

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA



TRABAJO DE FIN DE CARRERA (MTR280)

SISTEMA DE RECONOCIMIENTO DE ESTILO DE CONDUCCIÓN EN VEHÍCULOS DEL SISTEMA DE TRANSPORTE PÚBLICO DE LIMA

NOMBRE: Héctor David Aguirre Arista

PROFESORES: Ericka Patricia Madrid Ruiz
Elizabeth Roxana Villota Cerna

ASESORES: Jhon Portella Delgado
Elizabeth Roxana Villota Cerna

HORARIO: 10M2

CÓDIGO: 20131733

September 2018

Resumen

El reconocimiento de patrones usando algoritmos de machine learning e inteligencia artificial ha incrementado su popularidad debido a toda la información que actualmente tenemos disponible. Estas herramientas juegan un papel muy importante en el sector de transporte debido a su capacidad de poder optimizar el uso de recursos como el combustible o la energía eléctrica.

Esta optimización tiene beneficios en muchos ámbitos. Económicamente el usar de manera más eficiente el combustible o la energía eléctrica reduce costos. Ambientalmente el uso eficiente de combustible reduce las emisiones de CO₂, logrando reducir la contaminación en las ciudades. Por último, el estilo de conducción que usa eficientemente la energía esta asociado a un estilo de conducción menos agresivo y más seguro. Algo que es de vital importancia en países como el Perú en donde se producen cientos de muertes al año debido a accidentes de tránsito.

En la presente tesis, se buscará desarrollar un sistema capaz de reconocer el estilo de conducción de un chofer del transporte público de Lima, otorgándole feedback en tiempo real y monitorizándolo para que pueda mejorar su estilo de conducción y de esta manera se reduzca el costo del combustible utilizado, las emisiones generadas y se aumente la seguridad vial en la ciudad.

Este sistema será adaptable y escalable a cualquier tipo y número de vehículos, permitiendo así su implementación de una manera sencilla.

Índice

Lista de Figuras	vii
Lista de Tablas	ix
Introducción	1
1 Planteamiento de la Problemática	3
1.1 Motivación General	3
1.2 Objetivo general	4
1.3 Objetivos específicos	4
2 Estado del Arte	5
2.1 Términos relacionados con el estilo de conducción	5
2.2 Estado del arte según algoritmos usados	6
2.2.1 Algoritmos basados en reglas	6
2.2.2 Algoritmos basados en datos	8
2.3 Estado del arte según sensores usados	16
2.3.1 Acelerómetros de bajo costo	16
2.3.2 Smartphone	16
2.3.3 Inertial Measurement Unit	17
2.3.4 GPS	17
2.4 Feedback	17

3 Diseño Conceptual	19
3.1 Requerimientos del sistema	19
3.2 Modelo Black Box	22
3.3 Estructura de Funciones	22
3.3.1 Dominio mecánico	24
3.3.2 Dominio de energía	24
3.3.3 Dominio de sensores	25
3.3.4 Módulo de comunicación	25
3.3.5 Módulo de reconocimiento de estilo de conducción	25
3.4 Concepto de Solución	27
3.4.1 Dominio Mecánico	27
3.4.2 Dominio Energético	28
3.4.3 Dominios de Procesamiento y de Comunicación	29
3.4.4 Dominio de Sensores	29
3.4.5 Dominio de Interfaz	32
3.4.6 Dominio de Reconocimiento de estilo de conducción	32
3.5 Conceptos integrados de solución	33
3.5.1 Concepto integrado de solución 1	37
3.5.2 Concepto integrado de solución 2	38
3.5.3 Concepto integrado de solución 3	39
3.6 Evaluación de conceptos de solución	40
3.6.1 Evaluación técnica	40
3.6.2 Evaluación económica	41
3.6.3 Evaluación de soluciones	41
Bibliografía	43

Lista de Figuras

2.1	Terminología estilo de conducción	5
2.2	Resultados del sistema de reconocimiento de estilo de conducción	7
2.3	Tasa de consumo de combustible según diferentes estilos de conducción . .	9
2.4	Comparación de señales usando distancia Euclidiana y DTW	11
2.5	Algoritmo de <i>Dynamic Time Warping</i>	12
2.6	Topología de una Red Neuronal usada para Extreme Learning Machines . .	14
2.7	MPU-5050 de Invensense	17
3.1	Modelo de Black Box del sistema.	22
3.2	Estructura de Funciones.	23
3.3	Dominio mecánico de la estructura de funciones.	24
3.4	Módulo eléctrico de la estructura de funciones.	24
3.5	Módulo de sensores de la estructura de funciones.	25
3.6	Módulo de comunicación de la estructura de funciones.	26
3.7	Módulo de reconocimiento de estilo de conducción de la estructura de funciones.	26
3.8	Diagrama de evaluación	42

Lista de Tablas

2.1	Descripción de los clusters obtenidos	11
2.2	Porcentaje de reconocimiento de maniobras de conducción	13
2.3	Resumen de sensores usados en las investigaciones mencionadas	16
3.1	Lista de Requerimientos página 1.	20
3.2	Lista de Requerimientos página 2.	21
3.3	Matriz Morfológica del Dominio Mecánico	27
3.4	Conceptos de solución mecánicos propuestos	28
3.5	Matriz Morfológica del Dominio Energético	28
3.6	Conceptos de solución eléctricos propuestos	29
3.7	Matriz Morfológica de los Dominios de Procesamiento y Comunicación . .	30
3.8	Conceptos de solución propuestos del dominio de comunicación	30
3.9	Matriz Morfológica del Dominio de Sensores	31
3.10	Conceptos de solución propuestos del dominio de sensores	32
3.11	Matriz Morfológica del Dominio de Interfaz	32
3.12	Conceptos de solución propuestos del dominio de interfaz	33
3.13	Matriz Morfológica de Reconocimiento de estilo de conducción	34
3.14	Conceptos de solución propuestos del dominio de reconocimiento de estilo de conducción	35
3.15	Conceptos de solución propuestos	36
3.16	Evaluación técnica	40

3.17 Evaluación económica	41
3.18 Evaluación de soluciones	41

Introducción

El transporte es una actividad cotidiana que tiene un lugar muy importante en nuestras vidas. Sin embargo, en muchas ciudades esta actividad se realiza de manera ineficiente y esto tiene como consecuencia una larga lista de problemas, como el aumento de congestión vehicular, la disminución de la seguridad vial, el aumento de emisiones de CO₂, etc.

Uno de los principales motivos del transporte ineficiente es la conducta de los conductores al manejar. Según [1] el 80% de los peruanos conduce de manera agresiva y es el país con peor desempeño en conducción de 38 países analizados. Esta conducta agresiva está directamente relacionada con algunos de los problemas del transporte ineficiente.

Uno de ellos es el aumento de emisiones de CO₂. El transporte está liderado por el uso de combustibles fósiles y, aunque los autos eléctricos han empezado a ganar terreno en el sector automovilístico, estos aún representan tan solo el 1.34% de las ventas totales de autos en el mundo [2]. El transporte se registró como la fuente mas grande de emisión de CO₂ en el 2016 en Europa y Estados Unidos con el 27% y 28% de las emisiones de gases de efecto invernadero respectivamente. Entonces el transporte ya representa una gran parte de las emisiones de CO₂. Sin embargo, en [3] se demostró que un conductor agresivo puede consumir hasta 20 % más combustible que uno calmado, este consumo extra de combustible conlleva a un incremento del 50% en emisiones de CO₂. Se encuentra entonces en el estilo de conducción la oportunidad de reducir en gran medida las emisiones de este gas de efecto invernadero.

Por otro lado, otras de las consecuencias de la conducción agresiva es la seguridad vial. En el Perú en el año 2017 murieron un total de 772 personas debido a accidentes de tránsito [4]. Las mayores causas de los accidentes de tránsito en el Perú son el exceso de velocidad (32%) y la imprudencia del conductor (28%) [5]. Estas dos causas son errores humanos que están directamente relacionados con el estilo de manejo de los conductores.

Lo expuesto anteriormente convierte al estilo de conducción de un usuario en un factor relevante para la seguridad vial y para el cuidado del medio ambiente. Por lo que la presente

tesis desarrollará un sistema que sea capaz de caracterizar el estilo de conducción de un usuario para luego proporcionar un feedback que lleve al conductor a mejorar sus estilo de manejo y como consecuencia reducir las emisiones generadas, el combustible consumido e incrementar la seguridad durante la conducción.

1. Planteamiento de la Problemática

1.1. Motivación General

En un futuro, se espera que la mayoría de los autos sean autónomos, probablemente eléctricos y que el manejo del sistema de transporte este interconectado y sea inteligente. Para lograr esta integración del transporte se necesita un entendimiento profundo del comportamiento humano a la hora de realizar la tarea de conducir.

Este conocimiento no sólo será útil en el futuro, sino que en el presente nos ayudaría a tener vehículos que sean conducidos de una manera eficiente y más segura, reduciendo la cantidad de emisiones dañinas para el medio ambiente y la cantidad de accidentes de tránsito.

Por otro lado, actualmente no se cuenta con una forma de monitorizar el comportamiento de los conductores que trabajan en el transporte público. Una persona es capaz de darse cuenta fácilmente cuando el conductor realiza una maniobra temeraria y pone en riesgo la seguridad. Sin embargo, no existen muchos sistemas que puedan reconocer de forma automática o semiautomática este comportamiento.

El mismo hecho de monitorizar el estilo de manejo de un conductor de una manera objetiva y a lo largo de todo su recorrido hace que el conductor cambie su comportamiento al volante y tenga más cuidado a la hora de conducir.

Debido a todas las razones antes mencionadas, el desarrollo de un sistema de reconocimiento de estilos de conducción en Lima es importante.

1.2. Objetivo general

Se propone entonces el diseño de un módulo de reconocimiento de estilo de conducción que pueda adaptarse a cualquier vehículo y que entregue un feedback de acuerdo al estilo actual del conductor para su uso en el sistema de transporte público en Lima.

Este módulo utilizará algoritmos de reconocimiento de patrones y retornará una señal de feedback con el objetivo de advertir al conductor cuando este realice una maniobra agresiva que consuma energía innecesaria o que comprometa la seguridad vial. Además, el estilo de manejo de cada conductor será monitorizado enviando toda la información a un servidor para un análisis estadístico posterior.

Este sistema no sólo permitirá la adopción de un estilo de conducción más eficiente y seguro por parte de los conductores del sistema de transporte público; sino que además, brindará datos e información con la que no se cuenta actualmente y que podría ser utilizada para generar otros servicios tal como el seguimiento de los buses en tiempo real por parte de los usuarios.

1.3. Objetivos específicos

De acuerdo a lo expuesto anteriormente se establecen entonces los siguientes objetivos:

- Definir el concepto de estilo de conducción.
- Definir que parámetros de los vehículos se medirán y seleccionar los sensores que medirán estos parámetros.
- Diseñar un algoritmo o un conjunto de estos para reconocer el estilo de conducción de un conductor de acuerdo a la definición propuesta en esta tesis.
- Diseñar el mecanismo que será usado para entregar feedback al usuario.

2. Estado del Arte

2.1. Términos relacionados con el estilo de conducción

El concepto de "Estilo de conducción" no tiene una definición estándar que sea aceptada en todo el mundo. Al contrario, este concepto involucra una serie de factores que aumentan su complejidad y complican su definición. Debido a esto se usarán los conceptos definidos por Martinez et al. [6].

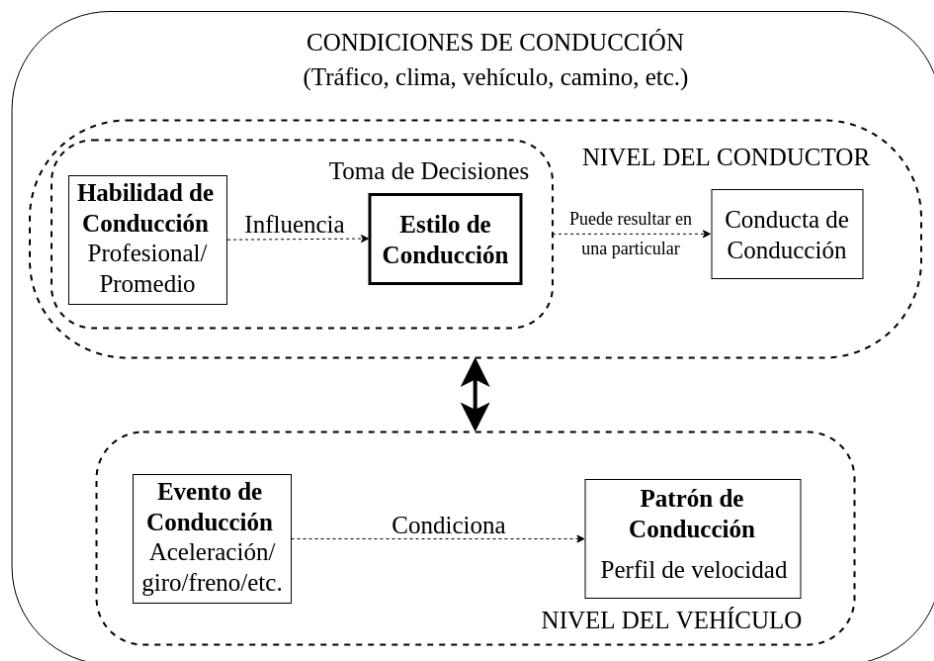


Fig. 2.1 Terminología relacionada con el estilo de conducción [6]

Como se aprecia en la Fig 2.1, se entiende por un evento de conducción como las maniobras que se usan durante la acción de conducir, como por ejemplo: acelerar, desacelerar y girar.

De la misma manera, El patrón de conducción se define como el resultado de los eventos de conducción sujetos a condiciones de manejo, como el clima o el tipo de calzada. Este resultado se puede expresar como un perfil de velocidades. en el que están incluidos todos los datos que se pueden obtener partiendo de este perfil de velocidad, como por ejemplo: duración del viaje, velocidad promedio y demanda de potencia calculada.

La habilidad de conducción es la habilidad que posee el conductor para controlar el vehículo. Este concepto se usa para diferenciar entre un conductor experimentado o profesional de un conductor promedio.

El estilo de conducción es más complejo de definir debido a que para algunos autores involucra factores subjetivos como la actitud del conductor, el humor o el cansancio. Para Dörr et al. [7], el estilo de conducción es la manera en la que la tarea de conducción es realizada. Esto se traduce en la forma en la que el conductor opera el vehículo (Pedal de aceleración, timón, freno, etc.). Esto se diferencia de el patrón de conducción tan solo porque no se asocia con un recorrido en específico sino con un conductor.

También, se puede expresar el estilo de conducción en niveles de agresividad como Aljaafreh et al. [8]. Debido a que la agresividad en los eventos de conducción esta asociada con un mayor consumo de combustible y a menor seguridad vial, definitivamente juega un papel importante dentro del concepto de estilo de conducción. Esta es la forma en la que se definirá el estilo de conducción para esta tesis.

2.2. Estado del arte según algoritmos usados

Se procederá a mostrar las implementaciones e investigaciones desarrolladas en la actualidad clasificados según el tipo de algoritmo que se uso para la caracterización de el estilo de conducción

2.2.1. Algoritmos basados en reglas

Dentro de esta categoría se encuentran algoritmos de clasificación basados en reglas que comprenden el uso de lógica difusa, lógica difusa adaptativa y algoritmos de agrupamiento. Estos algoritmos se caracterizan por estar definidos por *umbrales predefinidos* y son el enfoque más sencillo para tratar de clasificar los estilos de conducción.

Dörr et al. [7] desarrolló un sistema de reconocimiento de estilo de conducción online usando lógica difusa. Este sistema esta implementado usando *Matlab/Simulink* y es paramétrico.

Se puede configurar para ser usado en distintos tipos de vehículos. El sistema detecta la aceleración longitudinal, aceleración lateral, desaceleración, velocidad, brecha de tiempo y activación del sistema autónomo de velocidad crusero. Además determina a través de un Sistema de Navegación el tipo de calzada en el que se encuentra (Se distingue entre trocha, calles urbanas, carreteras pavimentadas y carreteras rurales). Esto lo realiza debido a que el tipo de calzada influencia en gran medida al estilo de conducción. Por ejemplo, en una trocha la mayoría de conductores manejarían a una velocidad suave para no dañar al vehículo, lo cual no necesariamente quiere decir que el mismo conductor conduzca de esa manera en otro tipo de escenarios. Usando Lógica Difusa se definieron 3 niveles de estilo de conducción: *Normal*, *Confortable* y *Deportivo*. Se probó el sistema en una simulación y se obtuvieron los siguientes resultados promedio (Fig. 2.2): 67.8% de clasificaciones correctas y solo 2% de clasificaciones incorrectas (El otro 30.2% pertenece a clasificaciones "Diferentes" que dan una clasificación contigua de estilo de manejo, por ejemplo cuando la clasificación real es *Normal* pero el sistema arroja como resultado *Confortable*, en cambio las clasificaciones incorrectas dan un resultado no contiguo al real)

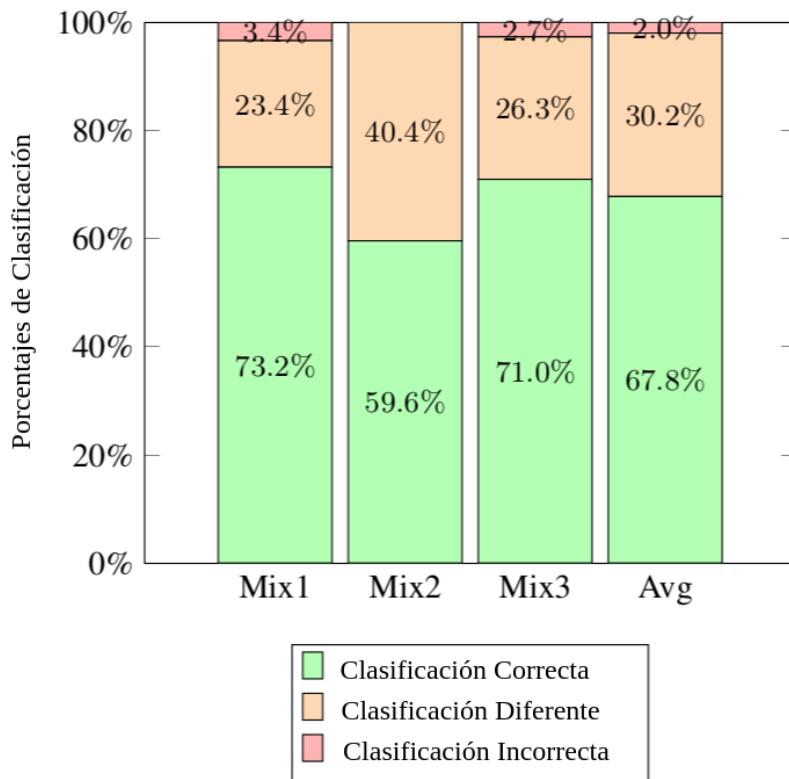


Fig. 2.2 Resultados del sistema de reconocimiento de estilo de conducción en [7]

Usar lógica difusa no es la única manera de clasificar estilos de conducción. Murphrey et al. [9] presentó un método de clasificación basado en el análisis del perfil de la sobreaceleración (la derivada de la aceleración respecto al tiempo). En su investigación define al estilo de manejo como un comportamiento dinámico, Un conductor puede manejar de forma calmada por un momento y de forma agresiva en otro momento. Por este motivo, el método de clasificación que propone predice el estilo de conducción del usuario en tiempo real. El algoritmo funciona de la siguiente manera:

1. Calcula el perfil de sobreaceleración durante una ventana de tiempo predefinido.
2. Calcula la desviación estándar de la sobreaceleración durante toda la ventana de tiempo.
3. Detecta el tipo de calzada actual y el nivel de congestión de tráfico usando el algoritmo propuesto en [10].
4. Calcula la proporción de sobreaceleración dividiendo la desviación estándar con un valor promedio que depende de el tipo de calzada y el nivel de tráfico actuales.
5. Dependiendo del resultado de la división realizada el conductor puede ser clasificado como *Calmado*, *Normal* o *Agresivo* usando reglas con umbrales predefinidos.

Este resultado dependerá mucho de la duración definida para la ventana de tiempo. Se recomendó usar una ventana de duración de 6 a 9 segundos para detectar los cambios de estilo de conducción oportunamente.

Se realizó también una comparación entre los 3 diferentes estilos de conducción con respecto a la tasa de consumo de combustible. Como se puede apreciar en la Fig. 2.3, los conductores clasificados como calmados están asociados a un menor consumo de combustible, mientras que los agresivos a un mayor consumo de combustible.

Se ha presentado dos algoritmos basados en reglas que usan tanto múltiples variables [7] como una sola [9] para predecir el estilo de conducción del usuario. Sin embargo, los resultados de estos sistemas dependen en gran medida del valor de los umbrales escogidos para realizar la clasificación.

2.2.2. Algoritmos basados en datos

Cuando se tiene una cantidad muy grande de datos, es difícil analizarlos para obtener las reglas que guíen al algoritmo. Es entonces cuando los algoritmos basados en datos entran en acción. Estos algoritmos, en comparación con los basados en reglas, pueden manejar

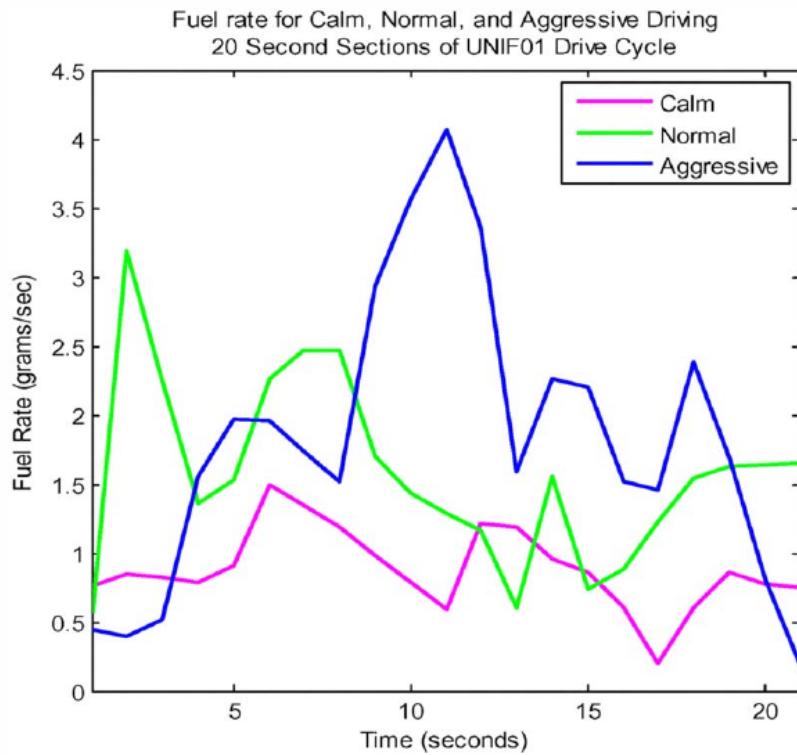


Fig. 2.3 Tasa de consumo de combustible según diferentes estilos de conducción [9].

una mayor cantidad de datos y deducir umbrales consistentes con estos datos sin que estos dependan de un experto en el tema. A continuación se presentarán los distintos tipos de algoritmos basados en datos y su uso en el reconocimiento de estilos de conducción.

A. Algoritmos de aprendizaje de máquinas no supervisados

Los algoritmos no supervisados no necesitan que los datos obtenidos por medio de los sensores en la actividad de conducción estén etiquetados. Es decir, si se registra los datos de un ciclo de conducción para un conductor, este registro no necesita estar asociado con el estilo de conducción que mantuvo este conductor. Esta clase de algoritmos busca patrones en los datos y los separa teniendo en cuenta sus similitudes y diferencias en diferentes grupos sin etiquetar.

Constantinescu et al. [11] propuso el análisis de el estilo de conducción por medio de dos algoritmos: Hierarchical Cluster Analysis (HCA) y Principal Component Analysis (PCA).

HCA es un algoritmo de *machine learning* que trata de categorizar a cada individuo en grupos con características similares. Es muy útil cuando no se conoce con exactitud el

número de grupos. Por medio de un análisis estadístico trata de minimizar la varianza dentro de cada grupo y a la vez aumentarla entre diferentes grupos. En este caso, el algoritmo de agrupamiento jerárquico consiste en empezar con grupos consistentes de un solo miembro y consecutivamente ir combinando los dos grupos más cercanos, hasta solo tener un grupo.

PCA es un método estadístico que usa una transformación para convertir un posible conjunto de variables relacionadas entre si en otro conjunto menor de variables linealmente independientes, en las que la varianza sea maximizada.

Para este caso en particular se trabajo con dispositivo GPS que se encargaba de medir la velocidad y aceleración del vehículo monitorizado a una frecuencia de muestreo de 1 Hz. De esta data se escogieron los siguientes parámetros:

1. Velocidad por encima de los 60 Km/h.
2. Velocidad promedio.
3. Desviación estándar de la velocidad.
4. Desviación estándar de la aceleración.
5. Promedio de la aceleración positiva.
6. Desviación estándar de la aceleración positiva.
7. Promedio de la aceleración negativa o frenado.
8. Desviación estándar del frenado.
9. Trabajo mecánico, que se calculó sumando toda la energía cinética positiva requerida para aumentar la velocidad del vehículo.

Luego del análisis realizado se encontraron 6 grupos o clusters y se describió cada uno de estos como se puede observar en el Tabla 2.1 usando los 5 componentes principales resultantes al aplicar PCA.

Los algoritmos no supervisados tienen una gran ventaja, no se necesitan conocer las etiquetas de clasificación *a priori*. Esto significa que la definición de etiquetas no limita o influencia de alguna manera a los resultados obtenidos, ya que los grupos o clusters formados dependen solo de los datos usados. Sin embargo, luego de hallar los grupos se necesita realizar un análisis para determinar su naturaleza.

Tabla 2.1 Descripción de los clusters obtenidos en [11].

Cluster	Agresividad	Velocidad	Aceleración	Frenado
1	Moderadamente baja	Baja-Moderada	Moderada	Suave-Moderado
2	Muy baja	Baja-Moderada	Baja-Moderada	Suave-Moderado
3	Moderadamente alta	Moderada	Moderada	Repentino
4	Neutral	Moderada	Alta	Moderado
5	Neutral	Moderada-Alta	Baja-Moderada	Moderado-Repentino
6	Alta	Alta	Alta	Repentino

B. Algoritmos de aprendizaje de máquinas supervisados

Los algoritmos supervisados, a diferencia de los no supervisados, necesitan tener un conocimiento previo de los datos que se tienen. Es decir, cada uno de los datos recolectados deben estar asociados a una etiqueta o clase ya predefinida.

Uno de los algoritmos más utilizados es el de K-Nearest Neighbors (k-NN), que se utilizó junto a Dynamic Time Warping (DTW) en el sistema de reconocimiento propuesto por Johnson and Trivedi [12]. El objetivo de este sistema es reconocer maniobras agresivas para dar un feedback al conductor y monitorizarlo, mejorando así la seguridad vial. Para lograrlo usaron los sensores presentes en un smartphone (acelerómetro, giroscopio, GPS y cámara frontal). Este sistema trabaja grabando la información de todos los sensores durante 5 minutos, luego analiza si ha ocurrido alguna maniobra potencialmente agresiva. Si una maniobra de este tipo ocurrió, conserva la información para un análisis posterior. En cambio, si no ocurrió ninguna maniobra de este tipo, borra la data para ahorrar espacio en disco.

Para la clasificación de cada maniobra detectada se usa Dynamic Time Warping (DTW), el cual es un algoritmo que es capaz de analizar la similaridad entre dos señales que no necesariamente tengan la misma duración y probablemente tengan un desfase. En la Fig. 2.4 se puede observar una comparación entre una comparación de una distancia Euclidiana y otra usando DTW en dos señales muy parecidas en forma pero que no se encuentran en fase.

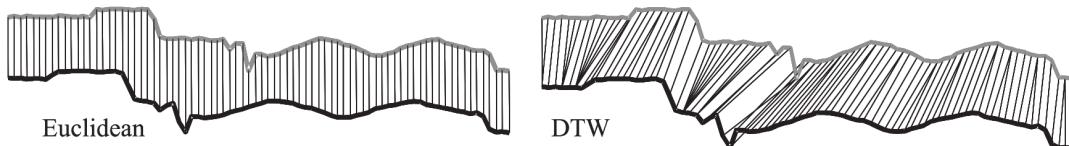


Fig. 2.4 Comparación de señales usando distancia Euclidiana y DTW [13].

El algoritmo consiste en crear una matriz de deformación $n \times m$, siendo n y m los números de puntos en cada señal. Esta matriz se llena calculando distancias entre cada punto, sin importar si se encuentran en el mismo instante temporal. Luego se dibuja el camino con la distancia más corta que une el inicio de las señales con el final de estas Fig. 2.5 (B). La suma de distancias en este camino se define como la distancia entre las dos señales analizadas y se puede usar para comparar la similitud y clasificar señales usando modelos. Información detallada sobre este algoritmo se puede encontrar en [13].

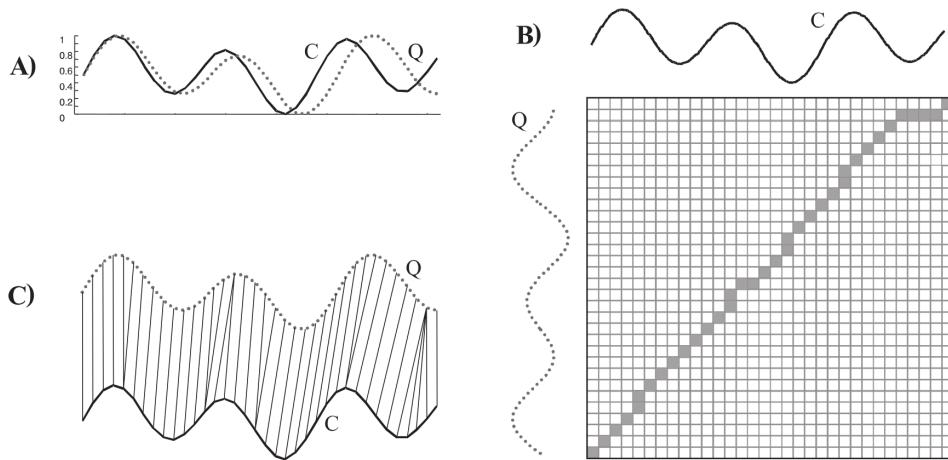


Fig. 2.5 **A)** Dos señales con forma muy similar pero desfasadas. **B)** Matriz de deformación usada para encontrar el camino de deformación. **C)** El resultado del alineamiento y deformación usando DTW. [13].

Para poder usar DTW se necesitó contar con señales modelo para cada tipo de maniobra a detectar. Se tuvo un total de 120 señales modelo en total, cada una asociada con una maniobra específica. Usando DTW se puede obtener una medida de distancia entre las señales a clasificar y las señales modelo. Luego se aplicó K-Nearest Neighbors (k-NN) para clasificar cada señal. Lo que hace k-NN es tener en guardadas las señales modelos y al momento de clasificar una nueva señal calcula la distancia con las señales modelos, selecciona las k distancias menores y dependiendo de la clase que tenga más señales cercanas, esta nueva señal es clasificada.

Para realizar la clasificación de señales solo se utilizó los datos de g_x , a_y y e_x (giroscopio en x , aceleración en y y ángulo de rotación de Euler en x respectivamente). Se puede observar en el Tabla. 2.2 los porcentajes de reconocimiento exitoso por cada maniobra.

Otro algoritmo aplicado usualmente al reconocimiento de estilos de manejo es el uso de *Redes Neuronales*. Echanobe et al. [14] utilizó redes neuronales para clasificar estilos de manejo y *Algoritmos Genéticos* para optimizar estas redes e incluso elegir los parámetros

Tabla 2.2 Porcentaje de reconocimiento exitoso de maniobras de conducción [12].

Maniobra	Porcentaje de reconocimiento exitoso (%)
Giro a la derecha (90°)	92
Giro a la izquierda (90°)	83
Vuelta en U (180°)	77
Giro a la derecha agresivo	100
Giro a la izquierda agresivo	100
Vuelta en U agresiva	100
Cambio de carril agresivo (derecha)	100
Cambio de carril agresivo (izquierda)	83

de entrada. El sistema que propuso se enfocaba en obtener un clasificador que pueda ser utilizado por un sistema embebido de potencia baja. Por lo cuál utilizo un tipo de red neuronal llamado *Extreme Learning Machines*. Este tipo de red neuronal, como se observa en la Fig. 2.6, se caracteriza por tener tan solo una capa oculta entre las entradas y las salidas. Además se eligen los pesos entre las entradas y la única capa oculta de manera aleatoria y nunca se actualizan. Esta red neuronal solo aprende los pesos entre la capa oculta y las salidas en una sola iteración. Por esta razón este tipo de red resulta siendo muy ligero y de entrenamiento muy veloz.

Echanobe et al. recolectó un conjunto de 21 variables de 11 conductores, pero no utilizó toda la información. En lugar de eso utilizó un algoritmo genético llamado *Non-Dominated Sorting Genetic Algorithm*. Este tipo de algoritmo genético realiza una optimización multiobjetivo. Lo cual significa que trata de optimizar más de una variable a la vez. En este caso se trata de minimizar 3 variables: el número de atributos de entrada para la red neuronal, el número de neuronas en la capa oculta y la tasa de error resultante de la red neuronal entrenada.

Normalmente no existe una solución que permita minimizar las 3 variables a la vez. Debido a esto, lo que se busca en realidad se conoce como *Soluciones Óptimas de Pareto*. Este término hace referencia a las soluciones en las que no se puede optimizar ninguna variable sin degradar a otra. Este tipo de algoritmos genéticos contiene el concepto de Soluciones óptimas de Pareto como función de fitness, de tal manera que permite optimizar las 3 variables antes mencionadas.

Finalmente, se escoge una solución que tiene un 90.1% de tasa de reconocimiento, 8 entradas para la red neuronal y 16 neuronas en la capa oculta.

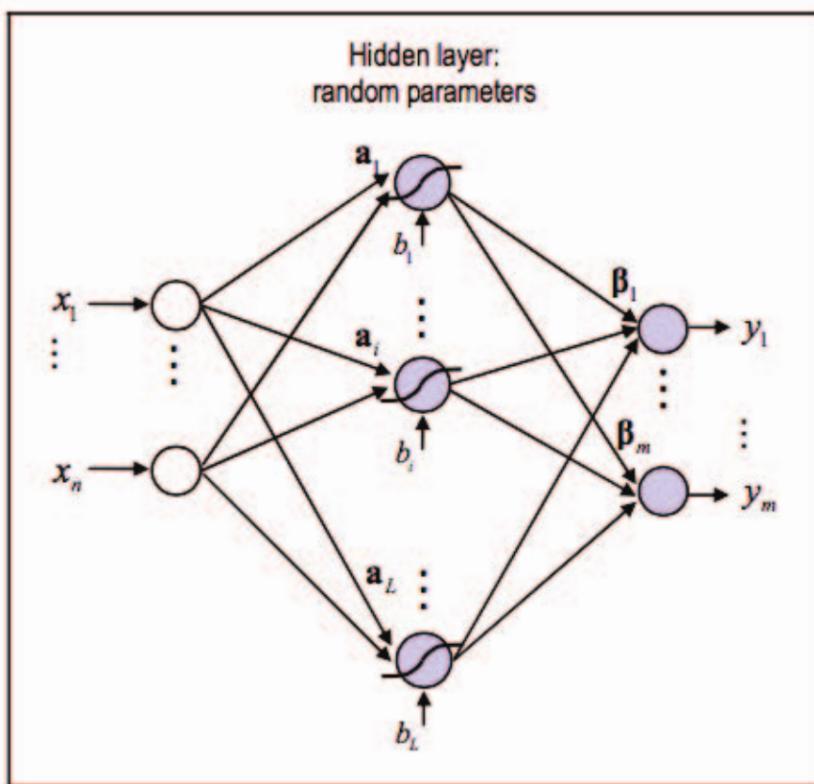


Fig. 2.6 Topología de una Red Neuronal usada para Extreme Learning Machines [14].

C. Algoritmos que combinan los enfoques supervisados y no supervisados

Los algoritmos supervisados y no supervisados aparecen como dos categorías totalmente separadas. Sin embargo, esto no significa que no se puedan usar juntos y combinar para mejorar sus ventajas o reducir sus desventajas.

Ly et al. [15] propuso un sistema que utiliza los sensores iniciales embebidos en un auto para identificar posibles maniobras inseguras y proporcionar un adecuado feedback hacia el conductor. Además también propone caracterizar y diferenciar conductores solo usando datos de los sensores iniciales, para poder diferenciar cuando dos personas distintas utilizan un mismo vehículo. Para lograr su objetivo primero analiza la información de los sensores para determinar un conjunto básico de maniobras: frenado, aceleración y giro.

Se utiliza el algoritmo de aprendizaje no supervisado *K-means Clustering* para identificar las maniobras. Este algoritmo consiste en generar k semillas, de preferencia no aleatorias ya que la inicialización tiene una gran influencia en los clusters resultantes, que representarán la media o el centroide de cada cluster, luego por medio de la definición de una métrica de distancia (Euclídea, citiblock, etc.) se asignan los datos al cluster del centroide más cercano. Luego de clasificar a todos los datos se vuelve a calcular el centroide de cada cluster (que muy probablemente cambiará de posición) y se repite los pasos hasta converger en una posición de centroides.

Al identificar las maniobras se usan dos fuentes de datos distintas: la primera cuenta con la información completa de los sensores, así como también un análisis estadístico de estos (mínimo, máximo, media, varianza, etc.); en cambio, la segunda solo cuenta con el análisis estadístico. Y los resultados que se obtienen son muy similares. El desempeño de el clasificador no se reduce al no incluir los datos completos de los sensores, sino que puede usar tan solo sus datos estadísticos.

Luego se utiliza el algoritmo supervisado *Support Vector Machines* (SVM). Este algoritmo usa los datos con sus respectivas clases y trata de crear un hiperplano que separe todos los datos pertenecientes a una clase de la otra. Mientras este hiperplano tenga una mayor distancia con los miembros más cercanos de ambas clases, se tendrá un menor error de generalización. El encontrar este hiperplano que separe a las dos clases en la mayoría de los casos no es posible en el espacio definido por los atributos actuales de los datos. Debido a esto se mapean los datos en un espacio con más dimensiones para lograr encontrar este hiperplano de una manera más sencilla.

2.3. Estado del arte según sensores usados

Para usar los algoritmos mencionados anteriormente se han usado diferentes clases de sensores para medir distintas variables, ya sea directa o indirectamente. En el Tabla 2.3 se puede apreciar los sensores utilizados en las investigaciones mencionadas.

Tabla 2.3 Resumen de sensores usados en las investigaciones mencionadas

Sensores usados	Referencias
Inertial Measurement Unit (IMU)	[9], [14], [15]
Acelerómetros de bajo costo	[8]
Smartphone	[12]
GPS	[11]
GPS e IMU	[7]

2.3.1. Acelerómetros de bajo costo

Los acelerómetros nos permiten medir la aceleración longitudinal y lateral, que son las que nos permiten medir eventos de conducción como el frenado y la aceleración o los giros. Con un acelerómetro de 2 ejes como el usado en 6294318 es suficiente para medir estas dos aceleraciones.

Estos acelerómetros pueden costar desde \$2, como el MMA6910KQ [16] que consta de 3 ejes y puede tener un rango de ± 3.5 o ± 5.0 g.

2.3.2. Smartphone

Los smartphones tienen cada vez más y más sensores que pueden ser usados para aplicaciones como las de la presente tesis. En [12] se usó los sensores integrados en un Iphone 4 para realizar la tarea de reconocimiento de estilo de conducción. Estos sensores fueron la cámara trasera (solo para registro), el acelerómetro de 3 ejes, el giroscopio de 3 ejes y el GPS.

2.3.3. Inertial Measurement Unit

Esta clase de sensores se puede encontrar bajo un rango de precios muy amplio. Existen unidades por debajo de los \$10 como el MPU-6050 (Fig. 2.7) que cuenta con un giroscopio de 3 ejes y un acelerómetro de 3 ejes; y funciona a un voltaje de 2.3 V a 3.4 V [17]

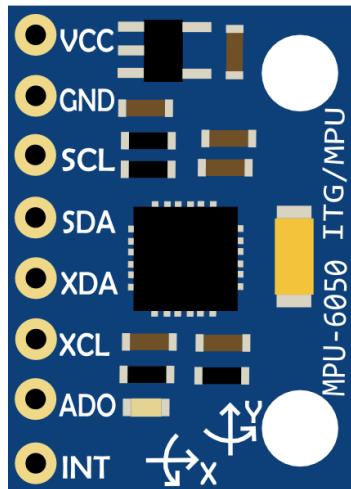


Fig. 2.7 MPU-6050 de Invensense [17].

2.3.4. GPS

El GPS es muy usado también al reconocer estilos de conducción ya que aporta información muy relevante al sistema, por ejemplo se puede usar la información del GPS para detectar el tipo de camino en el que se encuentra el auto en ese momento. Además se puede usar también para obtener la velocidad del auto, aunque es más usado en conjunto con un IMU para que al combinar la información de ambos sensores se usen algoritmos de *Sensor Fusion* para mejorar el resultado de las mediciones.

2.4. Feedback

Una de los elementos más importantes de los sistemas de reconocimiento de estilos de conducción antes mencionados son las señales de feedback que se le dan al conductor. Dependiendo de cuál sea el enfoque del reconocimiento de estilo de conducción (enfoque en ahorro de combustible o enfoque en agresividad de maniobras).

Hay 3 maneras principales de generar este feedback. De manera visual, háptica, audible o una combinación de estos. Cada una presenta ventajas y desventajas.

Para generar feedback de manera visual se pueden incluir Human Machine Interfaces (HMI), que básicamente son pantallas que permiten al conductor saber que estilo de conducción se encuentran ejerciendo en ese momento. Esto se podría lograr usando un smartphone o una tablet también. La desventaja de usar feedback visual es el conductor ya procesa mucha información visual al llevar a cabo la tarea de conducir, por lo que agregar más información requiere que el conductor preste atención al dispositivo y reduzca su atención en el camino. Esto podría presentar problemas de seguridad como se señala en [18]

En [19] se usó una tablet como método de feedback para entregar *Tips de Conducción* al usuario. El sistema funcionaba de la siguiente manera: Reconocía el estilo de conducción actual del conductor y se usaba *Learning Path Planning* para poder llegar desde el estilo de conducción detectado a uno óptimo en el que se consuma menos combustible. Obtenido esto, se procedían a mostrar los tips de conducción al conductor usando la tablet. En este sistema se lograba incrementar la economía del combustible en un 6.25%.

Para generar señales auditivas se puede utilizar un pequeño parlante que haga sonar una alarma cuando se registra un comportamiento no deseado del conductor (Estilo de conducción con alto consumo de combustible o muy agresivo). Este método

Por último, las señales hápticas han sido investigadas debido a que es el único canal de información por el cual los conductores no reciben mucha información de la tarea de conducir. Lo que quiere decir, que generaría menos distracciones que los otros tipos de feedback. En [20] se expone diferentes maneras de entregar esta señal (En el timón, el panel frontal del auto, el asiento , etc.).

3. Diseño Conceptual

3.1. Requerimientos del sistema

Uno de los requerimientos principales del sistema es el uso de una red de sensores modular y sencilla de instalar en distintos tipos de vehículos. Esto se explica fácilmente debido a que este sistema esta orientado a ser usado en el sistema de transporte público en Lima, el cual tiene una gran variedad y un gran número de vehículos. El sistema, entonces, debe ser adaptable a cualquiera de estos vehículos y no contar con un tiempo de instalación elevado.

También se tiene como requerimiento un grado de precisión alto en el reconocimiento de estilos y eventos de conducción. Esto es necesario debido a que este sistema se usará para monitorizar y mejorar la conducta de conducción de los choferes del servicio de transporte público. Los datos arrojados por este sistema necesitan ser confiables.

Por último, se necesita también de una adecuada infraestructura que permita que el sistema funcione en linea, entregando feedback al conductor, aún si se presentan condiciones como la falta temporal de conexión a Internet. Este requerimiento será importante al considerar donde realizar el procesamiento de los datos (dentro de un sistema embebido en cada vehículo o usando un servidor).

Tabla 3.1 Lista de Requerimientos página 1.

<i>Lista de requerimientos del Sistema</i>		Página 1 de 2 Edición: Rev. 1 (11/09/18)
Fecha de modificación	Descripción	
11/09/18	FUNCIÓN PRINCIPAL Reconocer el estilo de manejo de un conductor del sistema de transporte público de Lima y otorgar una señal de retroalimentación de acuerdo a su estilo de conducción actual.	
11/09/18	GEOMETRÍA <ul style="list-style-type: none"> • Alto < 30 cm. • Ancho < 30 cm. • Largo < 30 cm. 	
11/09/18	CINEMÁTICA No se tendrá movimiento.	
11/09/18	PESO Menor a 1 kg.	
11/09/18	MATERIALES Case: Plástico.	
11/09/18	ENERGÍA <ul style="list-style-type: none"> • Conexión de 12 VDC desde toma del auto. • 5 VDC para el microcontrolador. 	
11/09/18	SEÑALES Entrada: <ul style="list-style-type: none"> • Accionamiento de encendido y apagado. • Movimiento del vehículo. Salida: <ul style="list-style-type: none"> • Estilo de conducción. • Calor. 	
11/09/18	COMUNICACIONES <ul style="list-style-type: none"> • Conexión a Internet 3G, WiFi, etc. • Protocolo MQTT o CoAP con un servidor central. • Comunicación continua de datos de sensores. 	

Tabla 3.2 Lista de Requerimientos página 2.

<i>Lista de requerimientos del Sistema</i>		Página 2 de 2 Edición: Rev. 1 (11/09/18)
Proyecto:	Diseño de un sistema de reconocimiento de patrones de manejo	
Cliente:	Pontificia Universidad Católica del Perú	
Fecha de modificación	Descripción	
11/09/18	SENSORES <ul style="list-style-type: none"> • Sensores de movimiento. • Sensor de posición. 	
11/09/18	CONDICIONES DE OPERACIÓN Temperatura ambiente.	
11/09/18	SEGURIDAD <ul style="list-style-type: none"> • Se seguirá la norma ISO 16121 que regula el ambiente de la cabina de manejo. • El sistema no obstaculizará la visibilidad del conductor. • El sistema estará fijo y estará protegido de impactos que el vehículo podría tener. 	
11/09/18	ERGONOMÍA El sistema estará fuera de la vista del conductor.	
11/09/18	MANTENIMIENTO Mantenimiento preventivo cada 6 meses.	
11/09/18	SOFTWARE <ul style="list-style-type: none"> • Algoritmos de Machine Learning. • Programación usando C. 	
11/09/18	ENSAMBLE Instalación modular.	
11/09/18	COSTO Costo de fabricación: \$1000.	
11/09/18	PLAZO DE ENTREGA Entrega planeada para el 28/11/18.	

3.2. Modelo Black Box

En la Fig 3.1 se puede observar el modelo de Black Box o Caja Negra del sistema. En este modelo se pueden apreciar con mayor claridad las entradas y salidas del sistema

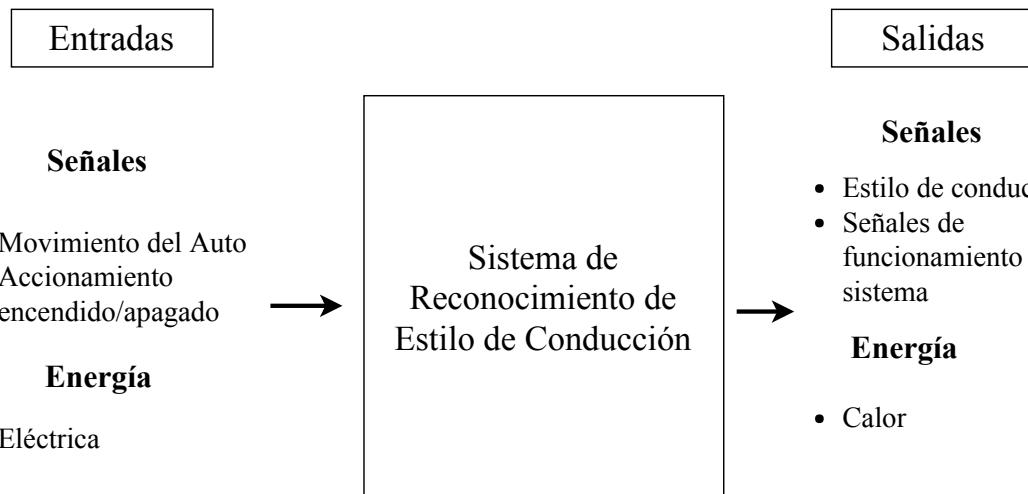


Fig. 3.1 Modelo de Black Box del sistema.

El sistema estará conectado al auto, de donde obtendrá la energía necesaria para funcionar. Debido a que el enfoque del sistema es ahorrar energía. Su consumo debe ser reducido.

Además de recibir la energía, este sistema procesará el movimiento del auto usando sensores obteniendo así toda la información necesaria para caracterizar el estilo de conducción del usuario. El estilo de conducción será almacenado y será representado como una señal de feedback al usuario. Además el sistema deberá indicar al usuario que se está funcionando.

3.3. Estructura de Funciones

La estructura de funciones del sistema se puede apreciar en la Fig 3.2. Para el elaboramiento de esta se han considerado 6 Dominios: Mecánica, Energía, Sensores, Comunicación, Reconocimiento del estilo de conducción e Interfaz.

Se describirá a continuación cada uno de estos dominios para así obtener una visión general del funcionamiento total del sistema.

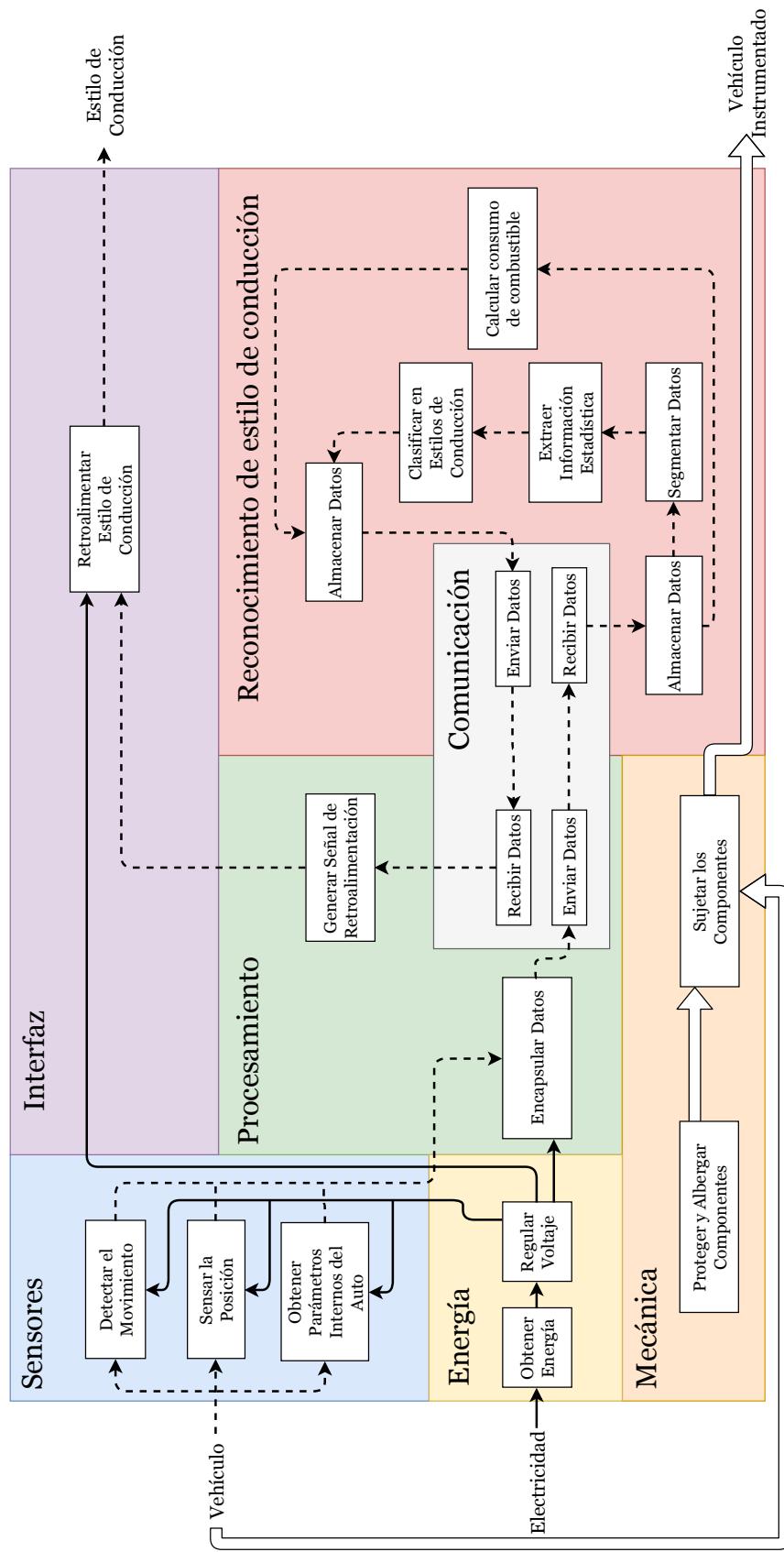


Fig. 3.2 Estructura de Funciones.

3.3.1. Dominio mecánico

En este dominio Fig 3.3 se encuentra la función de proteger y albergar todos los componentes del sistema. Se debe tener en cuenta que la velocidad relativa entre el vehículo y el sistema debe ser nula para que las medidas de los sensores sean precisas.

Además es importante también proteger al sistema de elementos como el polvo y la humedad. Por lo que se ejercerá la función de proteger al sistema.



Fig. 3.3 Dominio mecánico de la estructura de funciones.

3.3.2. Dominio de energía

Este dominio Fig 3.4 cumplirá la función de obtener la energía eléctrica necesaria para el funcionamiento del sistema y regular el voltaje, entregándole a cada módulo esta energía según sus requerimientos específicos. Los módulos que recibirán la energía serán los de Comunicación, Sensores e Interfaz.

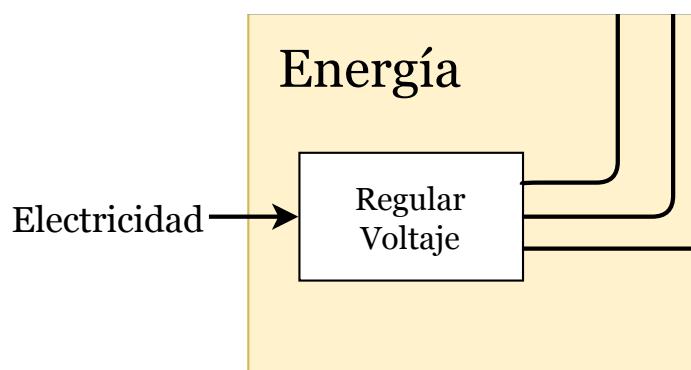


Fig. 3.4 Módulo eléctrico de la estructura de funciones.

3.3.3. Dominio de sensores

Este módulo Fig 3.5 realizará la función de recibir la información del movimiento del vehículo y transmitir esa información al módulo de comunicaciones que adquirirá todos los datos. Estos sensores serán tanto de movimiento, como de posición, pero además de estos se usarán los sensores internos del vehículo si estos se encuentran disponibles.

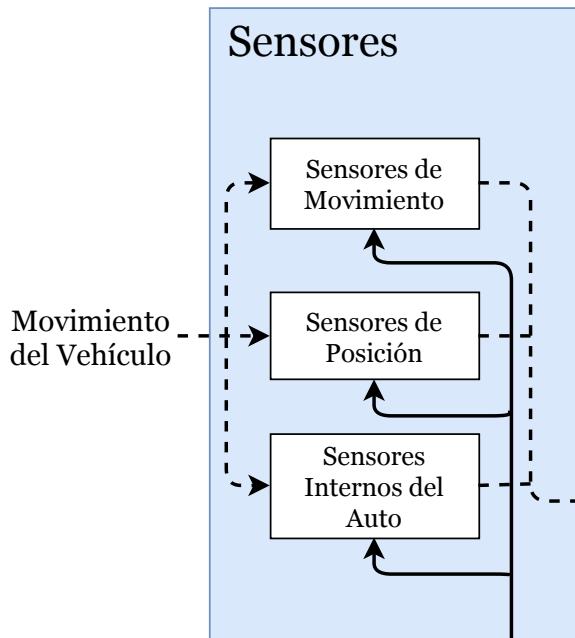


Fig. 3.5 Módulo de sensores de la estructura de funciones.

3.3.4. Módulo de comunicación

En este módulo Fig 3.6 se recopilaran los datos de

3.3.5. Módulo de reconocimiento de estilo de conducción

En este módulo se realizará el procesamiento de la data que fue enviada por el MCU. Para realizar esto se realizará un preprocesamiento de la data para luego procesarla usando algoritmos de machine learning.

Cuando la data haya sido procesada, se tendrá el estilo de conducción resultante que será enviado de vuelta al MCU por el servidor. Sin embargo, la data usada para obtener este resultado se conservará. Esta data se almacenará en el servidor para poder hacer análisis

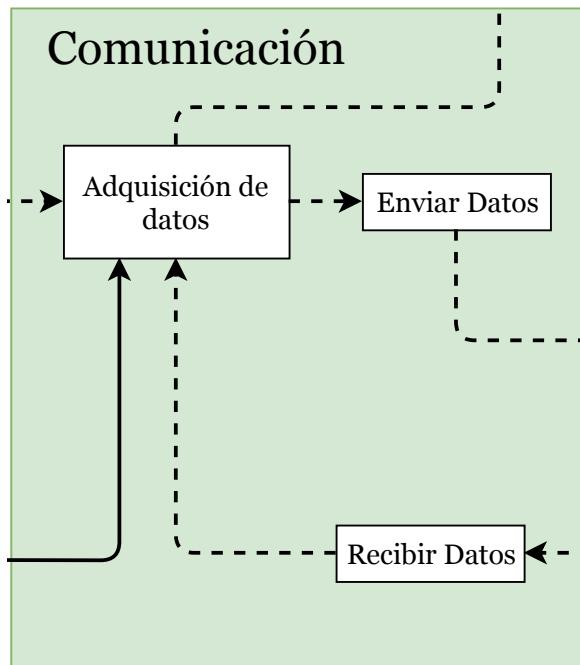


Fig. 3.6 Módulo de comunicación de la estructura de funciones.

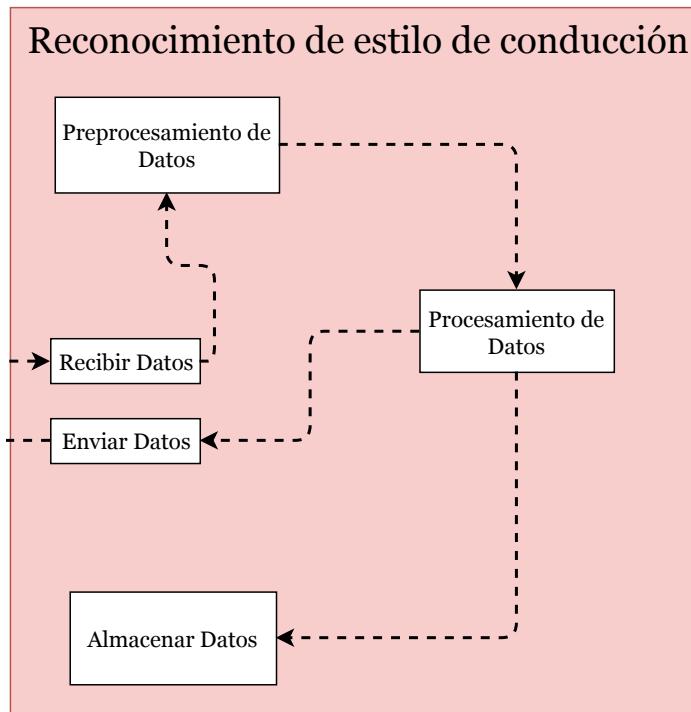


Fig. 3.7 Módulo de reconocimiento de estilo de conducción de la estructura de funciones.

offline o implementar otros servicios que puedan hacer uso de la información disponible, como un seguimiento en tiempo real de los buses, un monitoreo de los conductores, etc.

3.4. Concepto de Solución

En esta sección se propondrán conceptos de solución por cada dominio de la estructura de funciones mostrada anteriormente. Cada función tendrá distintas alternativas portadores de solución que al combinarse formarán una solución completa.

3.4.1. Dominio Mecánico

Se presenta a continuación la matriz morfológica del dominio mecánico en la Tabla 3.3 y la descripción de los conceptos de solución propuestos en la Tabla 3.4.

Tabla 3.3 Matriz Morfológica del Dominio Mecánico

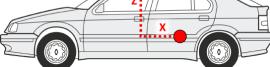
Funciones Parciales	Portadores de Solución		
	1	2	3
Proteger y Albergar Componentes	 Case en el panel frontal	 Case en el techo del vehículo	
Sujetar los Componentes	 Usando un sujetador para el case del dispositivo	 Atornillado a un sujetador que se posee una base magnética	

Tabla 3.4 Conceptos de solución mecánicos propuestos

<i>Conceptos de Solución Propuestos</i>	
Opción 1 →	Se posiciona el dispositivo en un case de plástico situado en el panel frontal del auto. Este case se sujetará por medio de un sujetador mecánico, para asegurar que no exista movimiento relativo entre el case y el vehículo. En el panel frontal se tiene una suficiente cobertura para las antenas de GPS y GPRS.
Opción 2 →	Se posiciona el dispositivo en un case de plástico situado en el techo del vehículo. Este case se sujetará siendo atornillado a sujetador que posee una base magnética que asegura su inmovilidad. Además esta solución es genial ya que permite que las antenas del GPS y GPRS esten en una óptima posición.
Opción 3 →	Se posiciona el dispositivo en el panel frontal del auto. Como en este caso el dispositivo es un smartphone, este será sujetado por un sujetador de smartphone como el de la opción 1.

3.4.2. Dominio Energético

Se presenta a continuación la matriz morfológica del dominio eléctrico en la Tabla 3.5 y la descripción de los conceptos de solución propuestos en la Tabla 3.6.

Tabla 3.5 Matriz Morfológica del Dominio Energético

Funciones Parciales	Portadores de Solución		
	1	2	3
Obtener Energía	 Puerto de 12V en los autos	 Batería de Litio	
Regular Voltaje	 Regulador Lineal	 Regulador Switching	 Cargador de Smartphone

Tabla 3.6 Conceptos de solución eléctricos propuestos

<i>Conceptos de Solución Propuestos</i>	
Opción 1 	Se usará el puerto de 12 V que se encuentra presente en la gran mayoría de los autos. A continuación se usará un regulador switching que se encargará de regular el voltaje a los requerimientos de los componentes de los dominios de comunicación e interfaz
Opción 2 	Se usará una batería de ion de litio que energizará a los componentes de los demás dominios. Se usará un regulador lineal para regular el voltaje que se entregará a los componentes de los demás dominios.
Opción 3 	Se usará el puerto de 12 V que se encuentra presente en la gran mayoría de los autos. Y se conectarán a este un cargador de smartphone. De esta manera se asegurará de que el smartphone siempre tenga energía.

3.4.3. Dominios de Procesamiento y de Comunicación

Se presenta a continuación la matriz morfológica del dominio de comunicación en la Tabla 3.7 y la descripción de los conceptos de solución propuestos en la Tabla 3.8.

3.4.4. Dominio de Sensores

Se presenta a continuación la matriz morfológica del dominio de sensores en la Tabla 3.9 y la descripción de los conceptos de solución propuestos en la Tabla 3.10.

Tabla 3.7 Matriz Morfológica de los Dominios de Procesamiento y Comunicación

Funciones Parciales		Portadores de Solución		
		1	2	3
Encapsular Datos	Hardware		Raspberry Pi Zero W	
Generar señal de retroalimentación		ESP32 Development Board	Raspberry Pi Zero W	Smartphone
Encapsular Datos	Software		CoAP	
Generar señal de retroalimentación		MQTT	CoAP	
Recibir/Enviar Datos	Hardware		Procesamiento interno (Los datos no se envían)	

Tabla 3.8 Conceptos de solución propuestos del dominio de comunicación

Conceptos de Solución Propuestos	
Opción 1 →	Se usará el microcontrolador ESP32 para recopilar y encapsular los datos que serán enviados a otros dominios (Interfaz y Clasificación de estilo de conducción). Este microcontrolador usará un módulo GPRS para conectarse a una red celular y poder comunicarse con la nube. Para encapsular los datos se usa el protocolo MQTT, que es adecuado para dispositivos de rendimiento limitado.
Opción 2 →	En esta solución se usará una Raspberry Pi Zero W que es una computadora de una sola tarjeta que, en este caso se encargará también de la ejecución de el módulo de clasificación de estilo de conducción. Es decir, el procesamiento se llevará a cabo localmente y no remotamente.
Opción 3 →	Se empleará un smartphone, el cual tendrá una aplicación que se encargue de recolectar la data de los sensores y luego conectarse a internet para enviar esta data al servidor. Esto se realizará haciendo uso del protocolo CoAP.

Tabla 3.9 Matriz Morfológica del Dominio de Sensores

Funciones Parciales	Portadores de Solución		
	1	2	3
Sensar el Movimiento	 IMU de 6 ejes	 IMU de 9 ejes	 Smartphone
Sensar la Posición	 GPS		
Sensar Parámetros Internos del Auto	 OBD2 + Bluethooth	 OBD2 + WiFi	

Tabla 3.10 Conceptos de solución propuestos del dominio de sensores

<i>Conceptos de Solución Propuestos</i>	
Opción 1 →	Para esta solución se usarán un acelerómetro, giroscopio y magnetómetro de 3 ejes cada uno, reunidos en un IMU. Además, se usará también un módulo GPS y un conector OBD2 que se pueda comunicar por medio de bluetooth con el microcontrolador.
Opción 2 →	En esta solución se usará un acelerómetro, giroscopio y magnetómetro, de dos ejes cada uno, reunidos en un IMU. Además se usará también un módulo GPS y un conector OBD2 que pueda comunicarse a través de WiFi con la computadora mencionada en el módulo de comunicación.
Opción 3 →	Para esta solución se usarán los sensores presentes en un smartphone (IMU y GPS) y además este smartphone se comunicará inalámbricamente con el conector OBD2 por medio de WiFi.

3.4.5. Dominio de Interfaz

Se presenta a continuación la matriz morfológica del dominio de interfaz en la Tabla 3.11 y la descripción de los conceptos de solución propuestos en la Tabla 3.12.

Tabla 3.11 Matriz Morfológica del Dominio de Interfaz

<i>Funciones Parciales</i>	<i>Portadores de Solución</i>		
	1	2	3
Indicar Funcionamiento			
Retroalimentar estilo de conducción	Tablet/Celular	Pantalla	

3.4.6. Dominio de Reconocimiento de estilo de conducción

Se presenta a continuación la matriz morfológica del dominio de reconocimiento de estilo de conducción en la Tabla 3.13 y la descripción de los conceptos de solución propuestos en la Tabla 3.15.

Tabla 3.12 Conceptos de solución propuestos del dominio de interfaz

<i>Conceptos de Solución Propuestos</i>	
Opción 1 	En esta solución se tendrá usará como pantalla que se conectara por medio de cables al ESP32 para poder mostrar la información relevante al conductor. Esta información incluye el estilo de conducción actual del conductor y parámetros de su consumo de combustible.
Opción 2 	En esta solución se mostrará la información en una tablet o celular. Usando para esto una aplicación android que se conectará por medio de bluetooth o WiFi con la raspberry pi. Esta pantalla mostrará al conductor información relevante con la tarea de conducción. Esta información contendrá el estilo de conducción del conductor actual, y parámetros de su consumo de combustible.
Opción 3 	En esta solución se mostrará la información en una tablet o celular. Usando para esto una aplicación android. Esta pantalla mostrará al conductor información relevante con la tarea de conducción. Esta información contendrá el estilo de conducción del conductor actual, y parámetros de su consumo de combustible.

3.5. Conceptos integrados de solución

En la siguiente tabla se puede observar un resumen de las soluciones escogidas por cada concepto de solución propuesto. Se describirán cada una de las soluciones para luego analizarlas con el fin de obtener la solución óptima.

Tabla 3.13 Matriz Morfológica de Reconocimiento de estilo de conducción

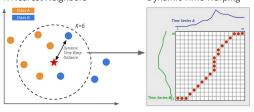
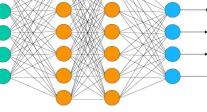
Funciones Parciales		Portadores de Solución		
		1	2	3
Almacenar Datos	Hardware	 Almacenamiento Local (Tarjeta SD)	 Almacenamiento Remoto (Servidor)	
	Software	 CSV	 Base de Datos	
Segmentar Datos		Ground Truth Rules	Changepoint Detection	
Extraer información estadística			Según algoritmo de clasificación a usar	
Clasificar Estilo de conducción		 DTW + k Nearest Neighbor	 Neural Networks	 Random Forest

Tabla 3.14 Conceptos de solución propuestos del dominio de reconocimiento de estilo de conducción

Conceptos de Solución Propuestos	
Opción 1 	Para esta solución se usará un servidor en la nube para almacenar los datos de los sensores en una base de datos y para ejecutar el algoritmo de procesamiento de datos. En este servidor se procederá a segmentar los datos en maniobras usando <i>Ground Truth Rules</i> , que son reglas que se generarán a través de conocimiento experto. Una vez la data se ha segmentado se extraerá información estadística como la media, la desviación estándar, etc. durante ese segmento. Estos valores estadísticos serán los features que alimentarán a la <i>Red Neuronal</i> que se encargará de clasificar cada segmento en un estilo de conducción agresivo o normal.
Opción 2 	En esta solución se usará una tarjeta SD para almacenar los datos de los sensores en un archivo csv y se ejecutará el procesamiento localmente. En este caso se usará el algoritmo de <i>Changepoint Detection</i> para segmentar los datos. Luego cada segmento se procesará a través de <i>k-Nearest Neighbor</i> para realizar la clasificación de estilo de conducción. Se usará <i>Dynamic Time Warping</i> (DTW) para comparar la similitud entre las señales. Para esto se necesitará un conjunto de <i>templates</i> que representen cada estilos de conducción posible y que se compararán con cada nueva señal para clasificar futuros segmentos.
Opción 3 	En esta solución se usa también un servidor en la nube para almacenar y procesar los datos de los sensores. En este caso el algoritmo a usar es el de <i>Random Forest</i> . Este algoritmo consiste en diferentes reglas que se usan para poder clasificar cada nuevo ejemplo que llega a partir de los datos de los sensores.

Tabla 3.15 Conceptos de solución propuestos

	Solución 1	Solución 2	Solución 3
Proteger y Albergar Componentes	Case en el panel frontal	Case en el techo del vehículo	Dispositivo en el panel frontal
Sujetar los Componentes	Usando un sujetador de smartphones	Atornillado a un sujetador que posee una base magnética	Usando un sujetador de smartphones
Obtener Energía	Puerto de 12 V	Batería de Ion Litio	Puerto de 12 V
Regular Voltaje	Regulador Switching	Regulador Lineal	Cargador de smartphone
Sensar el Movimiento	IMU de 9 ejes	IMU de 6 ejes	Smartphone
Sensar la Posición	GPS	GPS	Smartphone
Sensar Parámetros Internos del Auto	OBD2 + Bluetooth	OBD2 + WiFi	OBD2 + WiFi
Encapsular Datos	ESP32 + MQTT	Raspberry Pi Zero W + CoAP	Smartphone + MQTT
Generar Señal de Retroalimentación			
Enviar/Recibir Datos	Módulo GPRS	Procesamiento Internos	Smartphone
Retroalimentar Estilo de Conducción	Pantalla	Tablet/Smartphone	Tablet/Smartphone
Almacenar Datos	Almacenamiento Remoto (Servidor) + Base de Datos	Almacenamiento Local (Tarjeta SD) + CSV	Almacenamiento Remoto (Servidor) + CSV
Segmentar Datos	Ground Truth Rules	Changepoint Detection	Changepoint Detection
Clasificar Estilo de conducción	Neural Networks	DTW + k Nearest Neighbor	Random Forest

3.5.1. Concepto integrado de solución 1

El sistema se encuentra contenido en un case de plástico, que albergará todos los componentes electrónicos y que se sujetará usando un *phone holder* en el panel frontal del auto. Además, este sistema contará con una pantalla que estará posicionada de tal manera que el conductor pueda mirarla con facilidad. Este elemento será el que retroalimente el estilo de conducción al conductor.

El dispositivo se conectará a la toma de 12 V y a través de un regulador switching se alcanzarán los 3.3 V que necesita el ESP32 y los 5 V que necesitan algunos de los módulos. Estos módulos consisten en: Un acelerómetro de 3 ejes, un giroscopio de 3 ejes, un magnetómetro de 3 ejes (IMU), un módulo GPS y un módulo GPRS.

El conector OBD2 es el único módulo que no será energizado por el regulador switching, debido a que se conectará al puerto OBD2 del auto y obtendrá energía por este mismo medio. Este conector se comunicará por medio de bluetooth al microcontrolador y de esta manera obtendrá los parámetros internos del vehículo.

El microcontrolador entonces gestionará todo los datos recibidos y se encargará usar el protocolo MQTT para enviarlos. Usando el módulo GPRS que se conectará a internet, haciendo llegar los datos hasta el servidor. Luego los datos se almacenarán en una base de datos y se procesarán allí.

El primer paso para procesar los datos es segmentarlos en períodos en los que se realiza determinada maniobra con un solo estilo de conducción. Esto se puede lograr a través de reglas antes predefinidas usando conocimiento experto. Estas reglas son llamadas *Ground Truth Rules*.

El segundo paso es analizar el segmento estadísticamente. Se definirán una serie de *features* que representen adecuadamente al segmento de datos. Estos features podrían ser: media, desviación estándar, máximo , mínimo, etc. Estos elementos serán las entradas de la *red neuronal* que ya ha sido entrenada previamente con una cantidad significativa de datos.

La red neuronal procesará las entradas y arrojará en la salida la clase a la que pertenece este segmento, pudiendo ser estilo de conducción agresivo o estilo de conducción normal. Una vez que se tiene el estilo de conducción clasificado, se procede a enviarlo de regreso hacia el microcontrolador. Este se encargará de recibir la clasificación y de acondicionar esta información para mostrarla en la pantalla presente en el vehículo. Esta pantalla mostrará el estilo de conducción reconocido y un estimado de el consumo de combustible actual si esta disponible.

El cálculo de consumo de combustible se realizará también en el servidor usando parte de los datos que provienen del módulo OBD2. Sin embargo, esta información solo estará disponible si se tienen los parámetros adecuados desde el puerto OBD2 (Cada auto tiene disponibles diferentes parámetros).

3.5.2. Concepto integrado de solución 2

El sistema se encuentra contenido en un case de plástico, que albergará todos los componentes electrónicos y que se sujetará usando sujetador magnético en la parte superior del auto (techo). Además, este sistema contará con smartphone o tablet que este sujetada por un phone holder de tal manera que el conductor pueda mirarla con facilidad. Este elemento será el que retroalimente el estilo de conducción al conductor.

El dispositivo contará con una batería de ion litio con la que se encargará de obtener energía para su funcionamiento y a través de un regulador lineal se alcanzarán los 5 V que necesita la raspberry pi zero w y algunos de los módulos. Estos módulos consisten en: Un acelerómetro de 2 ejes, un giroscopio de 2 ejes, un magnetómetro de 2 ejes (IMU de 6 ejes), un módulo GPS y un módulo GPRS.

El conector OBD2 es el único módulo que no será energizado por el regulador lineal, debido a que se conectará al puerto OBD2 del auto y obtendrá energía por este mismo medio. Este conector se comunicará por medio de WiFi a la raspberry y de esta manera obtendrá los parámetros internos del vehículo.

La raspberry entonces gestionará todo los datos recibidos y se encargará de procesarlos localmente. El primer paso para procesar los datos es segmentarlos en periodos en los que se realiza determinada maniobra con un solo estilo de conducción. Esto se puede lograr a través de un algoritmo llamado *changepoint detection* que analiza el comportamiento de las señales y trata de dividirlas en segmentos en donde su comportamiento sea lineal o cercano a lineal.

El segundo paso es analizar el segmento y clasificarlo usando el algoritmo de *k-NN*. Este algoritmo consiste en comparar el segmento obtenido con segmentos ejemplo que se tienen almacenados, y otorgarle la clase de acuerdo a los ejemplos que sean mas similares a este segmento.

El algoritmo arrojará entonces la clase a la que pertenece este segmento, pudiendo ser estilo de conducción agresivo o estilo de conducción normal. Una vez que se tiene el estilo de conducción clasificado, se procede a enviarlo en la tablet o celular presente en el vehículo para retroalimentar la información al conductor. En el smartphone o tablet existirá una

aplicación que mostrará el estilo de conducción reconocido y un estimado de el consumo de combustible actual si esta disponible.

El cálculo de consumo de combustible se realizará en la raspberry pi usando parte de los datos que provienen del módulo OBD2. Sin embargo, esta información solo estará disponible si se tienen los parámetros adecuados desde el puerto OBD2 (Cada auto tiene disponibles diferentes parámetros).

3.5.3. Concepto integrado de solución 3

En esta solución el sistema usará los sensores internos del smartphone para recopilar los datos necesarios, por lo que se requerirá un sujetador de smartphone para autos en el panel frontal del auto que este posicionado de tal manera que el conductor pueda mirarla con facilidad, ya que el smartphone también mostrará los datos obtenidos luego de la clasificación.

El dispositivo se conectará a la toma de 12 V por medio del cargador de smartphone, para asegurarse de no perder energía y apagarse.

El conector OBD2 es el único módulo que no será energizado por el cargador de smartphone, debido a que se conectará al puerto OBD2 del auto y obtendrá energía por este mismo medio. Este conector se comunicará por medio de WiFi con el smartphone y de esta manera se obtendrán los parámetros internos del vehículo.

El smartphone entonces usará una aplicación android, que será la que gestionará todo los datos recibidos y se encargará usar el protocolo MQTT para enviarlos a un servidor remoto. Para esto se conectará a internet usando 3G, 4G o la línea que se encuentre disponible. Luego los datos se almacenarán en una base de datos y se procesarán en el servidor.

El primer paso para procesar los datos es segmentarlos en periodos en los que se realiza determinada maniobra con un solo estilo de conducción. Esto se puede lograr a través de un algoritmo llamado *changepoint detection* que analiza el comportamiento de las señales y trata de dividirlas en segmentos en donde su comportamiento sea lineal o cercano a lineal.

El segundo paso es usar el Random Forest para clasificar adecuadamente este segmento en un estilo de conducción agresivo o estilo de conducción normal. Una vez que se tiene el estilo de conducción clasificado, se procede a enviarlo de regreso hacia el smartphone. la aplicación, entonces, se encargará de recibir la clasificación y de acondicionar esta información para mostrarla en la tablet o celular presente en el vehículo. Esta aplicación mostrará también un estimado de el consumo de combustible actual si este esta disponible.

El cálculo de consumo de combustible se realizará también en el servidor usando parte de los datos que provienen del módulo OBD2. Sin embargo, esta información solo estará disponible si se tienen los parámetros adecuados desde el puerto OBD2 (Cada auto tiene disponibles diferentes parámetros).

3.6. Evaluación de conceptos de solución

Se evaluará en esta sección a las propuestas planteadas usando criterios técnicos y económicos. De esta manera se obtendrá la solución óptima.

3.6.1. Evaluación técnica

Tabla 3.16 Evaluación técnica

<i>Variantes de concepto</i>		<i>Solución 1</i>		<i>Solución 2</i>		<i>Solución 3</i>		<i>Solución ideal</i>	
<i>Criterios de evaluación</i>	<i>g</i>	<i>p</i>	<i>gp</i>	<i>p</i>	<i>gp</i>	<i>p</i>	<i>gp</i>	<i>p</i>	<i>gp</i>
<i>Buen uso de fuerza o energía</i>	2	3	6	1	2	3	6	4	8
<i>Seguridad</i>	2	3	6	2	4	3	6	4	8
<i>Rapidez</i>	3	3	9	1	3	2	6	4	12
<i>Estabilidad</i>	2	2	4	2	4	2	4	4	8
<i>Rigidez</i>	2	2	4	3	6	2	4	4	8
<i>Confiabilidad</i>	3	3	9	3	9	3	9	4	12
<i>Facilidad de manejo</i>	3	2	6	1	3	3	9	4	12
<i>Transportabilidad</i>	3	3	9	1	3	3	9	4	12
<i>Calidad de trabajo</i>	1	3	3	3	3	3	3	4	4
<i>Complejidad</i>	1	2	2	3	3	1	1	4	4
<i>Puntaje máximo</i>		26	58	20	40	25	57	40	88
<i>Valor técnico</i>		0.65	0.66	0.5	0.45	0.63	0.65	1	1

3.6.2. Evaluación económica

Tabla 3.17 Evaluación económica

Variantes de concepto		Solución 1		Solución 2		Solución 3		Solución ideal	
Criterios de evaluación	g	p	gp	p	gp	p	gp	p	gp
Número de piezas	3	3	9	3	9	3	9	4	12
Fácil adquisición de materiales	2	3	6	2	4	3	6	4	8
Costo de fabricación	1	3	3	3	3	3	3	4	4
Pocos desperdicios	1	3	3	3	3	3	3	4	4
Costo de tecnología	3	3	9	2	6	3	9	4	12
Facilidad de montaje	3	3	9	2	6	3	9	4	12
Facilidad de mantenimiento	3	2	6	2	6	3	9	4	12
Costos de operación	3	2	6	3	9	1	3	4	12
Puntaje máximo		22	51	20	46	22	51	32	76
Valor técnico		0.69	0.67	0.63	0.61	0.69	0.67	1	1

3.6.3. Evaluación de soluciones

Tabla 3.18 Evaluación de soluciones

	Valor técnico	Valor económico	Calificación
Solución 1	0.66	0.67	Buena solución
Solución 2	0.45	0.61	Buena solución
Solución 3	0.65	0.67	Solución deficiente

De acuerdo a lo observado en los análisis técnico y económico se consideran como buenas soluciones a las soluciones 1 y 3. Sin embargo, la solución 1 resulta la más adecuada. Por este motivo se escoge usar la solución 1 para el desarrollo de esta tesis.

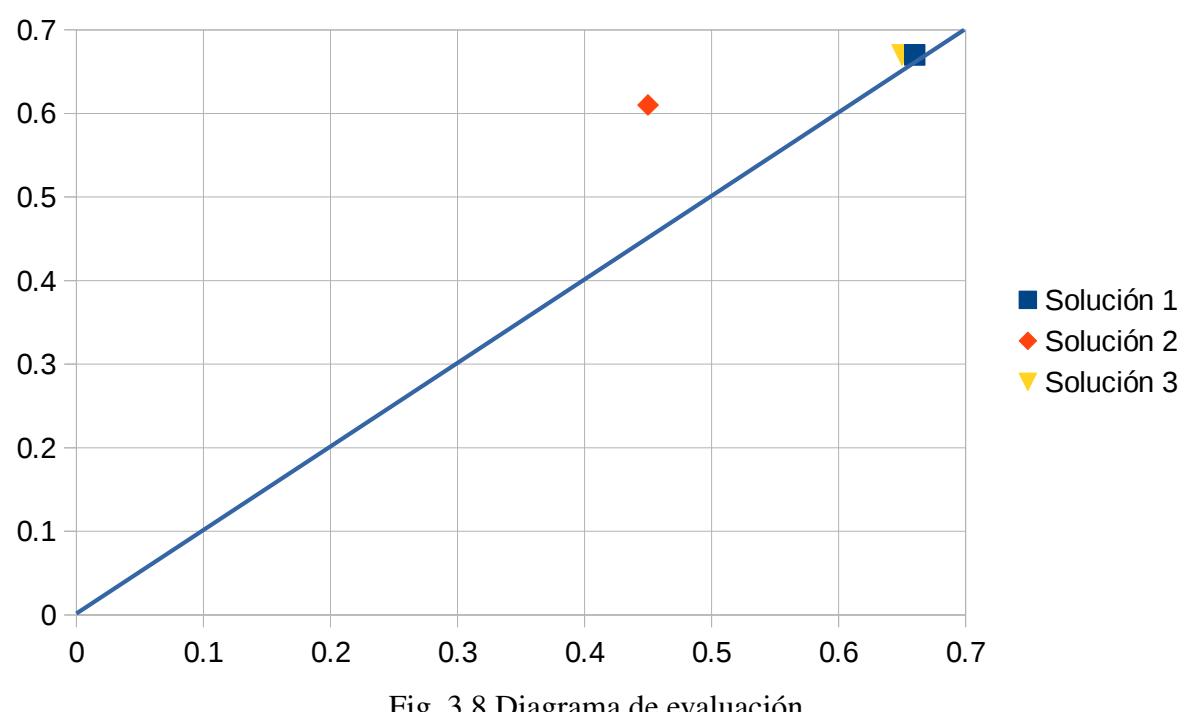


Fig. 3.8 Diagrama de evaluación

Bibliografía

- [1] “Conductores peruanos son los peores en manejo, según estudio,” *Publimetro*, Agosto 2017. [Online]. Available: <https://publimetro.pe/actualidad/noticia-conductores-peruanos-son-peores-manejo-segun-estudio-63608>
- [2] “Global plug-in sales for Q1-2018,” 2018, Ev-volumes. [Online]. Available: <http://www.ev-volumes.com/country/total-world-plug-in-vehicle-volumes/>
- [3] J. E. Meseguer, C. K. Toh, C. T. Calafate, J. C. Cano, and P. Manzoni, “Drivingstyles: a mobile platform for driving styles and fuel consumption characterization,” *Journal of Communications and Networks*, vol. 19, no. 2, pp. 162–168, April 2017.
- [4] H. Medrano Marin, “Este es el número de muertes por accidentes de tránsito en carreteras del Perú,” *El Comercio*, Febrero 2018. [Online]. Available: <https://elcomercio.pe/peru/numero-accidentes-transito-carreteras-peru-2017-noticia-499060>
- [5] J. L. Izaguirre Barrantes, “Siniestros Viales en el Perú: Reporte Estadístico PNP años 2012-2017 ,” *R2J: Sistemas Integrales Tecnológicos del Transporte*, Diciembre 2017. [Online]. Available: <http://r2jsitt.com/wp-content/uploads/2017/12/Siniestros-Viales-Perú-2012-2017-version-web-VL.pdf>
- [6] C. M. Martinez, M. Heucke, F. Wang, B. Gao, and D. Cao, “Driving style recognition for intelligent vehicle control and advanced driver assistance: A survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 666–676, March 2018.
- [7] D. Dörr, D. Grabengiesser, and F. Gauterin, “Online driving style recognition using fuzzy logic,” in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Oct 2014, pp. 1021–1026.
- [8] A. Aljaafreh, N. Alshabatat, and M. S. N. Al-Din, “Driving style recognition using fuzzy logic,” in *2012 IEEE International Conference on Vehicular Electronics and Safety (ICVES 2012)*, July 2012, pp. 460–463.
- [9] Y. L. Murphey, R. Milton, and L. Kiliaris, “Driver’s style classification using jerk analysis,” in *2009 IEEE Workshop on Computational Intelligence in Vehicles and Vehicular Systems*, March 2009, pp. 23–28.
- [10] *Highway Capacity Manual 2000*. Washington, D.C.: Transportation Research Board, 2000.

- [11] Z. Constantinescu, C. Marinoiu, and M. Vladoiu, “Driving style analysis using data mining techniques,” *International Journal of Computers Communications & Control*, vol. 5, no. 5, pp. 654–663, 2010.
- [12] D. A. Johnson and M. M. Trivedi, “Driving style recognition using a smartphone as a sensor platform,” in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, Oct 2011, pp. 1609–1615.
- [13] E. Keogh and C. A. Ratanamahatana, “Exact indexing of dynamic time warping,” *Knowledge and Information Systems*, vol. 7, no. 3, pp. 358–386, Mar 2005. [Online]. Available: <https://doi.org/10.1007/s10115-004-0154-9>
- [14] J. Echanobe, I. del Campo, and M. V. Martínez, “Design and optimization of a neural network-based driver recognition system by means of a multiobjective genetic algorithm,” in *2016 International Joint Conference on Neural Networks (IJCNN)*, July 2016, pp. 3745–3750.
- [15] M. V. Ly, S. Martin, and M. M. Trivedi, “Driver classification and driving style recognition using inertial sensors,” in *2013 IEEE Intelligent Vehicles Symposium (IV)*, June 2013, pp. 1040–1045.
- [16] Freescale, *High Accuracy Low g Inertial Sensor Rev. 5*, Agosto 2012.
- [17] InvenSense Inc, *MPU-6000 and MPU-6050 Product Specification Revision 3.4*, Septiembre 2013.
- [18] A. Benedetto, A. Calvi, and F. D’Amico, “Effects of mobile telephone tasks on driving performance: a driving simulator study,” *Advances in transportation studies*, vol. 26, pp. 29–44, 2012.
- [19] H. Ma, H. Xie, and D. Brown, “Eco-driving assistance system for a manual transmission bus based on machine learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 572–581, Feb 2018.
- [20] Y. Gaffary and A. Lécuyer, “The use of haptic and tactile information in the car to improve driving safety: A review of current technologies,” *Frontiers in ICT*, vol. 5, p. 5, 2018. [Online]. Available: <https://www.frontiersin.org/article/10.3389/fict.2018.00005>