

# Jornadas de Automática

## Detección de participantes del tráfico en entornos urbanos sobre imágenes RGB y nubes de puntos 3D

Montenegro, J.<sup>a,\*</sup>, García-Guillén, A.<sup>a</sup>, Castro, F. M.<sup>b</sup>, Martínez, J. L.<sup>a</sup>, Morales, J.<sup>a</sup>

<sup>a</sup>Dpto. de Ingeniería de Sistemas y Automática, Universidad de Málaga, Arquitecto Francisco Peñalosa, n° 6, 29071, Málaga, España

<sup>b</sup>Dpto. de Arquitectura de Computadores, Universidad de Málaga, Blvr. Louis Pasteur, n° 35, 29071 Málaga, España

**To cite this article:** Montenegro, J., García-Guillén, A., Castro, F.M., Martínez, J.L., Morales, J. 2024. Detection of Traffic Participants in Urban Environments from RGB images and 3D point clouds  
Jornadas de Automática, 45. <https://doi.org/10.17979/ja-cea.2024.45.10870>

### Resumen

Este artículo plantea el desarrollo de un entorno de pruebas para la detección de participantes del tráfico en entornos urbanos, mediante redes neuronales a partir del procesamiento de los datos procedentes de los sensores del vehículo: una cámara RGB y un sensor LiDAR 3D. Para ello se presenta la integración del simulador realista CARLA (*Car Learning to Act*), que permite la recreación de escenarios urbanos complejos, junto a ROS2 (*Robot Operating System*), que es un entorno para la creación de aplicaciones robóticas. En concreto, para el caso de imágenes RGB se evalúa cualitativamente el rendimiento de la red CNN (*Convolutional Neural Network*) YOLOv8 y la red transformadora especializada en detección DETR (*Detection Transformer*). De forma análoga, para la detección de participantes del tráfico en nubes de puntos se analizan las redes PV-RCNN (*PointVoxel Regional based Convolutional Neural Network*) y su evolución Part-A<sup>2</sup>-Net.

**Palabras clave:** Vehículos autónomos, sensores y actuadores automotrices, redes neuronales, programación y visión, percepción y detección, simulación.

### Detection of Traffic Participants in Urban Environments using RGB Images and 3D Point Clouds

#### Abstract

This article proposes the development of a test environment for the detection of traffic participants in urban environments using neural networks based on the processing of data from vehicle sensors: an RGB camera and a 3D LiDAR sensor. It presents the integration of the realistic simulator CARLA (*Car Learning to Act*), which allows the detailed recreation of complex urban scenarios, together with ROS2 (*Robot Operating System*), which is a framework for the development of robotic applications. Specifically, for the case of RGB images, the performance of the CNN (*Convolutional Neural Network*) YOLOv8 and the DETR (*Detection Transformer*) is qualitatively evaluated. Similarly, for the detection of traffic participants in point clouds, the PV-RCNN (*PointVoxel Regional based Convolutional Neural Network*) and its evolution Part-A<sup>2</sup>-Net are analysed.

**Keywords:** Autonomous vehicles, automotive sensors, neural networks, programming and vision, perception and sensing, simulation.

## 1. Introducción

La automatización se ha consolidado en una amplia variedad de sectores, entre ellos, la industria de la automoción (Biswas and Wang (2023)). En este sector, los vehículos autónomos están experimentando un continuo desarrollo y están ex-

pandiendo su presencia de manera progresiva en las carreteras.

En este ámbito, existen vehículos autónomos equipados con multitud de sensores, como es el caso del LiDAR 3D (Urmila. and Megalingam (2020)) o de las cámaras RGB (Moreau and Ibanez-Guzman (2023)), que proporcionan una gran

cantidad de datos del entorno junto al *software* de detección de obstáculos (Liu et al. (2023b)). Esto les permite alcanzar el nivel 3 de autonomía según la clasificación de la Sociedad de Ingenieros de la Automoción (SAE (2024)). En dicha clasificación se parte del nivel 1, correspondiente a sistemas de asistencia a la conducción como el mantenimiento de carril o el frenado automático de emergencia, hasta el nivel 5, que se corresponde con la completa autonomía del vehículo.

Además, cobra cada vez más fuerza el empleo de simuladores en el proceso de desarrollo de vehículos autónomos, lo que no solo reduce costes y tiempos, sino que también posibilita la realización de pruebas más seguras y versátiles (Nikolenko (2021)).

Tanto las clásicas Redes Neuronales Convolucionales (O’Shea and Nash (2015)) como las novedosas redes transformadoras (Carion et al. (2020)), aplicadas a la detección de objetos, demandan grandes cantidades de datos para su adecuado entrenamiento, verificación e implementación. En este contexto, los simuladores ofrecen la ventaja de generar datos fiables etiquetados automáticamente, junto a la posibilidad de modificar las condiciones de conducción.

En este artículo se presenta un entorno de pruebas para vehículos autónomos combinando ROS2 y CARLA para la detección de participantes del tráfico. Destaca la implementación y validación de diferentes modelos de redes neuronales convolucionales y transformadoras para el análisis de imágenes y nubes de puntos, comparando cualitativamente su rendimiento bajo diferentes condiciones.

El resto del artículo está organizado de la siguiente manera. El apartado 2 trata sobre el entorno empleado para generar y tratar las imágenes y nubes de puntos sintéticos. Tras esto, en la sección 3 se describen las redes neuronales empleadas para detectar participantes del tráfico. Posteriormente, en el apartado 4 se muestran los resultados de las detecciones realizadas por las redes neuronales. Por último, el artículo finaliza en la sección 5 con las conclusiones, agradecimientos y referencias.

## 2. Generación de imágenes y nubes de puntos sintéticos

La generación de imágenes y nubes de puntos sintéticos etiquetados constituye la base fundamental para el entrenamiento y la evaluación de sistemas de percepción y detección de obstáculos. En este contexto, los simuladores de un alto grado de realismo, como CARLA, permiten la creación de entornos urbanos complejos y dinámicos, donde cada elemento del escenario, denominado actor, puede ser controlado y modificado según se requiera para generar conjuntos de datos ricos y variados (Dosovitskiy et al. (2017)). Así se consigue la simulación de situaciones de tráfico realistas con diversos participantes como vehículos, peatones y obstáculos estáticos.

Dentro de las distintas posibilidades que ofrece CARLA, un aspecto a destacar es la generación de datos semánticamente etiquetados (Gannamaneni et al. (2021); Liu et al. (2023a)), una característica soportada tanto por las cámaras RGB como por los sensores LiDAR 3D integrados en CARLA. Estos datos etiquetados son cruciales para el entrenamiento de modelos de redes neuronales al proporcionar información precisa sobre la clasificación de objetos dentro de la escena simulada y generar automáticamente etiquetas semánticas para cada píxel de las cámaras RGB o para cada punto de las nubes

de puntos del LiDAR 3D. Todo ello elimina la necesidad de realizar laboriosos procesos de etiquetado manual, acelerando significativamente el ciclo de desarrollo de los algoritmos de percepción.

Una vez obtenidos los datos sintéticos, la herramienta *CARLA-ROS Bridge* ofrece una solución para integrar ROS2 con el simulador CARLA, facilitando la transferencia de datos entre el entorno simulado y ROS2. Gracias a la bidireccionalidad de esta herramienta, es posible tanto ajustar y actuar sobre parámetros del simulador, como obtener del mismo los datos de las cámaras RGB y del sensor LiDAR 3D.

En este escenario la utilidad de ROS2 radica en su capacidad para facilitar la recopilación, el procesamiento y la gestión de los datos generados por los sensores simulados (Fischer et al. (2021)), como son las cámaras RGB y los sensores LiDAR 3D. Además, esto se beneficia de la herramienta *ROS2 Bag*, que permite grabar y reproducir datos visualizados con RViz, facilitando la creación de bibliotecas de escenarios (*datasets*) de tráfico. Esto es útil para comparar enfoques de detección, asegurar la reproducibilidad de los resultados, y procesarlos posteriormente con redes neuronales para detectar participantes del tráfico. En la Figura 1 se muestran ejemplos de datos obtenidos mediante *CARLA-ROS Bridge* de los sensores del vehículo y visualizados con la herramienta RViz de ROS2.

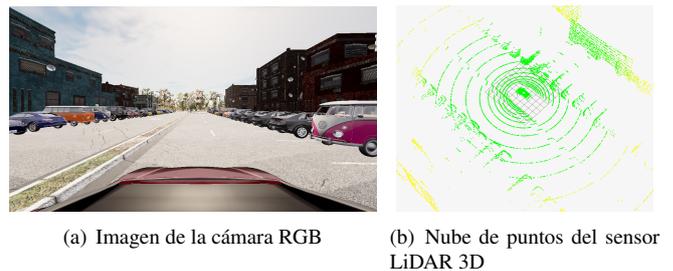


Figura 1: Ejemplo de visualización de los sensores con *rviz2*.

Por otro lado, en la Figura 2 se puede apreciar cómo el entorno ROS2 se comunica con CARLA permitiendo controlar los parámetros del simulador y teleoperar el vehículo sensorizado. Además, dicho simulador envía datos de los sensores a las redes neuronales para su procesamiento mediante ROS2. En el caso de las imágenes RGB, el procesamiento es directo por el paquete *ROS2 Object Detection*, mientras que para el caso del LiDAR 3D, se utiliza la herramienta *OpenPCDet* (OpenPCDet (2020)), que se comunica con ROS2 por medio del *wrapper PCDet.ROS2* (Pradhan (2023)).

## 3. Redes neuronales para la detección de objetos a partir de imágenes y nubes de puntos

Una de las aplicaciones que han adquirido importancia en los últimos años es la detección de participantes del tráfico para garantizar la seguridad y la eficacia de los vehículos autónomos en entornos urbanos (Balasubramaniam and Pasricha (2022)). En este contexto, las redes neuronales se han posicionado como herramientas fundamentales, capaces de procesar datos sensoriales y proporcionar información en tiempo

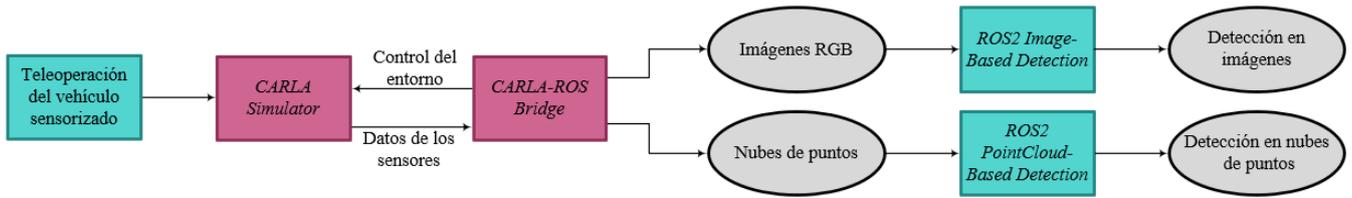


Figura 2: Esquema del entorno de trabajo. Los nodos de ROS2 se representan con un rectángulo azul, mientras que los tópicos con elipses.

real del entorno del automóvil. De entre los múltiples modelos de redes neuronales existentes, en este artículo se emplean las CNN (*Convolutional Neural Networks*) y DETR (*Detection Transformer*) para el procesamiento de imágenes RGB, mientras que para el caso de nubes de puntos se analizan las redes PV-RCNN (*PointVoxel Regional based Convolutional Neural Network*) y su evolución Part-A<sup>2</sup>-Net.

### 3.1. Detección de objetos a partir de imágenes RGB

Por un lado, las Redes Neuronales Convolucionales (O'Shea and Nash (2015)), compuestas por capas de convolución, de agrupación y una final donde todas las neuronas están conectadas entre sí, demuestran una capacidad excepcional para analizar imágenes capturadas por cámaras RGB, extrayendo características visuales significativas y clasificando eficazmente los objetos dentro de la escena (O'Shea and Nash (2015)). La estructura profunda de las CNNs y su habilidad para aprender patrones complejos las convierten en herramientas para la interpretación detallada de entornos urbanos. Sin embargo, el modelo original de CNN deja sin resolver múltiples problemas como la repetición de cajas de clasificación (*bounding boxes*). Es por ello que surgen algoritmos de alta eficacia pero cada vez más complejos, como es el caso de YOLO (*You Only Look Once*) (Redmon et al. (2016)).

Paralelamente, las redes transformadoras de detección representan una innovación significativa en el campo de la visión por computador, basándose en una arquitectura codificador-decodificador inicialmente diseñada para el procesamiento del lenguaje natural (NLP). Introducidas en 2017, las redes transformadoras revolucionaron los mecanismos de atención con su capacidad para realizar procesamiento paralelo, extendiendo posteriormente su aplicación al procesamiento de voz e imágenes. Esta evolución culminó en 2021 con el desarrollo de las DETR, diseñadas específicamente para tareas de detección de objetos (Carion et al. (2020)).

Además, existen variantes de las redes DETR, como las DETR-ResNet-50, DETR-ResNet-50-DC5 y DETR-ResNet-101, que se diferencian principalmente en su arquitectura y rendimiento (Carion et al. (2020)). Por un lado, la red DETR-ResNet-50 cuenta con 50 capas profundas, ofreciendo una precisión media (*Average Precision*)  $AP = 42\%$  y una velocidad de 28 FPS (*Frames Per Second*). Por otro lado, la red DETR-ResNet-50-DC5 incluye una dilatación en la quinta capa convolucional (C5), lo que ofrece una mejora en la precisión ( $AP = 43.3\%$ ) a costa de una menor velocidad (12 FPS). Finalmente, la red DETR-ResNet-101, cuenta con 101 capas profundas, ofreciendo una mejor precisión ( $AP = 43.5\%$ ) pero con una mayor velocidad que en el caso anterior (20 FPS).

La arquitectura de las redes DETR se caracteriza por su simplicidad y eficacia, dividida en tres componentes principales (Carion et al. (2020)): una CNN para la extracción de características, un codificador que, gracias a su propiedad llamada *atención*, establece relaciones entre diferentes partes de la imagen, y un decodificador que gestiona consultas de objetos para predecir clases y localizaciones. Esta estructura elimina la necesidad de mecanismos adicionales, solucionando de forma directa problemas asociados con las CNNs tradicionales. Sin embargo, un desafío de las DETR es su elevada demanda computacional para el entrenamiento, lo que podría limitar su aplicación en proyectos con recursos limitados. A pesar de este inconveniente, la disponibilidad de modelos preentrenados abre la posibilidad de integrar eficientemente estas redes en entornos como el simulador CARLA, ofreciendo una opción viable para el procesamiento avanzado de datos.

### 3.2. Detección de objetos a partir de nubes de puntos

En el ámbito de la detección de obstáculos en nubes de puntos tridimensionales (3D), para aplicaciones de conducción autónoma, se examinarán dos tipos de redes neuronales, como son las PV-RCNN (Shi et al. (2020a)) y las Part-A<sup>2</sup>-Free (Shi et al. (2020b)). ambas diseñadas para superar las limitaciones de los métodos convencionales en la detección de participantes del tráfico.

Las redes PV-RCNN, surgidas en 2019, representan una fusión entre 3D Voxel CNN y PointNet, propuesto en 2016 para el procesamiento directo de nubes de puntos mediante un enfoque basado en vóxeles (Shi et al. (2020a)). Esta red híbrida ha demostrado una capacidad sobresaliente para la detección de automóviles, superando a modelos existentes en precisión según evaluaciones realizadas con el repositorio KITTI (Geiger et al. (2013)), un estándar en la validación de algoritmos para vehículos autónomos. A pesar de su eficacia, la red PV-RCNN ha sido específicamente optimizada para la detección de vehículos, limitando su aplicación directa en la identificación de peatones y ciclistas, una consideración crucial para la navegación segura en entornos urbanos.

De forma similar, las redes Part-A<sup>2</sup>-Net, y su variante sin anclas (*anchor-free*), Part-A<sup>2</sup>-Free, introducidas también en 2019, abordan el análisis de nubes de puntos en dos etapas fundamentales: la detección de obstáculos y la agregación para el refinamiento de las *bounding boxes* (Shi et al. (2020b)). Este enfoque permite una detección y clasificación más efectiva de múltiples participantes de tráfico, incluidos vehículos, peatones, y ciclistas, ofreciendo una precisión media superior en comparación con otros modelos evaluados con KITTI. Además, la adaptación *anchor-free* del modelo Part-A<sup>2</sup>-Net demuestra una versatilidad y rendimiento mejorados.

#### 4. Pruebas y análisis. Resultados de los métodos de detección

Las pruebas se han realizado utilizando la versión 0.9.15. del simulador CARLA junto a la versión Humble de ROS2, dentro del sistema operativo *Ubuntu 22.04*, corriendo en un equipo con procesador Intel Core i7-13700F a 5.2 GHz, 16 núcleos y 24 MB L2 Cache, dos módulos de memoria RAM DDR5 16 GB 5600 MHz y una tarjeta gráfica GIGABYTE RTX 4070 TI GAMING OC con 12 GB.

En relación al vehículo utilizado en las simulaciones, se trata de un Toyota Prius con una cámara RGB estéreo, modelo ZED2i, instalada en el parabrisas frontal. Cabe destacar que, en este trabajo solo se utiliza un canal de la cámara estéreo. Adicionalmente, el sensor LiDAR 3D, modelo RS\_Helios 5515, está colocado sobre el techo del vehículo. Dichos sensores son simulados mediante la amplia gama de sensores virtuales que ofrece CARLA.

La metodología de pruebas incorpora diversas condiciones de simulación, que se muestran resumidas en la Tabla 1, donde se aprecian diferentes escenarios urbanos con la presencia de diferentes participantes de tráfico como peatones y vehículos, que incluyen coches, motos, ciclistas, camiones, furgonetas y autobuses. Esto permite examinar la robustez y flexibilidad de cada modelo de red neuronal frente a situaciones complejas y variadas que se presentan en la conducción autónoma. El rendimiento de los modelos se ha estimado en función de características como la confianza en la identificación de participantes bajo diversas condiciones de iluminación y visibilidad.

Tabla 1: Resumen de las condiciones de las simulaciones.

#	Vehículos	Peatones	Duración	Entorno	Mapa
1	30	25	3 min	Diurno y despejado	Town10
2	30	25	5 min	Nocturno y despejado	Town10
3	150	100	7 min	Diurno y despejado	Town15

##### 4.1. Simulación 1: Condiciones diurnas

En el primer escenario, para el caso de la detección de participantes del tráfico en imágenes RGB, algunos ejemplos se muestran en la Figura 3, mientras que algunos de los resultados de detección en la nube de puntos del LiDAR 3D se muestran en la Figura 4.

YOLOv8 demostró una alta velocidad de inferencia, aunque con menor precisión en la detección comparado con las redes DETR, que exhibieron un mejor rendimiento incluso con un umbral de confianza del 85 %. Entre las redes DETR, DETR-ResNet-50 y DETR-ResNet-101 destacaron por su rapidez, siendo DETR-ResNet-50-DC5 la que presentó un peor balance entre velocidad y potencia de detección.

La detección 3D reveló ciertas limitaciones en ambos modelos. Por un lado, la red Part-A<sup>2</sup>-Free demostró ser más efectiva en la detección de bicicletas, ofreciendo detecciones más precisas y frecuentes. Por el contrario, la red PV-RCNN, resultó ser más precisa en identificar autocares, mientras que encontró dificultades con las bicicletas. Sin embargo, ambas redes encontraron problemas para detectar peatones, lo cual

podría deberse a la falta de alteraciones discernibles en las nubes de puntos proporcionadas por el LiDAR.

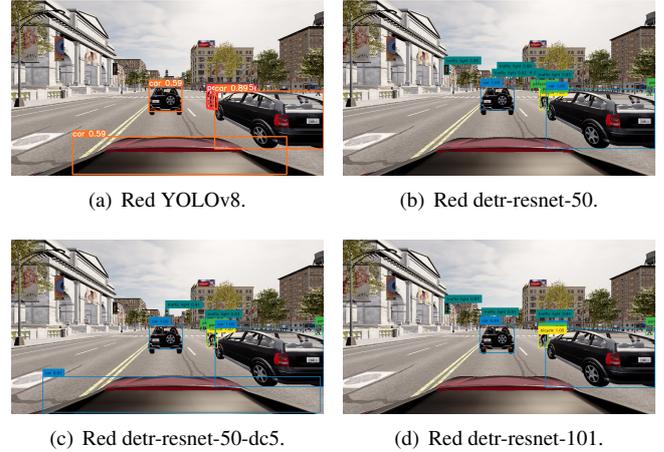


Figura 3: Procesamiento de las redes candidatas para imágenes RGB en la Simulación 1.

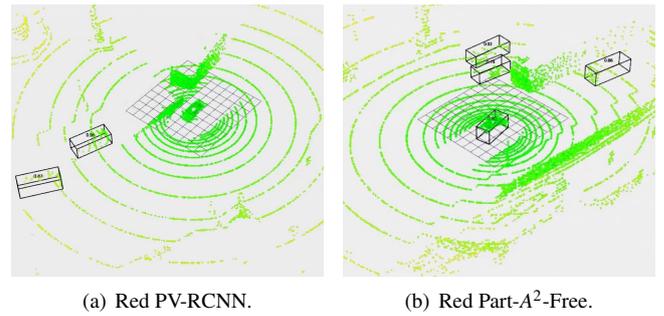


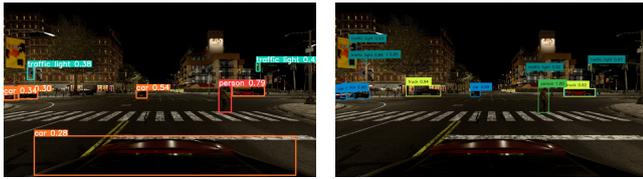
Figura 4: Procesamiento de las redes neuronales para nubes de puntos del LiDAR 3D en la Simulación 1.

##### 4.2. Simulación 2: Condiciones nocturnas

En la segunda simulación, para el caso de la detección de participantes del tráfico en imágenes RGB, algunos ejemplos se muestran en la Figura 5, mientras que algunos de los resultados de detección en la nube de puntos del LiDAR 3D se muestran en la Figura 6.

En cuanto a la detección usando las cámaras RGB, la transición a condiciones nocturnas no impactó significativamente en la eficacia de las detecciones 2D. Los resultados mostraron detecciones con un alto grado de confianza, de forma similar a lo obtenido en condiciones diurnas. Además, en el escenario nocturno se puede apreciar un mejor comportamiento de las redes transformadoras con densidades de tráfico moderadas, en contraposición a la red YOLOv8 que, a pesar de ser más rápida, mostró una potencia y fiabilidad inferiores en comparación con las redes DETR.

Respecto a la detección en nubes de puntos 3D, se obtuvieron los mismos resultados que en la primera simulación dado que, en el simulador CARLA, las variaciones meteorológicas no alteran el rendimiento del LiDAR.



(a) Red YOLOv8.

(b) Red detr-resnet-50.



(c) Red detr-resnet-50-dc5.



(d) Red detr-resnet-101.

Figura 5: Procesamiento de las redes neuronales para imágenes RGB en la Simulación 2.



(a) Red YOLOv8.



(b) Red detr-resnet-50.

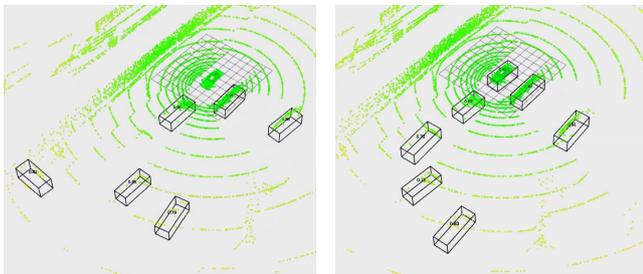


(c) Red detr-resnet-50-dc5.



(d) Red detr-resnet-101.

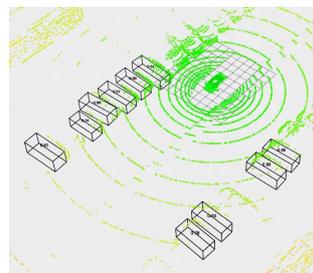
Figura 7: Procesamiento de las redes neuronales para imágenes RGB en la Simulación 3.



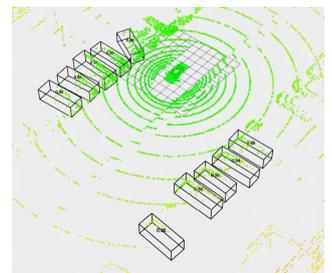
(a) Red PV-RCNN.

(b) Red Part-A<sup>2</sup>-Free.

Figura 6: Procesamiento de las redes neuronales para nubes de puntos del LiDAR 3D en la Simulación 2.



(a) Red PV-RCNN.



(b) Red Part-A<sup>2</sup>-Free.

Figura 8: Procesamiento de las redes neuronales para nubes de puntos del LiDAR 3D en la Simulación 3.

### 4.3. Simulación 3: Mapa ampliado con diversidad de tráfico

En la tercera simulación, a pesar de la alta cantidad de participantes, la densidad de tráfico resultó ser más baja en comparación con el mapa *Town10HD.Opt* debido a la mayor amplitud del mapa *Town15*. Para el caso de la detección en imágenes RGB, algunos ejemplos se muestran en la Figura 7, mientras que algunos de los resultados de detección en la nube de puntos del LiDAR 3D se muestran en la Figura 8.

Ante una menor densidad de tráfico, la diferencia de potencia y exactitud entre los distintos modelos de red se vio reducida en la detección 2D. Al igual que en los casos anteriores, YOLOv8 siguió mostrando ser la más rápida de las cuatro y la de menor potencia y exactitud. Por otro lado, el análisis con las cuatro redes Transformer proporcionó resultados muy similares entre ellas. De nuevo, el orden de rapidez en la inferencia de mayor a menor fue el siguiente: 1) YOLOv8, 2) DETR-ResNet-50, 3) DETR-ResNet-101, 4) DETR-ResNet-50-DC5.

En cuanto a la detección 3D, destaca la capacidad de las redes neuronales para generar las cajas de clasificación necesarias ante un gran número de detecciones, especialmente con vehículos estacionados que predominaron en la simulación. Sin embargo, en este escenario, se observó de forma más evidente que las detecciones ocurren con mucha más frecuencia por delante del vehículo, dándose en la parte trasera en menos casos.

## 5. Conclusiones

En este trabajo se ha presentado la integración de CARLA con ROS2 para facilitar el desarrollo de técnicas de detección de participantes del tráfico empleando redes neuronales. Para el caso de la detección en imágenes 2D, procedentes de cámaras RGB, sobresalió la red neuronal DETR en cuanto a precisión frente a la red YOLOv8, a cambio de una ligera disminución en la velocidad de procesamiento. En cuanto a la detección en nubes de puntos 3D, concretamente en el caso de bicicletas, destacó la red Part-A<sup>2</sup>-Free frente a la PV-RCNN, aunque se observó su incapacidad para detectar peatones de manera consistente. Finalmente, como línea futura de trabajo podría plantearse la extensión del presente trabajo incorporando una evaluación cuantitativa de las redes analizadas así como realizar una fusión multi-modal de las detecciones 2D y 3D.

## Agradecimientos

Este trabajo ha sido realizado gracias al Proyecto de Investigación de Excelencia de la Junta de Andalucía REMOVE (ProyExcel\_00684).

## Referencias

- Balasubramaniam, A., Pasricha, S., 2022. Object detection in autonomous vehicles: Status and open challenges. arXiv preprint arXiv:2201.07706. DOI: 10.48550/arXiv.2201.07706
- Biswas, A., Wang, H.-C., 2023. Autonomous vehicles enabled by the integration of IoT, edge intelligence, 5G, and blockchain. *Sensors* 23 (4). DOI: 10.3390/s23041963
- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-End Object Detection with Transformers. Springer International Publishing, pp. 213–229. DOI: 10.1007/978-3-030-58452-8\_13
- Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V., 2017. Carla: An open urban driving simulator. In: Proceedings of the 1st Annual Conference on Robot Learning. Proceedings of Machine Learning Research, pp. 1–16. DOI: 10.48550/arXiv.1711.03938
- Fischer, T., Vollprecht, W., Traversaro, S., Yen, S., Herrero, C., Milford, M., 2021. A robostack tutorial: Using the robot operating system alongside the conda and jupyter data science ecosystems. *IEEE Robotics and Automation Magazine*. DOI: 10.1109/MRA.2021.3128367
- Gannamaneni, S., Houben, S., Akila, M., 2021. Semantic concept testing in autonomous driving by extraction of object-level annotations from carla. In: 2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW). pp. 1006–1014. DOI: 10.1109/ICCVW54120.2021.00117
- Geiger, A., Lenz, P., Stiller, C., Urtasun, R., 2013. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research* 32 (11), 1231–1237. DOI: 10.1177/0278364913491297
- Liu, H., Gu, Z., Wang, C., Wang, P., Vukobratovic, D., 2023a. A lidar semantic segmentation framework for the cooperative vehicle-infrastructure system. In: 2023 IEEE 98th Vehicular Technology Conference (VTC2023-Fall). pp. 1–5. DOI: 10.1109/VTC2023-Fall160731.2023.10333790
- Liu, H., Wu, C., Wang, H., 2023b. Real time object detection using lidar and camera fusion for autonomous driving. *Scientific Reports* 13. DOI: 10.1038/s41598-023-35170-z
- Moreau, J., Ibanez-Guzman, J., 2023. Emergent visual sensors for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems* 24 (5), 4716–4737. DOI: 10.1109/TITS.2023.3248483
- Nikolenko, S. I., 2021. Synthetic data for deep learning. Vol. 174. Springer. DOI: 10.1007/978-3-030-75178-4
- OpenPCDet, 2020. OpenPCDet: An open-source toolbox for 3D object detection from point clouds. <https://github.com/open-mmlab/OpenPCDet>.
- O’Shea, K., Nash, R., 11 2015. An introduction to convolutional neural networks. ArXiv e-prints. DOI: 10.48550/arXiv.1511.08458
- Pradhan, S., 2023. ROS 2 wrapper for OpenPCDet. [https://github.com/pradhanshrijal/pcdet\\_ros2](https://github.com/pradhanshrijal/pcdet_ros2).
- Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788. DOI: 10.48550/arXiv.1506.02640
- SAE, 2024. Sociedad de ingenieros de la automoción. URL: <https://www.sae.org/>
- Shi, S., Guo, C., Jiang, L., Wang, Z., Shi, J., Wang, X., Li, H., 2020a. PV-RCNN: Point-voxel feature set abstraction for 3D object detection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10526–10535. DOI: 10.1109/CVPR42600.2020.01054
- Shi, S., Wang, Z., Shi, J., Wang, X., Li, H., 2020b. From points to parts: 3D object detection from point cloud with part-aware and part-aggregation network. *IEEE transactions on pattern analysis and machine intelligence* 43 (8), 2647–2664. DOI: 10.1109/TPAMI.2020.2977026
- Urmila, O., Megalingam, R. K., 2020. Processing of lidar for traffic scene perception of autonomous vehicles. In: 2020 International Conference on Communication and Signal Processing (ICCSP). pp. 298–301. DOI: 10.1109/ICCSP48568.2020.9182175