alberne Torres Restrepo**, con número de control M16CE090, de la Maestría en Ciencias de la Computación, le informamos que hemos revisado el trabajo de tesis profesional titulado "**Detección automática de eventos de riesgo en el manejo de un automóvil utilizando un teléfono inteligente y la plataforma FIWARE**" y hemos encontrado que se han realizado todas las correcciones y observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

DIRECTORA DE TESIS

Dra. Alicia Martínez Rebollar
Doctora en Informática
7399055

CO-DIRECTOR DE TESIS

Dr. Fabio Alberto Vargas Agudelo
Doctor en Ingeniería de Sistemas e
Informática

REVISOR 1

Dr. Joaquín Pérez Ortega
Doctor en Ciencias Computacionales
4795984

REVISOR 2

Dr. Máximo López Sánchez
Doctor en Ciencias de la Computación
7498547

C.p. M.T.I. María Elena Gómez Torres - Jefa del Departamento de Servicios Escolares.
Estudiante
Expediente

NACS/lmz



SEP
SECRETARÍA DE
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MEXICO

Centro Nacional de Investigación y Desarrollo Tecnológico

"2019, Año del Caudillo del Sur, Emiliano Zapata"

Cuernavaca, Mor., 5 de junio de 2019
OFICIO No. SAC/213/2019

Asunto: Autorización de impresión de tesis

ING. LEÓN ALBERNE TORRES RESTREPO
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS
DE LA COMPUTACIÓN
PRESENTE

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado "Detección automática de eventos de riesgo en el manejo de un automóvil utilizando un teléfono inteligente y la plataforma FIWARE", ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE
Excelencia en Educación Tecnológica®
"Conocimiento y tecnología al servicio de México"

DR. GERARDO VICENTE GUERRERO RAMÍREZ
SUBDIRECTOR ACADÉMICO

C.p. M.T.I. Guadalupe Garrido Rivera.- Jefa del Departamento de Servicios Escolares.
Expediente

GVGR/mcr



SEP TecNM
CENTRO NACIONAL
DE INVESTIGACIÓN
Y DESARROLLO
TECNOLÓGICO
SUBDIRECCIÓN
ACADÉMICA

cenidet[®]
Centro Nacional de Investigación
y Desarrollo Tecnológico

Interior Internado Palmira S/N, Col. Palmira, C. P. 62490, Cuernavaca, Morelos.
Tel. (01) 777 3 62 77 70, ext. 4106, e-mail: dir_cenidet@tecnm.mx

www.tecnm.mx | www.cenidet.edu.mx



A la “Flaca” mi esposa, Melissa y Mateo mis hijos, quienes son un gran soporte en mi presente. A Olga Elena Restrepo y Leonel de Jesús Torres mis padres, quienes ayudaron a formar lo que ahora es mi pasado e hicieron posible mi presente. A mis hermanos y seres queridos quienes siempre han creído en mí. A ustedes que Dios los bendiga siempre y les depare un futuro próspero lleno de felicidad y tranquilidad.

Agradecimientos

A quien hace todo posible y nada se mueve si así no lo quiere. Al Dios de los cielos que me sostuvo y puso las fuerzas para no abandonar cuando tuve todo en mi contra. A Jehová, quien enderezó mis caminos cuando todo estaba perdido, abrió puertas de bendición cuando las dificultades tocaron a mi puerta y me ha concedido hasta el más mínimo de mis anhelos. La gloria y la honra le sean por siempre.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el apoyo económico que permitió el desarrollo de esta tesis. además, de permitir que mi familia y yo aprendiéramos de la cultura de México.

Al Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) por la oportunidad de realizar una Maestría en Ciencias de la Computación. Al personal administrativo y académico, porque siempre me trataron con amabilidad y respeto.

Al INFOTEC quien proporcionó todos los recursos tecnológicos necesarios para desplegar los módulos de la plataforma FIWARE y brindó las capacitaciones necesarias para la participación en el proyecto SmartSDK y en especial, al doctor Hugo Estrada Esquivel, quien siempre me tuvo en cuenta para este tipo de capacitaciones.

A los responsables del proyecto SmartSDK por brindarme la oportunidad de hacer parte del equipo de trabajo y permitirme profundizar e interactuar con la plataforma FIWARE. En especial al equipo responsable del escenario SmartSDK Security quien financió la participación en varios eventos de carácter Nacional e Internacional.

A mi esposa quien me ha apoyado siempre, que tuvo el valor para asistir a todas mis presentaciones, aun cuando sabía que vendría una paliza, que tal vez sería mi último día en el plantel y uno más en la lista de mis fracasos. Gracias por no dejarme solo.

A mis hijos que siempre han estado ahí, aún en las dificultades, para traer felicidad a mi vida. Gracias por que me sorprenden siempre, como: la proeza de los dieces en la boleta de calificaciones de mi hija; o el grito de, “gooooool, somos campeones”, después del tiro de penal ejecutado por mí hijo. Gracias por las alegrías dadas en esta etapa de mi vida.

A la doctora Alicia, por todo lo que me enseñó y me permitió aprender. Gracias porque siempre se preocupó por mi bienestar. Gracias a su buena voluntad mis ojos vieron y mis pies tocaron lugares que nunca imaginé, los cuales no estoy seguro si tendré el placer de visitarlos nuevamente. Gracias por darnos la oportunidad a mí y mi familia de conocer un poco de la cultura de México y por no permitir que olvidemos que aún existen personas de buena voluntad.

Al Dr. Fabio Vargas por su acompañamiento en este proceso y por su apoyo incondicional desde que estaba en Colombia. Gracias por abrirme las puertas de su dependencia, en procura de que pudiera continuar con mi formación académica.

Un agradecimiento muy especial a Fernando Cipriano Ramírez, Fer o simplemente el “Compa” quien fue un compañero y amigo incondicional durante todo el período de la maestría, compañero de batallas en el proyecto SmartSDK Security y un gran apoyo en la fase de pruebas.

A los que acompañaron y apoyaron mi proceso formativo desde sus diferentes áreas del saber, ya que sin su ayuda, acompañamiento y asesoramiento hubiese sido muy difícil culminar este proceso de manera satisfactoria.

A todos los que de una u otra forma hicieron posible que obtuviera el presente logro... mi gratitud por siempre.

Resumen

Los accidentes de tránsito se encuentran dentro de las principales causas de muerte a nivel mundial [1], ubicándose por encima de factores como: el suicidio, el VIH y el homicidio. Es así, como estos accidentes y por ende la seguridad vial son una de las principales preocupaciones de las ciudades sostenibles o las ciudades inteligentes (smartcities).

Por otro lado, unido al concepto de ciudad inteligente se encuentra el concepto de internet de las cosas[2]. Este concepto ha dado paso a la existencia de objetos de uso diario equipados con una serie de sensores y algoritmos capaces de detectar automáticamente las características del entorno en que se encuentran[3]. Las capacidades del internet de las cosas han sido aprovechadas por las ciudades en diversas áreas para mejorar la calidad de vida de sus habitantes. Una de estas áreas es la movilidad que se enfrenta al problema de los accidentes de tránsito y en donde se están implementando diferentes servicios, aplicaciones y plataformas tecnológicas para proponer soluciones en la planificación de rutas, el control de tráfico, el reporte de accidentes y anomalías en la vía, etc.

Actualmente, existen varias propuestas de investigación que detectan de manera automática la presencia u ocurrencia de eventos de riesgo generadores de accidentes de tránsito [4], algunas de estas propuestas se apoyan en técnicas de procesamiento de señales y aprendizaje de máquina[5-9]. Sin embargo, El tema de la detección automática de este tipo de eventos de riesgo encierra un grado de complejidad alto; y pese al esfuerzo realizado por la comunidad a través de las propuestas existentes, aún no está cerrado, ni existen soluciones únicas, dado que no se ha alcanzado la madurez en cuanto a las técnicas de detección y a los algoritmos desarrollados para tal fin [6].

En este trabajo de investigación se propone un conjunto de algoritmos de software que permiten la detección automática de eventos de riesgo al momento de conducir un automóvil. Estos algoritmos utilizan la información obtenida de los sensores de un teléfono inteligente, el cual debe encontrarse dentro de un automóvil en marcha. Los eventos de riesgo detectados por estos algoritmos son: velocidad no permitida, desplazamiento en sentido contrario, cambios de velocidad, irregularidades en la vía y giros laterales. Además, estos algoritmos fueron implementados en un prototipo que utiliza la plataforma FIWARE, la cual, es una plataforma abierta para la construcción de aplicaciones de Internet del futuro. La evaluación de este trabajo se llevó a cabo con más de 100 recorridos realizados en diferentes localidades, carreteras y vehículos, de los cuales se obtuvieron más de 50 millones de registros, correspondientes a datos provenientes del acelerómetro, el giroscopio, el magnetómetro y el GPS de un teléfono inteligente.

Las pruebas de la aplicación propuesta dieron los siguientes resultados: velocidad no permitida 93%; desplazamiento en sentido contrario 93%; cambios de velocidad 94%; irregularidades en la vía 99%; y giros laterales 90%. Estos resultados muestran que, sí es posible detectar automáticamente eventos de riesgo al momento de conducir un automóvil, a partir de la utilización de la plataforma FIWARE y los datos recolectados por un teléfono inteligente.

Abstract

Traffic accidents are among the leading causes of death worldwide [1], ranking above factors such as: suicide, HIV and homicide. This is how these accidents and therefore road safety is one of the main concerns of sustainable cities or smart cities.

On the other hand, together with the concept of intelligent city is the concept of the internet of things [2]. This concept has given way to the existence of everyday objects equipped with a series of sensors and algorithms capable of automatically detecting the characteristics of the environment in which they are located [3]. The capabilities of the internet of things have been used by cities in various areas to improve the quality of life of its inhabitants. One of these areas is the mobility that faces the problem of traffic accidents and where different services, applications and technological platforms are being implemented to propose solutions in route planning, traffic control, accident and anomaly reporting on the road, etc.

Currently, there are several research proposals that automatically detect the presence or occurrence of risk events generating traffic accidents [4], some of these proposals are supported by techniques of signal processing and machine learning [5-9]. However, the issue of automatic detection of this type of risk events involves a high degree of complexity; and despite the effort made by the community through the existing proposals, it is not yet closed, nor are there any unique solutions, given that the maturity of the detection techniques and the algorithms developed for this purpose has not been reached [6] .

In this research work we propose a set of software algorithms that allow the automatic detection of risk events when driving a car. These algorithms use the information obtained from the sensors of a smartphone, which must be inside a moving car. The risk events detected by these algorithms are: speed not allowed, displacement in opposite direction, changes of speed, irregularities in the road and lateral turns. In addition, these algorithms were implemented in a prototype that uses the FIWARE platform, which is an open platform for the construction of Internet applications of the future. The evaluation of this work was carried out with more than 100 trips made in different locations, roads and vehicles, of which more than 50 million records were obtained corresponding to data from the gyroscope, the accelerometer, the magnetometer and the GPS of a smart phone.

The results of the proposed application gave the following results: speed not allowed 93%, displacement in the opposite direction 93%, speed changes 94%, irregularities in the road 99%, and lateral turns 90%. These results show that it is possible to automatically detect risk events when driving a car, with the use of the FIWARE platform and the data collected by a smartphone.

Tabla de contenido

Agradecimientos	v
Resumen.....	vii
Abstract.....	viii
Lista de figuras.....	xii
Lista de tablas	xiii
Capítulo 1. Introducción.....	1
1.1 Introducción	1
1.2 Planteamiento del problema.....	1
1.3 Objetivos	2
1.3.1 Objetivo general.....	2
1.3.2 Objetivos específicos	3
1.4 Metodología de solución.....	3
1.5 Estructura de la tesis.....	4
Capítulo 2. Marco conceptual.....	5
2.1 Causas generadoras de accidentes de tránsito.....	5
2.2 Usuarios del sistema vial.....	6
2.2.1 El conductor.....	6
2.2.2 Peatón.....	6
2.3 Vehículo	6
2.4 Espacio físico o entorno.....	6
2.5 Aspectos relacionados con el usuario, el vehículo y el entorno.....	7
2.5.1 Velocidad.....	7
2.5.2 Superficie de rodadura	7
2.5.3 Distancia de visibilidad de parada	7
2.5.4 Distancia de reacción	7
2.5.5 Distancia de frenado	8
2.5.6 Distancia de visibilidad de decisión.....	8
2.5.7 Distancia de visibilidad de rebase.....	8
2.6 Normatividad.....	9
2.7 Herramientas tecnológicas	9
2.7.1 Sensores	9

2.7.2	Teléfonos inteligentes	9
2.7.3	Tratamiento de señales digitales	9
2.7.4	FIWARE	10
2.7.5	Proyecto SmartSDK.....	10
Capítulo 3.	Estado del arte.....	11
3.1	Un método para la detección de eventos de conducción usando SAX con la exploración del uso de recursos en una plataforma de teléfonos inteligentes	11
3.2	Análisis y evaluación para el reconocimiento del comportamiento de conducción basado en un acelerómetro de 3 ejes utilizando el enfoque <i>Random Forest</i>	12
3.3	Identificación de interrupciones en la superficie de la carretera basadas en patrones del acelerómetro.....	12
3.4	Aprendiendo patrones de interrupciones en la superficie de la carretera usando la representación de bolsa de palabras	12
3.5	SASVi – Sistema de asistencia y seguridad vial	13
3.6	SenSafe: Un framework para la seguridad del tráfico basado en un teléfono inteligente para la detección del comportamiento de vehículos y peatones	14
3.7	Detección de eventos de conducción utilizando los datos sensoriales de un teléfono inteligente.....	14
3.8	Resumen.....	15
3.9	Tabla comparativa.....	16
Capítulo 4.	Detección automática de eventos de riesgo en el manejo de un automóvil.....	19
4.1	Selección de los eventos y los sensores	19
4.1.1	Selección de eventos	19
4.1.2	Selección de los sensores.....	20
4.1.3	Adquisición de los datos de prueba	21
4.2	Algoritmo base para la detección de desplazamiento a una velocidad no autorizada	21
4.3	Algoritmo base para la detección de desplazamiento en contrasentido	22
4.4	Algoritmo para la detección de cambio de velocidad y frenado	23
4.5	Algoritmo base para la detección de eventos basados en reconocimiento de patrones	24
4.6	Algoritmo base para la detección de giros laterales.....	26
4.7	Algoritmo base para detectar irregularidades en la vía.....	27
4.8	Modelo para la detección automática de eventos de riesgo en el manejo de un automóvil utilizando un teléfono inteligente y la plataforma FIWARE.....	28
4.8.1	Arquitectura	28

4.8.2	Módulo de detección de eventos (Teléfono).....	28
4.8.3	Módulo de monitoreo y visualización (FIWARE).	32
Capítulo 5.	Pruebas y resultados.....	35
5.1	descripción de las pruebas.....	35
5.1.1	Casos de estudio.....	35
5.1.2	Recursos utilizados	35
5.1.3	Procedimiento de pruebas	36
5.1.4	Frecuencia de muestreo.	36
5.1.5	Análisis inicial de los datos	37
5.1.6	Datos recolectados durante el periodo de pruebas.....	38
5.1.7	Prototipo para la realización de pruebas utilizando la plataforma FIWARE.....	39
5.2	Pruebas y resultados	40
5.2.1	Procesamiento de la señal	40
5.2.2	Cambios de velocidad.....	45
5.2.3	Detección de irregularidades en la vía.....	45
5.2.4	Algoritmos basados en técnicas matemáticas	47
5.2.5	Prueba en proyecto SmartSDK Security	50
Capítulo 6.	Conclusiones y trabajos futuros.....	51
6.1	Trabajos futuros	51
6.2	Publicaciones y logros.....	52
Referencias.....		54
Anexo 1 – Funciones utilizadas por los algoritmos base.....		56

Lista de figuras

Figura 1. Metodología de solución.	3
Figura 2. Estadísticas de los resultados obtenidos en la investigación realizada por el CESVI México.	20
Figura 3. Imágenes del prototipo y la visualización de los datos a partir del csv generado.	22
Figura 4. Arquitectura general del prototipo.	28
Figura 5. Opciones de parametrización que permite el módulo del teléfono inteligente.	29
Figura 6. Categorías y opciones de etiquetado de la vista toma de muestras.	30
Figura 7. Representación de una misma señal al aplicar el filtro de medias móviles.	31
Figura 8. Ejemplo de instancia del modelo de datos SignalDevice.	33
Figura 9. Sistema de referencia utilizado con respecto al vehículo y el teléfono.	36
Figura 10. intervalo en segundos entre la toma de un dato y el siguiente.	37
Figura 11. Archivo plano que contiene los datos adquiridos con el módulo de toma de señal.	37
Figura 12. Geolocalización de los datos presentados en la Figura 11.	38
Figura 13. Representación de los datos correspondientes al eje de desplazamiento vertical. ...	38
Figura 14. Análisis de una sola señal base a partir de sus derivadas con la ayuda del bloque de manipulación de señal y QuantumLeap.	40
Figura 15. Medias móviles para la clase camino irregular.	41
Figura 16. Tratamiento de la señal utilizando medias móviles de los modelos ARMA para camino regular.	41
Figura 17. Tratamiento de la señal utilizando medias móviles de los modelos ARMA para camino irregular.	42
Figura 18. Tratamiento de la señal utilizando medias móviles de los modelos ARMA para topes.	42
Figura 19. Registro de los datos catalogados como posibles eventos.	42
Figura 20. Visualización de una misma señal con diferentes manipulaciones sobre la señal. .	43
Figura 21. Nuevos puntos de la ruta después de aplicar un diezmado de 20 a la señal.	44
Figura 22. Nuevos puntos de la ruta después de aplicar un diezmado de 50 a la señal.	44
Figura 23. Mapa generado automáticamente a partir de los datos registrados por el GPS.	44
Figura 24. Ejemplo de las instancias presente en el conjunto de datos utilizados.	45
Figura 25. Primeras instancias del conjunto de datos de entrenamiento y prueba.	46
Figura 26. Recorrido completo (Tramo A).	47
Figura 27. Recorrido extraído (Tramo B).	47
Figura 28. Parámetros de prueba.	48
Figura 29. Resultados obtenidos para el tramo A.	48
Figura 30. Resultados obtenidos para el tramo B.	¡Error! Marcador no definido.
Figura 31. Línea entre el punto inicial y final del recorrido.	49
Figura 32. Aplicaciones del Proyecto SmartSDK Security.	50

Lista de tablas

Tabla 1. Tabla comparativa de trabajos listados anteriormente.....	17
Tabla 2. Relación entre causas generadoras de accidentes y fuente para la adquisición de los datos.....	20
Tabla 3. Categorías y eventos de etiquetado por cada subcategoría.....	21
Tabla 4. Listado de técnicas utilizadas de acuerdo con la causa generadora de un accidente de tránsito de interés.	35
Tabla 5. Frecuencia de muestreo por sensor utilizando un teléfono con SO Android	36
Tabla 6. Registros recolectados en la fase de pruebas.....	39
Tabla 7. Valores mínimos y máximos de aceleración y velocidad presentes en el conjunto de datos.....	39
Tabla 8. Resumen de resultados obtenidos al pasar los datos por los clasificadores.....	45
Tabla 9. Resultados obtenidos con Naive Bayes	46
Tabla 10. Resultados obtenidos con máquinas de vector soporte.....	47
Tabla 13. Características generales de la función distancia entre dos puntos.	56
Tabla 14. Características generales de la función distancia de un punto a una línea	56
Tabla 15. Características generales de la función línea a partir de dos puntos.....	57
Tabla 16. Características generales de la función punto dentro de un polígono.....	57
Tabla 17. Características generales de la función punto dentro de una línea	57

Capítulo 1. Introducción

1.1 Introducción

La prevención de los accidentes de tránsito es un tema importante tanto para entidades gubernamentales como no gubernamentales, ya que los costos derivados de las terapias y los tratamientos constituyen una carga económica para el estado, las aseguradoras y las familias de los afectados [10-12]. En este sentido, autoridades competentes y organizaciones adelantan diferentes acciones enfocadas en la prevención y la mitigación de los daños causados por este tipo de accidentes. Algunas de estas acciones utilizan diferentes mecanismos y herramientas con el fin de detectar causas o eventos de riesgo que puedan presentarse al momento de conducir un automóvil. Sin embargo, la detección automática de dichas causas o eventos es un tema complejo, que aún tiene diferentes áreas de trabajo que pueden ser abordadas o exploradas.

En este trabajo de investigación se aborda la detección de algunos de los eventos de riesgo que pueden ocurrir en el manejo de un automóvil y se presenta un conjunto de algoritmos base que pueden ser utilizados para la detección automática de dichos eventos. Los eventos abordados en este trabajo se tomaron con base en la investigación realizada por el Centro de Experimentación y Seguridad Vial de México [13] y son: exceso de velocidad, paradas repentinas, desplazamiento en contrasentido, irregularidades en la vía y giros laterales del vehículo. Para las pruebas y la implementación de los algoritmos seleccionados se propone la utilización de un teléfono inteligente y la plataforma FIWARE.

En términos generales, el entorno de pruebas propuesto utiliza los datos provenientes de los sensores de movimiento y localización de un teléfono inteligente que se encuentra dentro de un automóvil. Los datos adquiridos son procesados para obtener información con respecto al movimiento del vehículo. Esta información es utilizada para inferir la ocurrencia de un posible comportamiento que pueda desencadenarse en un accidente de tránsito. Cuando la ocurrencia de un evento de riesgo es detectada, se genera una notificación con los datos sobre el evento detectado, la cual es reenviada a los interesados que se encuentran registrados en la plataforma del entorno de pruebas desarrollado.

Los algoritmos base presentados en este trabajo se obtienen a partir del análisis y las pruebas desarrolladas en un periodo de más de un año. En los capítulos siguientes se describen los algoritmos seleccionados y las características del entorno en que se probó este conjunto de algoritmos.

1.2 Planteamiento del problema

La detección temprana de los eventos de riesgo o las causas generadoras de accidentes de tránsito son un aspecto importante al momento de implementar estrategias que busquen prever, prevenir o mitigar los daños y pérdidas causadas por este tipo de accidentes. Los accidentes de tránsito causaron cerca de 500.000 muertes y 14 millones de heridos en todo el mundo a finales de 2016 [14]. Estos se ubicaron como la principal causa de muerte en la

población menor de 45 años en México y una de las principales causas de muerte a nivel mundial [1] [15]. Los accidentes de tránsito están relacionados con errores humanos, fallas del automóvil o condiciones del entorno; siendo las causas más frecuentes: velocidad excesiva, no guardar distancia, invasión del carril contrario, giro indebido y distracciones [4][13]. Sin embargo, muchas de estas causas pueden ser detectadas por medio del uso de diferentes técnicas y tecnologías [10].

Para abordar el problema de la detección automática de los eventos de riesgo, al momento de conducir un automóvil se han realizado varios trabajos. Algunos de estos utilizan dispositivos embebidos en los vehículos como cámaras, sensores y radares que supervisan los comportamientos de conducción y alertan al conductor sobre posibles peligros en la vía [5][16]. Los nuevos modelos de autos usan el sistema ADAS para mejorar la seguridad y minimizar el riesgo de sufrir un accidente o colisionar contra otros vehículos [8][17]; algunas ciudades utilizan sistemas para supervisión y control colocados en las carreteras [18]; Así mismo, los teléfonos inteligentes son utilizados con más frecuencia en tareas relacionadas con la detección automática de los eventos de riesgo o las causas generadoras de accidentes de tránsito; esto se debe a que vienen equipados con múltiples sensores que permiten registrar los datos provenientes del vehículo [6-9] [19][20].

Sin embargo, el tema de la detección automática de eventos de riesgo que pueden desencadenar en un accidente al momento de conducir un automóvil encierra un grado de complejidad alto; y pese al esfuerzo realizado por la comunidad, aún no está cerrado, ni existen soluciones únicas, dado que no se ha alcanzado la madurez en cuanto a las técnicas de detección y los algoritmos desarrollados para tal fin [6].

Este trabajo de investigación se enfoca en el aspecto relacionado con el desarrollo y la optimización de algoritmos que permitan la detección automática de eventos de riesgo al momento de conducir un automóvil. En este trabajo se explora la factibilidad de utilizar un teléfono inteligente y la plataforma FIWARE para detectar automáticamente algunos de los eventos de riesgo que se presentan en el manejo de un automóvil. Los eventos de interés para el presente trabajo son: exceso de velocidad, paradas repentinas, giros laterales, irregularidades en la vía y el desplazamiento en sentido contrario en vías de un único sentido.

1.3 Objetivos

1.3.1 Objetivo general

Detectar automáticamente eventos de riesgo que se presentan en el manejo de un automóvil a través de la utilización de un teléfono inteligente y la plataforma FIWARE.

1.3.2 Objetivos específicos

- Seleccionar eventos, sensores y datos de prueba que serán utilizados para el desarrollo de los algoritmos base.
- Desarrollar algoritmos que permitan detectar si existe un evento de riesgo al momento de conducir un automóvil, a partir de los datos provenientes de un teléfono inteligente ubicado dentro de un automóvil.
- Realizar pruebas a los algoritmos desarrollados en un entorno controlado utilizando el ecosistema FIWARE.

1.4 Metodología de solución

La metodología de solución propuesta en este trabajo está compuesta por tres fases. La primera fase se centra en la selección de eventos, los sensores y la adquisición de datos necesarios para el desarrollo de los algoritmos base. La segunda fase abarca lo relacionado con la selección o el diseño de las técnicas y algoritmos que son utilizados para la detección de eventos de mal manejo e irregularidades en la vía. La tercera fase se enfoca en el desarrollo del prototipo utilizado para realizar las pruebas de las técnicas y algoritmos seleccionados. Ver Figura 1.

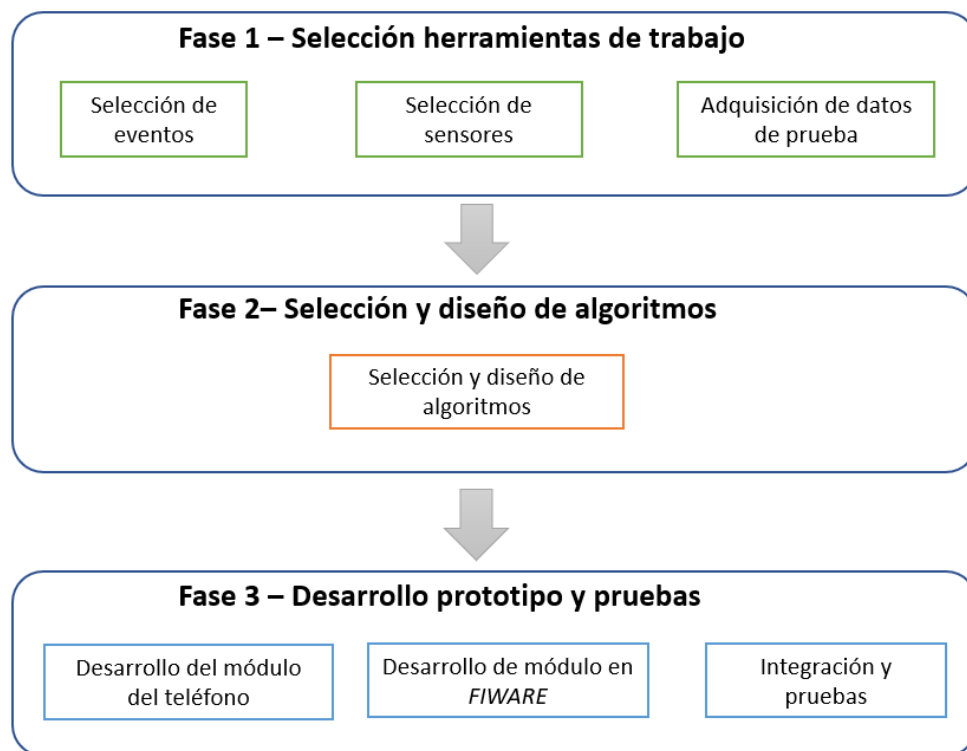


Figura 1. Metodología de solución. Fuente: elaboración propia.

1.5 Estructura de la tesis

La estructura de la tesis está formada por 6 capítulos. Estos son:

Capítulo 2. Estado del arte: En este capítulo se presentan los trabajos tomados como referencia para el desarrollo de los algoritmos base.

Capítulo 3. Marco conceptual: En este capítulo se presentan los conceptos básicos utilizados para la construcción de los algoritmos y el diseño del modelo para la detección de eventos de riesgo en el manejo de un automóvil.

Capítulo 4. Detección automática de situaciones de riesgo en el manejo de un automóvil: Este capítulo presenta el diseño y el desarrollo de los algoritmos base para la detección automática de eventos de riesgo en el manejo de un automóvil. El capítulo presenta una descripción detallada del prototipo de software desarrollado para la ejecución de las pruebas del algoritmo, el cual consta de una aplicación móvil desarrollada en la plataforma Android versión 4.0 y una aplicación web desarrollada utilizando componentes de la plataforma FIWARE.

Capítulo 5. Pruebas y resultados: En este capítulo se presenta una descripción de las pruebas realizadas y los resultados obtenidos a partir de los experimentos realizados en el entorno de pruebas.

Capítulo 6. Conclusiones y trabajos futuros: Por último, se presentan las conclusiones obtenidas a partir de los experimentos realizados y se mencionan los trabajos futuros que podrían realizarse a partir de los resultados obtenidos en este trabajo.

Capítulo 2. Marco conceptual

En este capítulo se presenta una descripción general de los conceptos tomados como referencia para el diseño o la selección del conjunto de algoritmos que se presentan en este trabajo. El tema abordado en este trabajo está enmarcado en el dominio del ecosistema vial (movilidad), enfocándose específicamente en el aspecto de la seguridad vial relacionado con los accidentes de tránsito. Si bien es cierto que la solución propuesta no tiene como finalidad la detección de un accidente de tránsito, también es cierto que los eventos e irregularidades abordados en esta solución pueden trascender hasta convertirse en accidentes de tránsito. Por lo tanto, la detección de la ocurrencia de uno de estos eventos puede apoyar a la toma de acciones que permitan la prevención de un accidente o la mitigación de los daños causados por el mismo. Un accidente de tránsito puede catalogarse como un percance vial que se presenta súbita e inesperadamente, determinado por condiciones y actos irresponsables potencialmente previsible, atribuidos a factores humanos, vehículos automotores, condiciones climatológicas, señalización y caminos [21]. En un mismo accidente de tránsito pueden intervenir varios factores. Aunque existen diferencias en las estadísticas sobre el porcentaje de accidentes que se presentan en cada factor, la mayoría concuerda en que los accidentes de tránsito se pueden atribuir a factores humanos, del vehículo o del entorno [11] [12] [13].

En las siguientes secciones se describen algunas observaciones relacionadas con los elementos antes mencionados, para tal fin se toma como referencia el manual de proyecto geométrico de carreteras de la dirección general de servicios técnicos [22] y el reglamento de tránsito en carreteras y puentes de jurisdicción federal de la secretaría de seguridad pública [23]. Estos documentos son desarrollados por instituciones públicas del estado mexicano. Los aspectos generales en los que se soporta este trabajo son: Causas generadoras de accidentes de tránsito, usuarios del ecosistema vial, el vehículo, el espacio físico, factores, la normatividad y las herramientas tecnológicas.

2.1 Causas generadoras de accidentes de tránsito

Las causas generadoras de accidentes de tránsito están generalmente relacionadas con factores humanos, el entorno o el vehículo, según la investigación realizada por el centro de experimentación y seguridad vial de México [13]. Los eventos abordados en este trabajo están asociados con el factor humano (exceso de velocidad, paradas repentinas, giros laterales, y desplazamiento en sentido contrario en vías de un único sentido.) y el factor del entorno (irregularidades en la vía). De acuerdo con la literatura revisada, el factor por el cual se generan más accidentes de tránsito es el humano y una de las causas que mayor accidentalidad genera es el exceso de velocidad [10-12]. Aunque los accidentes de tránsito son una realidad cotidiana, estos pueden ser predecibles y evitables; esta característica de los accidentes de tránsito ha dado lugar a que el concepto de seguridad vial tome fuerza. El principal objetivo de la seguridad vial es salvaguardar la integridad física de las personas que transitan por la vía, eliminando o disminuyendo los factores de riesgo [10].

2.2 Usuarios del sistema vial

Los usuarios del sistema vial son las personas que hacen uso de las vías. En este sentido, [10] [16] clasifican a los usuarios de la vía como: conductores, peatones, ciclistas, entre otros; mientras que [22] los clasifica como conductores, peatones o pasajeros y ciclistas; y en [23] se describe a los usuarios como conductores, pasajeros y peatones.

2.2.1 El conductor

El conductor es la persona que tiene el control y la responsabilidad del desplazamiento de un vehículo. Según [22], algunos aspectos relacionados con el diseño geométrico de la carretera son tomados a partir de algunas características relacionadas con el conductor. Además, las decisiones y acciones del conductor dependen de lo que le transmiten sus sentidos, siendo la vista uno de los sentidos más utilizados[24]. La visión es la limitación más importante para conducir; en particular la agudeza visual, la visión periférica, el deslumbramiento, la percepción de colores y la profundidad de percepción. Otros aspectos relacionados con el conductor son la expectativa, el tiempo de reacción, y el tiempo de respuesta [22].

2.2.2 Peatón

El peatón es la persona que transita a pie por las vías públicas [23]. La mayoría de los peatones consideran que no están sujetos a las leyes y reglamentos de tránsito; lo que hace difícil proyectar un movimiento peatonal ordenado y seguro. También se consideran peatones los que empujan cualquier otro vehículo sin motor de pequeñas dimensiones, o las personas con movilidad reducida que circulan con una silla de ruedas con o sin motor [22].

2.3 Vehículo

El vehículo es el medio de transporte que utiliza el usuario para circular por la carretera, y sus principales características son: tipo, dimensiones, peso y características de operación [22]. Un vehículo es el medio de transporte que está dotado de llantas que le permiten transitar, y es considerado un automotor cuando está dotado de medios de propulsión propios. [23]. Uno de los aspectos importantes relacionados con el vehículo son sus características de operación. Estas generalmente están asociadas con la dinámica del movimiento del vehículo y son de gran importancia tanto en el diseño de carreteras como el diseño y construcción de los vehículos. Algunas características relacionadas con la dinámica del movimiento del vehículo son: comportamiento dinámico del vehículo, comportamiento direccional del desplazamiento del vehículo, la deriva y la estabilidad [22].

2.4 Espacio físico o entorno

El espacio físico es el lugar en el cual las causas evolucionan hasta convertirse en accidentes de tránsito. En este sentido, el espacio más importante es la vía o la carretera por la cual transitan los automóviles. Sin embargo, el hecho de que una vía se pueda extender a lo largo del territorio y pueda cubrir grandes distancias, lleva a mirar la vía como un ente dinámico, que responde a factores como: las condiciones del terreno, el tipo de zona que

atraviesa (rural o urbana), características especiales de la zona (zonas residenciales, escolares, hospitales, entre otras) [22] [25] [26].

2.5 Aspectos relacionados con el usuario, el vehículo y el entorno.

En esta sección se presentan algunos conceptos que, por su naturaleza, involucran a más de uno de los elementos descritos hasta el momento. Estos conceptos son tomados del manual de proyecto geométrico de carreteras, documento oficial que establece los parámetros que deben considerarse para la construcción de carreteras en México[22].

2.5.1 Velocidad

La velocidad es la relación entre la distancia y el tiempo empleado para recorrerla, normalmente expresada en km/h. Es uno de los factores más importantes que intervienen en la selección de rutas y medios de transporte alternos del usuario. La velocidad es muy variable entre conductores y entre puntos de la carretera, por lo que conviene categorizarla como una variable aleatoria.

2.5.2 Superficie de rodadura

La superficie de rodadura es la superficie que está expuesta de manera directa a la acción de los neumáticos de los vehículos y posibilita su circulación; sus características influyen directamente en el comportamiento del vehículo en la carretera. Algunos aspectos relacionados con la superficie de rodadura son la uniformidad y la fricción.

2.5.3 Distancia de visibilidad de parada

La distancia de visibilidad de parada es la distancia de visibilidad necesaria para que un conductor, viajando a una determinada velocidad, sea capaz de detener su vehículo antes de alcanzar un objeto fijo ubicado en su trayectoria. La distancia de visibilidad de parada está compuesta por dos segmentos: la distancia de reacción y la distancia de frenado.

2.5.4 Distancia de reacción

La distancia de reacción es la distancia recorrida por el vehículo desde el instante en que el conductor reconoce la presencia de una situación peligrosa en la vía, hasta el instante en que opera los frenos. Un tiempo de reacción, a veces denominado de percepción-reacción, de 2.5 segundos, es adecuado para el 90% de las situaciones que pueden presentarse. Sin embargo, en situaciones complejas, como las que se presentan en intersecciones, pueden requerirse tiempos mayores. La distancia de reacción (d_R), en metros, se determina suponiendo que el vehículo circula a la velocidad (V), en km/h, durante el tiempo de reacción (t), en segundos:

$$d_R = \frac{V}{3.6} t \quad (1)$$

La constante de 3.6 aparece luego de hacer la conversión de km/h a m/s.

2.5.5 Distancia de frenado

La distancia de frenado (d_F), en metros, de un vehículo en carretera plana, que circula a la velocidad (V), en km/h, se determina igualando el trabajo efectuado por la fuerza de frenado a la energía cinética del vehículo, donde f_l es el coeficiente de fricción longitudinal:

$$d_F = \frac{V^2}{254 f_l} \quad (2)$$

La constante de 254 aparece luego de hacer la conversión de m/s a km/h y considerar un valor de 9.8 para la gravedad. Un enfoque más reciente para determinar la distancia de frenado, acorde a la tecnología automotriz actual, es considerar una tasa de deceleración (a) que resulte cómoda para la mayor parte de los conductores. Se ha considerado que tal tasa es 3.4 m/s², por lo que:

$$d_F = \frac{V^2}{88} \quad (3)$$

2.5.6 Distancia de visibilidad de decisión

Las distancias de visibilidad de parada resultan a menudo escasas cuando los conductores deben tomar decisiones complejas, cuando la información es difícil de percibir y procesar o cuando la maniobra es inesperada o inusual. En tales casos, conviene usar una distancia de visibilidad de decisión en lugar de una distancia de parada. La distancia de visibilidad de decisión es la distancia mínima necesaria para que un conductor pueda maniobrar el vehículo con anticipación, ante la presencia de una situación cuya complejidad demanda tiempos de percepción-reacción más grandes a los usualmente requeridos en la carretera. Esta distancia se determina con las mismas expresiones, pero considerando el siguiente tiempo de reacción (t), en segundos, para las maniobras de parada y cambio de velocidad o trayectoria:

- Parada, zona rural: 3 segundos.
- Parada, zona urbana: 9 segundos.
- Cambios, zona rural: 10 a 11 segundos.
- Cambios, zona suburbana: 12 a 13 segundos.
- Cambios, zona urbana: 13 a 14 segundos.

2.5.7 Distancia de visibilidad de rebase

Se dice que un tramo de carretera tiene distancia de visibilidad de rebase, cuando la distancia de visibilidad en ese tramo es suficiente para que el conductor de un vehículo pueda adelantar a otro que circula por el mismo carril, en presencia de un tercer vehículo que circula en sentido contrario. La distancia de visibilidad de rebase para el diseño de las carreteras en México es equivalente a la recorrida en 18 segundos. La distancia de visibilidad de rebase mínima, D_R en m, puede determinarse por la velocidad en km/h con la expresión:

$$D_R = 6.8 * V \quad (5)$$

2.6 Normatividad

La normatividad es el conjunto de reglas y leyes cuyo objetivo es proteger a los usuarios del ecosistema vial, y establecer parámetros de circulación que permitan el desplazamiento de los individuos en condiciones seguras. Para este trabajo, la normatividad es un aspecto muy importante, porque ofrece una visión general de los criterios y restricciones que deben tenerse en cuenta, para seleccionar las reglas y patrones para el diseño o selección de las técnicas; las cuales tendrán como objetivo la detección automática de la ocurrencia de eventos de mal manejo o la presencia de irregularidades en la vía.

2.7 Herramientas tecnológicas

Los esfuerzos que se están realizando en torno a la seguridad vial son diversos, y algunas de las estrategias se apoyan en herramientas tecnológicas de diferentes tipos. Estas herramientas pueden ser aplicaciones o dispositivos, utilizados para monitorear o controlar algunos comportamientos en la vía. En esta sección se describen algunas de las tecnologías utilizadas en este trabajo de investigación.

2.7.1 Sensores

Los sensores son dispositivos que detectan una acción externa y la transmiten de manera adecuada [27]. Un sensor convierte una señal física de un tipo en una señal física de otra naturaleza, para lo cual, se necesitan etapas adicionales de acondicionamiento como la amplificación y el filtrado de la señal. Algunos de los aspectos que caracterizan a los sensores son: la naturaleza de la señal que genera el sensor, la influencia de las señales de ruido, la calibración del sensor con respecto a la variable física, el tiempo de respuesta del sensor, entre otros [28].

2.7.2 Teléfonos inteligentes

Un teléfono inteligente es un dispositivo portátil que tiene gran capacidad de comunicación, procesamiento y almacenamiento. Además, del manejo de contenidos multimedia y facilidad para integrarse a redes inalámbricas [29]. Los teléfonos inteligentes cuentan con un conjunto de sensores para la detección de señales físicas como; aceleración, rotación, sonido, luminosidad, proximidad, campo magnético, entre otros. El empleo de teléfonos inteligentes ha sido propuesto en varias experiencias que abarcan temáticas de mecánica, electromagnetismo, óptica, oscilaciones y ondas entre otras [30].

2.7.3 Tratamiento de señales digitales

El procesamiento de una señal es la aplicación de una serie de operaciones lógicas y matemáticas a un conjunto de datos provenientes de la señal. Un teléfono inteligente contiene varios sensores que le permiten ser una fuente generadora de señales eléctricas. Estas señales pueden ser analizadas a través de diferentes algoritmos y procesos que las convierte en información útil para diferentes aplicaciones. Dentro de los aspectos más importantes asociados con la toma de señales se encuentran: el tipo de la señal, la frecuencia de muestreo y la conversión de la señal. Estos aspectos están directamente relacionados con el tipo de

sensores utilizados para la captura de la señal y el tipo de técnicas utilizadas para el procesamiento y manipulación de la misma [31].

2.7.4 FIWARE

FIWARE es una plataforma de software abierto para el desarrollo de aplicaciones en el entorno del internet del futuro, también conocido como el internet de las cosas [32]. El internet de las cosas reúne una serie de conceptos e ideas sobre productos y servicios presentes y futuros, que aprovechan las capacidades sensoriales y de conectividad de los objetos, para el desarrollo de servicios y aplicaciones [33]. La plataforma FIWARE proporciona un conjunto de herramientas y librerías denominadas *Generic Enablers* (GEs). Los GEs ofrecen interfaces que facilitan tareas como: la integración de los dispositivos IoT; el análisis y procesamiento de datos a mediana y gran escala (Big Data); y la incorporación de interfaces avanzadas para la interacción con los usuarios [34].

2.7.5 Proyecto SmartSDK

SmartSDK es un proyecto de colaboración entre México y Europa para promover la adopción y el uso de la plataforma FIWARE en México. Algunos de sus objetivos son: proporcionar modelos de datos y arquitecturas de servicios abiertos y reutilizables para el desarrollo de aplicaciones inteligentes; proporcionar los elementos básicos a los desarrolladores interesados en construir aplicaciones bajo la plataforma FIWARE, para escenarios de: ciudades inteligentes; salud inteligente; y seguridad inteligente [6].

Capítulo 3. Estado del arte

En este capítulo se presentan los trabajos de investigación que fueron tomados como punto de partida para la selección y el desarrollo de los algoritmos base. Los trabajos seleccionados cumplen con al menos uno de los siguientes criterios: el trabajo detecta alguno de los eventos de interés abordados en esta solución y/o el dispositivo usado para la toma de datos es un teléfono inteligente. Inicialmente se presentan los trabajos seleccionados y posteriormente se presenta una comparativa entre los trabajos que relacionan los eventos detectados y los sensores utilizados para la toma de datos.

3.1 Un método para la detección de eventos de conducción usando SAX con la exploración del uso de recursos en una plataforma de teléfonos inteligentes

Este trabajo de investigación presenta un método para la detección de eventos de conducción usando el algoritmo *SAX* (*Symbolic Aggregate approXimation*) y teléfonos inteligentes [8]. En este trabajo se describe una metodología que usa el algoritmo *SAX*. La metodología está compuesta por tres fases: en la primera, se realiza una transformación de los datos utilizando el algoritmo *SAX*; en la segunda se realiza el cálculo de las distancias; y en la tercera hace una comparación de los valores de distancia, con el fin de determinar el tipo de evento de conducción y el lugar en que se presentó el evento. El algoritmo *SAX* es utilizado porque aproxima una serie de tiempo a valores discretos, lo cual es apto para utilizar en series de tiempo con datos ruidosos, como es el caso de los datos recolectados de los sensores del teléfono inteligente. Además, este algoritmo tiene su propia función de distancia basada en el límite inferior de la distancia euclidiana. Los eventos que se detectan en este trabajo de investigación son: aceleración, giro a la izquierda y a la derecha, cambios de carril y giro en U. Para la recolección de los datos se utiliza el acelerómetro, el magnetómetro y el GPS (*Global Positioning System*) de un teléfono inteligente. De acuerdo con los resultados reportados, la metodología propuesta tarda tres segundos en detectar un evento y su precisión está entre el 50% y el 100% dependiendo del evento; la tasa de detección entre el 20% y el 37,5%. Los pasos seguidos por esta metodología son los siguientes:

1. Recibe una serie A de tamaño n
2. Define un tamaño de la palabra(w), el cual es el tamaño de la nueva serie.
3. Define el tamaño del alfabeto($sizeAlf$), que representa el número de rangos en los que se discretizará la serie
4. Establece el tamaño w de la nueva serie A'
5. Divide A en w segmentos de tamaño igual
6. Halla la media de los w segmentos y los lleva a'
7. Normaliza la serie A' (campana de Gauss)
8. Define las letras del alfabeto
9. Asigna cada letra del alfabeto a un área de la campana de Gauss
10. Asigna a cada segmento de A' la letra correspondiente de acuerdo con su valor y su equivalente en la campana de Gauss
11. Crea la palabra correspondiente a'
12. Calcula la distancia entre A' y los ejemplos

13. Determina el grado de similitud

3.2 Análisis y evaluación para el reconocimiento del comportamiento de conducción basado en un acelerómetro de 3 ejes utilizando el enfoque *Random Forest*

Este trabajo de investigación [35] propone la identificación automática de los eventos de conducción mediante un procedimiento que consta de dos etapas. En la primera etapa se toman los datos provenientes de un acelerómetro de tres ejes y se elimina el ruido de los datos aplicando tres filtros (filtro de *Kalman*, filtro de *Wavelet* y filtro de partículas). En la segunda etapa, se utiliza el algoritmo *Random Forest* (Bosques aleatorios) para clasificar los eventos. Los eventos que detecta son: arranque, frenado, giro a la izquierda, giro a la derecha, cambio de línea a la izquierda, cambio de línea derecha, giro en u y giros dobles derecha/izquierda. De acuerdo con los resultados reportados, el método propuesto en este trabajo tiene una precisión promedio del 98.1% y del 100% en la clasificación de eventos como: el frenado, el arranque y el giro a la derecha. En las pruebas se utiliza una base de datos con 2.400 eventos, y los resultados obtenidos se comparan con: árboles de decisión, C4.5, máquinas de vector soporte, *Naive Bayes*, *K-means* y *EM Clustering*. Una vez realizada la comparación, el algoritmo *Random Forest* es el que obtiene mejores resultados de detección. Aunque el trabajo reporta buenos resultados, este no expresa cuales son las 22 características que describen a los eventos, ni se especifica que ejes del acelerómetro utiliza. Además, no contempla el exceso de velocidad o las irregularidades en la vía, ni presenta un detalle sobre el método o los algoritmos implementados.

3.3 Identificación de interrupciones en la superficie de la carretera basadas en patrones del acelerómetro

El trabajo de investigación [36] propone la identificación automática de interrupciones o irregularidades en la superficie de la vía mediante la utilización de dos algoritmos: una red neuronal y el método de regresión logística. La red neuronal está programada en *python* y consta de una capa oculta y 5 neuronas. La regresión logística está implementada en *Matlab* y, utiliza las funciones propias del programa para optimizar el algoritmo. Aunque la implementación de la red neuronal y la regresión logística se hace en herramientas y lenguajes diferentes, los datos que se utilizan en las pruebas son los mismos. Los eventos que puede detectar son: topes, baches, boyas, pavimento regular y pavimento irregular. El porcentaje de exactitud reportado para la identificación de topes y boyas es mayor al 87%. Las pruebas indican que los tres ejes del acelerómetro aportan información de valor para la identificación de irregularidades en la vía y que el método que tiene mejores resultados es la red neuronal. El trabajo no aborda aspectos relacionados con el comportamiento del conductor o el exceso de velocidad.

3.4 Aprendiendo patrones de interrupciones en la superficie de la carretera usando la representación de bolsa de palabras

El trabajo de investigación [6] propone un algoritmo que permite la detección automática de: baches, topes de asfalto, topes metálicos, camino irregular y camino regular. Este

algoritmo utiliza la técnica *BoW* (Bolsa de palabras, *BoW* por el término inglés “*Bag of Words*”) para la extracción de segmentos cortos y representativos de la señal proveniente del acelerómetro. *BoW*, es una técnica utilizada principalmente en el procesamiento de lenguaje natural, que se ha extendido a otras áreas como la clasificación de imágenes. Los pasos que componen el algoritmo son: segmentación de todas las mediciones de entrenamiento de acuerdo con un tamaño de ventana predefinido; cada segmento se representa por un descriptor; todos los descriptores correspondientes a mediciones de una clase, se agrupan utilizando el algoritmo *k-means* en *k-clusters*. Según lo reportado en este trabajo, los resultados obtenidos por este algoritmo están por encima del 90%, y se obtienen valores cercanos al 100%. Además, el algoritmo discrimina con mayor precisión las subclases de un determinado evento. Al usar este algoritmo, la clasificación binaria supera resultados reportados por la literatura y, una red neuronal artificial obtiene la mayor precisión. Un aspecto a tener en cuenta es que previo a la realización de las pruebas, se realiza un recorrido para identificar y geo-localizar los lugares donde hay irregularidades en la vía. Algunos aspectos a resaltar de este trabajo son los siguientes:

- a) El algoritmo extrae todas las palabras presentes en un texto y las guarda en un vector al que llama diccionario.
- b) El diccionario no guarda palabras repetidas ni caracteres especiales
- c) A cada palabra se le asigna una posición única dentro del diccionario
- d) Para representar un texto se realiza el siguiente procedimiento:
 - i) Se crea un vector con tantas posiciones como palabras tenga el diccionario.
 - ii) Se recorre el texto palabra por palabra, se busca la posición asignada a la palabra y se incrementa su valor a 1
- e) El texto es representado en función de la frecuencia de aparición de las palabras que lo conforman, el orden en que se agregan las palabras al diccionario no es relevante, pero una vez asignadas, debe permanecer así.
- f) Para verificar la similitud o identificar la categoría de un texto, se utilizan diferentes técnicas como pueden ser las medidas de distancia.

Los pasos realizados por el algoritmo son los siguientes:

1. Toma de datos de los individuos o ejemplos (serie de tiempo)
2. Segmentación de la serie en trozos de tamaño n
3. Clasificación de segmentos en grupos utilizando *k-means*
4. Selección de los centroides como palabras del diccionario
5. Construcción del diccionario de palabras a partir de los centroides
6. Reescribir los ejemplos utilizando el algoritmo de código de palabras
7. Pasar los ejemplos a los clasificadores seleccionados.

3.5 SASVi – Sistema de asistencia y seguridad vial

SASVi [37] es un sistema de asistencia y seguridad vial que ofrece recursos y facilidades para notificar al conductor sobre posibles irregularidades en la vía. Esta plataforma es desarrollada para la ciudad de Chihuahua. Según lo reportado en este trabajo, la ciudad de Chihuahua registra el tercer lugar en accidentes viales a nivel nacional, siendo el exceso de

velocidad la principal causa de accidentalidad. La plataforma utiliza el GPS y el giroscopio de un teléfono inteligente para obtener la posición actual y la orientación del vehículo. La información proveniente del GPS se procesa para calcular la velocidad del automóvil y se compara contra las velocidades máximas permitidas. Además, la plataforma notifica al conductor sobre posibles encuentros con baches o reductores de velocidad. Según los resultados reportados, el sistema SASVi ofrece soporte de asistencia vial a través de notificaciones auditivas, cuando existe una posible irregularidad en el pavimento o cuando se ha rebasado el límite de velocidad para la vía por la cual se transita. La plataforma informa sobre la presencia de baches, topes, boyas y exceso de velocidad, pero no contempla la detección de conductas o maniobras peligrosas realizadas por parte del conductor.

3.6 SenSafe: Un framework para la seguridad del tráfico basado en un teléfono inteligente para la detección del comportamiento de vehículos y peatones

SenSafe, [16] es un marco de trabajo que utiliza un teléfono inteligente para detectar automáticamente eventos circundantes y proporcionar alertas a los conductores. Los eventos que detecta este marco de trabajo son: giro a la izquierda, giro a la derecha, cambio de carril a la izquierda, cambio de carril a la derecha, giro a la izquierda y giro a la derecha. Para la recolección de los datos se usan los sensores de un teléfono inteligente. Los sensores utilizados son: un acelerómetro de 3 ejes, el magnetómetro, el giroscopio y el GPS. Según los resultados reportados este marco de trabajo sólo utiliza un teléfono inteligente para detectar el comportamiento de conducción, intercambiar información y generar recordatorios y alertas. Éste utiliza la comunicación basada en eventos para proporcionar a los conductores indicaciones comunes, de los comportamientos de los vehículos circundantes y alertas de seguridad sobre posibles colisiones. Su porcentaje de detección esta entre el 84% y el 91%. Un aspecto importante de este trabajo es que está enfocado en mejorar la seguridad de todos los usuarios en la vía (conductores, peatones, ciclistas, entre otros). Una diferencia entre el marco de trabajo *SenSafe* y la plataforma SASVi, es que el marco de trabajo SenSafe detecta las conductas agresivas o riesgosas generadas por el conductor, mientras que la plataforma SASVi detecta las imperfecciones de la vía.

3.7 Detección de eventos de conducción utilizando los datos sensoriales de un teléfono inteligente

Este trabajo de investigación está enfocado en la detección automática de eventos de conducción agresivos, para lo cual utiliza tres algoritmos: algoritmo basado en reglas, algoritmo de concordancia de patrones y algoritmo de coincidencia de patrones auto desencadenados [4]. Para la detección de los eventos se utilizan los datos provenientes de los sensores de un teléfono inteligente. Los eventos de conducción que detecta son: frenado, frenado brusco, aceleración, aceleración brusca, giro a la izquierda, giro brusco a la izquierda, giro a la derecha, giro brusco a la derecha, cambio de carril a la izquierda, cambio de carril brusco a la izquierda, cambio de carril a la derecha y cambio brusco de carril a la derecha. Según los resultados obtenidos, el algoritmo basado en reglas no realiza una detección adecuada de los eventos de conducción en las direcciones laterales. La tasa de detección del

algoritmo de coincidencia de patrones oscila entre el 37,5% y el 100%, siendo el evento de frenado repentino el que presenta la tasa de detección más baja. El trabajo se centra en la detección de conductas agresivas y propone tres técnicas diferentes según sea el sensor que se utilice. Dentro de los trabajos realizados, es el único que utiliza el GPS para la detección de eventos, lo cual amplía la solución propuesta más allá del dominio de los teléfonos inteligentes.

3.8 Resumen

En la revisión del estado del arte se encontró que los trabajos enfocados en el reconocimiento y clasificación de eventos de riesgo al momento de conducir un automóvil constan de dos fases. La primera fase se enfoca en la toma de la señal, el filtrado del ruido y la transformación de la señal. La segunda fase se enfoca en la detección y la clasificación de los eventos que son de interés.

De acuerdo con la revisión realizada, generalmente las técnicas que se utilizan para el procesamiento de la señal son transformadas de Fourier, transformada Wavelet, cosenos, técnicas estadísticas, aproximación lógica simbólica, bolsa de palabras, etc. En los trabajos más recientes se ha optado por la utilización de las técnicas alternativas, porque son más fáciles de implementar y además se pueden complementar con técnicas de aprendizaje de máquina.

Por otro lado, en la fase de detección y clasificación de los eventos se puede observar que algunos trabajos se centran en la utilización y comparación de los resultados arrojados por los diferentes algoritmos, antes que en la modificación y ajuste del vector de características. Por otro lado, en los algoritmos de detección y clasificación, los mejores resultados reportados por la literatura son derivados de casos en los que se utilizan redes neuronales artificiales o árboles de decisión, en sus respectivas variantes.

La revisión del estado del arte permite inferir que la detección y clasificación temprana de un evento de riesgo al momento de conducir puede servir como apoyo a la prevención de accidentes de tránsito. Sin embargo, la literatura reporta otros aspectos importantes que influyen en los accidentes de tránsito tales como: violación de la normativa de tránsito, condiciones físicas y mentales del conductor o factores climáticos. Los trabajos revisados reportan buenos resultados de clasificación y detección de los diferentes eventos, sin embargo, cuenta con las siguientes deficiencias:

- Las pruebas se realizan en un ambiente controlado con el dispositivo en una posición fija, generalmente ubicados en el asiento del copiloto. El trabajo [6] realiza una gran aportación al realizar un estudio para identificar los lugares en los que generalmente un usuario puede llevar su dispositivo móvil mientras maneja.
- Algunos trabajos dejan el conjunto de datos con los que realizaron los experimentos a disposición de la comunidad científica. Sin embargo, en el conjunto de datos aportados no se incluyen todos los sensores o sus ejes, lo cual imposibilita la replicación de los experimentos. Por ejemplo, el trabajo [10] proporciona el conjunto de datos de prueba utilizado para identificar imperfecciones en la vía. Este trabajo

reporta que utilizó el acelerómetro, el giroscopio y el GPS, pero el conjunto de datos solo tiene información del eje Z del acelerómetro y no permite identificar las variables temporales o espaciales.

- Algunos de los trabajos realizados reportan que varios ejes pueden aportar información para un mismo evento, pero lo plantean para trabajos futuros.

La revisión del estado del arte también mostró que la velocidad puede ser obtenida a través de los datos recolectados por el GPS o los datos extraídos del acelerómetro, siendo más común la utilización del GPS. Los aspectos más relevantes extraídos de la revisión de la literatura y que aportan de manera significativa para este trabajo son:

- La identificación de técnicas y herramientas utilizadas en la solución de este problema. Además, permite identificar una tendencia en la utilización de teléfonos inteligentes como opción para resolver este problema, debido a aspectos como el costo-beneficio, la facilidad de implementación y la versatilidad en cuanto a cantidad de sensores presentes.
- Los sensores más utilizados para resolver este problema utilizando un teléfono inteligente son: acelerómetro, giroscopio, GPS y magnetómetro.
- Las pruebas realizadas hasta el momento se ejecutan con el teléfono ubicado en una posición fija.
- Un punto clave en la detección de los eventos, está relacionado con la selección y ajuste de los algoritmos encargados del procesamiento y filtrado de la señal, ya que, de la calidad de los datos obtenidos en esta etapa, depende el resultado de los algoritmos de detección y clasificación.
- Existen limitantes relacionadas con la capacidad de cómputo y el consumo de energía, que dificultan la ejecución de algunas técnicas de reconocimiento directamente en el teléfono.

3.9 Tabla comparativa

En esta sección se presenta una tabla comparativa en la cual se listan los trabajos antes descritos y se muestra la técnica algorítmica que utilizan, el número de vehículos utilizados en la fase de pruebas, los sensores que utiliza, los eventos que detecta, el número de eventos de prueba y si tiene o no un repositorio de datos disponible al público, ver Tabla 1.

Tabla 1. Tabla comparativa de trabajos listados anteriormente.

Trabajo de investigación	Técnica algorítmica	Nº vehículos usados	Sensores utilizados	Eventos detectados	Nº de eventos de prueba	Repo.
Análisis y evaluación para el reconocimiento del comportamiento de conducción basado en un acelerómetro de 3 ejes utilizando el enfoque <i>Random Forest</i> [35].	Algoritmo <i>Random Forest</i>	6	Acelerómetro	Frenado (100%), giro derecho (100%), giro izquierdo (> 90%), arranque (100%), giro en u (> 90%), cambio de carril a la derecha (> 90%), cambio de carril izquierda (> 90%).	2400	No
Un método para la detección de eventos de conducción usando SAX con la exploración del uso de recursos en una plataforma de teléfonos inteligentes [8].	Series de tiempo usando SAX	1	Acelerómetro, GPS, magnetómetro	Frenado (100%), frenado repentino (50%), aceleración repentina (66.67%), aceleración (66.67%), giro repentino derecha (60%), giro derecho (100%), giro izquierdo (100%), giro repentino izquierda (66.67%), giro en u (71.43%), cambio de carril a la derecha (100%), cambio de carril izquierda (50%).	120	No
Identificación de las interrupciones en la superficie de la carretera basadas en patrones del acelerómetro [36].	Red neuronal y regresión logística	2	Acelerómetro, GPS	Irregularidades en la vía entre 76% y 91%, no se observó un reporte por evento en particular.	201	Si**
Aprendiendo patrones de interrupciones en la superficie de la carretera usando la representación de bolsa de palabras [6].	Bolsa de palabras	12	Acelerómetro	Tope de metal (93.73%), tope de asfalto (95.9%), baches (95.27%), camino regular (94.42%), camino irregular (89.85%).	500	Si**
SASVi – Sistema de Asistencia y Seguridad Vial [37].	Red neuronal	No especificado	Acelerómetro, GPS, giroscopio	Baches (>85%), boyas (>85%), topes (>85%), velocidad (>85%).	121	No
Detección de eventos de conducción utilizando los datos sensoriales de un teléfono inteligente [4].	ABR, ABP, ACPA	1	Acelerómetro, GPS	Frenado (94.44%), frenado repentino (37.5), aceleración repentina (100%), aceleración (50%), giro repentino derecha (75%), giro derecha (100%), giro izquierda (100%), giro repentino izquierda (80%), cambio de carril a la derecha (50%), cambio de carril a la derecha brusco (50%), cambio de carril izquierda (100%), cambio de carril izquierda brusco (50%).	120	No

<p><i>SenSafe</i>: Un <i>Framework</i> para la seguridad del tráfico basado en un teléfono inteligente para la detección del comportamiento de vehículos y peatones [16].</p>	No especificado	No especificado	<p>Acelerómetro, giroscopio, GPS, magnetómetro</p>	<p>Frenado (100%), giro izquierdo (100%), giro derecho (100%), aceleración (100%), cambio de línea único (88%), cambio de línea múltiple (91%), retroceso (84%).</p>	No especificado	No
<p>Detección automática de eventos de riesgo en el manejo de un automóvil utilizando un teléfono inteligente y la plataforma FIWARE</p>	<p>Reconocimiento de patrones, técnicas estadística y matemáticas</p>	<p>más de 50 entre servicio público y particular</p>	<p>Acelerómetro, giroscopio, GPS, magnetómetro</p>	<p>Irregularidades en la vía (99), Exceso de velocidad (93%), cambio de velocidad (94%), desplazamiento en sentido contrario (93%), giros laterales (90%).</p>	13600	Si

Capítulo 4. Detección automática de eventos de riesgo en el manejo de un automóvil

El proyecto de investigación para la detección automática de eventos de riesgo en el manejo de un automóvil se enfoca en el diseño o selección de un conjunto de algoritmos base que permiten la detección automática de eventos como el exceso de velocidad o el desplazamiento en sentido contrario a través de un teléfono inteligente.

El diseño o la selección de los algoritmos se llevó a cabo a través de patrones y reglas que se identificaron a partir de uno o más de los siguientes casos: análisis de los datos recolectados por el dispositivo móvil en el proceso de experimentación; parámetros o variables contemplados en la norma; parámetros o variables que caracterizan los diferentes fenómenos físicos que intervienen mientras el vehículo se desplaza; algoritmos, técnicas o parámetros reportados por la literatura.

En este capítulo se presentan los siguientes tópicos: selección de eventos, selección de sensores, adquisición de datos para fase de pruebas, algoritmos base y un modelo para la detección automática de eventos de riesgo en el manejo de un automóvil utilizando un teléfono inteligente y la plataforma FIWARE

4.1 Selección de los eventos y los sensores

En esta sección se describen los aspectos generales relacionados con la selección de los eventos y la adquisición de los datos.

4.1.1 Selección de eventos

La selección de los eventos se realizó tomando en cuenta dos hechos importantes: el primero es la vinculación del CENIDET con el proyecto SmartSDK en el escenario SmartSDK Security y el segundo está relacionado con los datos estadísticos reportados por el centro de experimentación y seguridad vial de México (CESVI) y el instituto nacional de estadística y geografía (INEGI).

En el primer caso, una de las tareas asignadas a la institución en el marco del proyecto consistía en la detección de eventos de riesgo al momento de conducir un automóvil en un estacionamiento por medio de la utilización de la plataforma FIWARE y un teléfono inteligente. Lo eventos que se debían detectar son: velocidad no autorizada, desplazamiento en contrasentido y paradas repentinas.

En el segundo caso, al revisar el estudio realizado por el CESVI en el 2015[13] y el informe presentado por el INEGI en 2017 [38] se observó que las causas ubicadas en los primeros lugares son: velocidad excesiva, no guardar distancia, etc. ver similares figura 2. De esta manera, en este trabajo de investigación se decidió que los eventos de interés para analizar son: velocidad no permitida, desplazamiento en sentido contrario, cambios de velocidad, irregularidades en la vía y giros laterales

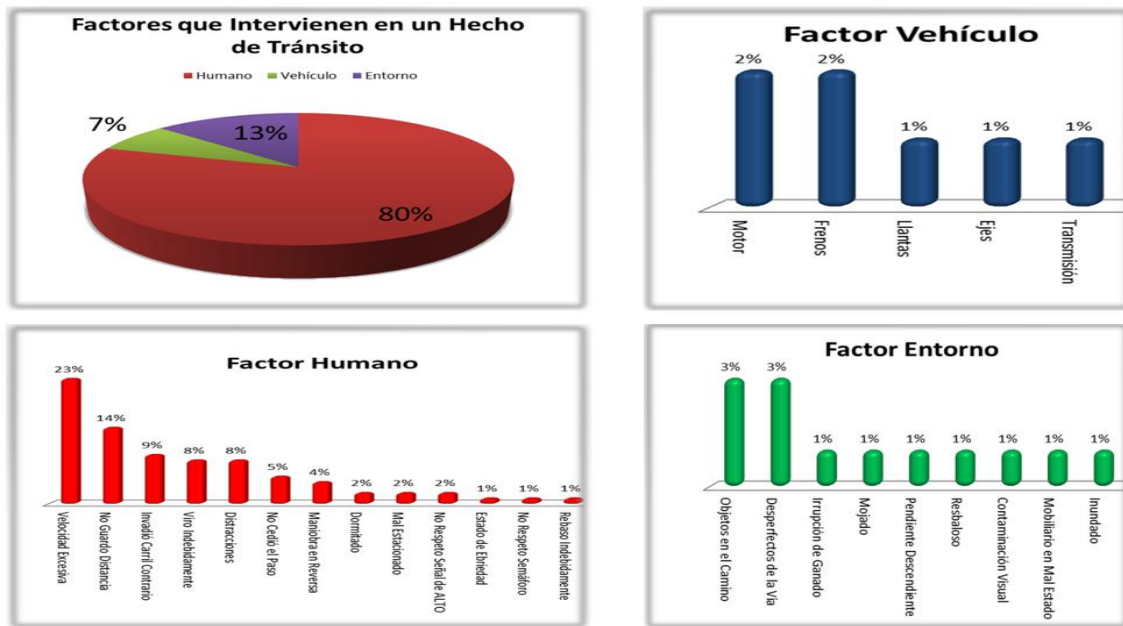


Figura 2. Estadísticas de los resultados obtenidos en la investigación realizada por el CESVI México. Fuente[13] parcialmente modificada .

4.1.2 Selección de los sensores

Los sensores seleccionados como fuente de datos para la selección de los algoritmos base enfocados en la detección de eventos de riesgo al momento de conducir un automóvil son: acelerómetro y GPS. La selección de los sensores se realizó a partir del conocimiento adquirido durante la revisión del estado del arte. En la Tabla 2 se muestra el listado de eventos seleccionados y el sensor que se utiliza para la obtención de los datos.

Tabla 2. Relación entre causas generadoras de accidentes y fuente para la adquisición de los datos

Relación entre las causas por las que se puede generar un accidente de tránsito y los sensores sobre que se genera la señal ante la ocurrencia del evento			
N°	Evento Seleccionados	Fuente	Observaciones
1	Exceso de velocidad	GPS	Este evento se detecta basado en la velocidad reportada por el GPS y la velocidad establecida para el segmento en que se desplaza el vehículo.
2	Giro indebido	Acelerómetro	Este evento solo determina si un movimiento a izquierda o derecha puede ser considerado como brusco, de acuerdo con los parámetros previamente establecidos.
3	Desperfectos en la vía	Acelerómetro	Este movimiento está asociado con el eje del movimiento vertical del vehículo.
4	Conducción en sentido contrario en vías de único sentido	GPS	Este evento se detecta con base en las coordenadas adquiridas del GPS.
5	Detenciones del vehículo y cambios de velocidad	GPS	Este evento se corresponde con el eje de desplazamiento longitudinal del vehículo.

4.1.3 Adquisición de los datos de prueba

La adquisición de datos de prueba se enfoca en la obtención de un conjunto de datos, que pueda ser utilizado para el análisis y diseño de los algoritmos. Para la recolección de los datos se desarrolló un prototipo en Android que permite obtener los datos provenientes de los sensores del teléfono inteligente. Los sensores utilizados para recolectar los datos son: GPS, acelerómetro, giroscopio y magnetómetro. Los datos del giroscopio y el magnetómetro se recolectan por si se desean complementar los algoritmos base en trabajos futuros. El prototipo desarrollado también permite etiquetar manualmente la ocurrencia de un evento, las categorías y opciones de etiquetado se muestran en la Tabla 3.

Para la recolección de los datos se colocó el teléfono en una posición semifija dentro del automóvil y se realizaron varios recorridos en los que se etiquetaba la ocurrencia de los eventos. Los datos fueron almacenados en archivos planos, en los cuales cada línea representa una lectura registrada por alguno de los sensores. Cada registro almacenado dentro del archivo plano está compuesto por los siguientes datos: identificador del registro, aceleración en el eje x, aceleración en el eje y, aceleración en el eje z, giro sobre el eje x, giro sobre el eje y, giro sobre el eje z, magnetómetro en el eje x, magnetómetro en el eje y, magnetómetro en el eje z, latitud, longitud, altitud, velocidad registrada por el GPS, precisión horizontal del GPS, Marca de tiempo en milisegundos, tipo de evento, tipo de desplazamiento, área del evento, tipo de espacio en el que se genera el desplazamiento, fecha y hora en la que se toma el registro.

Tabla 3. Categorías y eventos de etiquetado por cada subcategoría

Categoría	Opciones de etiquetado por categoría
Tipo de evento	Tope de metal, tope de asfalto, bache, parada, cambio de carril a la derecha, cambio de línea a la izquierda, giro a la izquierda, giro a la derecha, camino irregular
Medio de desplazamiento	Reposo, caminando, auto, ruta, autobús, moto, bicicleta, corriendo
Zona de desplazamiento	Urbana, rural, ciudad, pueblo, municipio, residencial, comercial, escolar
Tipo de vía	Pista, estatal, inter estatal, municipal, escaleras, circuito cerrado, andén

Además, los datos adquiridos con el prototipo se complementaron con un repositorio externo, el cual fue creado por [6]. Este repositorio externo cuenta con 500 ejemplos de irregularidades en la vía. El repositorio propio se construyó a partir de trabajo de campo y un prototipo de software creado para tal fin. En la Figura 3 se muestra la imagen del prototipo desarrollado. Así mismo, se muestran algunas imágenes relacionadas con uno de los análisis que se realizó sobre los datos contenidos en uno de los archivos tomados al azar.

4.1 Algoritmo base para la detección de desplazamiento a una velocidad no autorizada

El algoritmo base para la detección de desplazamiento a una velocidad no autorizada permite determinar si un automóvil se desplaza a una velocidad que se encuentra por fuera de los límites permitidos.

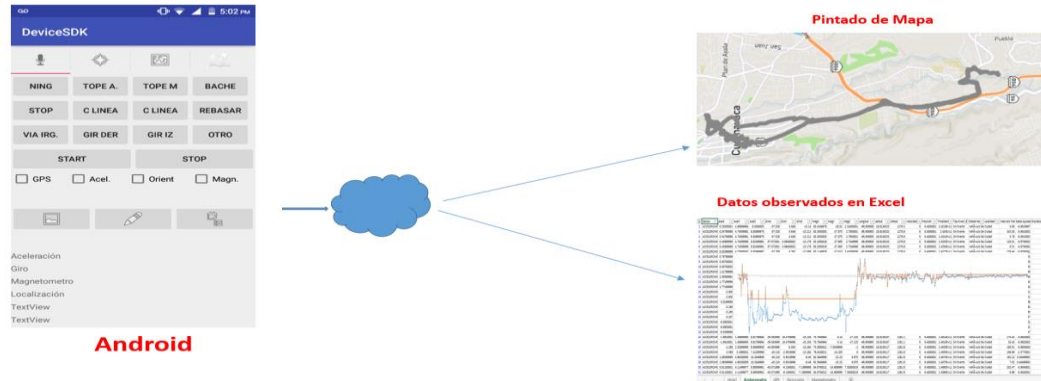


Figura 3. Imágenes del prototipo y la visualización de los datos a partir del csv generado. Fuente: elaboración propia.

El límite de velocidad es un concepto que se utiliza para referirse a los valores que permiten determinar si un automóvil se desplaza a una velocidad permitida de acuerdo con el contexto. El límite de velocidad está compuesto por un límite inferior y un límite superior que representan la velocidad mínima y máxima permitida respectivamente. La velocidad máxima está relacionada con el límite superior y se refiere a un valor limite a partir del cual se asume que el automóvil se desplaza a una velocidad que sobrepasa el valor máximo establecido. La velocidad mínima está relacionada con el límite inferior y se refiere a un valor limite a partir del cual se determina que el automóvil se desplaza a una velocidad por debajo del valor mínimo establecido.

El seudo código del algoritmo base es el siguiente:

```

velNoAutorizada (velActualAutomovil, posActualAutomovil, velMinAut, velMaxAut, [tolerancia=0])
  Si ( velMinAut > 0 ^ velMinAut > 0))
    Si ( velMinAut - tolerancia > velActualAutomovil v velMaxAut < velActualAutomovil +
tolerancia)
      Retornar objeto ["Vel. No Autorizada", posActualAutomovil, marca tiempo]
    Sino
      Retornar falso
  Sino
    Si (velMaxAut > 0)
      Si ( velMaxAut + tolerancia < velActualAutomovil)
        Retornar objeto ["Vel. No Autorizada", posActualAutomovil, marca tiempo]
      Sino
        Retornar falso
    Sino
      Si ( velMinAut - tolerancia > velActualAutomovil)
        Retornar objeto ["Vel. No Autorizada", posActualAutomovil, marca tiempo]
      Sino
        Retornar falso

```

4.2 Algoritmo base para la detección de desplazamiento en contrasentido

Este algoritmo permite determinar si un automóvil se desplaza en sentido contrario, para lo cual se utilizan principios de geometría. Inicialmente se parte de la premisa, que un

automóvil se desplaza por un segmento, el cual tiene un punto inicial y un punto final. El segmento tiene claramente definido un sentido de desplazamiento único, el cual va desde el punto inicial del segmento hasta el punto final del segmento. Dado que durante el desplazamiento del automóvil se están tomando sus coordenadas GPS, es posible determinar si el automóvil se acerca, se aleja del punto final o si está en el mismo punto. En el caso que el automóvil se aleje del punto final, entonces el automóvil se está desplazando en sentido contrario. El pseudo código del algoritmo base es el siguiente:

```

despSentidoContrario (posActualAutomovil, posAnteriorAutomovil, posInicialSegmentoVia,
posFinalSegmentoVia, [tolerancia=0])

tramo= crearLinea(posInicialSegmentoVia, posFinalSegmentoVia)

si( puntoEnLinea (posActualAutomovil,tramo, tolerancia)  $\wedge$  puntoEnLinea (posAnteriorAutomovil, tramo,
tolerancia))
    distanciaActual=distanciaEntrePuntos (posActualAutomovil, posFinalSegmentoVia)
    distanciaAnterior= distanciaEntrePuntos (posAnteriorAutomovil, posFinalSegmentoVia)
    si (distanciaActual< distanciaAnterior)
        retornar 1
    si(distanciaActual > distanciaAnterior)
        retornar objeto ["desplazamiento en sentido contrario", posActualAutomovil, marca tiempo]
    sino
        retornar 0
sino
    retornar 0

```

4.3 Algoritmo para la detección de cambio de velocidad y frenado

La detección del cambio de velocidad o el frenado, se determina a partir de los datos obtenidos en las dos últimas lecturas recibidas del GPS. Para los cálculos se utiliza la posición y la velocidad registrada por el GPS. Además, en los cálculos se utilizan los siguientes parámetros: coeficiente de fricción, el tiempo de percepción del conductor, el tiempo de reacción del conductor y el grado de inclinación. El pseudo código del algoritmo base es el siguiente:

```

cambioVelocidad (velActualAutomovil, posActualAutomovil, velAnteriorAutomovil, posAnteriorAutomovil,
[coeficienteFriccion=.675], [tiempoPercepcion=0.75], [tiempoReaccion=0.75], [gradoInclinación=0],
[tolerancia=0])

distanciaTotalRecorrida= distanciaEntrePuntos (posActualAutomovil, posAnteriorAutomovil)
distanciaReaccion = velAnteriorAutomovil * tiempo Reacción
distanciaPercepcion = velAnteriorAutomovil * tiempoPercepcion
distanciaFrenado=(Math.pow(velAnteriorCuerpo,2) * gradoInclinación) / (2* coeficienteFriccion *9.8)
distanciaTolerancia= (distanciaReaccion + distanciaFrenado+ distanciaPercepcion) * tolerancia;
distanciaMinimaAceptable= (distanciaReaccion + distanciaFrenado+ distanciaPercepcion -
distanciaTolerancia)
si (distanciaMinimaAceptable< distanciaTotalRecorrida)
    retornar objeto ["cambio de velocidad fuera de los parámetros", posActualAutomovil, marca
tiempo]
    sino
        retornar -1

```

4.4 Algoritmo base para la detección de eventos basados en reconocimiento de patrones

Este algoritmo está diseñado para la detección de eventos utilizando técnicas de reconocimiento de patrones con la ayuda de una herramienta externa. La razón para considerar una herramienta externa es que algunos teléfonos inteligentes no poseen la capacidad de computo que algunas de las técnicas de reconocimiento de patrones precisan. Aunque este algoritmo es flexible, depende de un vector de características previamente definido y un punto de acceso capaz de clasificar un vector de características de acuerdo con un determinado modelo. Los pasos básicos del algoritmo son los siguientes:

1. Se inicia el proceso de adquisición de datos del acelerómetro o el GPS.
2. Se generan vectores de características de acuerdo con el modelo indicado, de forma indefinida hasta que termine el proceso de adquisición de datos.
3. Se envía cada vector de características y el nombre del modelo al punto de acceso externo para que realice el proceso de clasificación.
4. Se recibe la respuesta del punto de acceso externo.
5. Finalmente, se envía una notificación en el caso de que la respuesta corresponda a un evento.

Las características específicas del algoritmo son las siguientes:

- Una vez que se inicia el proceso de adquisición de datos, este se ejecuta indefinidamente hasta que se decida terminar el proceso.
- Los datos pueden ser filtrados para suavizar el ruido, la técnica utilizada es la de medias móviles.
- El vector de características puede contener traslape, lo que significa que contiene los últimos registros del vector de características anterior, según se le indique.

Los pasos detallados del algoritmo se describen a continuación.

1. Se crea una variable p , que establece el valor de la ventana para suavizar la señal, un valor de p menor o igual a uno indica que no se suavizara la señal.
2. Se crea una estructura A de tipo cola que almacenara los últimos p datos recibidos del sensor, la cual se utilizara para suavizar la señal.
3. Se crea una variable suma que almacenara la suma total de los elementos contenidos en A .
4. Se crea una variable n que representa el número de datos de traslape de los datos, un valor inferior a uno indica que el vector de características no tiene datos traslapados.
5. Se crea una estructura B de tipo cola que almacena los datos de traslape y contiene los últimos n datos adquiridos del sensor.
6. Se crea una variable k que indica el tamaño total del vector de características.
7. Se crea una estructura *vectorCaracteristicas* de tipo array de tamaño k que representa al vector de características.
8. Se crea una variable *dato*, que contiene el ultimo valor obtenido del sensor.

9. Se crea una variable *valorFiltrado* que contiene el valor que será pasado al vector de características.
10. Se crea la variable *modelo*, que contiene el nombre del modelo con el cual se desea evaluar el vector de características.
11. Se inicia la aplicación y se empieza el proceso de adquisición de datos.
12. Se obtiene el valor proveniente del sensor y se almacena en la variable *dato*.
13. El valor *dato* es adicionado a la estructura *B* hasta alcanzar el tamaño *n*. cuando se alcanza el tamaño *n*, se retira el primer elemento de la cola *B* y se adiciona el valor de *dato*. Este proceso se repite hasta que se finalice la ejecución de la aplicación.
14. El valor de *dato* es adicionado a la estructura *A* hasta que alcance el tamaño *p*. Además, se incrementa la variable *suma* en el valor *dato*.
15. Cuando tamaño de *A* es igual al valor *p*, se divide la variable *suma* entre *p* y se asigna a la variable *valorFiltrado*, la cual se pasa al array *vectorCaracteristicas*.
16. Se retira el primer elemento de la cola *A* y se resta su valor de la variable *suma*.
17. Se agrega *dato* a la estructura *A* y se incrementa la variable *suma* en el valor *dato*.
18. Se repiten los pasos del 13 al 17 durante la ejecución de la aplicación.
19. El vector *vectorCaracteristicas* es llenado hasta alcanzar el tamaño *k*.
20. Una vez el vector *vectorCaracteristicas* completa sus *k* elementos es pasado al bloque de detección de eventos y se borran todos los elementos del vector *vectorCaracteristicas*.
21. Se remueve cada elemento de la estructura *B* y se asignan a la estructura *vectorCaracteristicas* hasta llegar a la posición *n* de la estructura *vectorCaracteristicas*.
22. Se repiten los pasos 19 a 21 durante la ejecución de la aplicación
23. La estructura *vectorCaracteristicas* y la variable *modelo* son pasados al punto de acceso externo de reconocimiento de patrones.
24. El punto de acceso externo de reconocimiento de patrones clasifica el vector de características con base en el nombre del modelo proporcionado y retorna una respuesta.
25. Cada vez que la respuesta recibida corresponde a un evento de acuerdo con el modelo seleccionado, se genera una notificación.

El pseudocódigo es el siguiente:

```

entero p // tamaño de la ventana para suavizar el ruido.
cola A // estructura para almacenar los últimos p elementos.
double suma // suma total de los elementos de A.
entero n // número de elementos de traslape
cola B // estructura para almacenar los últimos n elementos de traslape.
entero k // tamaño total del vector de características
array vectorCaracteristicas // estructura de tamaño k para almacenar los elementos del vector de características.
double dato // variable para almacenar el último dato adquirido del sensor.
Cadena modelo // variable que contiene el nombre del modelo que se utilizara para evaluar C.
double valorFiltrado // Valor que será pasado al vector de características

// Ciclo infinito para recolectar los datos provenientes del sensor.

mientras(verdadero){
    dato = leerSensor() // datos provenientes del sensor
    si (B.tamaño<n)
        B.adicionar(dato)

```

```

sino
  B.removerPrimero()
  B.adicionar(dato)

A.adicionar(dato)
si (A.tamaño<p)
  suma+=dato
sino
  valorFiltrado =suma/n
  llenarVectorCaracteristicas(valorFiltrado)
  suma -= A.removerPrimero()
  A.adicionar(dato)
  suma += dato
}
llenarVectorCaracteristicas(valorFiltrado){
  C.adicionar(valorFiltrado)
  si C.tamaño>=k
    detectarEvento (vectorCaracteristicas)
    C.vaciar()
  si n > 0
    entero i = 0
    mientras (B.tamano > 0)
      C[i] = B.removerPrimero()
      i += 1
}
detectarEvento(vectorCaracteristicas){
  respuestaEvento= llamarPuntoAccesoExterno(vectorCaracteristicas, modelo)
  si respuestaEvento != 'ninguno'
    enviarNotificacion() //
}

```

4.5 Algoritmo base para la detección de giros laterales

Este algoritmo permite identificar si un vehículo gira a la derecha o a la izquierda, basado en los datos provenientes del eje del acelerómetro que es transversal a los lados laterales del vehículo. La intensidad del giro se determina a partir de las lecturas realizadas por el acelerómetro en dicho eje y los valores de referencia obtenidos en la literatura. El pseudo código del algoritmo base es el siguiente:

```

giroLateral (posActualAutomovil, aceleracionLateral, listaValorReferencia)
esGiro=falso
Para categoria en listaValorReferencia
  Si aceleracionLateral pertenece categoria
    retornar objeto [categoria.clasificacion, posActualAutomovil, marca tiempo]
    esGiro=verdadero
fin para
si (esGiro)
  retornar objeto [categoria.clasificacion, posActualAutomovil, marca tiempo]
sino
  retornar 0

```

Lista valores de referencia para los giros tomando como unidad base la gravedad (g).

clasificación	Izquierda (+x)	Derecha(-x)	Enviar notificación
Ruido Blanco	$\geq 0g$ Y $< 0.1g$	$\geq 0g$ Y $< -0.1g$	No
Normal	$\geq 0.1g$ Y $< 0.3g$	$\geq -0.1g$ Y $< -0.3g$	No
Moderado	$\geq 0.3g$ Y $< 0.6g$	$\geq -0.3g$ Y $< -0.6g$	Sí
Fuerte	$\geq 0.6g$ Y $< 1g$	$\geq -0.6g$ Y $< -1g$	Sí
Muy Fuerte	$\geq 1g$	$\geq -1g$	Sí

4.6 Algoritmo base para detectar irregularidades en la vía

Este algoritmo permite identificar si existe una irregularidad en la vía basado en los datos provenientes del eje transversal a la vía por la que se desplaza el vehículo. En este trabajo solo se detecta si existe o no una irregularidad en la vía. La pronunciación de la irregularidad se clasifica a partir de las lecturas obtenidas por el sensor. Una lectura muy alta significa que la irregularidad es muy pronunciada o que el vehículo la cruzó de manera abrupta, ya sea, por la velocidad a la que se desplazaba o porque ingresó en ella de manera repentina, como puede ser el caso de los baches en días lluviosos. El pseudo código del algoritmo base es el siguiente:

```

giroLateral (posActualAutomovil, aceleracionVertical, listaValorReferencia)
esIrregularidad=falso
Para categoria en listaValorReferencia
    Si aceleracionVertical pertenece categoría
        retornar objeto [categoría.clasificacion, posActualAutomovil, marca tiempo]
        esIrregularidad =verdadero
fin para
if esIrregularidad
    retornar objeto [categoría.clasificacion, posActualAutomovil, marca tiempo]
sino
    retornar 0
    
```

Lista valores de referencia tomando como unidad base la gravedad (g).

Clasificación	Izquierda (+x)	Derecha(-x)	Enviar notificación
Ruido Blanco	$\geq 0g$ Y $< 0.2g$	$\geq 0g$ Y $< -0.2g$	No
Leve	$\geq 0.2g$ Y $3g$	$\geq -0.1g$ Y $< -0.3g$	No
Media	$\geq 0.3g$ Y $.6g$	$\geq -0.1g$ Y $< -0.3g$	Sí
Pronunciada	$\geq 0.6g$ Y $< 1g$	$\geq -0.6g$ Y $< -1g$	Sí
Muy Pronunciada	$\geq 1g$	$\geq -1g$	Sí

4.7 Modelo para la detección automática de eventos de riesgo en el manejo de un automóvil utilizando un teléfono inteligente y la plataforma FIWARE

En esta sección se describe el prototipo desarrollado a partir de los algoritmos base y técnicas seleccionadas para la detección automática de eventos de riesgo en el manejo de un automóvil. Esta sección contiene tres subsecciones: Arquitectura, módulo de detección de eventos y, módulo de monitoreo y visualización.

4.7.1 Arquitectura

En la Figura 4 se muestra la arquitectura general del prototipo para la detección automática de eventos de riesgo en el manejo de un automóvil utilizando la plataforma FIWARE, en el cual se implementan los algoritmos presentados en las secciones anteriores. El prototipo desarrollado está compuesto por dos módulos. El primer módulo es detección de eventos, el cual se ejecuta en el teléfono. El segundo módulo es monitoreo y visualización, el cual se ejecuta en la plataforma FIWARE.

4.7.2 Módulo de detección de eventos (Teléfono)

Este módulo está compuesto por dos componentes, ver Figura 4. El objetivo principal del módulo es detectar si se ha presentado la ocurrencia de uno de los siguientes eventos: velocidad no permitida, desplazamiento en contrasentido, cambios de velocidad, irregularidades en la vía y giros laterales. Para lograr este objetivo, el módulo cuenta con un componente que permite la manipulación y parametrización de la aplicación y otro que se encarga de la detección automática de los eventos.

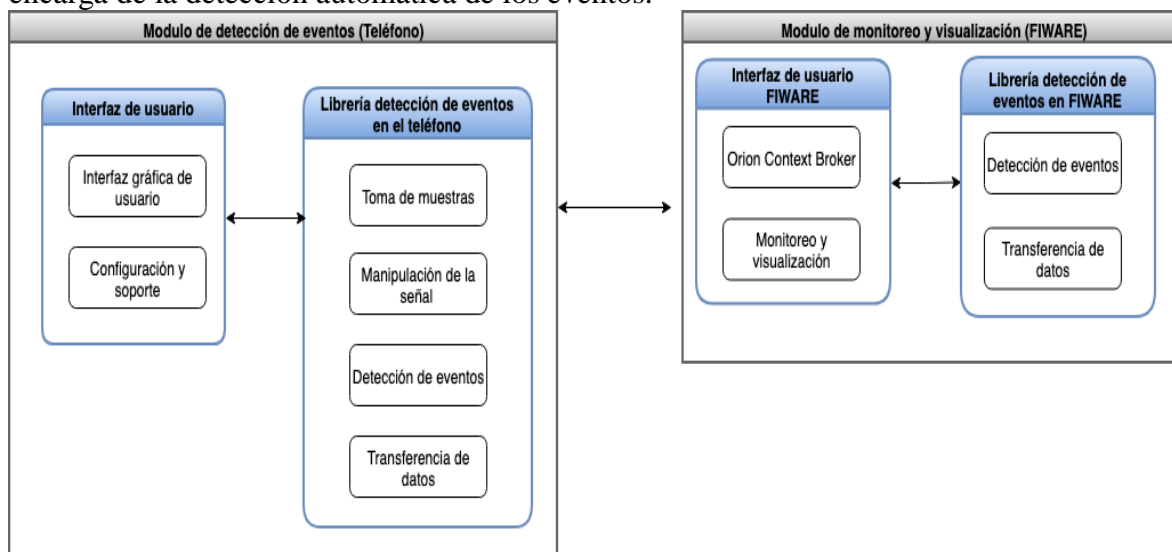


Figura 4. Arquitectura general del prototipo. Fuente: elaboración propia.

El módulo detección de eventos se ejecuta al lado del teléfono y está desarrollado en la versión 4.0 de Android, con un patrón de diseño modelo vista controlador. Todas las funcionalidades a excepción de la interfaz gráfica están desarrolladas en Java nativo e

incluidas en una librería única para facilitar su utilización en entornos diferentes a Android. Los sensores incluidos en esta versión son el acelerómetro, giroscopio o sensor de orientación, el magnetómetro y el GPS. A continuación, se describe cada uno de los componentes del módulo de detección de eventos.

4.7.2.1 *Componente interfaz gráfica de usuario*

Este componente está formado por dos bloques: Interfaz gráfica de usuario y, configuración y soporte. En las subsecciones siguientes se describe cada bloque.

Bloque de interface gráfica de usuarios. Este bloque está compuesto por un conjunto de vistas que permiten la interacción del usuario con la aplicación. En la Figura 5 se muestra la vista de la pantalla principal y algunas vistas de parametrización. Las vistas de parametrización permiten que el usuario ingrese los parámetros que la aplicación debe tener en cuenta al momento que se inicia el proceso de detección de eventos. Otra de las vistas importantes que tiene la ampliación es la de recolección de datos, la cual permite seleccionar los sensores a utilizar, inicializar el proceso de recolección de datos y etiquetar la ocurrencia de un evento. Como se puede observar en la Figura 6 , esta vista cuenta con diferentes opciones de etiquetado por categoría.

Bloque de configuración y soporte: Este bloque se encarga de gestionar los parámetros de configuración establecidos por el usuario en las vistas de parametrización. Además, gestiona todo lo relacionado con la información solicitada por la interfaz gráfica del usuario, los modelos de datos y algunas utilidades requeridas por los demás bloques.

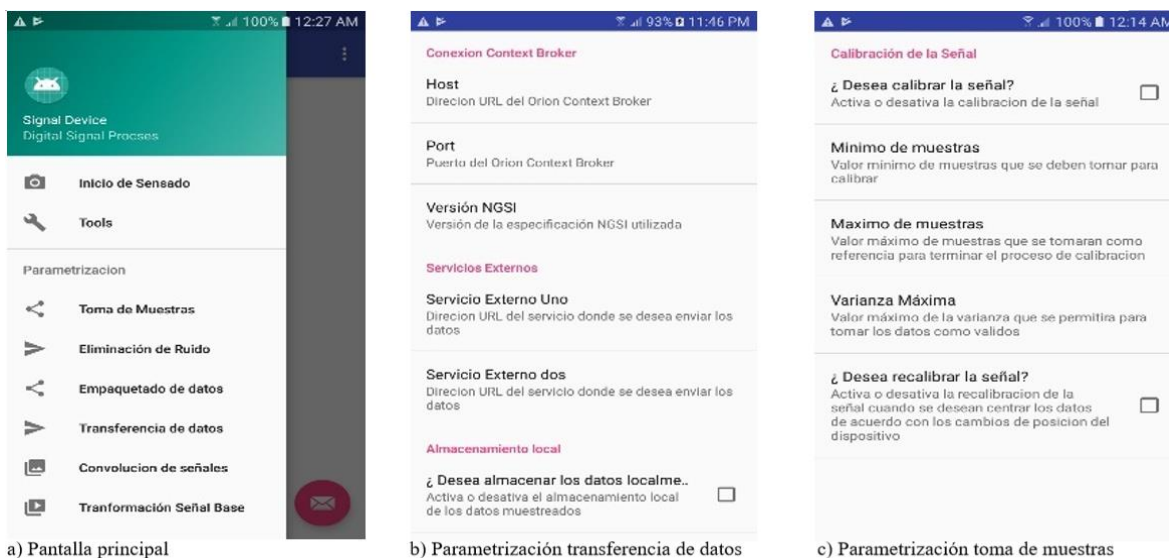


Figura 5. Opciones de parametrización que permite el módulo del teléfono inteligente. Fuente: elaboración propia.

4.7.2.1 *Componente detección de eventos*

Este componente es una librería que está compuesta por cuatro bloques: toma de muestras, manipulación de la señal, detección de eventos y transferencia de datos. Este es el componente más importante del módulo porque es donde se lleva a cabo todas las tareas

necesarias para la detección de los eventos. En los apartados siguientes se describe cada bloque.

Bloque de toma de muestras. Este bloque es el encargado de obtener todas las señales provenientes de los sensores del teléfono. En este componente se realizan dos procesos importantes: obtención de la señal y calibración. El proceso de toma de señal se inicializa una vez que el usuario lo indica y que el proceso de calibración indica que el teléfono se encuentra en una posición válida para empezar la toma de las señales. Una posición será válida cuando la varianza máxima del tipo de señal a obtener se encuentra dentro de los parámetros establecidos en el componente de configuración. La calibración puede además ser omitida, si el usuario lo considera pertinente, ver la Figura 5.

Bloque de manipulación de la señal. Este bloque se encarga de transformar las señales tomadas de los sensores. Su objetivo es transformar la señal que será entregada al bloque de detección de eventos, de acuerdo con la parametrización establecida por el usuario. Algunas de las tareas que se realizan en este bloque son las siguientes:



Figura 6. Categorías y opciones de etiquetado de la vista toma de muestras. Fuente: elaboración propia.

- Centrado de la señal: esta función se encarga de tomar el dato proveniente del bloque de toma de señales y centrarla con base en un valor de referencia, el cual puede ser proporcionado por el usuario o centrado teniendo en cuenta la media de los datos calculada en el proceso de calibración.
- Reducción de ruido: esta funcionalidad permite reducir el ruido de la señal, el método de medias móviles es el utilizado para la reducción del ruido. En la Figura 7 se muestran diferentes medias móviles aplicada a una misma señal base. En este caso se utilizó el repositorio externo, una característica importante de este reportorio es que todos los ejemplos son de tamaño diferente, por lo que no se pueden pasar a un algoritmo de clasificación directamente. En la Figura 7 se muestra la gráfica para el ejemplo de la clase camino irregular, al cual se le aplicaron las siguientes medias móviles: MA(5), MA (10), MA(15), MA(20), MA(25), MA(30), MA(40), MA(50), MA(75), MA(100). Como se puede observar en la Figura 7, a medida que aumenta el valor de la media móvil, los picos de señal van desapareciendo. La utilización de esta técnica permite tener una representación más limpia de la señal y facilita el trabajo con los datos provenientes del acelerómetro. Además, permite la disminución del ruido blanco que se genera producto de la superficie de desplazamiento y los diferentes sistemas del vehículo.

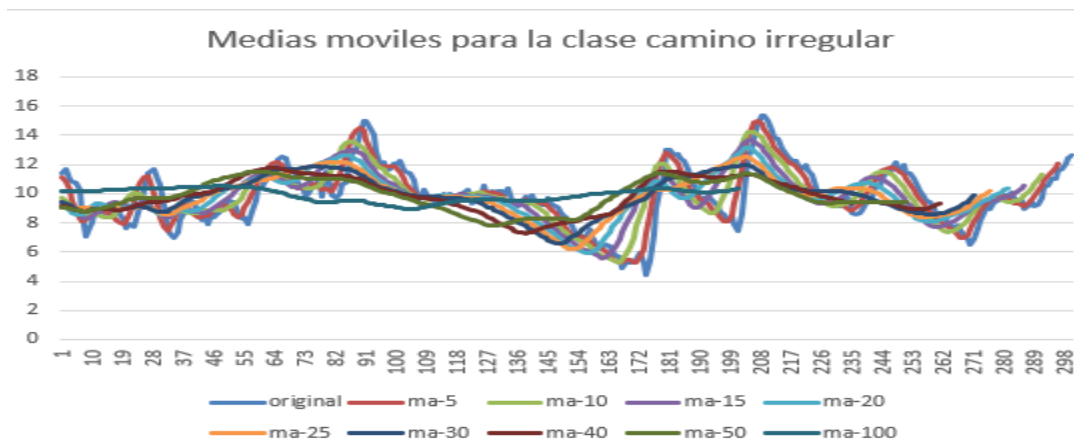


Figura 7. Representación de una misma señal al aplicar el filtro de medias móviles. Fuente: elaboración propia.

- Diezmado e interpolación de la señal: Esta función permite aplicar diferentes valores de diezmado e interpolación a una señal base, para realizar este proceso el usuario debe establecer en las opciones de parametrización, el número de señales derivadas que desea obtener y los valores de interpolación o diezmado para cada señal.
- Empaquetado de la señal: Esta funcionalidad permite que los datos sean enviados de uno en uno, o que el usuario seleccione el número de datos que tendrá una muestra.
- Ventanas de tiempo: Esta opción permite manejar diferentes ventanas de tiempo, ya sea para el cálculo de medias móviles, para el empaquetado de datos o para el uso en funciones que dependen de valores anteriores o futuros para el cálculo del valor actual.

Bloque de detección de eventos: En este bloque se realizan todas las tareas relacionadas con la identificación o clasificación de los eventos. Su objetivo principal es determinar si se presentó la ocurrencia de un evento y generar una notificación en el caso de detectar algún evento. En este bloque se realizan cuatro procesos: recepción de datos, identificación de los eventos, identificación del área de ocurrencia del evento y notificación de la ocurrencia de un evento.

- El proceso de recepción de datos se encarga de distribuir los datos a los diferentes algoritmos de detección. los datos se distribuyen teniendo en cuenta el sensor del cual provienen los datos, por ejemplo, los datos provenientes del GPS son pasados a los algoritmos de exceso de velocidad, desplazamiento en la señal y paradas repentinas.
- El proceso de identificación de eventos se encarga de ejecutar el algoritmo seleccionado en el paso anterior y determinar si se presentó la ocurrencia de un evento. En el caso que se identifique la ocurrencia de un evento se pasa al siguiente proceso, en caso contrario, no se realiza ningún otro proceso.
- El proceso de identificación área de ocurrencia del evento sólo se realiza cuando se ha detectado un evento, su objetivo principal es obtener información sobre el área en la que se presentó dicho suceso. Una vez que se detecta el área, se continua con el proceso de notificación de eventos.
- El proceso de notificación de la ocurrencia de un evento se encarga de empaquetar toda la información obtenida en los procesos anteriores y la pasa al bloque de transferencia de datos.

Bloque de transferencia de datos: Este bloque permite la comunicación entre el módulo de detección de eventos y el módulo de monitoreo y visualización. La función principal de este componente es la gestión del almacenamiento y distribución de los datos. El componente empaqueta los datos en objetos (modelos de datos) de tipo JSON basados en la especificación NGSI de FIWARE. En este trabajo se propone un nuevo modelo de datos bajo la especificación NGSI, al cual hemos llamado *SignalDevice*. Este es un modelo de datos flexible que permite el envío de una o varias señales. En la Figura 8 se muestra la estructura del objeto JSON del modelo de datos propuesto. Este modelo de datos se caracteriza por tener una estructura básica que identifica de manera única al tipo de señal que se está procesando. Los campos id, type, refDevice, timestamp, dateModifieds y SD1 son de tipo obligatorio. La estructura SD representa la señal que se está procesando, siempre se debe identificar si se trata de la señal original o se trata de una señal derivada.

4.7.3 Módulo de monitoreo y visualización (FIWARE).

El módulo de monitoreo y visualización es el encargado de permitir la visualización de los datos provenientes del módulo de detección de eventos. Este módulo está compuesto por dos componentes: interfaz de usuario en FIWARE y librería para la detección de eventos. En el componente de interfaz de usuario en FIWARE se utilizan dos módulos genéricos, los cuales hacen parte de la plataforma de FIWARE (Orion Context Broker y Quantum Leap). El componente de la librería para detección de eventos en FIWARE está desarrollado en Python. Una de las razones para seleccionar el lenguaje de programación Python es que cuenta con un gran número de herramientas, las cuales facilitan la ejecución de actividades

relacionadas con el reconocimiento de patrones, machine learning, procesamiento de señales, entre otros. En este apartado se describen los elementos que conforman el interfaz de usuario FIWARE.

Módulo Orion Context Broker. Este es un módulo genérico de la plataforma FIWARE. El objetivo de este módulo genérico en este trabajo de investigación es recibir la información de contexto generada por el módulo de detección de eventos del teléfono y enviarla al módulo genérico QuantumLeap. Para la transferencia de información entre los diferentes elementos del prototipo se utilizan los modelos de datos Device [39], Alert [40] y SignalDevice. Los dos primeros modelos de datos hacen parte de la especificación NGSI y el último es un nuevo modelo de datos que se propone en este trabajo de investigación.

Monitoreo y visualización. Esta es la etapa final del prototipo y su principal objetivo es permitir que el usuario visualice los eventos detectados o las gráficas de las señales que se están generando desde las diferentes fuentes de toma de señales que se están utilizando. Para la visualización de las gráficas se utiliza el módulo genérico QuantumLeap, el cual permite trabajar con series de tiempo y visualizar sus graficas en una interfaz WEB.

```

{
  "type": "SignalDevice",
  "id": "SD_E",
  "refDevice": {
    "value": "Device_Prueba_1",
    "type": "Text"
  },
  "SD-Acel_Y": {
    "value": 9.8,
    "metadata": {
      "description": {
        "value": "Base signal"
      }
    }
  },
  "SD-Acel_X": {
    "value": 0.8,
    "metadata": {
      "description": {
        "value": "Base signal"
      }
    }
  },
  "SD-Acel_Z": {
    "value": 0.35,
    "metadata": {
      "description": {
        "value": "Base signal"
      }
    }
  },
  "timestamp": {
    "value": 1523804024221,
    "type": "Integer"
  },
  "dateModified": {
    "value": "2018-04-15T07:21:24.285Z",
    "type": "DateTime"
  }
}

{
  "type": "SignalDevice",
  "id": "SD_E",
  "refDevice": {
    "value": "Device_Prueba_1",
    "type": "Text"
  },
  "SD1": {
    "value": 9.8,
    "metadata": {
      "description": {
        "value": "Base signal"
      }
    }
  },
  "SD2": {
    "value": 0.8,
    "metadata": {
      "description": {
        "value": "Centered signal"
      }
    }
  },
  "timestamp": {
    "value": 1523804024221,
    "type": "Integer"
  },
  "dateCreated": {
    "value": "2018-04-15T07:21:24.285Z",
    "type": "DateTime"
  },
  "dateModified": {
    "value": "2018-04-15T07:21:24.285Z",
    "type": "DateTime"
  }
}

```

Figura 8. Ejemplo de instancia del modelo de datos SignalDevice. Fuente: elaboración propia.

4.7.3.1 *Componente de la librería detección de eventos en FIWARE*

Este componente tiene un comportamiento similar al componente de detección de eventos del teléfono, su diferencia principal es que fueron desarrollados para ejecutarse en la plataforma FIWARE. la razón por la cual no se incluyen los bloques de toma de muestra y manipulación de la señal es porque estos procesos son ejecutados en el módulo de detección de eventos alojado en el teléfono inteligente.

Capítulo 5. Pruebas y resultados

En esta sección se presenta una descripción de las pruebas realizadas y los resultados obtenidos. La fase de pruebas se realizó por un periodo de 14 meses. Durante este periodo se realizaron más de 100 recorridos en zonas rurales y urbanas; y se recolectaron más de 50 millones de registros provenientes del acelerómetro, el GPS, el magnetómetro y, el giroscopio o el sensor de orientación. En las subsecciones siguientes se presenta una descripción general de las pruebas realizadas y los resultados obtenidos. Esta sección está distribuida de la siguiente forma: primero se realiza una descripción de las pruebas y luego se presentan las pruebas y resultados obtenidos.

5.1 descripción de las pruebas

En esta subsección se presenta una descripción general de las pruebas realizadas para las cuales se utilizó el prototipo de software descrito en la sección 4.8. la estructura de la presente subsección es la siguiente: en primer lugar, se presentan los casos de estudio, después se listan los recursos utilizados y por último se describe el procedimiento realizado.

5.1.1 Casos de estudio

Las pruebas se realizaron con el objetivo de encontrar defectos en los algoritmos para identificar aspectos relacionados con el porcentaje de acierto en la detección de la ocurrencia de un evento. Los algoritmos desarrollados se enfocan en la detección de la ocurrencia de un evento relacionado con alguna de las causas relacionadas en la Tabla 4.

Tabla 4. Listado de técnicas utilizadas de acuerdo con la causa generadora de un accidente de tránsito de interés.

Nº	Causa	Técnica	Fuente de información
1	Velocidad no permitida	Principios matemáticos	GPS y normativa de tránsito
2	Giros laterales	Principios matemáticos	Acelerómetro
3	Desperfectos en la vía	Reconocimiento de patrones - Principios matemáticos	Acelerómetro
4	Conducción en sentido contrario en vías de único sentido	Técnicas matemáticas basadas en geometría analítica	GPS e y normativa de tránsito
5	Cambios de velocidad	Reconocimiento de patrones - Principios matemáticos	GPS y Acelerómetro

5.1.2 Recursos utilizados

Los recursos utilizados para la evaluación de nuestra aplicación fueron: tres teléfonos inteligentes (ZTE Blade V6, un Samsung Galaxy J4 y un Motorola Moto G5), una computadora portátil para Gamer ASUS (Core i7, 8GB, 1TB, NVIDIA GeForce GTX 1050)

y dos máquinas virtuales alojadas en la plataforma FIWARE Lab nodo México. Además, para la toma de muestras se utilizaron diferentes medios de transporte público y particular.

5.1.3 Procedimiento de pruebas

El procedimiento para la realización de las pruebas es el siguiente:

Lo primero que se hizo fue colocar el teléfono en una posición fija dentro de tal forma que los ejes del teléfono coincidieran en lo posible con los ejes de referencia del vehículo. En la Figura 9. Sistema de referencia utilizado con respecto al vehículo y el teléfono.



Figura 9. Sistema de referencia utilizado con respecto al vehículo y el teléfono. Fuente [6].

Luego se seleccionaron los sensores y una vez que el vehículo está en marcha se inicia manualmente, el proceso de toma de datos para la detección de eventos. Durante el recorrido, el copiloto etiqueta la ocurrencia de alguno de estos eventos: giros laterales, paradas e irregularidades en la vía; además los datos son guardados automáticamente en el formato indicado en la sección 4.1.3. Después de la toma de la señal se procedió con la limpieza, la transformación y el análisis de los datos. Finalmente, los datos fueron pasados a los diferentes algoritmos de detección.

5.1.4 Frecuencia de muestreo.

La frecuencia de muestreo es importante porque es la que indica cuantos datos puede tomar el sensor en un tiempo determinado, e influye directamente en el uso que puede darse a la información detectada en un determinado momento. El tiempo de muestreo de los sensores en los teléfonos inteligentes está limitado por el fabricante del sistema operativo, es así, como los dispositivos con sistema operativo Android permiten una frecuencia de muestreo teórica para el acelerómetro hasta de 200Hz y los dispositivos con sistema operativo IOS hasta de 100Hz. Para el desarrollo de este trabajo de investigación se utilizó la máxima frecuencia de muestreo permitida por el dispositivo. En la Tabla 5 se muestran los resultados promedio obtenidos.

Tabla 5. Frecuencia de muestreo por sensor utilizando un teléfono con SO Android

Sensor	Total de datos	Lectura más Rápida	Lectura más tardada	Promedio lectura	Frec. Muestreo Promedio
Acelerómetro	1´048.559	0.0015 seg	0.9527seg	0.0067 seg	149.24 Hz
GPS	43.143	0.0027 seg	651.9987 seg	1.2502 seg	0.8 Hz
Giroscopio	1´048.555	0.0011 seg	38.4544 seg	0.0061 seg	162.76 Hz
Magnetómetro	122.111	0.0013 seg	1.7964 seg	0.1314 seg	7.61 Hz

Los valores promedios obtenidos para la frecuencia de muestreo indican que, si bien es cierto que, la frecuencia máxima permitida de muestreo para el acelerómetro es de 200 Hz en el caso de los dispositivos *Android*, en la practica la frecuencia de muestreo está próxima a 150Hz, esto puede asociarse a varios factores como son la calidad del dispositivo y los procesos adicionales que el dispositivo estaba ejecutando en ese momento. Otro aspecto importante observado fue el relacionado con el espacio de tiempo que existe entre dato y dato. El cual puede variar por diversos factores. En la Figura 10 se presenta un fragmento de un archivo en el cual muestra que para el GPS la frecuencia de muestreo está muy cercana a 1Hz. Sin embargo, existen algunos momentos en los que este intervalo cambia por el comportamiento propio del GPS

9725	GPS	Sin Registro	Sat Mar 10 14:09:48 GMT+01:00 2018	Sin Registr	0.99996906
9726	GPS	Sin Registro	Sat Mar 10 14:09:49 GMT+01:00 2018	Sin Registr	1.0239274
9727	GPS	Sin Registro	Sat Mar 10 14:09:50 GMT+01:00 2018	Sin Registr	0.97561516
9728	GPS	Sin Registro	Sat Mar 10 14:09:51 GMT+01:00 2018	Sin Registr	1.03984771
9729	GPS	Sin Registro	Sat Mar 10 14:09:52 GMT+01:00 2018	Sin Registr	0.99204401
9730	GPS	Sin Registro	Sat Mar 10 14:10:06 GMT+01:00 2018	Sin Registr	13.9693825
9731	GPS	Sin Registro	Sat Mar 10 14:10:07 GMT+01:00 2018	Sin Registr	1.05412047
9732	GPS	Sin Registro	Sat Mar 10 14:10:08 GMT+01:00 2018	Sin Registr	0.94555302
9733	GPS	Sin Registro	Sat Mar 10 14:10:09 GMT+01:00 2018	Sin Registr	0.99861552
9734	GPS	Sin Registro	Sat Mar 10 14:10:10 GMT+01:00 2018	Sin Registr	1.01568109
9735	GPS	Sin Registro	Sat Mar 10 14:10:11 GMT+01:00 2018	Sin Registr	0.98496161
9736	GPS	Sin Registro	Sat Mar 10 14:10:12 GMT+01:00 2018	Sin Registr	1.00053115
9737	GPS	Sin Registro	Sat Mar 10 14:10:13 GMT+01:00 2018	Sin Registr	0.99932339
9738	GPS	Sin Registro	Sat Mar 10 14:10:14 GMT+01:00 2018	Sin Registr	1.00486479
9739	GPS	Sin Registro	Sat Mar 10 14:10:15 GMT+01:00 2018	Sin Registr	0.99595328
9740	GPS	Sin Registro	Sat Mar 10 14:10:16 GMT+01:00 2018	Sin Registr	0.9942026
9741	GPS	Sin Registro	Sat Mar 10 14:10:17 GMT+01:00 2018	Sin Registr	1.00902969
9742	GPS	Sin Registro	Sat Mar 10 14:10:18 GMT+01:00 2018	Sin Registr	0.9967799
9743	GPS	Sin Registro	Sat Mar 10 14:10:19 GMT+01:00 2018	Sin Registr	1.02234396
9744	GPS	Sin Registro	Sat Mar 10 14:10:20 GMT+01:00 2018	Sin Registr	0.977265
9745	GPS	Sin Registro	Sat Mar 10 14:10:23 GMT+01:00 2018	Sin Registr	3.00347026
9746	GPS	Sin Registro	Sat Mar 10 14:10:24 GMT+01:00 2018	Sin Registr	1.01690849
9747	GPS	Sin Registro	Sat Mar 10 14:10:33 GMT+01:00 2018	Sin Registr	8.98099755
9748	GPS	Sin Registro	Sat Mar 10 14:10:34 GMT+01:00 2018	Sin Registr	1.00179896
9749	GPS	Sin Registro	Sat Mar 10 14:10:35 GMT+01:00 2018	Sin Registr	1.0005251
9750	GPS	Sin Registro	Sat Mar 10 14:10:36 GMT+01:00 2018	Sin Registr	0.99591802
9751	GPS	Sin Registro	Sat Mar 10 14:10:37 GMT+01:00 2018	Sin Registr	1.004175

Figura 10. intervalo en segundos entre la toma de un dato y el siguiente. Fuente: elaboración propia.

5.1.5 Análisis inicial de los datos

El análisis inicial de los datos se realizó con el fin de verificar y limpiar los datos que serían tomados como referencia para el diseño de los algoritmos. La identificación y verificación de los eventos etiquetados se llevó a cabo de manera manual. Lo primero que se hizo fue tomar al azar algunos de los archivos planos obtenidos con el prototipo de adquisición de datos y cargarlos individualmente en Excel. Una vez que se cargaba el archivo se procedía a buscar los registros que habían sido etiquetados. En la Figura 11 se muestra extracto de uno de los archivos analizados en el cual se etiquetó la presencia de un tope.

1552	GPS	0.465	-2.94199991	9.29500008	4.46400023	46.4399986	-3.92400003	353.394989	15.5600004	2.375	-98.9452333	18.8429933	1322.6	15.8205023	6.80000019	1.8381E+12	Sin evento
1553	GPS	0.96899998	-2.32500005	10.2110004	11.9519997	58.0320015	-25.8479996	347.470001	13.4650002	2.20499992	-98.9453583	18.8430817	1322.6	15.4596815	6.90000001	1.8391E+12	Tope
1554	GPS	0.88800001	-3.8210001	9.12699986	7.38000011	49.2120018	-9.35999966	349.505005	17.7049999	5	-98.9454717	18.843165	1322.6	15.056654	7	1.8401E+12	Tope
1555	GPS	1.94200003	-0.98199999	10.1350002	5.796	45.5040016	-3.45600009	350.980011	4.2550011	7.5	-98.9455883	18.84325	1322.5	14.9219968	7.09999991	1.8411E+12	Tope
1556	GPS	0.46000001	1.87100005	13.0150003	4.35599995	43.8120003	-0.28799999	353.619995	-2.03999996	5.22499991	-98.9456767	18.843335	1322.5	14.3319254	7.19999981	1.8426E+12	Tope
1557	GPS	0.56099999	0.83200002	7.54899979	4.10400009	43.0919991	1.04400003	354.079987	-1.91499999	7	-98.9457517	18.8434133	1322.5	13.5305033	7.19999981	1.8432E+12	Tope
1558	GPS	0.148	0.324	10.3940001	3.77999997	42.2639999	1.69200003	354.630005	-0.20999999	2.25	-98.945845	18.8434933	1322.5	13.3174076	7.4000001	1.8442E+12	Tope
1559	GPS	-0.27700001	-1.097	10.8070002	3.81599999	41.723999	2.66400003	354.63501	8.26600023	-0.03	-98.9459533	18.84358	1322.4	12.9684267	7.4000001	1.8451E+12	Tope
1560	GPS	0.83600003	0.054	10.3999996	3.63599992	41.9760017	2.9519999	354.734985	3.14000011	0.685	-98.9460317	18.8436333	1322.4	11.8936863	7.19999981	1.8462E+12	Tope
1561	GPS	0.343	-0.875	10.0340004	3.77999997	41.9760017	3.27600002	354.725006	0.73000002	-1	-98.9461017	18.8436767	1322.3	10.8112259	7.09999991	1.8471E+12	Tope
1562	GPS	0.104	-0.113	8.73600006	3.77999997	41.6160011	2.77200007	354.350006	1.70000005	0.09	-98.9461733	18.843725	1322.3	10.2692232	7	1.8482E+12	Tope
1563	GPS	-0.41400001	-0.52999997	9.12699986	3.852	41.4360008	3.16799998	354.299988	1.87000001	-2.5	-98.9462467	18.8437767	1322.3	9.8945055	6.90000001	1.8492E+12	Tope
1564	GPS	-0.176	-0.50800002	10.007	3.81599999	41.0040016	3.09599996	354.11499	3.85999999	-1	-98.94633	18.8438417	1322.2	9.86928368	6.80000019	1.8502E+12	Tope
1565	GPS	-0.19	-0.755	10.5010004	3.81599999	41.4360008	2.66400003	354.350006	3.46499991	1.25	-98.9464167	18.8438983	1322.2	9.97274303	6.69999981	1.8512E+12	Tope
1566	GPS	-0.37200001	-0.91100001	9.40100002	3.92400003	41.2919998	2.91599989	354.154999	6.43499994	-1.35000002	-98.9465017	18.8439533	1322	10.1251011	6.80000019	1.8522E+12	Tope
1567	GPS	0.602	-0.76700002	10.5450001	3.99600005	41.2560005	2.59200001	354.059998	4.69000006	2.70000005	-98.9465867	18.84401	1321.9	10.304739	6.80000019	1.8532E+12	Tope
1568	GPS	0.375	-0.45899999	10.3500004	3.70799995	41.7960014	2.12400007	354.470001	2.80999994	3.45499992	-98.94668	18.844065	1321.8	10.4977598	6.80000019	1.8542E+12	Tope
1569	GPS	0.80299997	-0.80900002	10.7370005	4.53599978	42.7319984	-8.53199959	352.445007	2.43000007	0.07499981	-98.9467767	18.8441233	1321.6	10.7479143	6.80000019	1.8551E+12	Sin evento
1570	GPS	0.236	-0.94999999	9.93599987	3.852	42.0480003	-1.65600002	354.065002	3.70499992	0.375	-98.9468733	18.8441917	1321.5	11.12497692	6.80000019	1.8561E+12	Sin evento

Figura 11. Archivo plano que contiene los datos adquiridos con el módulo de toma de señal. Fuente: elaboración propia.

En el caso de las irregularidades en la vía se trató de hacer una verificación visual, para lo cual se utilizó una de las características de Google Maps. Dado que ya se había identificado la sección del archivo que tenía etiquetado el evento, se tomó la coordenada inicial y final del evento, y luego se geolocalizaron en el mapa como se muestra en la Figura 12.

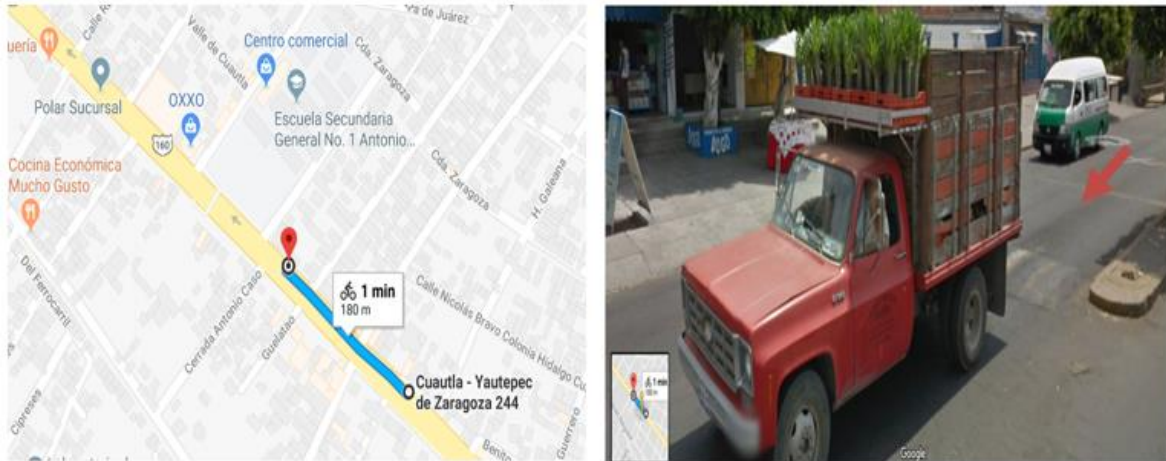


Figura 12. Geolocalización de los datos presentados en la Figura 11. Fuente: elaboración propia.

Por último, se graficaron los datos correspondientes al eje z o eje de desplazamiento vertical registrados por el acelerómetro como se muestra en la Figura 13 y se compararon con los reportados por la literatura. Este proceso se realizó con todos los archivos generados al azar.



Figura 13. Representación de los datos correspondientes al eje de desplazamiento vertical. Fuente: elaboración propia.

5.1.6 Datos recolectados durante el periodo de pruebas

Los datos recolectados durante el periodo de prueba suman un total de 66'965.564 registros, los cuales corresponden a más de 100 recorridos realizados en zonas rurales y urbanas. Dentro del conjunto de datos obtenido se etiquetaron 13.649 eventos distribuidos de la siguiente manera: 777 giros laterales, 12.654 detenciones del vehículo y 218 irregularidades en la vía. Es importante resaltar que los eventos correspondientes a exceso de velocidad y dirección en contra sentido no fueron etiquetados por dos motivos: primero, los datos fueron tomados en entornos reales con el tráfico normal de la vía; y segundo, los valores

de referencia para establecer estos comportamientos se pueden adquirir a partir de la norma. En la Tabla 6 se muestra el total de datos por sensor.

Tabla 6. Registros recolectados en la fase de pruebas

Sensor	Núm. registros
Acelerómetro	54'452,442
GPS	391,388
Magnetómetro	5'006,596
Orientación / Giroscopio	7'115,138
Total	66'965,564

Así mismo, en la Tabla 7 se presentan los valores mínimos y máximos para la aceleración y la velocidad obtenidos dentro del conjunto de datos.

Tabla 7. Valores mínimos y máximos de aceleración y velocidad presentes en el conjunto de datos

Parámetro	Valor Mínimo	Valor Máximo	Número de Registros	Valor Promedio
Aceleración en eje longitudinal(Y)	-38.77409744	38.85669708	54117673	2.8899959
Aceleración en eje lateral(X)	-37.24180603	39.03147507	54117673	0.2233966
Aceleración en eje vertical (Z)	-37.32321167	38.90577698	54117673	5.79880077
Velocidad	0	270.7999878	391388	22.8593062

5.1.7 Prototipo para la realización de pruebas utilizando la plataforma FIWARE.

El prototipo para la realización de pruebas tomó el componente base del prototipo realizado para la recolección de los datos y agregó las funcionalidades descritas en el capítulo 4. En la práctica, este prototipo está compuesto por un conjunto de módulos los cuales se comunican entre sí. De manera global, un módulo se ejecuta en el teléfono inteligente y otro se ejecuta en la plataforma FIWARE. En cuanto al reconocimiento automático de los eventos, el prototipo cuenta con dos bloques principales del lado del teléfono: El primero enfocado en la manipulación de la señal proveniente de los sensores y el segundo en la detección de los eventos. Es importante resaltar que para las pruebas se realizaron en teléfonos inteligentes que tenían corriendo además del prototipo de pruebas, las aplicaciones que el usuario a bien tenía. Esto se hizo con el fin de tener unos datos lo más cercanos posibles a un escenario real. En la Figura 14 se muestra una de las pruebas realizadas referente a la manipulación, una señal base y su visualización en el módulo desarrollado en la plataforma FIWARE.

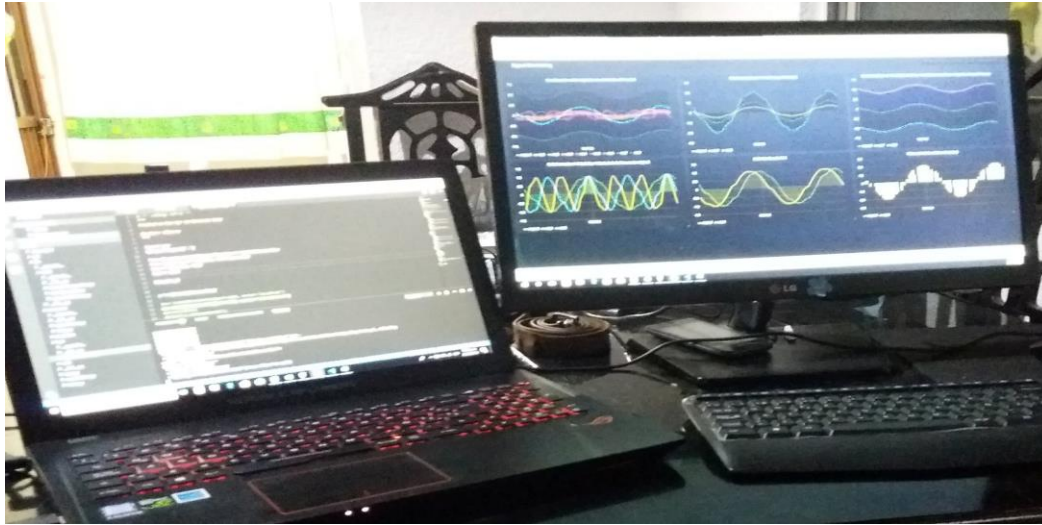


Figura 14. Análisis de una sola señal base a partir de sus derivadas con la ayuda del bloque de manipulación de señal y QuantumLeap. Fuente: elaboración propia.

5.2 Pruebas y resultados

Las pruebas se realizaron acorde a los conocimientos adquiridos en la etapa de análisis y selección o diseño de los algoritmos. En esta etapa se encontró que los trabajos enfocados en el reconocimiento y clasificación de eventos basados en los datos provenientes de algún tipo de sensor constan de dos etapas: procesamiento de la señal y, detección o clasificación de los eventos que son de interés. En este apartado se describen inicialmente las pruebas relacionadas con el procesamiento de la señal y posteriormente las pruebas relacionadas con las pruebas de los algoritmos propuestos.

5.2.1 Procesamiento de la señal

Las pruebas relacionadas con el procesamiento de la señal se realizaron con la ayuda de los componentes de tema de muestra y manipulación de la señal. Las pruebas relacionadas con el procesamiento de la señal se centraron en dos procesos independientes, pero necesarios para la fase de detección de eventos: tratamiento de la señal y construcción de mapas.

5.2.1.1 Tratamiento de la señal

El tratamiento de la señal consta de una serie de procesos: filtrado, preanálisis, empaquetado entre otras. En esta sección se describen algunos de los procesos realizados. Para esta prueba se utilizó el repositorio externo el cual corresponde a señales tomadas con el acelerómetro. Las clases presentes en este repositorio son: tope de asfalto, tope de metal, bache, camino regular y camino irregular. El repositorio cuenta con 100 ejemplos de cada clase y un factor importante es que todos los ejemplos son de tamaño diferente, por lo que no se pueden pasar directamente a algunos algoritmos de clasificación.

El primer paso que se realizó fue la eliminación de ruido blanco o filtrado de la señal, para esto se utilizó la técnica de medias móviles $MA(p)$, donde p se refiere al tamaño de la ventana.

Para este proceso se tomó al azar un ejemplo de cada clase y se ejecutó el script para obtener nuevos vectores a partir del vector original. El script se ejecutó para calcular las siguientes medias móviles: $MA(5)$, $MA(10)$, $MA(15)$, $MA(20)$, $MA(25)$, $MA(30)$, $MA(40)$, $MA(50)$, $MA(75)$, $MA(100)$. Una vez obtenido los nuevos vectores se procedió a observar sus gráficas. En la Figura 15, se muestra la gráfica para el ejemplo de la clase camino irregular.

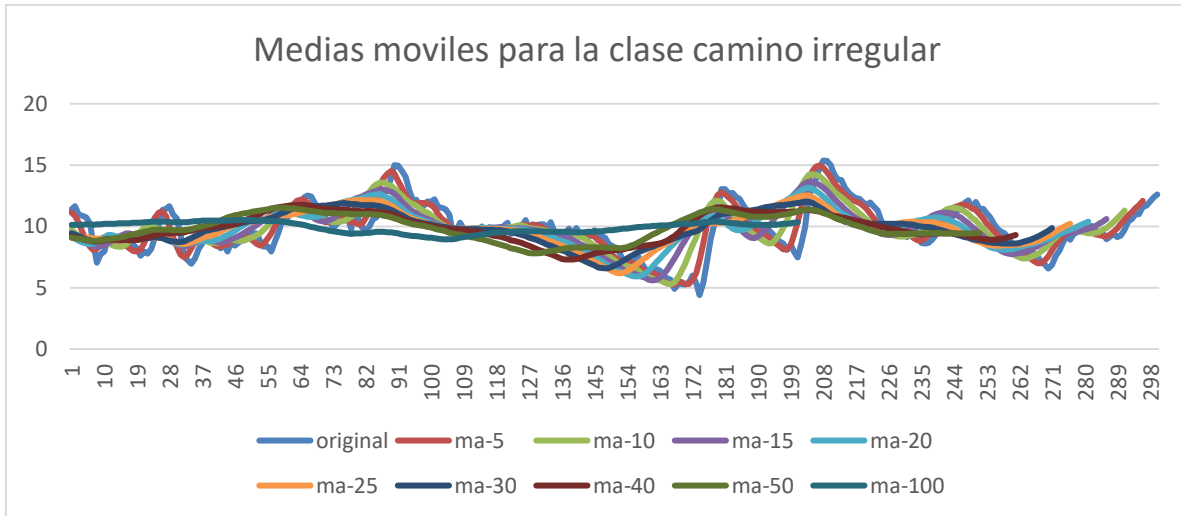


Figura 15. Medias móviles para la clase camino irregular. Fuente: elaboración propia.

El nuevo vector resultante de aplicar el modelo de medias móviles $MA(p)$ tiene $p-1$ elementos menos que el vector original. De acuerdo con los resultados obtenidos al observar varias gráficas, se determinó que en este trabajo se utilizarían medias móviles con p menor o igual a 15 para el proceso de filtrado. Sin embargo, se utilizará un p más alto para el caso de calibración de la señal.

La Figura 16 muestra la gráfica para un ejemplo de la clase “camino regular”, el cual está compuesto por un total de 298 muestras, como se puede observar en la gráfica las diferencia que existe entre la gráfica para $MA(15)$ y $MA(50)$ es muy poca. Esto se debe a que el comportamiento de un camino regular es muy similar al comportamiento del ruido blanco, por lo que su media va a tender a cero o a un valor de referencia. En este caso se aproxima al valor de la gravedad por cuanto la señal no está centrada.

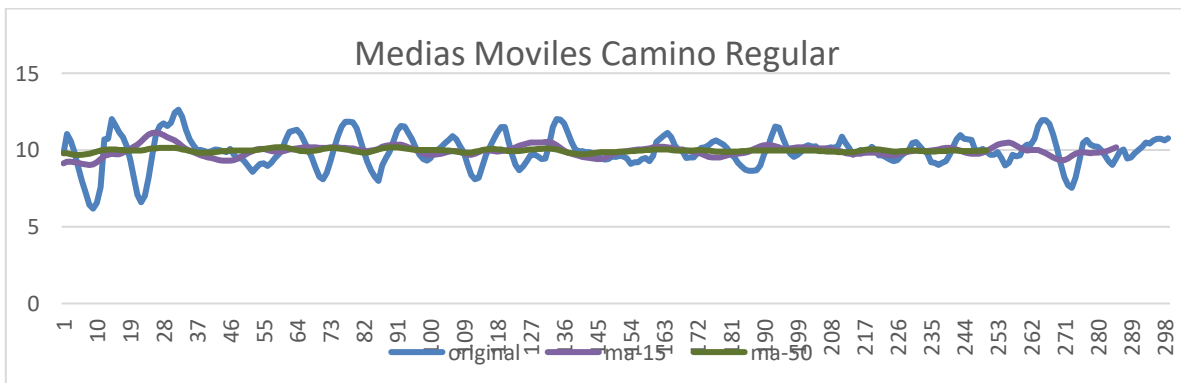


Figura 16. Tratamiento de la señal utilizando medias móviles de los modelos ARMA para camino regular. Fuente: elaboración propia.

En la imagen anterior puede observarse que los datos corresponden al *eje x* y el *eje y* del acelerómetro. Lo que indica que los eventos corresponden a giros o cambios de velocidad por fuera de los parámetros de referencia establecidos. Los vectores se empiezan a crear cada vez que se encuentra un valor diferente de cero y se terminan después de la aparición de la cantidad de ceros consecutivos establecidos en un parámetro de referencia. Una vez que un vector finalizado, se pasa al bloque de empaquetado. El bloque de empaquetado procesa la señal de acuerdo con los parámetros preestablecidos y los distribuye generalmente al bloque de detención de eventos y al Orion Context Broker. En la Figura 20 se muestra algunas de las tareas que el módulo de empaquetado puede hacer. En este caso se utiliza el modelo de datos SginalDevice para el envío de la señal base y sus derivadas; y se utiliza el módulo de QuantumLeap para la visualización de las señales. Además, se utiliza una función senoidal como señal base.



Figura 20. Visualización de una misma señal con diferentes manipulaciones sobre la señal. Fuente: elaboración propia.

En la figura anterior, se pueden observar la potencialidad de los componentes desarrollados cuando se integran con el ecosistema FIWARE. En el panel 1 se muestran todas las señales; en el panel 2 se muestra la señal original, además, de las señales equivalentes al escalar la señal original en un factor de $\frac{1}{2}$ y un factor de 2. En el panel 3, se muestra la señal original y sus equivalentes al desplazarla ± 5 unidades con respecto al eje horizontal. En el panel 4 se muestra la señal original y su equivalente al multiplicar la señal original por un factor de $\frac{1}{2}$ y $\frac{1}{3}$, o lo que es igual al muestrear la señal 2 y 3 veces más rápido respectivamente. En el panel 5, se muestra la señal original y su equivalente al desfazar la señal en $\frac{1}{4}$ y por último en el panel 6, se muestra un diezmado por 2 de la señal original. Con el proceso de empaquetado y distribución de los datos concluye el proceso de adquisición y tratamiento de la señal.

5.2.1.2 Generación Automática de rutas

Este proceso se centra en el tratamiento de los datos provenientes del GPS. Los datos provenientes del GPS juegan un papel muy importante porque son la base de algunos de los algoritmos y, además, son utilizados en todos los casos para la geolocalización del lugar en

el que se detecta el evento. En esta actividad se creó un conjunto de algoritmos capaz de crear un archivo HTML que traza la ruta registrada. Además, apoyados en los algoritmos desarrollados para el tratamiento de la señal, es posible diezmar los datos correspondientes a una misma ruta. Las rutas diezgadas son utilizadas para obtener la geometría de la ruta por medio de los servicios de Google Maps. El módulo de generación de rutas también es capaz de construir automáticamente la cantidad de tramos que puede tener una ruta, con base en la geometría que marca el conjunto de datos GPS recolectado. En las Figura 21 y la Figura 22 se muestran los tramos generados para una misma ruta, después de aplicar un diezmo de 20 y 50 muestras respectivamente.

```

app.py | velocidad.py | gps_02-Mar-2018_cords-dz_20.csv x
1 -99.22182116|18.87985551|[18.87985551,-99.22182116][[-99.22182116,18.87985551][[lat:18.87985551 , lng:-99.22182116]|18.87985551,-99.22182116|-99.22182116|18.87985551]|0.17000000178813934|4221645499323.0|Fri
2 -99.2216809|18.87970734|[18.87970734,-99.2216809][[-99.2216809,18.87970734][[lat:18.87970734 , lng:-99.2216809]|18.87970734,-99.2216809|-99.2216809|18.87970734]|0.46000000834465027|4265430793681.0|Fri Mar 02
3 -99.2217101|18.87966973|[18.87966973,-99.2217101][[-99.2217101,18.87966973][[lat:18.87966973 , lng:-99.2217101]|18.87966973,-99.2217101|-99.2217101|18.87966973]|0.38999998569488525|4285458412788.0|Fri Mar 02
4 -99.22173552|18.87966756|[18.87966756,-99.22173552][[-99.22173552,18.87966756][[lat:18.87966756 , lng:-99.22173552]|18.87966756,-99.22173552|-99.22173552|18.87966756]|0.430543205666.0|Fri Mar 02 08:04:58
5 -99.22174489|18.87963916|[18.87963916,-99.22174489][[-99.22174489,18.87963916][[lat:18.87963916 , lng:-99.22174489]|18.87963916,-99.22174489|-99.22174489|18.87963916]|0.5600000023841850|4325432211627.0|Fri M
6 -99.22176448|18.87980804|[18.87980804,-99.22176448][[-99.22176448,18.87980804][[lat:18.87980804 , lng:-99.22176448]|18.87980804,-99.22176448|-99.22176448|18.87980804]|0.7799999713897705|4345432536047.0|Fri Mar 02
7 -99.22180453|18.87970962|[18.87970962,-99.22180453][[-99.22180453,18.87970962][[lat:18.87970962 , lng:-99.22180453]|18.87970962,-99.22180453|-99.22180453|18.87970962]|0.6200000047683716|436542192289.0|Fri M
8 -99.22176817|18.87965968|[18.87965968,-99.22176817][[-99.22176817,18.87965968][[lat:18.87965968 , lng:-99.22176817]|18.87965968,-99.22176817|-99.22176817|18.87965968]|1.1200000047683716|4371426083068.0|Fri M
9

```

Figura 21. Nuevos puntos de la ruta después de aplicar un diezmo de 20 a la señal. Fuente: elaboración propia.

```

app.py | velocidad.py | gps_02-Mar-2018_cords-dz_50.csv x
1 [-99.22182116|18.87985551|[18.87985551,-99.22182116][[-99.22182116,18.87985551][[lat:18.87985551 , lng:-99.22182116]|18.87985551,-99.22182116|-99.22182116|18.87985551]|0.17000000178813934|4221645499323.0|Fri
2 -99.22171473|18.87965515|[18.87965515,-99.22171473][[-99.22171473,18.87965515][[lat:18.87965515 , lng:-99.22171473]|18.87965515,-99.22171473|-99.22171473|18.87965515]|0.360000001430511475|4295429571274.0|Fri
3 -99.22176448|18.87980804|[18.87980804,-99.22176448][[-99.22176448,18.87980804][[lat:18.87980804 , lng:-99.22176448]|18.87980804,-99.22176448|-99.22176448|18.87980804]|0.7799999713897705|4345432536047.0|Fri Mar 02
4 -99.22176817|18.87965968|[18.87965968,-99.22176817][[-99.22176817,18.87965968][[lat:18.87965968 , lng:-99.22176817]|18.87965968,-99.22176817|-99.22176817|18.87965968]|1.1200000047683716|4371426083068.0|Fri M

```

Figura 22. Nuevos puntos de la ruta después de aplicar un diezmo de 50 a la señal. Fuente: elaboración propia.

Como puede observarse en la Figura 22, cada registro contiene las coordenadas iniciales y finales. Además, cada coordenada se presenta en diferentes formatos. Esto se hace con el objetivo de facilitar la obtención de las coordenadas en diferentes situaciones, ya que en algunos casos se puede desear visualizar los datos en Google Maps, pero en otros casos se puede optar por otro proveedor que pida un formato diferente. En otros casos vamos a necesitar la latitud y longitud por separado. Un factor importante a tener en este caso, es que las coordenadas GPS solo están compuestas por latitud y longitud. Se omite la latitud, porque tanto los servicios como librerías utilizadas en este trabajo no requieren o no contemplan la latitud. Para finalizar, en la Figura 23 se muestra el mapa generado automáticamente a partir de los datos obtenidos.



Figura 23. Mapa generado automáticamente a partir de los datos registrados por el GPS. Fuente: elaboración propia.

5.2.2 Cambios de velocidad

Esta tarea se centró en detectar el comportamiento del desplazamiento del vehículo a partir de la velocidad registrada por el GPS. Para este caso se seleccionó un recorrido en particular, se particionó en grupos de 5 elementos consecutivos de acuerdo con tiempo de ocurrencia, donde $v01$ y $v02$ corresponden a los dos últimos valores registrados en el vector de características y $v1-v3$ corresponden a nuevos registros de velocidad. Además, se agregó la media y la varianza de las velocidades registradas; luego se realizó un etiquetado manual de cada uno de los ejemplos. En total se etiquetaron 391 ejemplos. Estos datos fueron pasados por tres clasificadores y los resultados obtenidos se muestran en la Tabla 8. Así mismo, En la Figura 24 se muestran ejemplos de las primeras instancias.

Tabla 8. Resumen de resultados obtenidos al pasar los datos por los clasificadores.

Técnica	Clasificados Correctamente		Clasificados Incorrectamente	
	Cantidad	%	Cantidad	%
Naive Bayes	281	71.867 %	110	28.133 %
RandomForest	368	94.1176 %	23	5.8824 %
J48	367	93.8619 %	24	6.1381 %

Como puede observarse en la Tabla 8, el algoritmo con el que se obtuvieron mejores resultados fue con *RandomForest*, lo cual es coherente con lo reportado en la literatura.

id	v01	v02	v1	v2	v3	promedio	varianza	estado	
1	0	0	0	3	3	4	1.4	3.04	InicioMarcha
2	3	4	3	3	3	3	3.2	0.16	VelocidadVariable
3	3	3	4	4	4	4	3.6	0.24	IncrementoVelocidad
4	4	4	4	5	5	5	4.4	0.24	IncrementoVelocidad
5	5	5	5	5	5	5	5	0	VelConstante
6	5	5	5	5	5	4	4.8	0.16	DecrementoVelocidad
7	5	4	4	4	4	4	4.2	0.16	DecrementoVelocidad
8	4	4	4	3	3	3	3.6	0.24	IncrementoVelocidad
9	3	3	3	3	3	3	3	0	VelConstante
10	3	3	2	2	2	2	2.4	0.24	DecrementoVelocidad
11	2	2	0	0	0	0	0.8	0.96	Parada
12	0	0	0	0	0	0	0	0	Detenido
13	0	0	0	0	0	0	0	0	Detenido
14	0	0	0	0	0	0	0	0	Detenido
15	0	0	0	0	3	0.6	1.44	1.44	InicioMarcha
16	0	3	3	0	0	0	1.2	2.16	ParadasIntermitente
17	0	0	0	3	3	4	1.4	3.04	InicioMarcha
18	3	4	5	5	5	5	4.4	0.64	IncrementoVelocidad

Figura 24. Ejemplo de las instancias presente en el conjunto de datos utilizados Fuente: elaboración propia.

5.2.3 Detección de irregularidades en la vía

El procedimiento realizado para esta prueba fue similar al anterior. En este caso se utilizaron los datos provenientes del acelerómetro y dado que la velocidad de muestreo es más rápida que la del GPS, se tomó una ventana de 15 muestras sin traslape. Adicional al

promedio y la varianza se agregó la distancia que existe en puntos entre el valor mínimo y máximo de las 15 muestras registradas. En la Figura 25 se muestran ejemplos de las primeras instancias.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15	promedio	varianza	distancia	tipo
2	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	-0.06667	0.062222	1	Irregularidad-Minima
3	0	-1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0.133333	2	Irregularidad-Minima
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Tramo-Normal
5	0	0	1	0	0	0	0	0	0	0	0	1	-1	0	0	0.066667	0.195556	2	Irregularidad-Minima
6	0	1	-1	1	0	1	0	0	-1	0	-1	2	0	-1	-1	0	0.8	3	Irregularidad-Leve
7	0	0	0	0	0	0	0	1	1	-1	0	0	-1	-1	-1	-0.13333	0.382222	2	Irregularidad-Minima
8	-1	0	0	0	0	-1	0	0	0	0	1	2	0	0	0	0.066667	0.462222	3	Irregularidad-Leve
9	0	0	0	1	0	0	0	0	-1	2	0	0	0	0	0	0.133333	0.382222	3	Irregularidad-Leve
10	0	0	0	0	0	0	0	1	0	0	0	1	-1	1	1	0.2	0.293333	2	Irregularidad-Minima
11	-2	5	-2	2	-1	2	-1	0	0	0	0	0	0	0	0	0.2	2.826667	7	Irregularidad-Grave
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Tramo-Normal
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Tramo-Normal
14	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0.133333	0.115556	1	Irregularidad-Minima
15	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	-1	-0.13333	0.115556	1	Irregularidad-Minima
16	0	-1	0	0	1	-1	0	-2	-1	0	-1	0	0	1	1	-0.2	0.693333	3	Irregularidad-Leve
17	0	0	0	1	0	1	1	0	0	0	1	0	0	0	0	0.266667	0.195556	1	Irregularidad-Minima
18	1	0	0	0	-1	1	0	0	0	0	0	1	0	1	1	0.266667	0.328889	2	Irregularidad-Minima
19	0	0	-1	1	-1	0	0	0	0	-1	0	1	-1	0	0	-0.13333	0.382222	2	Irregularidad-Minima
20	-1	0	0	1	-1	0	0	-1	0	0	0	0	0	1	1	0	0.4	2	Irregularidad-Minima
21	0	0	1	0	0	0	0	1	-1	0	1	0	-1	3	3	0.466667	1.315556	4	Irregularidad-Leve
22	0	-2	2	-1	0	0	1	0	0	0	0	0	0	-1	-1	-0.13333	0.782222	4	Irregularidad-Leve
23	2	0	0	-1	0	0	-1	0	-1	0	0	1	0	-1	-1	-0.13333	0.648889	3	Irregularidad-Leve

Figura 25. Primeras instancias del conjunto de datos de entrenamiento y prueba Fuente: elaboración propia.

En este caso se utilizó el algoritmo de *Naive Bayes* con un porcentaje de clasificación del 52.4% y máquinas de vector soporte con un porcentaje del 99.2%. En la Tabla 9 y la

Tabla 10 se muestra la matriz de confusión para cada uno de los clasificadores.

Tabla 9. Resultados obtenidos con *Naive Bayes*

	Irregularidad-Grave	Irregularidad-Leve	Irregularidad-Media	Irregularidad-Mínima	Irregularidad-Muy-Grave	Tramo-Normal
Irregularidad-Grave	4	0	0	0	1	0
Irregularidad-Leve	0	47	0	6	0	0
Irregularidad-Media	0	1	18	0	0	0
Irregularidad-Mínima	0	0	0	80	0	0
Irregularidad-Muy-Grave	0	0	0	0	0	0
Tramo-Normal	2	29	3	200	0	118

Tabla 10. Resultados obtenidos con máquinas de vector soporte

	Irregularidad-Grave	Irregularidad-Leve	Irregularidad-Media	Irregularidad-Mínima	Irregularidad-Muy-Grave	Tramo-Normal
Irregularidad-Grave	6	0	0	0	0	0
Irregularidad-Leve	0	74	0	1	0	0
Irregularidad-Media	0	0	21	0	0	0
Irregularidad-Mínima	0	3	0	285	0	0
Irregularidad-Muy-Grave	0	0	0	0	1	0
Tramo-Normal	0	0	0	0	0	118

5.2.4 Algoritmos basados en técnicas matemáticas

En esta sección se describen los resultados obtenidos con los algoritmos basados en técnicas matemáticas, los cuales son: velocidad no autorizada, detección de cambios de aceleración, conducción en contrasentido. Para realizar las pruebas se utilizó un recorrido compuesto por 5.090 puntos (*tramo A*). De este conjunto de datos se extrajo un conjunto de 490 puntos (*tramo B*) que cumplía con la característica de ser un segmento de la vía con muy pocas curvas, donde el tráfico vehicular y la presencia de irregularidades en la vía no es tan alto. El objetivo de extraer este segmento es observar cómo influye la definición de los segmentos en el comportamiento de los algoritmos. En la Figura 26 y la Figura 27 se muestra la ubicación geoespacial de los recorridos antes descritos.

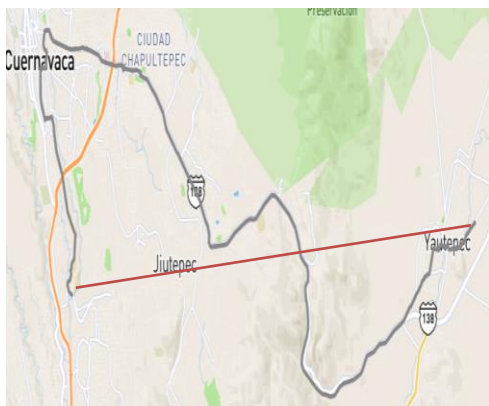


Figura 26. Recorrido completo (Tramo A). Fuente: elaboración propia.

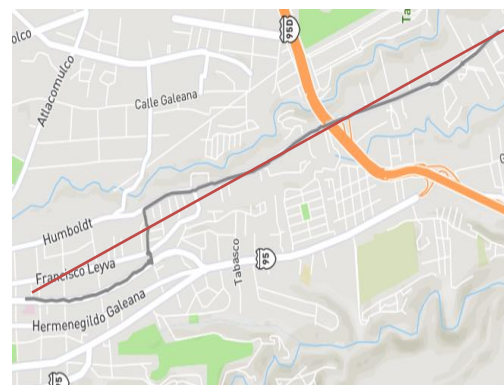


Figura 27. Recorrido extraído (Tramo B). Fuente: elaboración propia.

En la Figura 28 se muestran los parámetros de referencia utilizados para esta prueba.

```

tiempoPercepcion=0.75
tiempoReaccion=0.75
gravedad=9.8
coeficienteFriccion=0
inclinacionSegmento=0
percentageToleranciaFrenado=0
velMax=50
velMin=0
toleranciavel=0

```

Figura 28. Parámetros de prueba. Fuente: elaboración propia.

El desplazamiento se realizó siempre en sentido correcto y sin tener segmentaciones en ninguno de los tramos. En la Figura 29 se muestran algunos de los valores obtenidos para el *tramo A*. El porcentaje de detección en sentido contrario está cercano al 70%. Con respecto a la velocidad, los resultados no son muy alentadores, ya que casi un 40% de las lecturas obtenidas fueron clasificadas como desplazamiento a una velocidad no permitida. Los resultados obtenidos en las pruebas realizadas sobre el tramo A, son realmente bajas. Lo cual deja dos opciones abiertas, la primera que los algoritmos utilizados no son los adecuados para este tipo de situaciones y la segunda que los segmentos o las variables no fueron debidamente parametrizados.

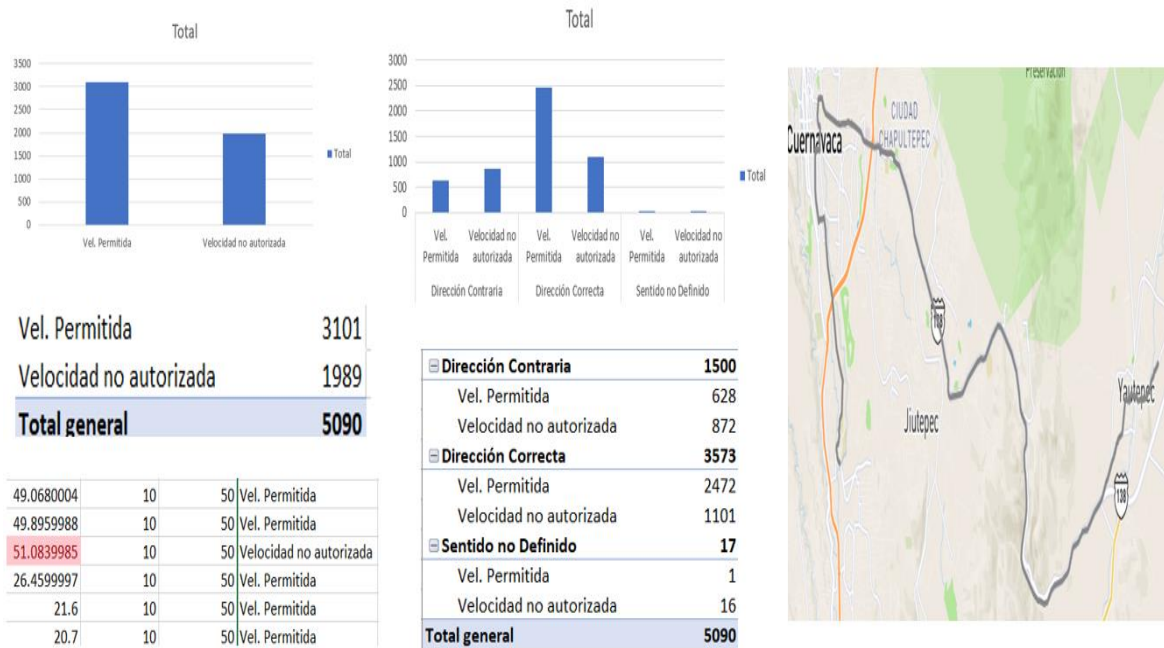


Figura 29. Resultados obtenidos para el tramo A. Fuente: elaboración propia.

Luego se ejecutaron los algoritmos con los mismos parámetros, pero sobre el *tramo B*. Los resultados obtenidos mejoran con respecto a los obtenidos en el *tramo A* y se obtienen unos valores por encima del 93% y no se detecta ningún exceso de velocidad. Es importante recordar que el *tramo B* es un segmento del *tramo A* por lo tanto los valores del *tramo B* son exactamente iguales a los contenidos para ese mismo segmento en el *tramo A*.

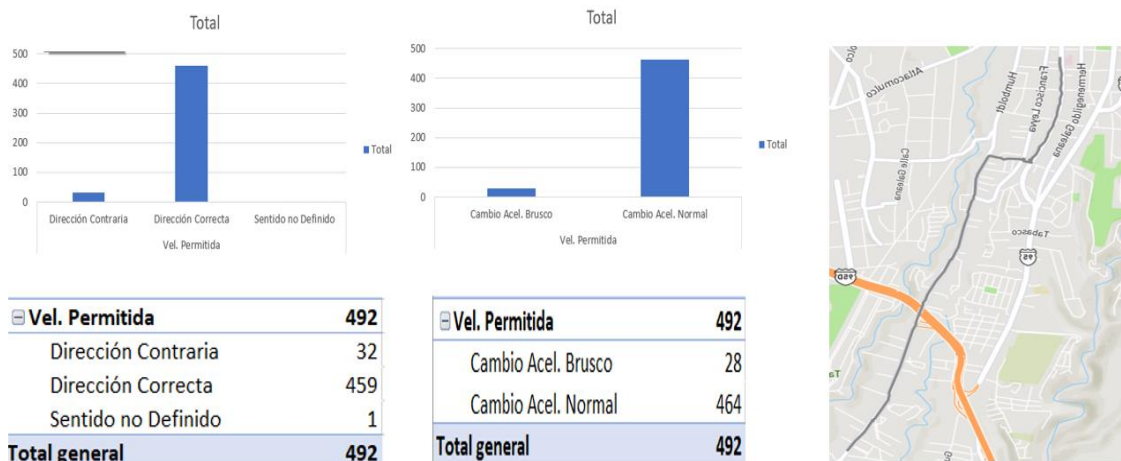


Figura 30. Resultados obtenidos para el tramo B. Fuente: elaboración propia.

Lo datos más acercados a la realidad del recorrido realizado se presentan en el *tramo B*. una de las razones por lo que esto ocurre está relacionada con los algoritmos seleccionados. Por ejemplo, en el caso de conducción en dirección contraria el *tramo B* presenta menos falsos positivos que el *tramo A*, esto se debe a que este algoritmo está fundamentado en el concepto de distancia entre dos puntos el cual se calcula con la distancia esférica, pero, aun así, este no representa en realidad la trayectoria que un vehículo tiene que seguir para llegar hasta un punto. De este modo, al *tramo B* presenta un mejor resultado, dado que todos sus puntos están más cerca de la línea recta que hay entre el punto inicial y el punto final, lo cual puede observarse en la Figura 31. Los valores relacionados con la detección de velocidad no permitida se deben a que en términos de geometría y ubicación espacial el *tramo B* es más homogéneo que el *tramo A*, el *tramo B* se encuentra en una zona urbana por lo que el límite de velocidad de 50 k/h se encuentra dentro del rango normal. El *tramo A* no corresponde a un tramo homogéneo en cuanto a su geometría y su ubicación espacial, ya que algunas partes corresponden a zonas urbanas, otras a zonas rurales. Algunas partes corresponden a vías secundarias o terciarias, mientras que otras corresponden a vías primarias con velocidades permitidas de 80 k/h, velocidad que está muy por encima de la establecida en los parámetros de referencia.

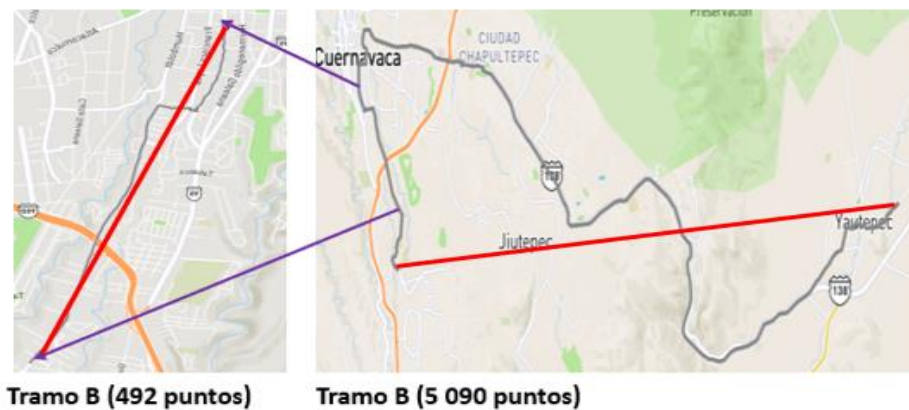


Figura 31. Línea entre el punto inicial y final del recorrido. Fuente: elaboración propia.

5.2.5 Prueba en proyecto SmartSDK Security

Además de las pruebas realizadas con datos históricos, los algoritmos base fueron probados en el marco del proyecto SmartSDK Security. La aplicación desarrollada en el marco del proyecto SmartSDK Security está compuesta por un conjunto de módulos interconectados entre sí. En la Figura 32 se muestran algunas de las vistas desarrolladas, en este caso se alcanza a observar una vista del módulo que corre del lado del teléfono inteligente que además de detectar los eventos, puede emitir y recibir alertas de los mismos. Y una vista del módulo alojado en la web que corre sobre la plataforma FIWARE, el cual permite que un administrador o un guarda de seguridad pueda observar los eventos que ocurren en un campus determinado. En este punto es muy importante aclarar, que algunos de los algoritmos base desarrollados en este trabajo fueron implementados en el proyecto SmartSDK Security. Sin embargo, las aplicaciones mostradas en la **¡Error! No se encuentra el origen de la referencia.** corresponden a desarrollos propios del equipo SmartSDK Security y no a los módulos que para efectos de toma, procesamiento y visualización se realizaron durante este trabajo. Esto soporta la independencia y flexibilidad de los módulos desarrollados en el presente trabajo. El comportamiento de los algoritmos fue similar al comportamiento presentado con los datos históricos.

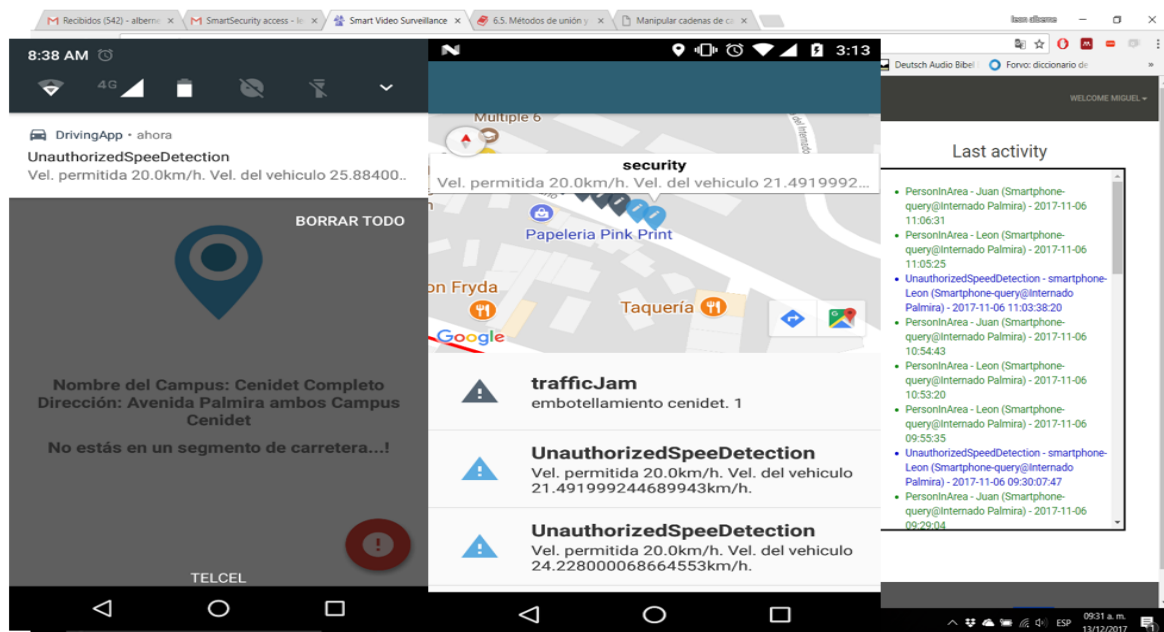


Figura 32. Aplicaciones del Proyecto SmartSDK Security. Fuente: elaboración propia.

Capítulo 6. Conclusiones y trabajos futuros

En este trabajo se muestra que a partir de los datos obtenidos de un teléfono inteligente y la utilización de la plataforma FIWARE es factible la detección automática de los eventos de velocidad no permitida, desplazamiento en contrasentido, cambios de velocidad, irregularidades en la vía y giros laterales al manejar un automóvil.

Con base en los resultados de las pruebas realizadas, los mejores resultados obtenidos son los siguientes: velocidad no permitida 93%, desplazamiento en contrasentido contrario 93%, cambios de velocidad 94%, irregularidades en la vía 99%, y giros laterales 90%. Estos resultados son similares a los reportados por otros autores. Sin embargo, es importante tener en cuenta que la calidad y precisión de los algoritmos van a depender de la calidad de los datos, la adecuada parametrización de los segmentos de la vía y las características técnicas del teléfono inteligente.

La implementación del prototipo de prueba muestra que la utilización de la plataforma FIWARE, en conjunto de la especificación NGSI facilitan la implementación de soluciones orientadas a las ciudades inteligentes y el internet de las cosas. Sin embargo, también es importante resaltar que aún existen algunas limitaciones a nivel de infraestructura y de seguridad que impiden realizar un mejor aprovechamiento de las características del internet de las cosas y las ciudades inteligentes.

En este trabajo se presenta un conjunto de algoritmos base que pueden ser una alternativa de solución para la detección automática de eventos que pueden convertirse en situaciones de riesgo al momento de conducir un automóvil; La profundización en el tema permitió establecer que la detección automática de eventos de riesgo que puedan desencadenar en un accidente de tránsito encierra un grado de dificultad alto, ya que involucra conceptos de varias disciplinas. Por lo cual, el planteamiento de soluciones únicas o completas requiere del trabajo de un equipo interdisciplinario no solo del área de ingeniería, sino también de áreas relacionadas con el comportamiento social y la fisonomía de la persona.

Con base en el trabajo realizado también se pudo constatar que el reconocimiento y la clasificación de eventos basados en los datos provenientes de algún tipo de sensor constan de dos fases o etapas: la primera fase está relacionada con la toma de la señal, el filtrado del ruido y la transformación de la señal de acuerdo con el dominio del problema. La segunda fase está relacionada con la detección y la clasificación de los eventos que son de interés. Siendo la primera etapa en la que más se centran algunos trabajos, ya que para la segunda se hace uso de alguna herramienta de clasificación o reconocimiento de patrones.

6.1 Trabajos futuros

Los resultados obtenidos en este trabajo son solo una base que puede ser utilizada como punto de partida para trabajos futuros relacionados con el procesamiento de señales y la detección de eventos en el dominio del internet de las cosas. Un área de oportunidad puede

estar relacionada con la geolocalización, dada que las características propias de los datos tomados con el GPS pueden tener diferencias de más de un segundo entre dato y dato. Se puede utilizar el acelerómetro y el giroscopio para dar una ubicación aproximada sobre la localización del vehículo, mientras el GPS está fuera de línea.

Otro trabajo que se puede realizar es agregar más variables a los algoritmos desarrollados, como podrían ser las relacionadas con el entorno o con los sistemas de amortiguación del vehículo dado que en ese trabajo se contempla al vehículo como un sistema rígido.

Los algoritmos incluidos en este trabajo son cinco. Sin embargo, con los repositorios creados, se podrían detectar eventos como pendiente ascendente o vía resbalosa ya que estos datos dependen en gran parte de la información de contexto y una geolocalización del tramo el segmento de la carretera que cuenta con estas características.

Otro trabajo futuro puede estar relacionado con la detección automática del tipo de irregularidad, ya que en este trabajo se detecta la presencia de dicha irregularidad, pero no se especifica si corresponde a un tope o a un bache.

Otro trabajo futuro sería probar el comportamiento del módulo de toma de datos y el de procesamiento de señales en otros dominios.

Finalmente, los módulos desarrollados no se probaron con señales provenientes de sistemas embebidos, lo cual abre un área de oportunidad para trabajos futuros.

6.2 Publicaciones y logros

Publicaciones

Jiménez-Builes, J., Mejía, J., Soto, D., Vargas, F., & Torres, L. Método para crear un módulo de minería de datos que apoye la interpretación de resultados. Investigación e Innovación en Ingeniería de Software. Colombia. ISBN: 978-958-59687-6-9, Ed. Sello Editorial TdeA, 2017.

Martinez, A., Ramirez, F., Estrada, H. & Torres, L. A.: Generic Module for Collecting Data in Smart Cities on Remote Sensing & Spatial Information Sciences, pp. 65 -72, Puebla, Mexico, DOI:10.5194/isprs-archives-XLII-4-W3-65-2017, 2017.

Martinez, A., Torres, L., Vargas, F., & Ortiz-Hernández J.: Adquisición de señales digitales utilizando teléfonos inteligentes y FIWARE. In: COMIA 2019, Nayarit, Mexico, 2019. Aceptado para publicar.

Logros

Obtención de beca y asistencia al congreso *2nd International Conference on Smart Data and Smart Cities* realizado en Puebla en octubre del 4 al 7

Asistencia a la reunión general del proyecto SmartSDK la cual se denominó “SmartSDK General Assembly Meeting and R&D Collaboration Meeting” organizado por Martel en Zúrich del 5 al 9 marzo 2018.

Integrante de la comisión del equipo de trabajo Smarsdk que asistió al evento FIWARE Summit, realizado en la ciudad de Málaga, España del 13 al 15 de diciembre de 2016.

Referencias

- [1] Organización Mundial de la Salud, “Informe Sobre La Situación Mundial De La Seguridad Vial 2015.” pp. 1–12, 2015.
- [2] M. Díaz, C. Martín, and B. Rubio, “State-of-the-art, challenges, and open issues in the integration of Internet of things and cloud computing,” *J. Netw. Comput. Appl.*, vol. 67, no. C, pp. 99–117, May 2016.
- [3] J. S. Rueda and J. M. Talavera Portocarrero, “Similitudes y diferencias entre Redes de Sensores Inalámbricas e Internet de las Cosas: Hacia una postura clarificadora,” *Rev. Colomb. Comput.*, vol. 18, no. 2, pp. 58–74, 2018.
- [4] C. Saiprasert, T. Pholprasit, and S. Thajchayapong, “Detection of Driving Events using Sensory Data on Smartphone,” *Int. J. Intell. Transp. Syst. Res.*, no. June 2016, pp. 1–12, 2015.
- [5] B. Fernandes, M. Alam, V. Gomes, J. Ferreira, and A. Oliveira, “Automatic accident detection with multi-modal alert system implementation for ITS,” *Veh. Commun.*, vol. 3, pp. 1–11, 2016.
- [6] L. C. Gonzalez, R. Moreno, H. J. Escalante, F. Martinez, and M. R. Carlos, “Learning Roadway Surface Disruption Patterns Using the Bag of Words Representation,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, 2017.
- [7] H. M. Ali and Z. S. Alwan, “Car Accident Detection and Notification System Using Smartphone,” *Int. J. Comput. Sci. Mob. Comput.*, 2015.
- [8] P. Chaovalit, C. Saiprasert, and T. Pholprasit, “A method for driving event detection using SAX with resource usage exploration on smartphone platform,” *EURASIP J. Wirel. Commun. Netw.*, 2014.
- [9] A. Bin Faiz, A. Imteaj, and M. Chowdhury, “Smart vehicle accident detection and alarming system using a smartphone,” in *1st International Conference on Computer and Information Engineering, ICCIE 2015*, 2016.
- [10] OMS, *Paquete de medidas técnicas de seguridad vial*. Ginebra: Organización Mundial de la Salud, 2017.
- [11] Organización Mundial de la Salud, “Accidentes de tránsito -Lesiones causadas por el tránsito,” OMS, 2018. [Online]. Available: <https://bit.ly/2LqJlqr>. [Accessed: 16-May-2019].
- [12] Fundación Carlos Slim, “Las causas más comunes en accidentes de tránsito - Seguridad Vial.” [Online]. Available: <https://bit.ly/2z9kDBq>. [Accessed: 16-May-2019].
- [13] CESVI, “Factores que generan los accidentes de tránsito,” 2015.
- [14] Icebike, “Real time traffic accident statistics.” [Online]. Available: <https://bit.ly/2QfsUMw>. [Accessed: 20-May-2019].
- [15] INEGI, “Mortalidad - ¿De qué mueren los mexicanos?” [Online]. Available: <https://bit.ly/1TYgZPe>. [Accessed: 20-May-2019].
- [16] Z. Liu, M. Wu, K. Zhu, and L. Zhang, “SenSafe: A smartphone-based traffic safety framework by sensing vehicle and pedestrian behaviors,” *Mob. Inf. Syst.*, vol. 2016, 2016.
- [17] CEA Seguridad Vial, “El Sistema ADAS ayuda a prevenir accidentes de tráfico.” [Online]. Available: <https://bit.ly/2Hrk0bG>. [Accessed: 20-May-2019].
- [18] K. Rostamzadeh, H. Nicanfar, N. Torabi, S. Gopalakrishnan, and V. C. M. Leung, “A context-aware trust-based information dissemination framework for vehicular networks,” *IEEE Internet Things J.*, vol. 2, no. 2, pp. 121–132, 2015.
- [19] N. Kattukaran, A. George, and T. P. M. Haridas, “Intelligent accident detection and alert system for emergency medical assistance,” in *2017 International Conference on Computer Communication and Informatics, ICCCI 2017*, 2017.
- [20] L.-J. Kau and C.-S. Chen, “A Smart Phone-Based Pocket Fall Accident Detection, Positioning, and Rescue System,” *IEEE J. Biomed. Heal. Informatics*, vol. 19, no. 1, pp. 44–

- 56, 2015.
- [21] INEGI, “Accidentes de tránsito terrestre en zonas urbanas y suburbanas - Glosario - Accidente.” [Online]. Available: <https://bit.ly/2Wj3SRU>. [Accessed: 21-May-2019].
 - [22] Secretaría de comunicaciones y Transporte, “Manual de proyecto geométrico de carreteras,” 2018.
 - [23] Secretaria de seguridad publica, “Reglamento de Tránsito en Carreteras y Puentes de Jurisdicción Federal.” [Online]. Available: <https://bit.ly/2Jx4oFN>. [Accessed: 20-May-2019].
 - [24] Asociación Argentina de Carreteras, “Vialidad Nacional 80° Aniversario.” [Online]. Available: <https://bit.ly/2VNH9Tg>. [Accessed: 20-May-2019].
 - [25] J.-R. Quintero-González, “Del concepto de ingeniería de tránsito al de movilidad urbana sostenible,” *Ambient. y Desarro.*, vol. 21, no. 40, p. 57, 2017.
 - [26] Comisión Nacional de los Derechos Humanos Mexico, *Movilidad, vivienda y derechos humanos*. Ciudad de México, México, 2016.
 - [27] Real Academia Española (RAE), “Definición sensor,” 2017. [Online]. Available: <http://dle.rae.es/?id=XaY03jU>. [Accessed: 20-May-2019].
 - [28] J. Robert D. Frey, “Apparatus and method for inertial sensor calibration.” [Online]. Available: <https://patents.google.com/patent/US9857198B2/en>. [Accessed: 20-May-2019].
 - [29] J. Organista-Sandoval, L. McAnally-Salas, and G. Lavigne, “El teléfono inteligente (smartphone) como herramienta pedagógica,” *Apertura*, vol. 5, pp. 6–19, 2013.
 - [30] J. Wahlstrom, I. Skog, and P. Handel, “Smartphone-Based Vehicle Telematics: A Ten-Year Anniversary,” *IEEE Transactions on Intelligent Transportation Systems*, 2017.
 - [31] Universidad Politécnica de Madrid, “Capítulo 12 : Introducción al Procesamiento digital de Señales.” [Online]. Available: <https://bit.ly/1euSPrj>. [Accessed: 20-May-2019].
 - [32] Grupo fractalia, “Las plataformas de código abierto en las Smart Cities.” [Online]. Available: <https://bit.ly/2w9tGRZ>. [Accessed: 20-May-2019].
 - [33] Centro de estudios en libertad de expresión y acceso a la información, “El ‘internet de las cosas’: más internet que otra cosa.” [Online]. Available: <https://bit.ly/2VVZMjv>. [Accessed: 02-May-2017].
 - [34] Sergio García Gómez *et al.*, “Fiware: Una plataforma abierta y estándar para Ciudades Inteligentes,” *ESMARTCITY*, 2015. [Online]. Available: <https://bit.ly/2WjNel0>. [Accessed: 20-May-2019].
 - [35] W. Cao, X. Lin, K. Zhang, and L. Zhang, “Poster Abstract : Analysis and Evaluation of Driving Behavior Recognition Based on a 3-axis Accelerometer Using a Random Forest Approach,” pp. 303–304, 2017.
 - [36] F. Martínez, L. Carlos Gonzalez, and M. Ricardo Carlos, “Identifying Roadway Surface Disruptions Based on Accelerometer Patterns,” *IEEE Lat. Am. Trans.*, vol. 12, no. 3, pp. 455–461, 2014.
 - [37] F. Martínez, L. C. González, and M. R. Carlos, “SASVi – Sistema de Asistencia y Seguridad Vial,” *Res. Comput. Sci.*, vol. 76, pp. 51–60, 2014.
 - [38] D. G. de S. T. DGST, “Estadística accidentes de tránsito 2017, república mexicana,” 2018.
 - [39] FIWARE, “Device - Fiware-DataModels.” [Online]. Available: <https://bit.ly/2VYTdN7>. [Accessed: 21-May-2019].
 - [40] FIWARE, “Alert - Fiware-DataModels.” [Online]. Available: <https://bit.ly/30s5eZM>. [Accessed: 20-May-2019].

Anexo 1 – Funciones utilizadas por los algoritmos base

En esta sección se describen algunas de las funciones utilizadas por los algoritmos base. Las características especificadas para cada función son las siguientes:

- Entradas: se refiere a los datos que se deben ingresar para que el algoritmo pueda ejecutarse, los parámetros de entrada que son opcionales se incluyen dentro de corchetes.
- Salida: se refiere a la información que el algoritmo retorna, una vez que termina su ejecución.
- Tipo de desarrollo: Especifica si el algoritmo hace parte de una librería o es desarrollado dentro del proyecto.
- Firma: Forma como se hace mención al elemento en el pseudocódigo.

6.2.1.1 *Distancia entre dos puntos*

Esta función se utiliza para obtener la distancia que hay entre dos puntos ubicados en el sistema de referencia, la cual es de utilidad al momento de generar alertas, identificar la proximidad a una determinada área, calcular la distancia de frenado, etc. En la Tabla 11 se presentan las características generales de esta función.

Tabla 11. Características generales de la función distancia entre dos puntos.

Característica	Descripción
Parámetros de entrada	Punto A y Punto B
Parámetro de salida	Valor <i>d</i> : Distancia entre el punto A y el punto B
Tipo de desarrollo	Librería externa
Firma	distanciaEntrePuntos (<i>Punto A</i> , <i>Punto B</i>)

6.2.1.2 *Distancia de un punto a una línea.*

Esta función se utiliza para obtener la distancia más corta que hay entre una línea y un punto. Esta distancia se obtiene a partir de puntos geolocalizados. En la Tabla 12 se presentan las características generales de esta función.

Tabla 12. Características generales de la función distancia de un punto a una línea

Característica	Descripción
Parámetros de entrada	<i>Punto</i> , <i>línea</i> , [<i>tolerancia en metros</i>]
Parámetro de salida	Valor <i>d</i> : distancia en metros
Tipo de desarrollo	Librería externa
Firma	distanciaPuntoLinea (<i>Punto A</i> , <i>línea</i> , [<i>tolerancia en metros</i>])

6.2.1.3 Crear línea a partir de un conjunto de puntos.

Esta función permite crear una línea a partir de dos puntos, donde el primer y último punto se corresponden con el punto inicial y final de la línea; sus características generales se presentan en la Tabla 13.

Tabla 13. Características generales de la función línea a partir de dos puntos

Característica	Descripción
Parámetros de entrada	<i>punto inicial y punto final</i>
Parámetro de salida	Valor <i>l</i> : Línea recta definida por dos puntos
Tipo de desarrollo	Librería externa
Firma	<i>crearLinea (puntoInicial, puntoFinal)</i>

6.2.1.4 Punto dentro de un polígono.

Esta función permite establecer si un punto se encuentra dentro de un polígono el cual puede ser regular o irregular. La tolerancia en este caso se refiere a la cantidad de metros máxima fuera del polígono con respecto a su borde más cercano, en la cual el punto se considerará como dentro del polígono. En la Tabla 14 se presentan las características generales de la función.

Tabla 14. Características generales de la función punto dentro de un polígono

Característica	Descripción
Parámetros de entrada	<i>Punto, polígono, [tolerancia en metros]</i>
Parámetro de salida	verdadero si el punto está dentro del polígono y falso en cualquier otro caso
Tipo de desarrollo	Librería externa
Firma	<i>puntoEnPoligono (Punto, línea, [tolerancia=0])</i>

6.2.1.5 Punto dentro de una línea.

Esta función permite establecer si un punto se encuentra dentro de una línea o segmento. La tolerancia en este caso se refiere a la distancia máxima en metros desde el punto a la línea, en la cual el punto se considerará como dentro de la línea. En la Tabla 15 se presentan las características generales de la función.

Tabla 15. Características generales de la función punto dentro de una línea

Característica	Descripción
Parámetros de entrada	<i>Punto, línea, [tolerancia en metros]</i>
Parámetro de salida	verdadero si el punto está dentro del polígono y falso en cualquier otro caso
Tipo de desarrollo	Librería externa
Firma	<i>puntoEnLinea (Punto, línea, [tolerancia=0])</i>