






Análisis de un control crucero adaptable en condiciones de poca visibilidad Analysis of an adaptive cruise control under low visibility conditions

G. A. Bañuelos-Peña ^a, A. C. Coronado-Andrade ^{a,*}, J. E. Velázquez-Velázquez ^a, J. D. Rivera-Fernández ^b,
D. A. Fabila-Bustos ^b

^a Instituto Politécnico Nacional, Unidad Profesional Interdisciplinaria de Ingeniería Campus Hidalgo, 42162, San Agustín Tlaxiaca, Hidalgo, México.

^b Laboratorio de Optomecatrónica, UPIIH, Instituto Politécnico Nacional, Distrito de Educación, Salud, Ciencia, Tecnología e Innovación, San Agustín Tlaxiaca, 42162, Hidalgo, México.

Resumen

Este trabajo se presenta el análisis de un sistema de control crucero adaptable para vehículos robots. El sistema de control crucero adaptable propuesto ayuda a evitar choques por alcance en escenarios de poca visibilidad. La colisión se previene con el ajuste de velocidad, que está en función de la distancia del vehículo que se encuentre enfrente del vehículo robot. Se implementa con el uso de la visión artificial, el control adaptable, los sistemas embebidos, y el diseño asistido por computadora. Se analiza el funcionamiento del sistema propuesto mediante pruebas en terrenos a distintas inclinaciones, así como en escenarios controlados con visión favorable, y con visión deficiente. La visión deficiente es escenario clave para este trabajo, ya que, en caso de poca visibilidad en el ambiente, el sistema de visión artificial dejara de detectar vehículos. Entonces los sensores LiDAR y radar pasarán a medir distancia y velocidad del vehículo circulando al frente, y con estos datos, reducir o mantener la velocidad del vehículo.

Palabras Clave: modelo de referencia, control crucero adaptable, visión artificial, vehículo robot.

Abstract

This work presents the analysis of an adaptive cruise control system for robotic vehicles. The proposed adaptive cruise control system assists in preventing rear-end collisions in low-visibility scenarios. Collision avoidance is achieved through speed adjustment, which is dependent on the distance from the vehicle in front of the robotic vehicle. It is implemented using computer vision, adaptive control, embedded systems, and computer-aided design. The functionality of the proposed system is analyzed through tests on terrains with varying inclinations, as well as in controlled scenarios with good visibility and poor visibility. Lack of visibility is a critical scenario in this work because, in case of limited visibility in the environment, the computer vision system will stop detecting vehicles. Then, LiDAR and radar sensors will measure the distance and speed of the vehicle ahead, and based on this data, reduce or maintain the vehicle's speed.

Keywords: reference model, adaptive cruise control, computer vision, robotic vehicle.

1. Introducción

Los vehículos con sistema de control crucero adaptable (ACC por sus siglas en inglés) trabajan en dos modos de operación: el primer modo actúa como un control crucero convencional (CCC por sus siglas en inglés), donde el conductor establece una velocidad deseada y el vehículo la mantiene constante y cuando no haya ningún automóvil en el mismo carril. El segundo modo de operación se activa cuando el vehículo se acerca a otro con menor velocidad. En este caso,

el sistema mantiene una distancia preconfigurada para evitar posibles colisiones, controlando la velocidad según la información proporcionada por los sensores. La distancia de vehículo a vehículo es configurable, pero se limita a la velocidad de frenado del vehículo en específico. El cambio entre los dos modos de operación se realiza de manera automática (Shakouri, Czebot, & Ordys, 2015).

La primera generación de sistemas de control crucero adaptable se implementó en Japón en 1995, con la finalidad de aplicarlo como un sistema avanzado de ayuda al conductor

*Autor para la correspondencia: acoronado1800@alumno.ipn.mx

Correo electrónico: gbanuelos1800@alumno.ipn.mx (Gabriel Alejandro Bañuelos-Peña), acoronado1800@alumno.ipn.mx (Allan Christopher Coronado-Andrade), jvelazquezv@ipn.mx (Juan Eduardo Velázquez-Velázquez), jdriveraf@ipn.mx (Josué Daniel Rivera-Fernández), dfabilab@ipn.mx (Diego Adrián Fabila-Bustos).

(ADAS por sus siglas en inglés) a evitar accidentes debido a la creciente cantidad de automóviles en los caminos. El sistema ACC en un vehículo, no solo se trata del confort y seguridad, también se ha demostrado que el tráfico se mueve con más fluidez y se reduce el consumo de combustible (Shakouri et al., 2015).

Los sistemas ACC suelen utilizar un sensor radar que proyecta un haz directamente hacia la parte delantera del vehículo equipado con el sistema, a fin de detectar los objetos que estén por delante de él, así como un conjunto de sensores LiDAR y una o más cámaras de apoyo que ayudan a tener una visión de campo más completa, además de un acelerómetro que le permite conocer la pendiente del camino (Rajamani & Zhu, Semi-autonomous adaptive cruise control systems, 2002).

En este trabajo se analiza el funcionamiento de una estrategia ACC en escenarios de poca visibilidad. El sistema propuesto utiliza como principal método de detección y seguimiento de vehículos la visión artificial implementada en una cámara. En situaciones de poca visibilidad, un radar y un LiDAR asumirán el papel de sensores principales en caso de que la detección de vehículos a través de la visión artificial se pierda.

La organización del artículo es como sigue: en la sección 2 se presenta el diseño del ACC de un vehículo robot; en la sección 3 se abordan los aspectos de la visión artificial. El diseño del prototipo requerido para implementar el sistema de ACC se presenta en la sección 4.

2. Control crucero

La composición de los sistemas ACC cuentan con muchas vertientes, por esto, es primordial realizar un control crucero convencional en el modelo dinámico del vehículo.

2.1. Modelo dinámico longitudinal del vehículo

Es necesario realizar un análisis del modelo longitudinal vehicular, ya que se requiere una ecuación como modelo para usar como planta en el control. Este modelo debe mostrar la fuerza suficiente que se debe aplicar sobre las ruedas para su movimiento constante. Tomando en cuenta las dimensiones del vehículo, así como la inclinación de la pendiente y factores externos.

Considerando un vehículo en movimiento sobre un camino en un plano inclinado con fuerzas externas actúan sobre él, se produce un equilibrio de fuerzas a lo largo del eje longitudinal del vehículo.

De la Figura 1, se puede determinar la ecuación fundamental del movimiento longitudinal del vehículo como:

$$m\ddot{x} = F_{xf} + F_{xr} - F_{aero} - R_{xf} - R_{xr} - mg \sin \theta, \quad (1)$$

donde

- F_{xf} es la fuerza longitudinal de la rueda frontal,
- F_{xr} es la fuerza longitudinal de la rueda trasera,
- F_{aero} es la fuerza de resistencia aerodinámica,
- R_{xf} es la fuerza de resistencia a la rodadura frontal,

- R_{xr} es la fuerza de resistencia a la rodadura trasera,
- m es la masa del vehículo,
- g es la aceleración de la gravedad,
- θ es el ángulo de inclinación del camino.

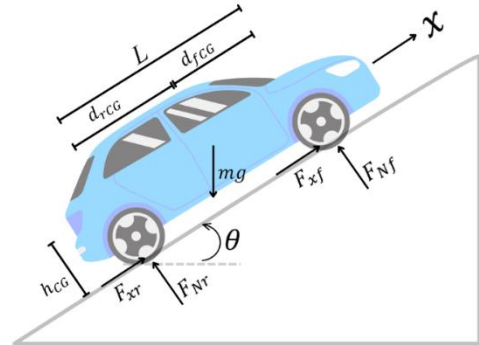


Figura 1: Vehículo longitudinal.

Las fuerzas longitudinales de las ruedas F_x , son la primera ayuda para que el vehículo se mueva y son dependientes de la relación entre la velocidad angular de la rueda $r_{eff}\omega_w$ y la velocidad longitudinal del vehículo \dot{x} . La velocidad angular de la rueda es altamente influenciada por los componentes pertenecientes al tren motriz, los cuales transmiten la potencia proveniente del motor a las ruedas y las ruedas les regresan carga. La fuerza de arrastre aerodinámica longitudinal en un vehículo F_{aero} puede ser definida como:

$$F_{aero} = \frac{1}{2} \rho C_d A_F (\dot{x} + V_{wind})^2, \quad (2)$$

donde

- ρ es la densidad de la masa del aire,
- C_d es el coeficiente de arrastre aerodinámico,
- A_F es el área frontal del vehículo,
- $V_x = \dot{x}$ es la velocidad longitudinal del vehículo,
- V_{wind} es la velocidad del viento (positiva si es en contra y negativa si es a favor del movimiento).

F_{xf} y F_{xr} , son fuerzas longitudinales de fricción que actúan sobre las ruedas. Estudios experimentales determinan que cada fuerza depende de:

- El radio de deslizamiento.
- La fuerza normal sobre la rueda.
- El coeficiente de fricción entre la rueda y el terreno.

Tomando en cuenta dichos factores se obtiene lo siguiente:

$$F_x = F_{xf} + F_{xr}, R_x = R_{xf} + R_{xr} = fmg \cos \theta, \quad (3)$$

donde f es el coeficiente de resistencia a la rodadura que toma valores en un rango de $[0.01, 0.04]$. Sustituyendo y simplificando en (1), se obtiene el modelo simplificado:

$$\ddot{x} = \frac{F_x - F_{aero} - R_x - mg \sin \theta}{m} \tag{4}$$

Dado que las fuerzas longitudinales de la rueda dependen de las mismas variables, las podemos representar como una sola fuerza llamada F_x , tal y como se muestra en (3), resultando (4) como el modelo matemático simplificado del vehículo.

A fin de lograr el controlador adaptable, es necesario el diseño de un sistema control que pueda llevar la respuesta de la planta al comportamiento deseado, es decir que el vehículo pueda llegar a la velocidad solicitada sin sobrepasar un margen de error (Rajamani, *Vehicle Dynamics and Control*, 2011).

El el modelo matemático general para un vehículo arbitrario de (4), se debe representar en espacio de estados. En el caso del vehículo robot, la fuerza aerodinámica no es significativa. Considerando como variables de estado a $X_1 = x$ y $X_2 = \dot{x}$. En (5), se muestra el resultado del espacio de estados.

$$\begin{aligned} \dot{X}_1 &= X_2, \\ \dot{X}_2 &= \frac{1}{m}(F_x - mg(\sin\theta + f\cos\theta)). \end{aligned} \tag{5}$$

2.2. Simulación de la respuesta de la planta sin controlador

Se simula la planta utilizando el modelo simplificado (5), y se le aplica un estímulo de tipo escalón para analizar su comportamiento. La Figura 2, muestra la configuración del modelo dinámico en la simulación, que recibe tanto la entrada del estímulo como una perturbación que representa una pendiente del 20%.



Figura 2: Simulación de la planta a partir del modelo dinámico.

El estímulo tipo escalón provoca el aumento de velocidad que sobrepasa la velocidad deseada porque el sistema está en lazo abierto. En el momento en que la perturbación se activa, la velocidad desciende de forma abrupta como se muestra en la Figura 3.

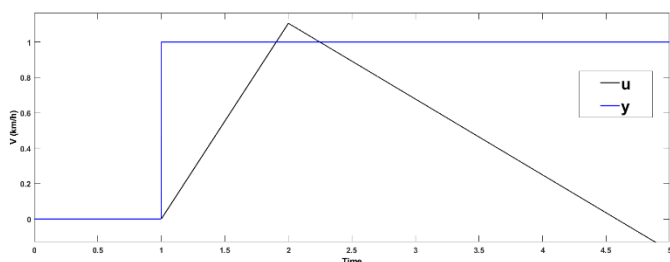


Figura 3: Respuesta de la planta a una entrada escalón (sin controlador).

2.3. Simulación de la planta controlada

La planta por sí sola no es capaz de mantener la respuesta esperada. Si se tratara de un vehículo, este comenzará a ir cada vez más rápido en pendientes negativas, y a menos que se encuentre con una pendiente positiva que haga que su propio peso supere la fuerza ejercida (lo frene) este no alcanzará ni mantendrá la velocidad deseada.

Para alcanzar la velocidad deseada se requiere de una fuerza variable que se ajuste a las necesidades del vehículo en el momento de su aplicación. Esto se logra mediante el uso de un controlador. En el presente trabajo se utilizó un controlador proporcional-integral (PI), con ganancias de $K_p = 25$ y $K_i = 4$, obtenidas por sintonización heurística buscando una respuesta rápida y con poco sobrepaso. La Figura 4, muestra la configuración del sistema en lazo cerrado (diagrama de simulación), así como la respuesta ante una entrada escalón.

Al igual que el sistema sin controlar, la planta, que contiene el modelo dinámico, sufre una perturbación simulando un ángulo, la cual causa una alteración en la respuesta. Esta perturbación se compensa por el controlador PI. A medida que el tiempo tiende a infinito, el error se aproxima a cero. La métrica numérica para calcular el error en este caso fue el error medio (ME), ya que esta considera valores positivos y negativos.

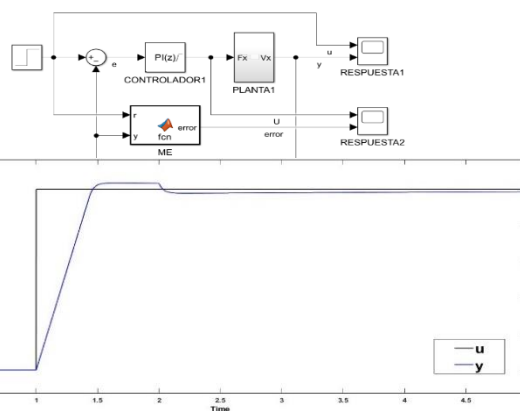


Figura 4: Respuesta de la planta a una entrada escalón (con controlador).

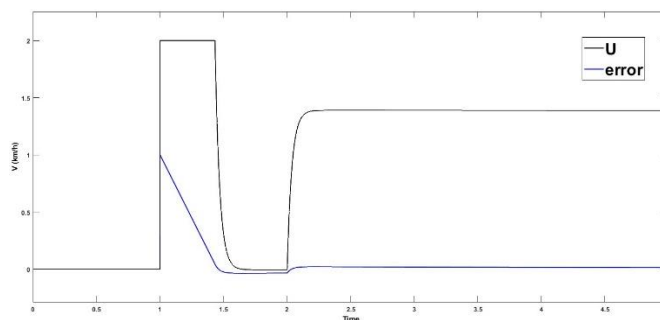


Figura 5: Dinámica de la acción de control y el error.

En la Figura 5 se observa que, aunque el controlador diseñado es válido para mantener la velocidad solicitada, no es capaz de adaptarse, esta capacidad es proporcionada por otro tipo de controlador, el cual se aborda a continuación.

2.4. Control adaptable por modelo de referencia (MRAC)

Un control adaptable es aquel con la capacidad de modificar su comportamiento en función de las circunstancias generadas por el cambio en la dinámica de un sistema y las perturbaciones presentes. Puede ser visto como un tipo de control con dos tipos de cambios ocurriendo a distintas velocidades, donde el cambio rápido corresponde al comportamiento del sistema ordinario y el cambio lento a la modificación de parámetros que gobiernan el comportamiento del sistema ordinario.

El control adaptable por modelo de referencia se puede considerar como una aproximación al control adaptable, justificado por la hipótesis que para cualquier conjunto de valores posibles en los parámetros de una planta existe un controlador lineal de complejidad fija (Rubio & López Sánchez, 1996).

Regla del MIT. Sea $G_m(S,p)$ un modelo de referencia y $G_a(S,\hat{p})$ un sistema que se debe ajustar para seguir al modelo de referencia disminuyendo el error entre sus respuestas. Se define J como la función de costo del error cuadrático medio:

$$J = \frac{1}{2} \int e^2 dt; e = y_m - y_a, \tag{6}$$

donde

- S es la variable de Laplace,
- p es el parámetro del modelo de referencia,
- \hat{p} es el parámetro por ajustar,
- e es el error de ajuste,
- y_m es la salida del modelo de referencia,
- y_a es la salida del sistema ajustable.

Mediante el método del gradiente para la optimización se obtiene la regla de adaptación:

$$\Delta \hat{p}(e, t) = -K \nabla(J) = -K \frac{dJ}{d\hat{p}}, \tag{7}$$

donde

- K es la ganancia de adaptación,
- $\Delta \hat{p}$ es la variación con respecto al último valor de \hat{p} .

Dado que es una regla conocida el desarrollo de esta resulta en la siguiente ley de adaptación (Rubio & López Sánchez, 1996):

$$\dot{\hat{p}} = K_1 e y_m. \tag{8}$$

Para lograr que el vehículo responda de la manera deseada, es necesario un sistema que se encargue de manejar el comportamiento de los motores que lo propulsan, esta tarea es llevada a cabo por el sistema de control.

Modelo de referencia. El modelo de referencia es un sistema lineal de tercer orden que se pide sea Hurwitz para asegurar la estabilidad. La matriz de estado A , de entrada B , y de salida C , están dadas por:

$$A = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1.3 & -18 & -6 \end{pmatrix}, \tag{9}$$

el resto de las matrices para el modelo propuesto se presentan en (10):

$$B = \begin{pmatrix} 0 \\ 0 \\ 0.98 \end{pmatrix}, \tag{10}$$

$$C = (1.3 \quad 18 \quad 0).$$

Siguiendo el procedimiento de la regla del MIT, se obtiene que la ley de adaptación:

$$\dot{\hat{p}} = -1 \times 10^8 y_m e, \tag{11}$$

donde:

- y_m es la respuesta del modelo,

- e es la diferencia entre la respuesta de la planta y la respuesta del modelo.

Note que la ganancia K_1 se eligió lo suficientemente grande para que el error de seguimiento en estado estable sea menor al 2%. Con esta ganancia y el modelo de referencia (9)-(10) el sistema en lazo cerrado es estable asintóticamente, lo cual se puede determinar al considerar la función candidata de Lyapunov: $V(e) = \frac{1}{2} e^2$.

Simulación del control adaptable. Se realiza la simulación de la respuesta del modelo de referencia a una entrada escalón, haciendo cambiar la amplitud de la entrada para comprobar que el modelo se mantiene estable ante los cambios de la entrada, en la Figura 6 se muestra la respuesta del modelo.

El modelo presenta una dinámica más lenta a la de la planta, requisito necesario para poder ser implementado en el control MRAC, asimismo, se muestra una respuesta dentro del 2% de error con respecto a la entrada.

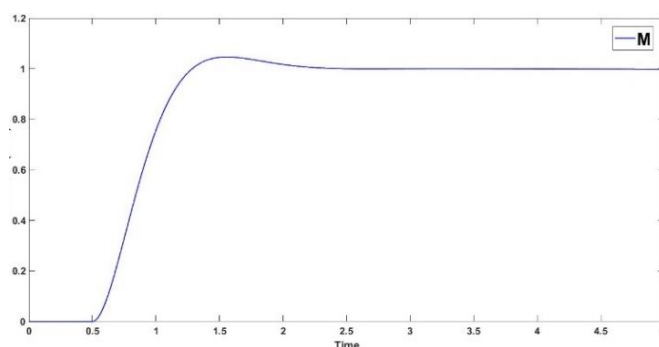


Figura 6: Respuesta del modelo de referencia a una entrada escalón con cambio de amplitud.

La configuración en paralelo del control por modelo de referencia indica la manera adecuada de conectar el modelo y la planta con control de velocidad simple por medio de un mecanismo de ajuste. Esto permitirá al sistema completo adaptarse a los cambios en las perturbaciones. Véase en Figura 7.

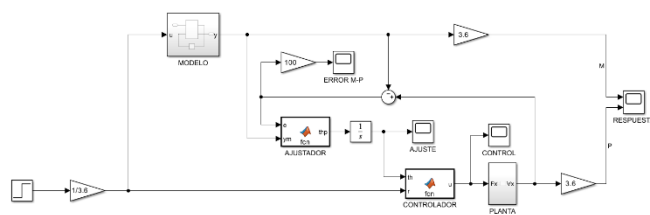


Figura 7: Diagrama de simulación del control MRAC.

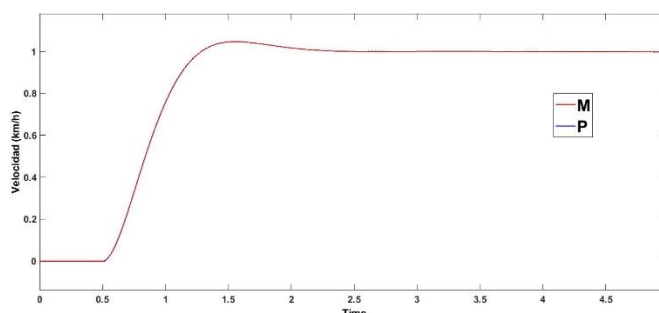


Figura 8: Respuestas de la planta controlada y el modelo de referencia.

Se mantienen los cambios de amplitud en el escalón y la entrada de perturbaciones a la planta, obteniendo la respuesta del sistema donde la planta siempre tiende al comportamiento del modelo como se muestra en la Figura 8.

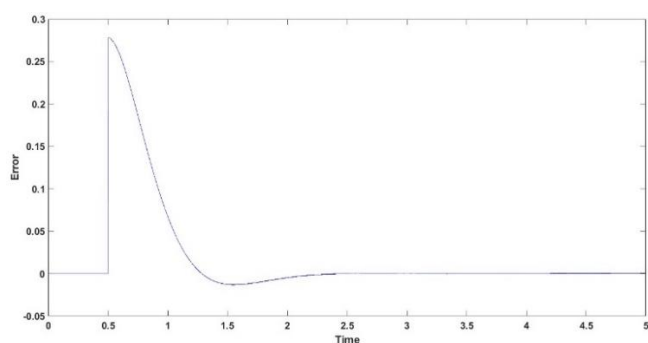


Figura 9: Dinámica del error de la planta controlada.

El error de la planta respecto a la entrada se calculó por medio de la métrica numérica del error medio, y se muestra en la Figura 9. Este comportamiento se encuentra dentro de un margen aceptable, por lo que el control adaptable puede llevarse a la implementación.

3. Visión artificial

Su objetivo es crear un modelo del mundo real a partir de imágenes. Un sistema de visión artificial recopila información útil de un escenario partiendo de la proyección en dos dimensiones de este, sin embargo, la información no está disponible directamente al ser recopilada, se requiere de una o varias estrategias de mapeo que permitan su lectura (Jain, Kasturi, & Schunck, 1995).

Mediante el análisis de una imagen obtenida por una cámara como una matriz de puntos, se obtiene una función dependiente de dos variables $f(x,y)$ que representan las coordenadas de cada punto en la imagen y el valor obtenido es el brillo de ese punto en la imagen. Para el caso de imagen a color se utilizan tres arreglos ordenados en el modelo de color RGB.

La estimación de diferencias entre píxeles adyacentes puede identificar patrones o características que se clasifican en comportamientos específicos. Para la extracción de la información de la imagen se deben considerar los defectos posibles de encontrar en esta.

Previo a la clasificación de la información se debe realizar la transformación de la imagen a un dominio óptimo en que se pueda desarrollar el trabajo, eliminando información redundante y manteniendo solo la información de utilidad, siempre buscando la reducción de la información a procesar (Sanabria S & Archila D, 2011).

3.1. Procesamiento de visión

El método por el que se obtiene la información de una imagen es conocido como procesamiento de visión y que puede llevarse a cabo con sistemas embebidos especializados. Para lograrlo es necesario realizar los siguientes pasos: Adquisición de la imagen, optimización de la imagen,

localización de características específicas, mediciones, comparación de características específicas, toma de decisión, comunicación de resultados (Cognex Corporation, 2016).

3.2. Detección y seguimiento de imágenes

La detección de objetos implica identificar un objeto previamente entrenado en miles de imágenes, sin importar su distancia, velocidad o tamaño (Chandan G. et al., 2018). El seguimiento de objetos utiliza esta detección para determinar la ubicación, dirección y velocidad en el cuadro anterior. Esto permite predecir con precisión el movimiento en el siguiente cuadro (Chandan, Jain, & Jain, 2018).

3.3. OpenCV (Open-Source Computer Vision)

Es una biblioteca gratuita multiplataforma escrita en C++ basada en C de funciones de programación dirigida mayormente para detección de imágenes en tiempo real. Es compatible con los marcos de aprendizaje profundo, tales como Tensorflow, Torch/PyTorch y caffe (Saxena, Pathak, Singh, & Shukla, 2019).

3.4. YOLO (You Only Look Once)

Es un algoritmo de detección rápida de objetos usando el aprendizaje profundo, es decir, mediante redes neuronales convolucionales (CNN).

Las redes neuronales tradicionales tienen una capa de entrada, una de salida y neuronas artificiales ocultas completamente conectadas. Las CNN, por otro lado, tienen lo mismo, pero también múltiples capas de agrupamiento, convolucionales, de normalización y totalmente conectadas, lo que mejora la detección y seguimiento de objetos de manera eficiente.

YOLO tiene 29 capas de CNN y dos capas completamente conectadas. Se alternan las capas convolucionales para reducir gradualmente el espacio con las características del objeto. Las iniciales extraen las características de las imágenes y las capas completamente conectadas predicen probabilidades y coordenadas de salida del objeto.

YOLO-V4 es una versión mejorada del original, con un aumento del 10% en precisión y del 12% en velocidad en comparación con YOLO-V3. YOLO-V4 se compone de CSPDarknet53 como una línea de transferencia de datos que tiene ramificaciones más pequeñas (backbone). CSPDarknet53 es una backbone que ayuda a mejorar la capacidad de aprendizaje de las redes neuronales convolucionales (CNN) (I. Serrano & López Peña, 2017), (Redmon, Divvala, Girshick, & Farhadi, 2016), (Bochkovskiy, Wang, & Liao, 2020).

El entrenamiento se realiza recopilando un conjunto de imágenes donde se encuentre el objeto a detectar, sin importar si aparece una o más veces en la imagen, posteriormente se especifica la ubicación de cada objeto que se quiera detectar presente en la imagen mediante la ayuda de la herramienta LabelImg (proceso de etiquetado). Con la cámara del vehículo robot implementada fueron recopiladas mediante un script que captura bloques de 70 fotografías por cada ejecución, un total de 3369 imágenes de vehículos, de las cuales el 80% (2695) se utilizó para entrenamiento y el resto como validación (674). La Figura 10, muestra la comparación entre las imágenes

obtenidas con la cámara implementada y una de mayor calidad.



Figura 10: Comparación de la calidad de las cámaras.

Con el fin de reducir los errores en la percepción de la cámara, todas las imágenes fueron tomadas con la misma cámara a emplear en el prototipo final. Se capturaron imágenes en escenarios con lluvia y sin lluvia, de día y de noche.

Una vez etiquetadas todas las imágenes, estas son divididas en las proporciones mencionadas con anterioridad para el entrenamiento y validación de la red convolucional de YOLOv4-Tiny.

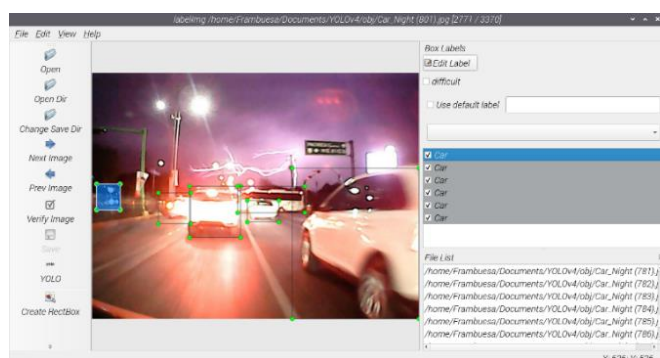


Figura 11: Proceso de etiquetado de imágenes en LabelImg.

3.5. Entrenamiento

Al ser un proceso que demanda una gran cantidad de recursos, el entrenamiento no pudo ser realizado directamente en la Raspberry, por lo que se optó por la herramienta Google Colab, utilizando los pesos de la red previamente entrenada del autor Aleksey Bochkovskiy, yolov4-tiny, misma que es una red convolucional de 29 capas.

El funcionamiento deseado en esta parte del sistema es ajustar el setpoint de velocidad del vehículo una vez sea detectado un automóvil al frente en el mismo carril, por esto se delimita la sección de medición, haciendo que, si un vehículo está fuera del rango de detección del carril, no será tomado en cuenta, por otro lado, si se encuentra dentro del rango, el vehículo será considerado.

4. Diseño de prototipo

Antes de la implementación del prototipo se requiere de un diseño de todas las etapas de detección, medición y actuadores, esta sección mostrará cada uno de los sistemas presentes en el robot.

El sistema de instrumentación debe tener la capacidad de adquirir mediciones por diferentes medios que permitan la operación incluso cuando uno de los sensores no logre obtener una medición, esta es la razón del uso de visión artificial como método sensor principal, el uso de LiDAR como método secundario y radar como último método de adquisición de la distancia al obstáculo frente al vehículo, dato solicitado para el cálculo del setpoint, lo anterior es ilustrado en la Figura 12.

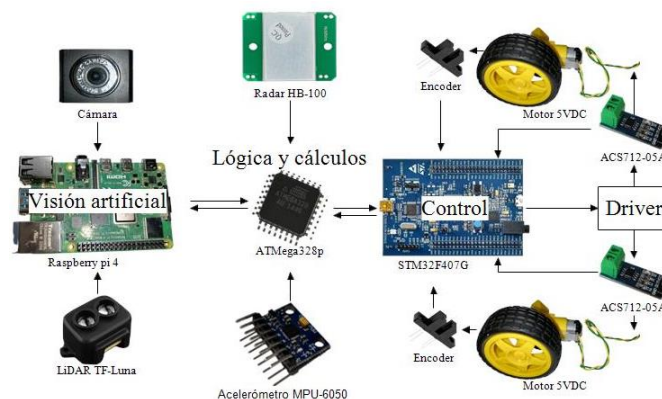


Figura 12: Diagrama general del prototipo

4.1. LiDAR

Nombrado en algunas ocasiones como radar láser, consiste en un método de medición y mapeo de objetos de gran precisión. Su funcionamiento se basa en la comparación de las características del reflejo de una señal óptica con la propia señal, iluminando el objetivo por medio de un láser.

Pese a la existencia de muchos modelos de LiDAR, el de estado sólido es el más empleado en el sector automotriz, ya que es un sensor que no contiene partes móviles, lo que le permite tener una menor cantidad de ruido en la señal y se puede considerar que no sufre ningún desgaste con el uso (Khader & Cherian, 2020).

Dado que el vehículo robot es pequeño, la cantidad de peso que puede cargar es limitada, incluyendo dispositivos, motores y baterías. Ante esta limitación, se considera el uso del sensor LiDAR TF-Luna, que se destaca por su bajo consumo eléctrico. Según su hoja de datos su potencia es de menos de 0.35 W, lo que representa un ahorro sustancial en comparación con otras alternativas. Por otro lado, un LiDAR de largo alcance tiene un consumo significativamente mayor (100W en su operación normal), lo que requeriría de baterías de mayor capacidad, aumentando el peso del vehículo y potencialmente afectando su capacidad de carga, rendimiento y maniobrabilidad (Aboujja, Chu, & Bean, 2022).

El LiDAR TF-Luna trabaja junto en conjunto con la cámara en la tarjeta Raspberry Pi 4B. Su comunicación está dada por el protocolo de transmisión serial, el sensor puede entregar mediciones extras que ayudan a conocer su estado de funcionamiento, en este caso solo tomando los valores de distancia.

Recordando que la visión está limitada a un área de detección del automóvil, en la Figura 13, desde a), hasta c) se muestran lo que sucede cuando un automóvil se encuentra frente al vehículo robot. La distancia no se tiene en cuenta en función de su ubicación, sino que se calcula el promedio entre la distancia obtenida por el LiDAR y la distancia calculada

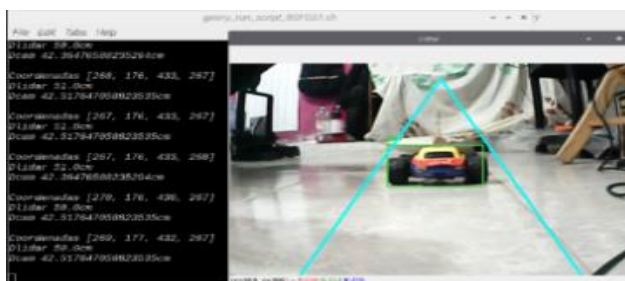
con las coordenadas en el fotograma de detección, si el objeto se encuentra dentro de la zona delimitada. En caso de no haber detección por parte de la cámara, se utilizará la distancia del LiDAR para evitar colisiones con objetos no detectados. Esto también se aplica si la detección se encuentra fuera de la zona.



a)



b)



c)

Figura 13: Medición de distancia con LiDAR.

4.2. Acelerómetro

Es una herramienta de medición que detecta la aceleración y vibraciones presentes en un cuerpo. Aplicaciones avanzadas utilizan el acelerómetro con fines de navegación, midiendo los cambios de velocidad en un rango de tiempo conocido, se determina la aceleración o desaceleración del cuerpo. Adicionalmente, si se llevan a cabo mediciones en diferentes direcciones se puede determinar la inclinación del cuerpo en el espacio, capacidad que es comúnmente conocida como giroscopio (Faisal, Purboyo, & Ansori, 2020).

La señal es enviada en 2 registros de 8 bits por cada eje, a pesar de esto, la medición directa del sensor no entrega el ángulo de inclinación, para conocerlo deben unirse ambas partes de la lectura y realizar algunas transformaciones, en (12) se muestra la manera en que se unen los registros.

$$a_x = \frac{(Re\ g_H < 8) | Re\ g_L}{16384}, \tag{12}$$

donde:

- a_x es la aceleración sobre el eje x (punto flotante),
- Reg_H es el registro alto (8bits),
- Reg_L es el registro bajo (8bits).

Cuando han sido obtenidas todas las aceleraciones de los ejes, para conocer el ángulo en el eje de interés, se aplica un nuevo cálculo mediante (13) que involucra a los dos ejes restantes, para este caso, el eje de interés es el x.

$$\theta_{xm} = \left(\frac{180}{\pi}\right) \left[\tan^{-1} \left(\frac{a_y}{\sqrt{a_x^2 + a_z^2}} \right) \right]. \tag{13}$$

Se comunica con el microcontrolador ATmega328p por medio de comunicación I²C. Es necesario para el funcionamiento del sistema, debido a que se obtiene medición de ángulo del vehículo, este sensor es principal ayuda del siguiente sensor radar.

4.3. Radar

Uno de los atributos más convenientes del uso de radar es que pueden tener una gran exactitud incluso operando en ambientes con poca o mucha luz, así como en lugares con poca visibilidad como en condiciones de lluvia, nieve, niebla, neblina, nubes de polvo. La distancia de medición depende del tipo de radar, estos se pueden clasificar por sus bandas de frecuencia.

La frecuencia juega un rol demasiado importante en estos dispositivos, ya que, al funcionar con el principio del efecto Doppler. Si hay un movimiento en el objetivo a medir, se producirá un cambio en la frecuencia debido a dicho efecto. Este cambio es proporcional a la velocidad del objeto en relación con el radar (Santiago Cruz, 2016).

Tabla 1: Denominación de bandas de operación de radar.

Banda	Frecuencia nominal	Frecuencia específica
C	4,000 – 8,000 MHz	5,250 – 5,925 MHz
X	8,000 – 12,000 MHz	8,500 – 10,680 MHz
K_u	12 – 18 GHz	13.4 – 14, 15.7 – 17.7GHz

En la Tabla 1 se especifican algunas de las bandas y su denominación de operación de los radares. En especial la banda de interés es la banda X, porque sus módulos son de los más comunes, y a su vez, de las opciones más económicas. La banda X es usada en sistemas de radar donde el tamaño de la antena implica una limitación física. Se usan en sistemas que requieren detección fina de objetos (Barton, 2005).

Se usó un radar HB-100 de efecto Doppler, el cual tiene como salida el parámetro de frecuencia, con amplitud muy pequeña y con bastante ruido. La Figura 14 muestra el oscilograma obtenido a partir de esa señal.

La solución que se da a este problema es la inclusión una etapa de amplificación y comparación por medio de amplificadores operacionales, ilustrado en la Figura 15.

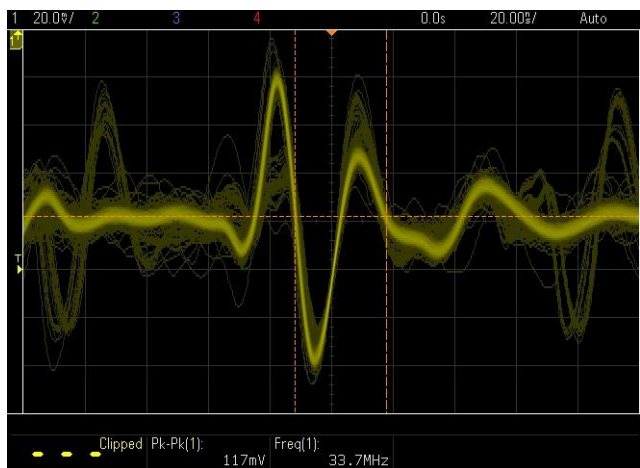


Figura 14: Señal a la salida del sensor.

La salida del circuito acondicionador muestra una señal que entrega una onda cuadrada con una frecuencia fácilmente medible mediante el módulo de captura del Timer de un microcontrolador.

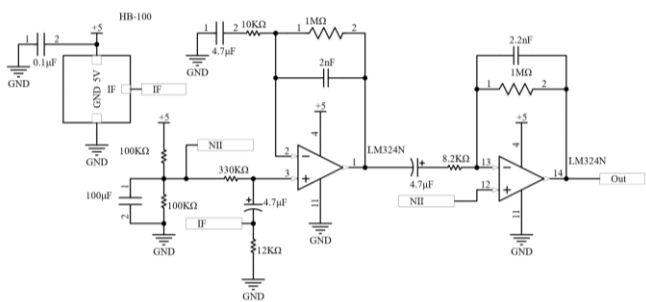


Figura 15: Circuito acondicionador de señal del sensor radar HB-100.

Utilizando el microcontrolador ATmega328p, se realiza la lectura de la señal con el método mencionado con anterioridad, el microcontrolador realiza la captura de la frecuencia emitida por el sensor que por medio de (14) se obtiene la velocidad lineal relativa a la que se mueve el objeto frente al sensor.

$$v = \frac{f_D C_0}{2f_0 \cos \alpha'} \tag{14}$$

donde:

- f_D es la frecuencia Doppler (medida),
- f_0 es la frecuencia de transmisión del sensor,
- C_0 es la velocidad de la luz,
- α es el ángulo relativo del objetivo,
- v es la velocidad relativa del objetivo.

La comprobación del funcionamiento de la captura del microcontrolador se realizó ingresando frecuencias conocidas producidas por un generador de señales. La Figura 16, muestra la frecuencia de la señal generada medida desde el osciloscopio.

Al realizarse pruebas con diferentes valores de frecuencia se mantuvieron resultados consistentes cada vez, lo que permite concluir que el script funciona adecuadamente presentando una desviación estándar por muestra $S = 0.71$. Con la certeza que la obtención de la lectura sería correcta se

realizaron nuevas pruebas leyendo la señal del sensor esta vez, obteniendo nuevamente resultados favorables.

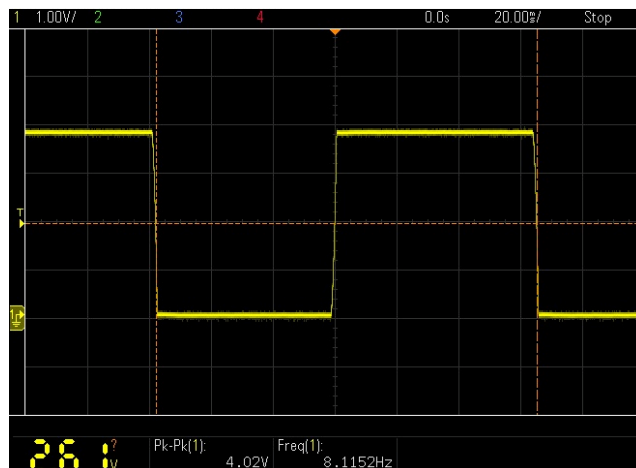


Figura 16: Medición de frecuencia de onda del osciloscopio.

La Tabla 2, enumera algunos de los valores obtenidos por el microcontrolador y el osciloscopio manteniendo un error menor a 2 Hz.

Tabla 2: Comparación de mediciones.

Medición	Microcontrolador (Hz)	Osciloscopio (Hz)
1	8.95	8.11
2	24.13	25.26
3	37.76	39.27

4.4. Encoder

La selección del opto-interruptor ITR8102 fue hecha por su bajo costo y funcionamiento sencillo. Consta de un diodo emisor de luz infrarroja frente a un fototransistor, entre ambos dispositivos se hace girar un disco con 20 ranuras movido por eje del motor.

En la configuración de la Figura 17, se mantiene encendido el diodo emisor de luz infrarroja y el transistor se queda a la espera de la interrupción de luz, dando como resultado una señal de onda cuadrada que es leída por el microcontrolador STM32F407G, donde se realizan los cálculos mostrados con anterioridad en el apartado de diseño de control.

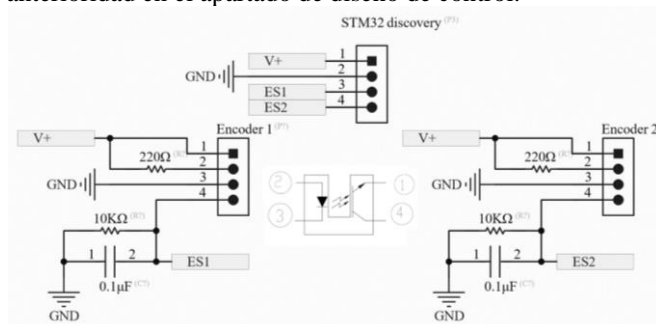


Figura 17: Diagrama eléctrico de conexiones del encoder.

4.5. Sensor de corriente

Es necesario medir la corriente para el funcionamiento del control de motores. Un sensor interrumpe una de las líneas de alimentación del motor, haciendo que la corriente circule a través de este, logrando el cambio en el campo magnético

detectado por un sensor de efecto Hall que se encarga de realizar la medición, a este método se les denomina sensor invasivo, ya que se tiene que interrumpir una terminal para hacerla pasar por el sensor.

El sensor de corriente seleccionado es ACS712-05A. Este sensor de efecto Hall entrega una referencia de voltaje en función de la corriente. La razón de conversión de señales se muestra en (15).

$$V_{out} = \frac{V_{cc}}{2} + \frac{168mV}{A} \tag{15}$$

4.6. Control de corriente

La implementación de un control de corriente es necesaria para un vehículo que opera con motores eléctricos de corriente directa. Para lograr esto, se utiliza una transformación de la señal de control adaptable a una señal de corriente válida para los motores. La integración del control de corriente se realiza mediante una interrupción DMA enlazada al ADC3 del microcontrolador STM32F407G y la interrupción por actualización del PWM del Timer 8. Esto permite sincronizar una frecuencia de muestreo de 3 KHz y ajustar la transmisión de datos entre las partes del sistema dentro y fuera de la interrupción.

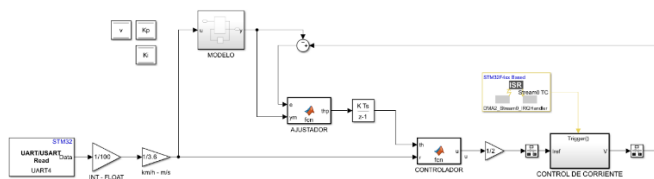


Figura 18: Integración del control de corriente en el control adaptable.

Dentro de la interrupción, se llevan a cabo dos procesos independientes pero simultáneos: la lectura de corriente en cada motor mediante el módulo ADC3 y la obtención de la medición de velocidad de las ruedas mediante el pin 12 del puerto D. Se utiliza un controlador proporcional-integral (PI) para manejar las corrientes y entregar el ciclo de trabajo del PWM a los Timers 1 y 8. El diseño de este controlador es genérico y solo se enfoca en la conexión entre lo electrónico y lo mecánico, sin profundizar en su desarrollo.

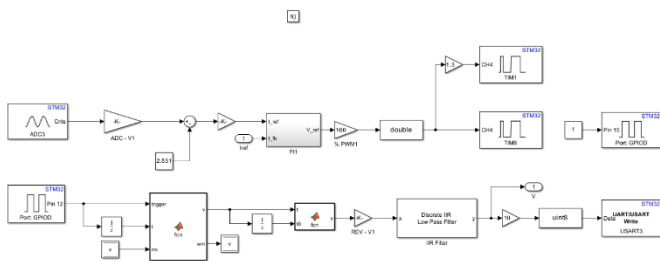


Figura 19: Funcionamiento interno del control de corriente.

4.7. Comunicación

La tarjeta Raspberry Pi 4B usada para el reconocimiento y lectura de distancia con el LiDAR, debe ser capaz de comunicar esa información al microcontrolador ATmega328p, el cual tiene la tarea de dar la lectura del radar y el acelerómetro, así como realizar los cálculos pertinentes

para obtener el setpoint de velocidad, que será transmitido al microcontrolador STM32F407G que realiza el control y lecturas de sensores de corriente y encoders utilizados para cerrar el lazo del controlador.

El procesamiento de la visión artificial demanda demasiados recursos que implican un tiempo de respuesta de 950 ms, lo que alentaría más el proceso si se aplicara directamente el control y la visión artificial en una sola microcomputadora Raspberry. De igual manera, el microcontrolador STM32F407G consume gran parte de sus recursos solo en la aplicación del control, por lo que requiere que un tercero realice los cálculos relacionados al acondicionamiento de la velocidad de referencia, es aquí donde se integra el microcontrolador ATmega328p, La razón de esta serie de dispositivos fue por su accesibilidad económica y fácil programación e implementación.

Comunicación entre Raspberry Pi 4B Y ATmega328p.

Esta sección del sistema asigna la velocidad del control adaptable. Se verifica la presencia de obstáculo y se mide la distancia y velocidad. Ambos dispositivos se comunican mediante el protocolo SPI a una velocidad de 4 MHz. Esta velocidad evita saturación en el procesamiento y asegura una respuesta rápida.

Comunicación entre ATmega328p Y STM32F407G.

La segunda mitad del proceso de comunicación transmite el setpoint de velocidad calculado a través del protocolo de comunicación serial a una velocidad de transmisión estándar de 9600 baudios/s, con el fin de mantener una transmisión estable y con un error controlado del 0.2%, que ofrece el microcontrolador en esta configuración, tomando en cuenta que el oscilador conectado al mismo es de 16 MHz.

4.8. Implementación del prototipo

Cuando se realiza un análisis teórico para el diseño de un controlador, se suelen tomar consideraciones de diseño en las cuales idealmente el sistema operará al ser implementado, sin embargo, las condiciones de funcionamiento en la realidad suelen discrepar de lo estipulado en la teoría.

Se busca llevar a la realidad la aplicación del control en un vehículo robot de dimensiones 0.25 m x 0.14 m y velocidad máxima de rueda de aproximada de 180 RPM, equipado con la cámara, LiDAR, radar, acelerómetro, sensores de corriente y encoders necesarios para una operación adecuada.

El diseño de una tarjeta electrónica para la instrumentación del control de velocidad crucero adaptable es indispensable para evitar falsos contactos que ocasionan errores en medición y de operación.

Altium Designer es una herramienta de desarrollo que brinda un extenso entorno de desarrollo de tarjetas de circuitos eléctricos, esta herramienta facilita el diseño completo del sistema embebido partiendo desde el diagrama esquemático pasando por el acomodo de componentes, pistas de conexión, tamaño de tarjeta, llegando hasta la generación de archivos de fabricación (Gerber, pick and place y active BOM).

La Figura 21 muestra componentes enrutados para evitar ruido entre señales. La tarjeta tiene dimensiones de 79 mm x 77 mm, por lo que se optimiza el espacio usando dos capas. La capa superior tiene componentes "through-hole" y la capa inferior tiene componentes SMD. Se enrutan las conexiones en ambas capas, principalmente en la inferior. Las pistas

tienen distintos calibres según la señal que transportan. Las pistas más anchas soportan corrientes altas o frecuencias elevadas, mientras que las más delgadas se usan para corrientes o frecuencias bajas. Los puntos de cruce entre capas permiten la conexión y continuidad de las pistas.

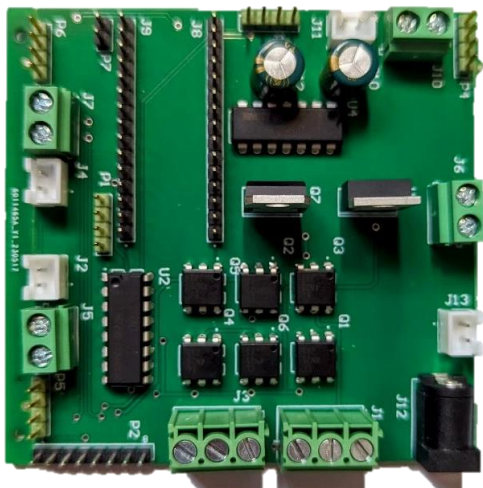


Figura 20: Tarjeta ensamblada.

Diseño e impresión de piezas en 3D. Para el montaje en el vehículo robot, es necesario el diseñar una pieza personalizada para evitar ruido eléctrico, así como desconexiones a causa del movimiento. Cuenta con bases para colocar todos los sensores empleados, por esto surge la motivación de diseñar un conjunto de piezas a medida impresas en 3D que sostengan los sensores en su posición.



Figura 21: Piezas 3D.

5. Resultados

En las Figuras 22-30 se muestra la velocidad obtenida por los sensores sin filtrar y filtrada para los diferentes experimentos realizados. El ruido es causado por la deficiente calidad de los sensores. Se aplican etapas de filtrado para tener una mejor medición, es decir, para tener una velocidad cercana a la real y poder retroalimentarla en el sistema de control. Para la retroalimentación de la velocidad en el programa del sistema, se agregó un filtro de respuesta infinita al impulso (IIR), ya que fue el que presentó mejores resultados al ser calibrado manualmente. También es importante destacar que el filtro no implica una alta carga de cómputo. Para la

obtención de las gráficas, se programó un filtro de media móvil. Este filtro calcula un promedio de las mediciones obtenidas por medio de comunicación serial desde el STM32F407G, que lee la salida de los encoders.

5.1. Regulación de velocidad

La regulación de velocidad del vehículo es el objetivo principal de un sistema de control crucero, ya sea tradicional o adaptable. En cada prueba, el setpoint de velocidad se establece en 1 y se modifica en situaciones donde el vehículo se encuentra con obstáculos en el camino.

Sin pendiente. En esta prueba, el vehículo robot comienza su recorrido en una superficie lisa y sin pendiente. En esta situación, muestra un desempeño aceptable, ya que mantiene una velocidad relativamente estable y en un momento se alcanzó la velocidad solicitada. La Figura 22, ilustra lo descrito anteriormente.

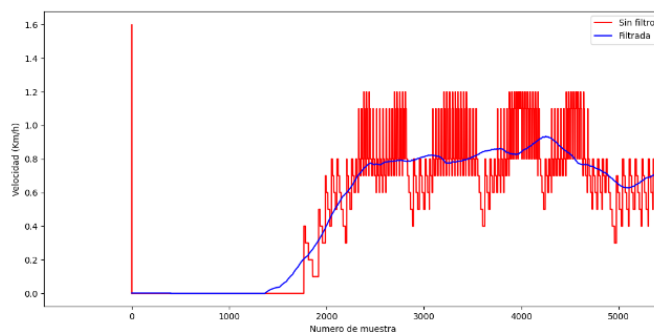


Figura 22: Gráfica de regulación de velocidad (sin pendiente).

Pendiente positiva 2%. Los resultados en esta ocasión superaron las expectativas, ya que el vehículo pudo alcanzar la velocidad deseada y mantenerla de manera estable dentro de un margen de error aceptable. El hecho de que la respuesta en una pendiente positiva fuera mejor que prueba la anterior puede resultar contraintuitivo. La justificación de este resultado se debe al lugar donde se llevó a cabo la prueba. En la primera prueba, realizada en un espacio reducido, la pared se detectó como un obstáculo al frente, lo que provocó que sistema realice un ajuste en la velocidad para evitar la colisión. Por otro lado, en la segunda prueba, la respuesta obtenida se muestra la Figura 23.

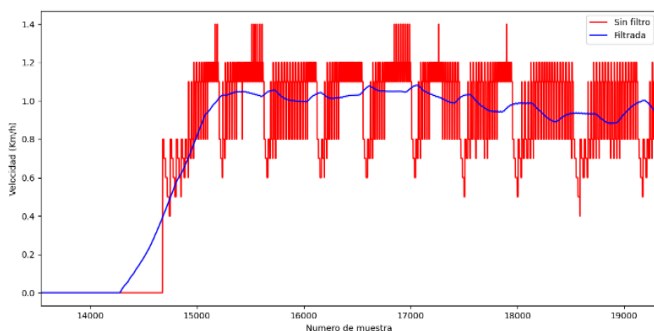


Figura 23: Gráfica de regulación de velocidad (pendiente positiva 2%).

Pendiente positiva 10%. Debido a las limitaciones mecánicas del vehículo, esta prueba resultó ser la más desafiante. Los motores tienen una capacidad de carga reducida, incluso con caja reductora, lo que significa que el peso del robot afecta la respuesta del controlador. Aunque la

respuesta es estable, no logra alcanzar la velocidad deseada, sino aproximadamente el 50% de esta. En la Figura 24, se observa que al controlador le resulta difícil mantener la velocidad actual, incluso cuando es menor a la solicitada.

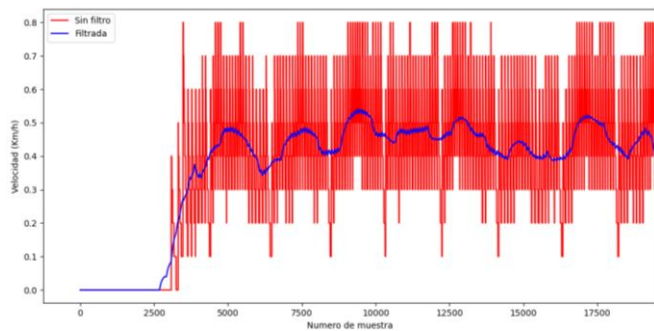


Figura 24: Gráfica de regulación de velocidad (pendiente positiva 10%).

Pendiente negativa 10%. En esta ocasión, se presenta una situación opuesta que la pendiente positiva, donde la velocidad supera la deseada. Aunque el control intenta regularla durante un instante, no lo consigue y solo ocasiona el detenimiento del vehículo. Al intentar compensarlo nuevamente, el vehículo vuelve a sobrepasar el setpoint. El comportamiento se puede observar en la gráfica de la Figura 25.

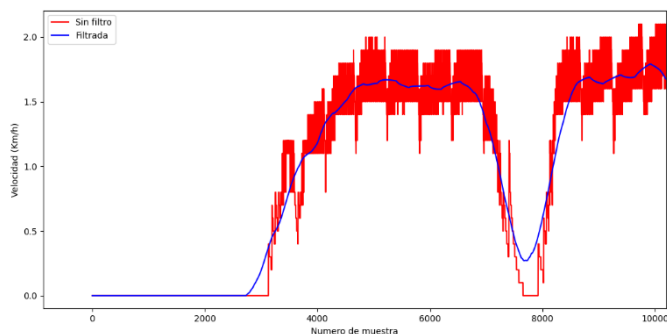


Figura 25: Gráfica de regulación de velocidad (pendiente negativa 10%).

5.2. Detenimiento y reanudación de avance

En un sistema de control adaptable, además de regular la velocidad, es necesario contar con la capacidad de detener el avance de ser necesario y, en algunos casos, reanudarlo una vez que el vehículo que inicialmente lo obstruía haya dejado de bloquear el camino.

Sin pendiente. Retomando la primera prueba realizada, posterior al avance un segundo vehículo se cruza en el trayecto del primero, obligándolo a detenerse hasta que el camino esté libre y pueda continuar. La Figura 26, muestra como el vehículo robot tiene una velocidad establecida cuando se encuentra con un obstáculo y se ve forzado a detenerse, posteriormente, el segundo vehículo deja de obstruir el camino y puede reanudarse el trayecto por un instante hasta llegar a la pared, la cual representaría el final del camino.

Pendiente positiva 2%. De la manera similar, después de lograr la estabilización de la velocidad en una pendiente moderada, se interrumpe el avance debido a la presencia de un vehículo en el camino. Sin embargo, una vez el vehículo se retira, se reanuda la trayectoria hasta llegar al final del camino, donde el robot se detiene nuevamente. La Figura 27 expone el

comportamiento del vehículo robot al momento de suceder lo antes explicado.

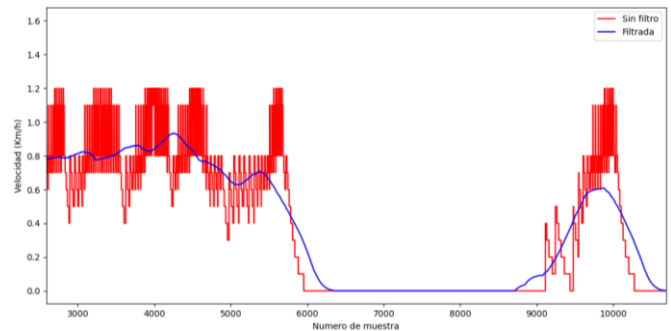


Figura 26: Gráfica de velocidad con detenimiento de trayectoria (sin pendiente).

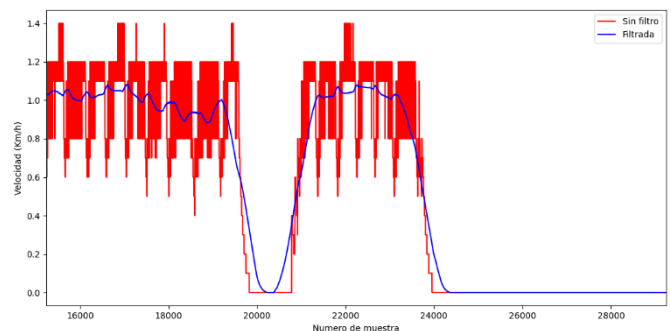


Figura 27: Gráfica de velocidad con detenimiento de trayectoria (pendiente positiva 2%).

Pendiente negativa 10%. En la prueba previa con dicha pendiente, el vehículo superó el setpoint de velocidad. No obstante, el detenimiento y la reanudación de la trayectoria indica un resultado plenamente favorable, tal como se ilustra en la Figura 28. Como se indicó anteriormente, cuando el vehículo se ve interrumpido por el obstáculo mientras se desplaza a una velocidad establecida, responde de manera oportuna y se detiene. Posteriormente, cuando se retira el obstáculo, continúa con la velocidad crucero. Aunque esta respuesta sea la esperada, es importante resaltar que al final de la prueba (al llegar al final del camino), el sistema es capaz de percatarse de dicha situación y reduce suavemente la velocidad de manera anticipada.

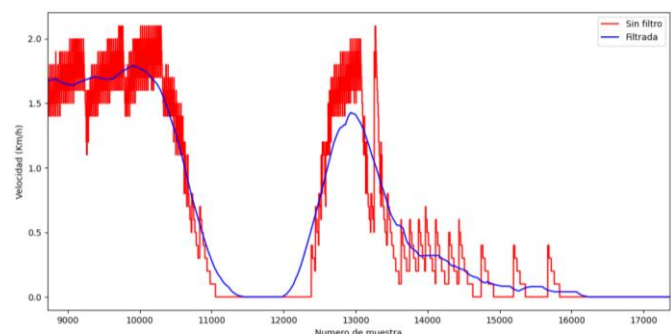


Figura 28: Gráfica de velocidad con detenimiento de trayectoria (pendiente negativa 10%).

5.3. Operación con poca visibilidad

Uno de los objetivos de este trabajo es que el sistema sea capaz de operar incluso cuando la visibilidad sea poca, por esto se realizó una prueba en la que el robot nuevamente debe

ir por una trayectoria en la que un segundo vehículo se interpone y posteriormente se retira. Esta prueba fue realizada en condiciones donde no había iluminación y los resultados obtenidos se muestran en la Figura 29.

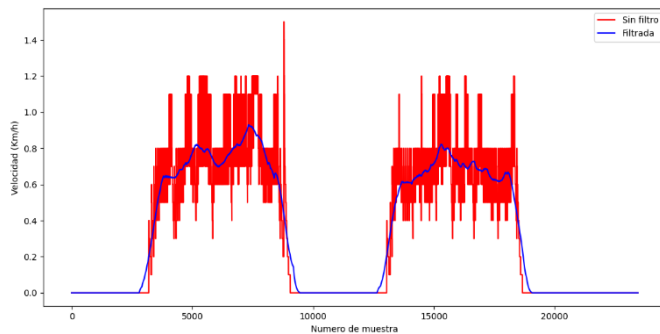


Figura 29: Gráfica de velocidad con poca iluminación.

5.4. Análisis general de resultados

Se observaron variaciones en el tiempo de respuesta, error y distancia de frenado durante cada prueba realizada. Sin embargo, a pesar de estas variaciones, el comportamiento del controlador fue aceptable. Se pudo notar que el desempeño del controlador se vio afectado en su mayoría por las limitaciones mecánicas de los motores y las características físicas del propio robot. Estas limitaciones, combinadas con las condiciones del entorno, propiciaron un ambiente desafiante para el controlador. A pesar de estas dificultades, el controlador se mantuvo estable, reflejando únicamente un error mayor en las pruebas más exigentes.

Seguimiento. Al finalizar todas las pruebas, se llevó a cabo una prueba final de seguimiento de vehículo que representó el funcionamiento continuo que el controlador debe soportar. Los resultados de esta prueba permiten realizar un análisis general del comportamiento obtenido, el cual se puede observar en la Figura 30, donde resaltan algunos puntos.

Al inicio de la prueba, el vehículo robot alcanza la velocidad del automóvil guía y, posteriormente, se ve forzado a reducirla ligeramente para ajustarse a una disminución de velocidad debido a un obstáculo en frente. Luego, se detuvo tras haber alcanzado la distancia de seguridad a causa del detenimiento momentáneo del obstáculo. En el siguiente instante el movimiento se ve reanudado nuevamente, limitado por la velocidad del vehículo que se encontraba al frente.

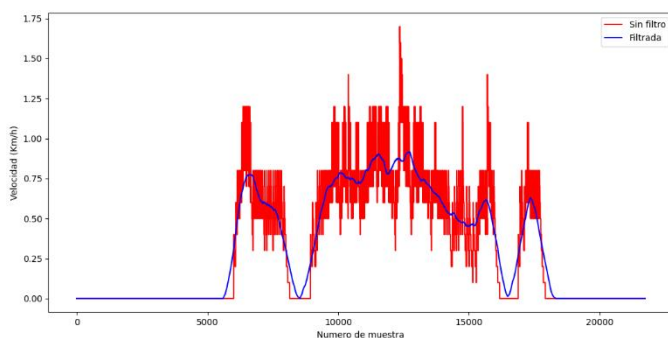


Figura 30: Gráfica de velocidad en seguimiento continuo de un vehículo.

En la tercera parte de la prueba, el automóvil guía supera la velocidad de setpoint, lo que permitió que el robot alcanzara dicha velocidad solo por un corto periodo, ya que el vehículo

guía comienza a reducir su velocidad nuevamente y el vehículo robot se ve forzado a ajustarse a su velocidad hasta el detenimiento. En la parte final de la prueba, después de una breve pausa en el avance, el obstáculo sale del camino, lo que permite que el vehículo robot pueda acelerar nuevamente justo antes de detenerse al finalizar la prueba.

6. Discusión

El sistema ACC se puede implementar en distintas vertientes del control, resultando interesante el análisis comparativo entre el control crucero adaptable por modelo de referencia (MRAC) y un control predictivo basado en modelo (MPC) como el que se menciona en (Andan & Ohol, 2022). Es interesante debido a que es implementable como bloque en Simulink, pero dicho análisis conlleva procedimientos importantes. La discretización es uno de ellos, ya que el modelo longitudinal del vehículo, en este caso tendría que ser discretizado, así como la optimización de la función de costo. Por su parte los dispositivos implementados en cada caso deberán cumplir con las características necesarias para llevar a cabo una implementación.

Las limitaciones de cómputo, así como el uso de diversos microcontroladores y microcomputadoras, produce un retardo en la velocidad de respuesta. Una alternativa prometedora radica en la incorporación de un FPGA, aprovechando su capacidad para ejecutar múltiples tareas de manera simultánea. Esto podría conducir a un sistema integrado en un solo dispositivo, manteniendo una respuesta rápida y eficiente ante situaciones inesperadas. Dado que un FPGA puede otorgar prestaciones según las necesidades específicas del sistema. Se considera que usar un FPGA daría lugar a un sistema superior de desarrollo en comparación con las alternativas mencionadas anteriormente, véase (Malle, Nyboe, & Ebeid, 2022).

7. Conclusiones

Siempre existen discrepancias entre el diseño teórico y la implementación real. Se presentaron situaciones como las limitaciones mecánicas que, en la simulación del diseño teórico, no representaron ningún problema. Incluso con las limitaciones se concluye que el controlador cumple con su objetivo, siendo estable, tal y como se analizó en el modelo de referencia y la ley de adaptación. También fue consistente durante su operación en diferentes escenarios mencionados en el apartado de recopilación y etiquetado de imágenes.

El controlador tarda aproximadamente 500 ms. La visión artificial del sistema tuvo un rendimiento de 950 ms debido a las limitaciones de la Raspberry Pi 4. Sin embargo, este tiempo de respuesta no afecta la velocidad de operación del robot. Por lo tanto, se requeriría de dispositivos con mejores prestaciones si se desea implementar en vehículos más rápidos.

El error en estado estacionario es posiblemente el punto más débil del sistema, ya que es el más afectado por las características mecánicas de los motores. Esto se hizo evidente en los terrenos con pendientes pronunciadas, donde la respuesta se desfasó del setpoint de velocidad, ya sea hacia arriba o hacia abajo, en función del sentido en que se transitaba la pendiente. En cambio, para pendientes de menor inclinación, el error se mantuvo dentro de un rango de $\pm 12\%$,

siendo capaz de ajustarse constantemente al nuevo setpoint de velocidad.

El controlador tuvo un desempeño destacable, viéndose limitado únicamente por el tiempo de respuesta que podía proporcionar la visión artificial. A su vez, este límite fue subsanado por la respuesta del resto de los sensores equipados en el sistema, que brindan una medición considerablemente más rápida y permiten que el controlador se adapte con mayor tasa de actualización.

Con respecto a los subsistemas de medición necesarios para la implementación del controlador, el sistema embebido diseñado demostró cumplir con su objetivo de manera óptima, aprovechando al máximo el espacio y recursos disponibles. La autonomía, debido a la capacidad energética de las baterías, fue su única limitante. No obstante, esto no forma parte interna de la circuitería diseñada.

El sistema es capaz de soportar condiciones de poca iluminación, ya que fue entrenado con imágenes en entornos oscuros y lluviosos. Los resultados de velocidad varían entre 0.7 km/h y 0.9 km/h durante su avance, mientras la detección de vehículos sigue operando correctamente. Si la cámara no detecta el vehículo delante, los sensores LiDAR y radar pueden continuar con la operación del sistema.

Agradecimientos

Los autores agradecen al Programa Institucional de Formación de Investigadores (PIFI) por el apoyo económico, así como a la Secretaría de Investigación y Posgrado del Instituto Politécnico Nacional SIP-IPN, proyectos: 20231269, 20232196 y 20230367.

Gabriel Alejandro Bañuelos Peña. A mis padres, a quienes expreso mi gratitud por todo el esfuerzo y sacrificio que han hecho para apoyarme a seguir mis sueños, por sus consejos y por ser mi mayor fuente de inspiración, muchas gracias, los amo y los admiro. A mis familiares por su apoyo y fraternidad en los mejores y en los más difíciles momentos, gracias por estar siempre a mi lado. A mi novia y a mis amigos, quienes siempre que tienen la oportunidad de ayudar lo hacen y por su paciencia, acompañamiento y palabras de apoyo.

Allan Christopher Coronado Andrade. Extiendo mi más sincero agradecimiento a mis padres y familiares que me apoyaron en cada proceso. Gracias a mis familiares, amigos, profesores y a mi novia por siempre creer en mí. Cada persona me inspiró para seguir avanzando.

Referencias

- Aboujja, S., Chu, D., & Bean, D. (2022). Multi-junction long-wavelength laser diode in long range LiDAR for high speed autonomous vehicles. *En Laser Radar Technology and Applications* (págs. 23-24). Orlando: SPIE. doi:10.1117/12.2618326
- Andan, S., & Ohol, S. S. (Octubre de 2022). Modelling and simulation of adaptive cruise control and overtake assist system. *Materials Today: Proceedings. ScienceDirect*, 72, 1353-1360. doi:10.1016/j.matpr.2022.09.330
- Barton, D. K. (2005). *Radar System Analysis and Modeling*. Norwood: ARTECH HOUSE, INC.
- Bochkovskiy, A., Wang, C. Y., & Liao, H. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv preprint*. doi:10.48550/arXiv.2004.10934
- Chandan, G., Jain, A., & Jain, H. (Julio de 2018). Real time object detection and tracking using deep learning and OpenCV. *En 2018 (ICIRCA)*, (págs. 1305-1306). Coimbatore: IEEE. doi:0.1109/ICIRCA.2018.8597266
- Cognex Corporation. (2016). Introducción a la visión artificial.
- Faisal, I. A., Purboyo, T. W., & Ansori, A. S. (2020). A Review of Accelerometer Sensor and Gyroscope Sensor in IMU Sensors on Motion Capture. *J. Eng. Appl. Sci*, 15(3), 826-829. doi:10.36478/jeasci.2020.826.829
- I. Serrano, A. S., & López Peña, A. M. (2017). YOLO Object detection for onboard driving images. *ddd. uab. cat*.
- Jain, R., Kasturi, R., & Schunck, B. G. (1995). *Machine Vision* (Vol. 5). New York: McGraw-hill.
- Khader, M., & Cherian, S. (2020). An Introduction to Automotive lidar. *Texas instruments*.
- Malle, N. H., Nyboe, F. F., & Ebeid, E. (2022). Onboard Powerline Perception System for UAVs Using mmWave Radar and FPGA-Accelerated Vision. *IEEE Access*, 10, 113543-113559. doi:0.1109/ACCESS.2022.3217537
- Rajamani, R. (2011). *Vehicle Dynamics and Control*. Minneapolis: Springer Science & Business Media. doi:10.1007/978-1-4614-1433-9
- Rajamani, R., & Zhu, C. (Septiembre de 2002). Semi-autonomous adaptive cruise control systems. *IEEE Transactions on Vehicular Technology*, 51(5), 1186-1192. doi:10.1109/TVT.2002.800617
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-time object detection. *En 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (págs. 779-788). Las Vegas: IEEE. doi:10.1109/CVPR.2016.91
- Rubio, F. R., & López Sánchez, M. J. (1996). Control adaptativo. *En Control adaptativo y robusto* (Vol. 9, págs. 10-16). Sevilla: Secretariado de publicaciones de la universidad de Sevilla.
- Sanabria S, J. J., & Archila D, J. F. (Diciembre de 2011). Detección y análisis de movimiento usando visión artificial. *Scientia Et Technica*, 16(49), 180-188.
- Santiago Cruz, J. A. (2016). *Radar detector de objetos*. Instituto Politécnico Nacional, Tesis de licenciatura. ESIME Zacatenco. Ciudad de México: IPN. Obtenido de <https://tesis.ipn.mx/handle/123456789/22009>
- Saxena, M. R., Pathak, A., Singh, A. P., & Shukla, I. (2019). Real time object detection using machine learning and OpenCV. *Int J Inform Sci Appl (IJISA)*, 11(1), 0974-225.
- Shakouri, P., Czczeczot, J., & Ordys, A. (2015). Simulation Validation of Three Nonlinear Model-Based Controllers in the Adaptive Cruise Control System. *J Intell Robot Syst*, 80, 207-229. doi:10.1007/s10846-014-0128-4