

# Plataforma de desarrollo para vehículos autónomos con visión computacional

Alejandro D.J. Gómez F.

ajgomezf@unal.edu.co

Universidad Nacional de Colombia Sede Medellín

Facultad de Minas

**Resumen**—El desarrollo de vehículos autónomos cada día es mas popular. Podemos ver empresas que actualmente se dedican a la fabricación de autos como BMW o Tesla, que actualmente buscan la manera de desarrollar algoritmos de navegación que faciliten la conducción de carros sobre diferentes entornos con el fin de otorgarle la tarea a máquinas inteligentes. Y definimos como máquinas inteligentes a aquellas que poseen una gran variedad de sensores, entre ellos cámaras que le permiten tener una visión de lo que ocurre alrededor y poder tomar una decisión a partir de la información capturada.

**Keywords**—autonomous cars, computer vision.

## I. INTRODUCCIÓN

La actual demanda de vehículos eléctricos cada vez aumenta por necesidad de transporte y a su vez, por los nuevos desarrollos a nivel industrial, que muestra una gran variedad de aplicaciones en el área de transporte, mensajería, asistencia, exploración, y demás, donde un vehículo que presente las características adecuadas para la navegación puede derivar en diversas aplicaciones, mas allá del uso actual. Observamos que, en la actualidad, el desarrollo de vehículos para la mensajería de paquetes y alimentos se hace cada vez mas popular. Tal es el caso de aplicaciones como Kiwi Campus [1], que realizan entrega de pedidos de alimentos a través de vehículos autónomos sobre un área específica. Y como característica principal observamos que, en las aplicaciones industriales, existe un sinfín de vehículos en diferentes tamaños que son representados por el sistema de movimiento Ackerman, para los sistemas de movilidad y navegación. La necesidad de un modelo a escala, surge al momento de buscar una manera de probar e implementar las diversas soluciones que los vehículos eléctricos puedan tener, siendo la mas popular en la actualidad, el desarrollo de vehículos capaces de navegar en espacios cerrados o abiertos para realizar diversas tareas, muchas de ellas con la finalidad de evitar la presencia humana cuando esta puede ser reemplazada por un vehículo, ya sea por temas de riesgo o automatización de tareas.

El presente trabajo tiene como finalidad, desarrollar un vehículo que sirva de plataforma de desarrollo para aplicaciones que requieran un sistema de visión computacional integrado para la navegación en diferentes entornos, partiendo de piezas que son accesibles de manera comercial, con la finalidad de desarrollar aplicaciones que permitan el desarrollo de algoritmos de navegación.

## II. DESARROLLO

Para la construcción de este vehículo, se establece primero los siguientes requisitos:

- Contar con piezas comerciales.
- Contar con una cámara.
- Poseer una unidad de procesamiento con GPU
- Sistema de alimentación con baterías eléctricas.
- Sistema de tracción Ackerman.
- Conectividad wifi.

### II-A. Chasis Mecánico

El vehículo a seleccionar es un carro a control remoto de la marca EXCEED-RC de escala 1:10, el cual posee un sistema de tracción similar al de un vehículo real, con el sistema de tracción Ackerman, presentado en la Figura 1. Este tamaño, 1:10 presenta beneficios al momento de necesitar espacio y tracción para mover las partes que integrarán al vehículo. En la Figura 2 se puede apreciar el interior del vehículo.



Figura 1: Carro RC Escala 1:10.

Este sistema cuenta con un control y receptor de radio frecuencia, donde podemos apreciar que para el caso del receptor, este opera con señales PWM para el manejo de la dirección y velocidad, tal como se muestra en la Figura 3, donde se observan los actuadores que reciben esta señal.

### II-B. Sistema de Cómputo

Se seleccionó la tarjeta Jetson Nano la cual se presenta en la Figura 4, de la marca nvidia, cuya tabla de especificaciones se muestra en la Figura 5. Vemos que es una tarjeta conveniente



Figura 2: Interior del vehículo.

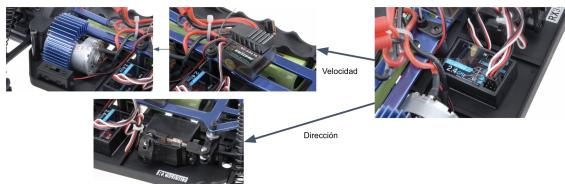


Figura 3: Receptor que opera con señales PWM para la velocidad y dirección.

para el uso de cámaras para el desarrollo de algoritmos que requieran visión computacional, por la GPU que viene integrada en la tarjeta.



Figura 4: Jetson Nano.

### II-C. Control Velocidad y Dirección

Empleando la Jetson Nano conectado a una tarjeta PCA9685, es posible manipular las señales PWM que se hicieron mención antes. La conexión con esta tarjeta, que funciona a través del protocolo i2c, se presenta en la Figura 6

### II-D. Sistema de alimentación

El chasis cuenta con un sistema de alimentación a través de una batería NiMh, con su respectivo cargador incluido. La capacidad de la batería originalmente es de 3300mAh entregando un voltaje máximo de 8,4V, el cual entrega hasta 20 minutos de operación con el vehículo a una velocidad de

Developer Kit Technical Specifications	
<b>GPU</b>	128-core Maxwell
<b>CPU</b>	Quad-core ARM A57 @ 1.43 GHz
<b>Memory</b>	4 GB 64-bit LPDDR4 25.6 GB/s
<b>Storage</b>	microSD (not included)
<b>Video Encoder</b>	4K @ 30   4x 1080p @ 30   9x 720p @ 30 (H.264/H.265)
<b>Video Decoder</b>	4K @ 60   2x 4K @ 30   8x 1080p @ 30   18x 720p @ 30 (H.264/H.265)
<b>Camera</b>	2x MIPI CSI-2 DPHY lanes
<b>Connectivity</b>	Gigabit Ethernet, M.2 Key E
<b>Display</b>	HDMI and DP
<b>USB</b>	4x USB 3.0, USB 2.0 Micro-B
<b>Others</b>	GPIO, I2C, I2S, SPI, UART
<b>Mechanical</b>	100 mm x 80 mm x 29 mm

Figura 5: Especificaciones Jetson Nano.

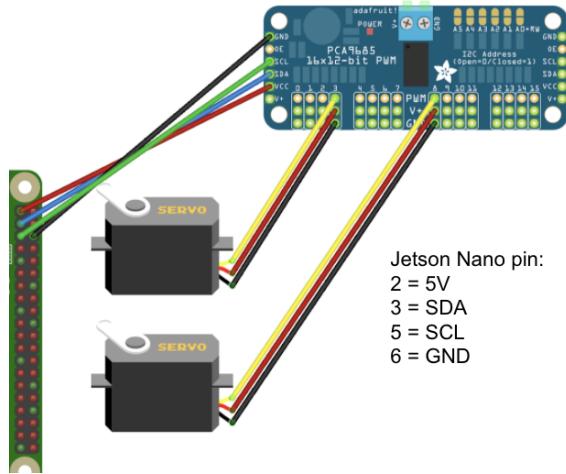


Figura 6: Conexión entre Jetson Nano y PCA9685.

5m/s. Para extender la duración de la batería y poder alimentar la tarjeta, se reemplazó la misma, por una de mayor capacidad, de 5000mAh.

Adicionalmente se empleó un conversor DC-DC de 5V y 2A el cual se ilustra en la Figura 7 para alimentar la Jetson Nano de la misma batería del vehículo.

### II-E. Cámara

La cámara usada para el sistema de visión computacional es la Logitech C920, la cual se puede apreciar en la Figura 8

### II-F. Conectividad

La Jetson Nano posee un socket para conectar una tarjeta de red, adicionalmente al contar con un puerto ethernet y una salida HDMI, adicional a los puertos USB, es posible

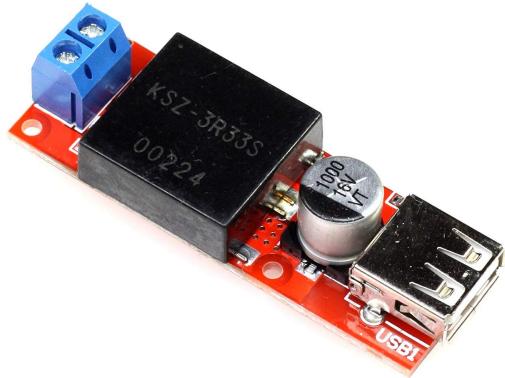


Figura 7: Conversor dc-dc KSZ-3R33S.



Figura 8: Cámara Logitech C920.

realizar la configuración usando una conexión de red, teclado y monitor.

La imagen que se presenta en la Figura 9 muestra el vehículo en su etapa de desarrollo incluyendo las partes ya mencionadas. En la Figura 10 se muestra el ensamble final.

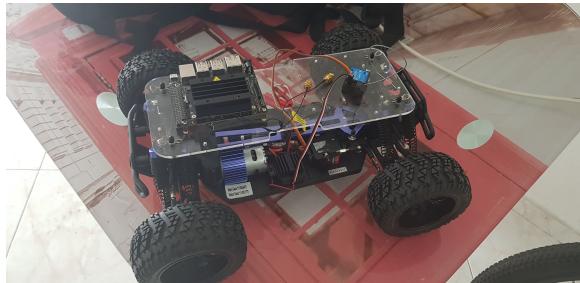


Figura 9: Vehículo en desarrollo



Figura 10: Vehículo con todas las partes ensambladas

#### REFERENCIAS

- [1] "Kiwibot | Food delivery robot." [Online]. Available: <https://www.kiwibot.com/>