

# Asistente Inteligente de Conducción Autónoma para Evitar Obstáculos y Colisiones

Santiago J. Cachumba <sup>1</sup>, Pablo A. Briceño <sup>2</sup>, Víctor H. Andaluz <sup>3</sup>, Germán Erazo <sup>4</sup>.

sjcachumba@espe.edu.ec, pabriceno1@espe.edu.ec, vhandaluz1@espe.edu.ec,  
wgerazo@espe.edu.ec

<sup>1</sup> Universidad de las Fuerzas Armadas Espe, 171103, Sangolquí, Ecuador.

DOI: 10.17013/risti.n.pi-pf

**Resumen:** En este artículo se presenta la implementación de un asistente inteligente de conducción autónomo, con el fin de prevenir colisiones a causa de la desatención o impericia de los conductores. Para lograr lo requerido, se desarrolla un entorno gráfico en Unity 3D, en el que puede interactuar el usuario mediante los dispositivos hápticos con el vehículo. Para el vehículo se toma como referencia el modelo cinemático de un robot tipo car-like, modelado en Matlab, este permite recrear el comportamiento de un vehículo real. La evasión de obstáculos se da cuando un objeto se encuentra en la trayectoria del vehículo y sobrepasa la distancia mínima de maniobra, en tal caso el controlador toma el mando del vehículo, realiza la evasión, posiciona el vehículo dentro de la trayectoria original y le devuelve el control al conductor. Durante este proceso de evasión se tiene una retroalimentación de fuerza en el volante, que gira de acuerdo a la posición del obstáculo y el pedal de freno, que se acciona según la velocidad del automóvil.

**Palabras-clave:** Asistente de conducción; Interacción humano-computadora; Prevención de colisión; Realidad Virtual, Unity 3D

## *Intelligent Autonomous Driving Assistant to Avoid Obstacles and Collisions*

**Abstract:** This article presents the implementation of an intelligent autonomous driving assistant, with the purpose of preventing collisions due to the lack of carelessness of drivers. To achieve what is required, a graphic environment is developed in Unity 3D, in which the user can interact through haptic devices with the vehicle. For the vehicle, the kinematic model of a car-like robot, modeled in Matlab, is taken as a reference, which allows it to recreate the behavior of a real vehicle. Obstacle avoidance occurs when an object is in the vehicle's trajectory and exceeds the minimum maneuvering distance, in

which case the controller makes the evasion, positions the vehicle within the original trajectory and returns control to the driver. During this evasion process there is a force feedback on the steering wheel, which rotates according to the position of the obstacle.

**Keywords:** Collision Prevention; Driver's Assistant; Human-Computer Interaction; Unity 3D, Virtual Reality.

## 1. Introducción

En la sociedad actual uno de los problemas mundiales más preocupantes sin una solución definitiva son los accidentes de tránsito; a nivel mundial alrededor de 1,35 millones de personas pierden su vida en accidentes de tránsito, de treinta a cincuenta millones padecen traumatismos no mortales que causan algún tipo de discapacidad, este el alto número de accidentes de tránsito tienen impacto sobre la economía de cada país, estas pérdidas económicas equivalen al 3% del producto interno bruto de la mayoría de países en desarrollo; las principales causas de estos accidentes son cansancio, distracción, impericia y el consumo de estupefacientes o bebidas alcohólicas, todas estas causas se deben a errores humanos, (OMS, 2018). Los sistemas de seguridad modernos están enfocados en la reducción del error humano antes mencionado, a través de diferentes sistemas de alerta o asistentes de conducción

En la Figura 1, se puede observar los países con el mayor número de accidentes con consecuencias mortales. Las zonas resaltadas en color rojo pertenecen a los países con mayor tasa de decesos por accidentes de tránsito y se evidencia que en las regiones de África, el sur de Europa, Asia y América del Sur existe la mayor cantidad de accidentes con víctimas mortales, (OMS, 2018)

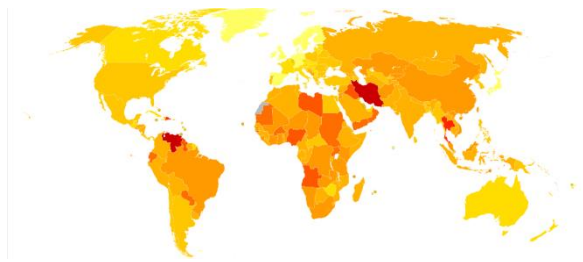


Figura 1 – Índice de accidentes mortales por países (OMS, 2018)

En la República del Ecuador se tiene un alto número de accidentes de tránsito, en comparación a países que están más desarrollados, realizando un análisis cuantitativo, se obtiene como resultado alrededor de 1058 fallecidos en 12460

accidentes durante el año 2018, estas cifras han ido aumentando durante los últimos 4 años, (Cruz, 2018); es evidente la necesidad de medidas que permitan reducir estas cifras. En ciertas ocasiones un accidente de tránsito no puede evitarse, pero se pueden reducir las pérdidas fatales producidas por el mismo, mejorando y modificando los sistemas de seguridad activos y pasivos en los vehículos y la normativa de regulación en aspectos de seguridad para los vehículos que ingresan al Ecuador, ya que ambos factores inciden directamente en la cantidad de accidentes producidos en el país.

Gracias a la implementación de sistemas inteligentes de seguridad en los vehículos, se pueden categorizar en un nivel de conducción autónoma según la SAE, como se identifica en la Figura 2.

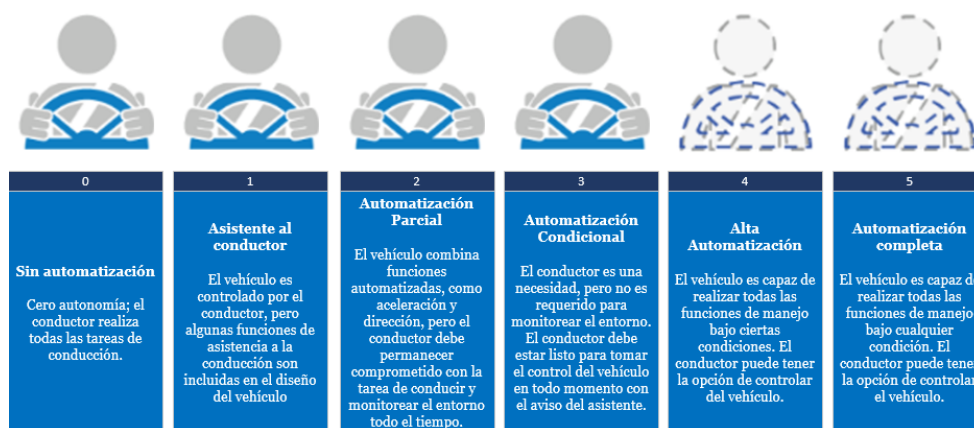


Figura 2 – Niveles de conducción autónomo.

Tomando en consideración estos niveles, varias empresas multinacionales de automotores, han optado por implementar estos sistemas de seguridad, como lo es Mercedes Benz, con los siguientes asistentes de conducción: Asistencia de distancia activa DISTRONIC, Asistencia activa de cambio de carril, Asistente de límite de velocidad activa, Asistencia de parada de emergencia activa, asistencia de dirección evasiva, Asistencia de señales de tráfico, Asistente de parqueo, Asistente de aparcamiento remoto. Estos sistemas implementados especialmente en los últimos modelos de Mercedes, lo colocan en el nivel 2 de conducción autónoma, aunque se prevé que para el 2020 o 2021 Mercedes dispondrá de vehículos con nivel 3 de conducción autónoma. Otras marcas de igual manera han implementado sistemas aún más complejos que los han llevado a alcanzar niveles de 3 hasta 4, como lo son BMW, con su modelo i3, Audi, con su modelo Elaine Concept y Tesla, con sus modelos S, 3 y X, entre los más reconocidos.

En el ámbito académico, se han desarrollado investigaciones enfocadas en seguridad vial con el fin de evitar o al menos disminuir la cantidad de accidentes de tránsito ocasionados por la impericia, desatención o el cansancio de los conductores. Entre las investigaciones más relevantes sobre asistentes de conducción, se puede determinar cuatro grupos primordiales, como lo son: i) Asistente de parqueo, que ayudan al conductor a determinar el espacio de estacionamiento y prevenir roces o contactos con muros o coches, en otros casos, el asistente de parqueo es remoto y no necesita que el conductor esté dentro del vehículo para que realice la acción (Wang, Y., & Zhu, X., 2012); ii) Asistente de frenado, que sirven para prevenir colisiones con objetos que se presenten en el camino, ya sean fijos o móviles, ya que el sistema evalúa el riesgo de colisión y advierte al conductor o toma una acción inmediata de frenado (Kusano, K. D., & Gabler, H. C., 2012); iii) Asistente de conducción para corrección de camino, los cuales se basan en el comportamiento del conductor para tomar la acción de redirigir el vehículo y tomar la ruta correcta, con el fin de evitar accidentes en la vía (Bunces, E., & Zambrano, D., 2018). iv) Asistente que permite evadir obstáculos, fijos o móviles, estos evalúan los riesgos de colisión y toman una ruta alternativa en caso de que el conductor no responda a las advertencias sonoras u ópticas que el sistema emite a través de redes inteligentes de control, que se confirman mediante simulaciones (Schaub, A., Baumgartner, D., & Burschka, D., 2016).

La propuesta de desarrollo de esta investigación consiste en realizar un entorno gráfico en realidad virtual juntamente con un vehículo, dentro del cual se implementa un asistente de conducción autónomo, enfocado a la evasión de obstáculos presentes en la trayectoria del vehículo, mediante un controlador; el algoritmo inteligente toma la decisión de maniobrar para evadir el obstáculo de acuerdo a los datos de posición del mismo; el proceso sistemático de control del automóvil es, realizar la maniobra evasiva, evitar el obstáculo, volver el vehículo hacia su trayectoria original y devolver el control del vehículo al usuario para que este siga su camino, una vez terminado el asistente de conducción el aporte de su implementación será significativo en el aumento de seguridad activa de los vehículos al ser capaz de prevenir colisiones contra objetos que se encuentren obstaculizando el paso del mismo.

El presente artículo está dividido en cinco capítulos; El capítulo I es la introducción. Capítulo II especifica la estructura del sistema. Capítulo III detalla el entorno virtual de la aplicación. Capítulo IV es el desarrollo del asistente autónomo de conducción. Capítulo V son los resultados experimentales de la aplicación en diferentes condiciones de funcionamiento.

## **2. Estructura del Sistema**

El objetivo del sistema es lograr la interacción entre el usuario con el entorno virtual, para lo cual se simula la conducción de un vehículo a través de los dispositivos

hápticos y el desarrollo de un ambiente virtual en el que se desplaza el vehículo. De esta manera se estimulan sensaciones en el usuario que le permiten desarrollarse con naturalidad y pueda ejecutar acciones a través del volante y pedales para conducir el vehículo.

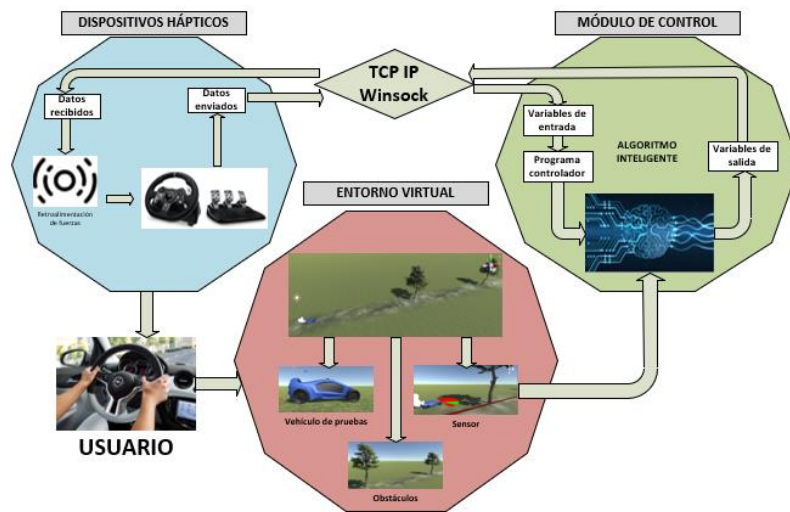


Figura 3 – Estructura del Sistema

Tal como se describe en la Figura 3, la implementación del entorno en realidad virtual brinda la experiencia de inmersión, lo que permite que el usuario se adentre en la simulación y concentre sus sentidos dentro de la misma. La comunicación entre el entorno virtual con el módulo de control se realiza a través del envío de datos de un sensor frontal de tipo Raycast que está implementado dentro del entorno virtual, estos datos son enviados a través del método de comunicación TCP IP y establecen las acciones de corrección de trayectoria necesarias para evitar colisiones con obstáculos, estas acciones provocan retroalimentación de fuerzas en el volante que son percibidas por el usuario.

Este trabajo está subdividido en cuatro áreas principales, la implementación del entorno mediante realidad virtual, el desarrollo de la estrategia para la identificación de obstáculos, retroalimentación hacia los dispositivos hápticos y pruebas experimentales de utilidad y eficiencia del sistema.

### 3. Entorno Virtual

Mediante Unity 3D se desarrollan los gráficos que dan vida al entorno virtual, entre los principales, se tiene el terreno, texturas, obstáculos, vehículo en 3D y el sensor

que es el encargado de enviar la señal al controlador implementado en Matlab, el cual receipta, procesa la información y envía valores que activan la retroalimentación de fuerza de los dispositivos hápticos para que realice la maniobra evasiva, que se ve reflejada en la pantalla tal como se muestra en la Figura 4.

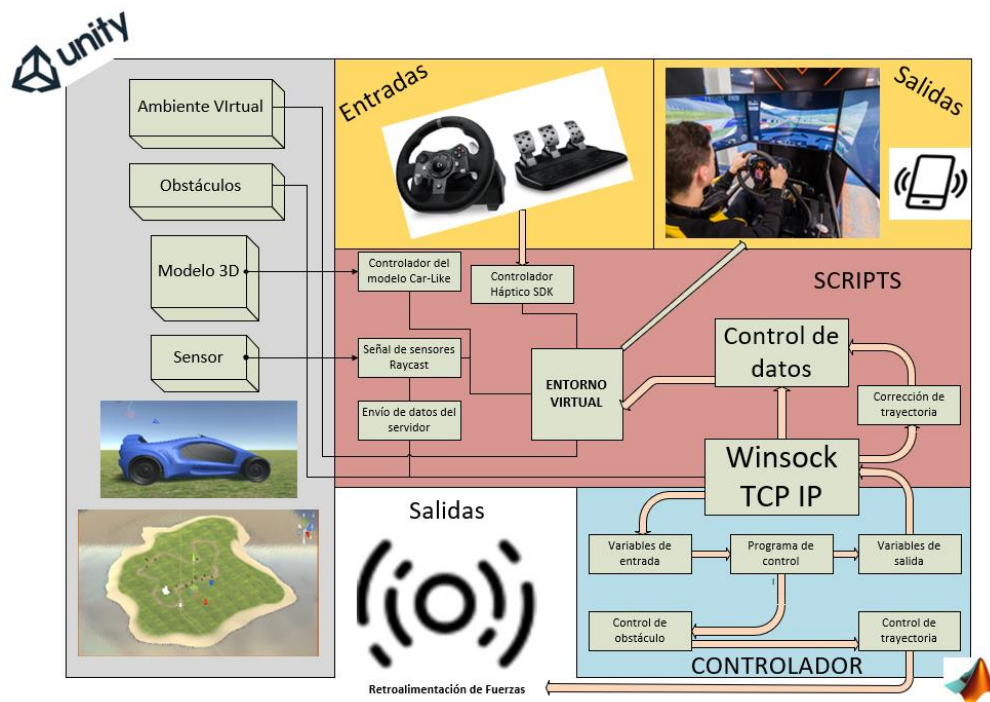


Figura 4 – Desarrollo del entorno virtual

La ruta por la que se desplaza el vehículo se establece en Matlab, mediante 3 ecuaciones que describen diferentes trayectorias, las cuales son: línea recta, senoidal y parábola. Al iniciar la aplicación en Unity 3D y en Matlab, la ruta preseleccionada se dibuja en el entorno virtual. Una vez dibujada la ruta hay la opción de colocar obstáculos en distintas posiciones de acuerdo a la ruta seleccionada. El conductor del vehículo puede maniobrar en diferentes direcciones, ya que el asistente solo corrige la trayectoria si el vehículo se encuentra dentro de la ruta preestablecida y existe un obstáculo de por medio caso contrario el único responsable de la conducción es el usuario.

Dentro de los Scripts implementados en Unity, se tiene el encargado del control y maniobrabilidad del vehículo, en donde se reúne todos los aspectos físicos, dinámicos y cinemáticos para realizar la animación de movilidad del mismo, el

encargado de la transmisión de datos en tiempo real desde el entorno virtual de Unity hacia el controlador en Matlab y viceversa; los datos enviados desde Matlab a Unity, corresponden a la trayectoria del vehículo, su velocidad y orientación, que son reflejados en el entorno virtual; en el caso de que se presente un obstáculo en medio de la trayectoria dentro del entorno virtual, se capta una señal mediante el sensor y se envía el valor de la distancia que existe entre el vehículo y el obstáculo hacia el controlador en Matlab, para que el algoritmo inteligente, pueda procesar la información y tome la decisión de enviar una ruta alternativa, disminuyendo la velocidad, para que no sea brusco el cambio de dirección en el vehículo de la simulación en Unity, a fin de evitar una colisión inminente.

La detección de obstáculos se realiza a través de un sensor láser tipo Raycast que permite asignar longitud, dirección y un punto de partida de cada rayo; el sensor se ubica en la parte frontal-central del vehículo y cada rayo se encuentra dispuesto cada 1 grado, contando con 180 rayos en total como se puede observar en la Figura 6a; esta disposición permite obtener una mayor exactitud al entrar en contacto con un obstáculo en el frente del vehículo. La detección depende de la ubicación del obstáculo y su distancia; si el objeto está demasiado cerca, la fuerza de retroalimentación será más notoria.

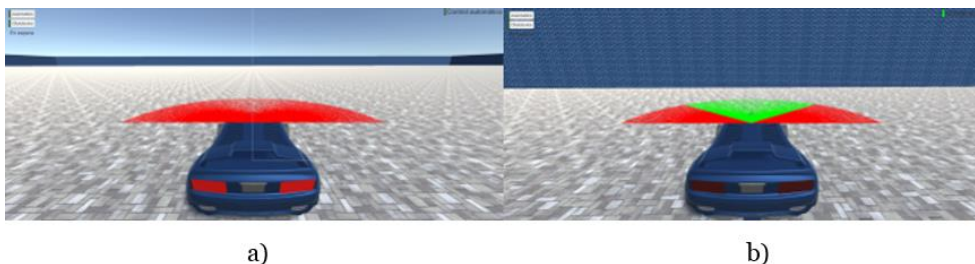


Figura 5 – a) Disposición de los Raycast b) Detección de obstáculos.

Tal como se muestra en la Figura 5b, los rayos más cercanos al obstáculo cambian de color a verde, obteniendo valores de distancia y dirección, son procesados en Unity 3D mediante el Script de control de trayectoria, luego son enviados hacia el controlador de Matlab, en donde se procesan y si el conductor no realiza ninguna acción evasiva, se toma la decisión de quitar el control al conductor y evadirlo de manera automática, cambiando la dirección del vehículo y dando la retroalimentación de fuerza en el volante. Una vez evadido este obstáculo se le devuelve el control al conductor del vehículo permitiéndole seguir su camino o detener el vehículo si el usuario lo decide.

Además, dentro del entorno virtual se ha implementado una opción de piloto automático, como se puede ver en la Figura 6, en donde al activarse el manejo se

realiza de manera autónoma por el asistente a través de la ruta seleccionada previamente y el conductor puede frenar el vehículo en cualquier momento.

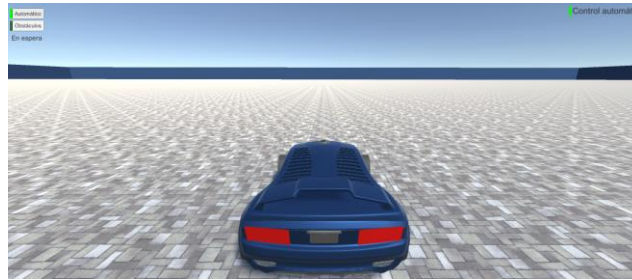


Figura 6 - Activación de opción automático.

#### 4. Asistente Autónomo

El modelado cinemático del automóvil corresponde al de un robot tipo Car-Like, permite recrear el comportamiento de un vehículo real, partiendo de la distancia entre ejes  $D$ , un centro de giro  $ICR$  que se ve reflejado en el ángulo de las ruedas  $\varphi$ , también un ángulo  $\theta$  que representa la posición angular del vehículo con respecto al eje  $X$ , y como último dato importante se tiene la distancia  $a$ , que es el punto en donde se realiza el análisis de control; la velocidad lineal  $u$  y angular  $\omega$ , son los datos que permiten al vehículo seguir la trayectoria establecida.

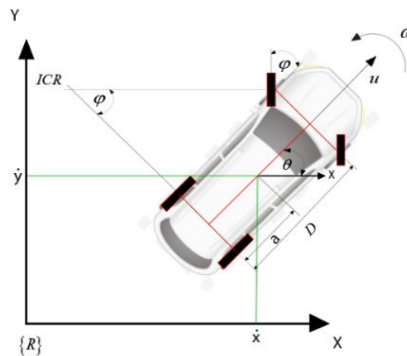


Figura 7 – Modelo cinemático car - like.

En base a la Figura 7 se puede definir el modelo cinemático del vehículo como se describe en las ecuaciones (1) y (2):



$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -a \sin(\theta) \\ \sin(\theta) & a \cos(\theta) \end{bmatrix} \begin{bmatrix} u \\ \omega \end{bmatrix} \quad (1)$$

$$\dot{\theta} = \omega = \frac{u}{D} \tan(\varphi) \quad (2)$$

El desarrollo del control de trayectoria se realiza a través de la ley de seguimiento de camino (3) establecida como:

$$\mathbf{v}_{ref} = \mathbf{J}^{-1} \left[ \mathbf{V}_d + K \tanh(\tilde{h}) \right] \quad (3)$$

El vector de velocidad deseada  $\mathbf{V}_d$  es tangente a la trayectoria del automóvil y el ángulo  $\beta$  representa la dirección del camino con respecto a la orientación del vehículo; esto sirve para determinar las componentes de la velocidad en los ejes  $X$  y  $Y$ , como se muestra en la Figura 8, logrando así que el vehículo siga la trayectoria asignada.

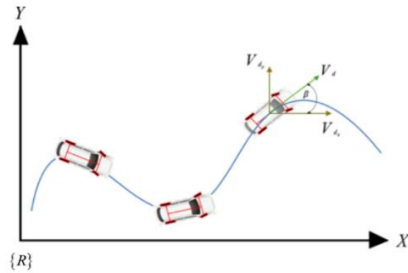


Figura 8 – Control de seguimiento de camino.

Una vez aplicada la ley de seguimiento de camino, se adiciona la detección de objetos mediante el sensor tipo Raycast implementado en Unity, el cual envía una serie de datos que son receptados por el controlador y procesador, para que se modifique la trayectoria y se evada el obstáculo en cuestión, así como se muestra en la Figura 9.

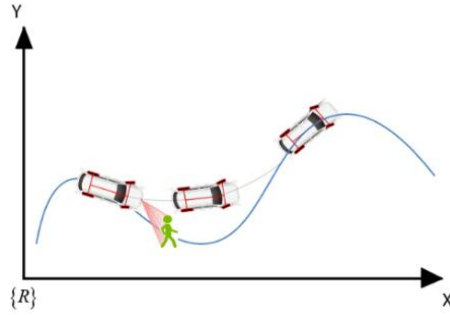


Figura 9 – Control de seguimiento de camino.

Por lo tanto, la evasión de obstáculos está dada por la ecuación (4):

$$\mathbf{v}_{ref} = \mathbf{J}^{-1} \left[ \mathbf{V}_d + K \tanh(\tilde{\mathbf{h}}) \right] + \mathbf{V}_{obs} \quad (4)$$

Los valores de la matriz  $\mathbf{V}_{obs}$  son el resultado de los datos recibidos por el sensor, que son de distancia (5) y dirección (6) y son representados de la siguiente manera:

$$\mathbf{u}_{obs} = \mathbf{Z}^{-1} \left( K_{uobs} (r-d) \left( \frac{\pi}{2} - |\alpha| \right) \right) \quad (5)$$

$$\boldsymbol{\omega}_{obs} = \mathbf{Z}^{-1} \left( K_{\omega obs} (r-d) (\alpha) \left( \frac{\pi}{2} - |\alpha| \right) \right) \quad (6)$$

Los datos recibidos por el sensor en tiempo real son la distancia del vehículo hacia el obstáculo y el ángulo de posición del rayo del sensor.

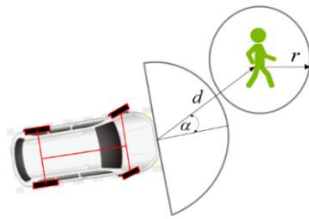


Figura 11 – Control de seguimiento de camino.

Para realizar el proceso de evasión, es se modifica la velocidad del vehículo para que el movimiento no sea agresivo, para lo cual se calcula el nuevo módulo de velocidad (7) de la siguiente manera:

$$V_d = \frac{V_{\max}}{1 + K_1(\rho) + K_2(u_{\text{obs}})} \quad (7)$$

Las ecuaciones planteadas anteriormente han sido implementadas en código de programación en Matlab para el control cinemático del vehículo y la evasión de obstáculos.

## 5. Resultados

Para la etapa de finalización del asistente de conducción, se realizan pruebas experimentales en el ambiente virtual, a fin de evaluar el correcto funcionamiento del mismo, con trayectorias distintas, para verificar que el sistema cumple con las expectativas deseadas en diferentes condiciones de servicio.

Para realizar las pruebas del asistente, se utiliza un computador de escritorio (Intel (R) Core (TM) i7-4770, 8 GB en memoria RAM, con tarjeta de video NVIDIA GeForce GTX 650, junto a los dispositivos hápticos, como lo son el volante y pedales con retroalimentación de fuerzas (Logitech G920).

Como se mencionó las rutas preestablecidas son las que se ven en la Figura 13, de acuerdo a la forma de la ruta se generan obstáculos que se encuentren en las mismas para poder detectarlos y evadirlos.

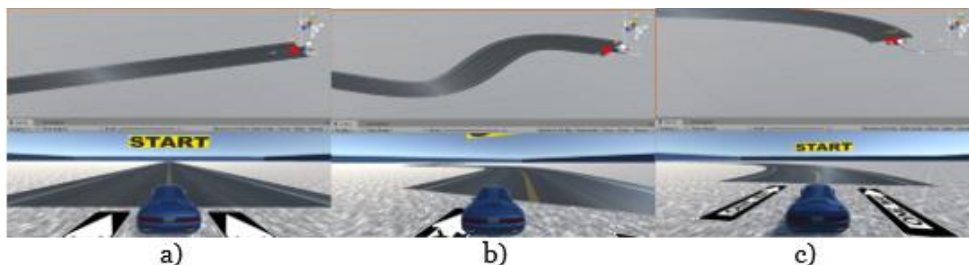


Figura 12- a) Trayectoria recta; b) Trayectoria senoidal; c) Trayectoria parabólica

Para realizar la prueba se seleccionó la trayectoria senoidal y se observa el comportamiento del vehículo y la retroalimentación de fuerzas cuando se acerca a un objeto, obteniendo los siguientes resultados.

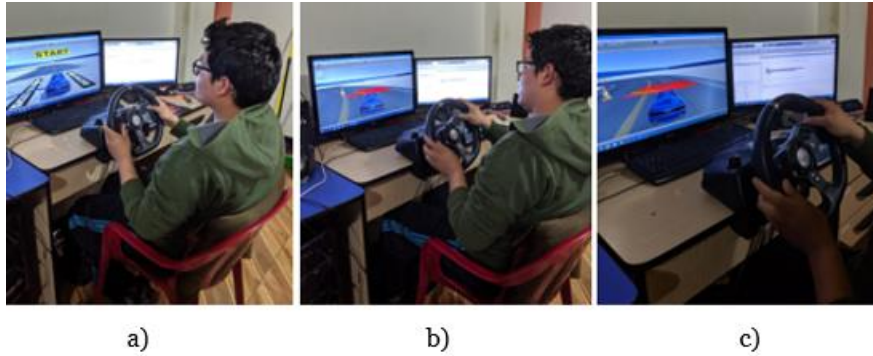


Figura 13- a) Inicio de pruebas; b) Detección y evasión de obstáculo; c) Detección del obstáculo sin evasión

En la figura 14-a) se evidencia el inicio de la prueba el volante y los pedales responden de manera correcta, en la figura 14-b) se encuentra el primer obstáculo que esta dentro de la zona de censado del Raycast, el asistente quita el control del vehículo al usuario, evade el obstáculo y luego de evadirlo devuelve el control al conductor, en la figura 14-c) el controlador detecta un obstáculo pero puesto que este se encuentra fuera de la zona mínima de evasión el asistente no quita el control al usuario y este puede seguir manejando el vehículo sin ningún problema.

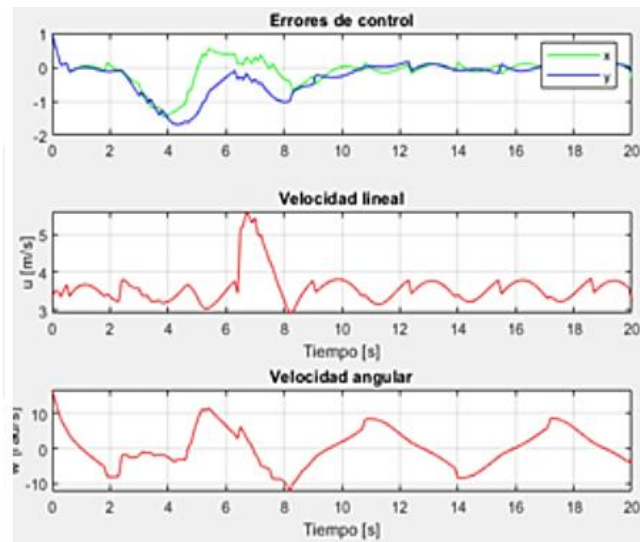


Figura 14- Gráficas obtenidas en la prueba de ruta senoidal

Los resultados después de la simulación muestran el comportamiento del controlador al tener diferentes obstáculos en el camino. En la Figura 15 se muestran tres gráficas importantes como son: i) La gráfica del error que aunque comienza en un valor positivo y con respecto al tiempo varia hasta valores negativos más altos debido a los obstáculos, siempre los valores al final del error tienden a cero, ya que esta es la función del controlador y mediante la gráfica se comprueba que el controlador está correctamente implementado; ii) la gráfica de la velocidad lineal muestra el comportamiento del vehículo en su desplazamiento lineal, pero debido a que la trayectoria es un senoide, la velocidad varía en una escala entre 0 a 4 aproximadamente, siendo la más baja cuando el vehículo se encuentra en la cresta de la función y la velocidad más alta cuando el vehículo retoma el camino; en la gráfica se muestra una cresta que llega a una velocidad de 5.5 aproximadamente, cuando el vehículo encuentra un obstáculo en el camino y realiza la maniobra evasiva; iii) la gráfica de la velocidad angular representa la velocidad de giro del vehículo, que a comparación de la gráfica anterior, varía en una escala más amplia entre 0 a 10 aproximadamente debido a que la trayectoria tiene muchas curvas; entre la escala de tiempo de 4 a 8 aproximadamente donde se encuentra el obstáculo más notorio, se puede observar como la curva se interrumpe y realiza un cambio de velocidad más perpendicular que los otros, demostrando como el vehículo evade el obstáculo exitosamente y después retoma la trayectoria predeterminada.

## **6. Conclusiones**

En el presente documento se presentó el diseño de un asistente autónomo de conducción para evitar colisiones con objetos, el diseño del asistente se basa en un modelo cinemático car-like con un controlador de seguimiento de camino, el asistente de conducción se activa en caso de existir un obstáculo y el conductor no lo está evadiendo.

Cuando el asistente entra en funcionamiento realiza 3 procesos, toma el control del auto, evade el obstáculo y devuelve el control al usuario, esto permite asegurar mayor seguridad tanto para conductores de vehículos que sean principiantes o avanzados como también para transeúntes o demás vehículos que se encuentren en la vía, permitiendo reducir el número de accidentes por descuidos o impericia de la gente.

El uso de realidad virtual permite realizar las pruebas de funcionamiento en un entorno seguro, al ser un simulador se puede declarar que el resultado de las pruebas virtuales sería aproximado a res

Para posibles trabajos futuros se puede incluir un algoritmo de control que incluya inteligencia artificial para que el sistema continúe aprendiendo en base de sus errores y permita obtener una mejor optimización, además de realizar la construcción de un prototipo de vehículo autónomo en el que se pueda incluir este asistente para verificar su importancia.

## Referencias

- Bunces, E., & Zambrano, D. (2018, November). Autonomous Assistance Control Based on Inattention of the Driver When Driving a Truck Tract. In International Conference on Social Robotics (pp. 11-24). Springer, Cham.
- Cruz, E. (17 de Agosto de 2018). Tasa de fallecidos por accidentes de tránsito en Ecuador ha aumentado en el 2018. *El Universo*, pág. 10.
- Du, X., & Tan, K. K. (2014). Autonomous reverse parking system based on robust path generation and improved sliding mode control. *IEEE Transactions on Intelligent Transportation Systems*, 16(3), 1225-1237.
- Eckert, A., Hartmann, B., Sevenich, M., & Rieth, P. (2011, June). Emergency steer & brake assist: a systematic approach for system integration of two complementary driver assistance systems. In 22nd International Technical Conference on the Enhanced Safety of Vehicles (ESV) (pp. 13-16).
- Gao, Y., Lin, T., Borrelli, F., Tseng, E., & Hrovat, D. (2010, January). Predictive control of autonomous ground vehicles with obstacle avoidance on slippery roads. In ASME 2010 dynamic systems and control conference (pp. 265-272). American Society of Mechanical Engineers.
- Gupta, A., Divekar, R., & Agrawal, M. (2010, December). Autonomous parallel parking system for Ackerman steering four wheelers. In 2010 IEEE International Conference on Computational Intelligence and Computing Research (pp. 1-6). IEEE.
- Kusano, K. D., & Gabler, H. C. (2012). Safety benefits of forward collision warning, brake assist, and autonomous braking systems in rear-end collisions. *IEEE Transactions on Intelligent Transportation Systems*, 13(4), 1546-1555.
- Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S., ... & Sokolsky, M. (2011, June). Towards fully autonomous driving: Systems and algorithms. In 2011 IEEE Intelligent Vehicles Symposium (IV) (pp. 163-168). IEEE.
- Liniger, A., Domahidi, A., & Morari, M. (2015). Optimization-based autonomous racing of 1: 43 scale RC cars. *Optimal Control Applications and Methods*, 36(5), 628-647.
- Mayorga, O. A., & Andaluz, V. H. (2018, November). Autonomous Control Through the Level of Fatigue Applied to the Control of Autonomous Vehicles. In International Conference on Social Robotics (pp. 115-126). Springer, Cham.
- National Highway Traffic Safety Administration, "Automated Vehicles for Safety", 2018. [En línea]. Disponible en: <https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety>.

- Organización Mundial de la Salud, "Accidentes de tránsito", 2018. [En línea]. Disponible en: <https://www.who.int/es/news-room/fact-sheets/detail/road-traffic-injuries>.
- Schaub, A., Baumgartner, D., & Burschka, D. (2016). Reactive obstacle avoidance for highly maneuverable vehicles based on a two-stage optical flow clustering. *IEEE Transactions on Intelligent Transportation Systems*, 18(8), 2137-2152.
- Segura, R., Urrutia, P., Salazar, F., & Buele, J. (2019). Virtual Environment for Remote Control of UGVs Using a Haptic Device. *Developments and Advances in Defense and Security, Smart Innovation, System and Technologies* 152.
- Trösterer, S., Wurhofer, D., Rödel, C., & Tscheligi, M. (2014, September). Using a parking assist system over time: Insights on acceptance and experiences. In *Proceedings of the 6th International Conference on Automotive User Interfaces and Interactive Vehicular Applications* (pp. 1-8). ACM.
- Wang, Y., & Zhu, X. (2012, June). Design and implementation of an integrated multi-functional autonomous parking system with Fuzzy logic controller. In *2012 American Control Conference (ACC)* (pp. 2928-2933). IEEE.