

Universidad de Alcalá

Escuela Politécnica Superior

Grado en Ingeniería Informática

Trabajo Fin de Grado

Algoritmos de detección de objetos 3D basados en LiDAR:
comparación entre técnicas PCL clásicas y Deep Learning

Autor: Javier de la Peña García

Tutores: Luis Miguel Bergasa Pascual y Carlos Gómez Huélamo

2021

UNIVERSIDAD DE ALCALÁ
ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería Informática

Trabajo Fin de Grado

**Algoritmos de detección de objetos 3D basados en LiDAR:
comparación entre técnicas PCL clásicas y Deep Learning**

Autor: Javier de la Peña García

Directores: Luis Miguel Bergasa Pascual y Carlos Gómez Huélamo

Tribunal:

Presidente: Felipe Espinosa Zapata

Vocal 1º: Fernando Naranjo Vega

Vocal 2º: Rafael Barea Navarro

Calificación:

Fecha:

A nuestros alumnos pasados, presentes y futuros...

“Empieza haciendo lo necesario, luego haz lo posible y de pronto empezarás a hacer lo imposible.”

Francisco de Asís

Agradecimientos

A todos los que la presente vieron y entendieron.

Inicio de las Leyes Orgánicas. Juan Carlos I

Este trabajo es el fruto de muchas horas de trabajo, tanto de los autores últimos de los ficheros de la distribución como de todos los que en mayor o menor medida han participado en él a lo largo de su proceso de gestación.

Mención especial merece Manuel Ocaña, el autor de la primera versión de las plantillas de proyectos fin de carrera y tesis doctorales usadas en el Departamento de Electrónica de la Universidad de Alcalá, con contribuciones de Jesús Nuevo, Pedro Revenga, Fernando Herránz y Noelia Hernández.

En la versión actual, la mayor parte de las definiciones de estilos de partida proceden de la tesis doctoral de Roberto Barra-Chicote, con lo que gracias muy especiales para él.

También damos las gracias a Manuel Villaverde, David Casillas, Jesús Pablo Martínez, José Francisco Velasco Cerpa que nos han proporcionado secciones completas y ejemplos puntuales de sus proyectos fin de carrera.

Finalmente, hay incontables contribuyentes a esta plantilla, la mayoría encontrados gracias a la magia del buscador de Google. Hemos intentado referenciar los más importantes en los fuentes de la plantilla, aunque seguro que hemos omitido alguno. Desde aquí les damos las gracias a todos ellos por compartir su saber con el mundo.

Resumen

Este documento ha sido generado con una plantilla para memorias de trabajos fin de carrera, fin de máster, fin de grado y tesis doctorales. Está especialmente pensado para su uso en la Universidad de Alcalá, pero debería ser fácilmente extensible y adaptable a otros casos de uso. En su contenido se incluyen las instrucciones generales para usarlo, así como algunos ejemplos de elementos que pueden ser de utilidad. Si tenéis problemas, sugerencias o comentarios sobre el mismo, dirigidlas por favor a Javier de la Peña García <j.pena@edu.uah.es>.

Palabras clave: Plantillas de trabajos fin de carrera/máster/grado y tesis doctorales, L^AT_EX, soporte de español e inglés, generación automática.

Abstract

This document has been generated with a template for Bsc and Msc Thesis (trabajos fin de carrera, fin de máster, fin de grado) and PhD. Thesis, specially thought for its use in Universidad de Alcalá, although it should be easily extended and adapted for other use cases. In its content we include general instructions of use, and some example of elements than can be useful. If you have problemas, suggestions or comments on the template, please forward them to Javier de la Peña García <j.pena@edu.uah.es>.

Keywords: Bsc., Msc. and PhD. Thesis template, L^AT_EX, English/Spanish support, automatic generation.

Resumen extendido

Con un máximo de cuatro o cinco páginas. Se supone que sólo está definido como obligatorio para los TFGs y PFCs de UAH.

Índice general

Resumen	ix
Abstract	xi
Resumen extendido	xiii
Índice general	xv
Índice de figuras	xix
Índice de tablas	xxi
Índice de listados de código fuente	xxiii
Índice de algoritmos	xxv
Lista de acrónimos	xxvii
1 Introducción	1
1.1 Sistemas de conducción autónomos	1
1.2 Sistemas de percepción	2
1.2.1 Principales sensores para la percepción en vehículos autónomos	2
1.2.1.1 Cámara	3
1.2.1.2 Radar	4
1.2.1.3 LiDAR	4
1.2.2 Sistemas de detección	5
1.2.3 Sistemas de seguimiento	6
1.2.4 Fusión sensorial	6
1.3 Deep Learning	6
2 Propuesta de trabajo	7

3	Sistemas clásicos de percepción con LiDAR	9
3.1	Voxelización	9
3.2	Ransac-3D	9
3.3	KD-tree	9
3.4	Filtrado posterior a la detección	9
4	Sistemas de percepción con LiDAR basados en Deep Learning	11
4.1	Principales datasets	11
4.1.1	Kitti	11
4.1.2	Waymo	11
4.1.3	nuScenes	11
4.1.3.1	nuScenes-devkit	11
4.1.4	Comparativa entre los diferentes datasets	11
4.2	Estado del arte en detección utilizando LiDAR	11
4.2.1	PointPillars	11
4.2.2	SECOND	11
4.2.3	PointRCNN	11
4.2.4	PV-RCNN	11
4.2.5	CBGS	11
4.3	OpenPCDet	11
5	Desarrollo realizado	13
5.1	Estado del proyecto T4AC	13
5.1.1	ROS	13
5.1.2	Docker	13
5.2	Implementación en CARLA	13
5.2.1	CARLA	13
5.2.2	Funcionamiento del LiDAR en CARLA	13
5.2.3	Tratado de datos	13
5.3	Fusión sensorial	13
5.4	Vehículo del proyecto T4AC	13
5.5	Implementación sobre el vehículo T4AC	13
6	ad-devkit	15
6.1	Estado del arte en evaluación de vehículos autónomos	15
6.2	Obtención del ground truth	15
6.3	Evaluación de los modelos	15

7	Resultados obtenidos	17
7.1	Análisis cuantitativo en Kitti	17
7.2	Análisis cuantitativo en nuScenes	17
7.3	Análisis cualitativo del modelo clásico en CARLA	17
7.4	Análisis cualitativo de CBGS en CARLA	17
7.4.1	Comparativa con PointPillars en CARLA	17
7.5	Análisis cuantitativo de CBGS en CARLA	17
7.6	Análisis cualitativo de CBGS sobre el vehículo T4AC	17
7.6.1	Comparativa con PointPillars sobre el vehículo T4AC	17
8	Conclusiones	19
8.1	Modelos estudiados	19
8.2	Comparativas adicionales	19
8.2.1	Ajuste de modelos basados en Kitti a nuScenes	19
8.2.2	Número de PCL de entrada en modelos evaluados sobre nuScenes	19
8.2.3	Tamaño del voxel en modelos basados en redes neuronales	19
8.3	Futuros trabajos	19
	Bibliografía	21
A	Herramientas y recursos	23

Índice de figuras

1.1	Ejemplo de arquitectura de un sistema de conducción autónoma.	1
1.2	Niveles de autonomía.	2
1.3	Ejemplo de cámara utilizado en vehículos autónomos.	3
1.4	Uso de cámara con lluvia.	3
1.5	Ejemplo de Radar utilizado en vehículos autónomos.	4
1.6	Ejemplo de LiDAR utilizado en vehículos autónomos.	5
1.7	Ejemplo de detecciones 2D utilizando cámara.	5

Índice de tablas

7.1	Rendimiento medio de CBGS PointPillars Multihead en nuScenes.	17
7.2	Análisis por clase de CBGS PointPillars Multihead en nuScenes.	18
7.3	Análisis por clase de CBGS PointPillars Multihead en CARLA.	18

Índice de listados de código fuente

Índice de algoritmos

Lista de acrónimos

ADAS Advanced Driver Assistance System.

ADS Automated Driving System.

FPS Frames per Second.

GNSS Global Navigation Satellite System.

GPS Global Positioning System.

IMU Inertial Measurement Unit.

LiDAR Light Detection and Ranging.

Radar Radio Detection and Ranging.

Sonar Sound Navigation and Ranging.

Capítulo 1

Introducción

No te conformes con el mundo que has heredado. Nunca se ha resuelto un desafío sin personas que pensasen diferente.

Tim Cook

1.1 Sistemas de conducción autónomos

En los últimos años gracias a una mejora en los sensores, en la capacidad de computo principalmente por la aceleración por hardware, la visión por computador, el deep learning y el desarrollo de técnicas de comunicación, ha propiciado que nos encontremos en una carrera por la creación de sistemas de conducción autónomos. Empresas de sectores de la automoción y la tecnológicas como ArgoAI, Audi, Baidu, Cruise, Mercedes-Benz, Tesla, Uber o Waymo entre otras, invierten enormes cantidades de dinero para el desarrollo de estas tecnologías [1].

Para la obtención de sistemas de conducción autónoma es necesario tener un buen entendimiento del entorno y hacer uso de un buen control en tiempo real, para ello son utilizados sensores que puedan aportar información al vehículo como son cámaras, [LiDAR](#), [Radar](#), [IMU](#), [GPS](#) o hasta [Sonar](#).

En adición a los sensores también es necesario la utilización de sistemas de localización tanto global como local, mapeado del entorno, toma de decisiones y control del vehículo.

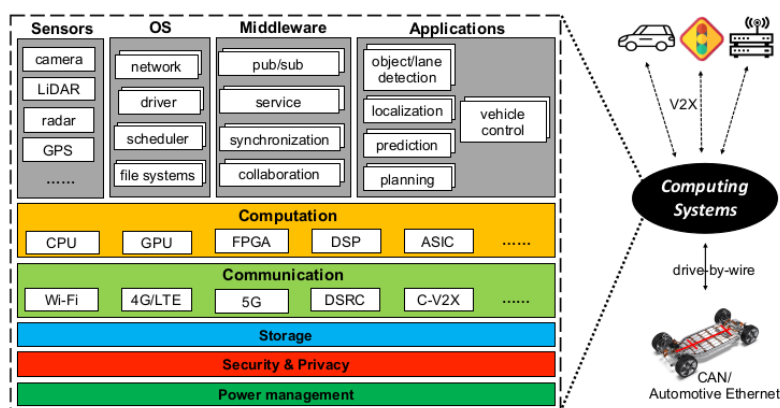


Figura 1.1: Ejemplo de arquitectura de un sistema de conducción autónoma.

La evolución continua de estos sistemas trata de ofrecer un mayor nivel de seguridad al volante con Advanced Driver Assistance System (ADAS), para que en un futuro puedan ser remplazados por Automated Driving System (ADS).

Para analizar el avance de estos sistemas y para poder compararlos, se ha dividido según su nivel de autonomía, por lo que se tiene desde un nivel 0 a un nivel 5. El nivel 0 indica que el coche no tiene ningún tipo de autonomía, en el nivel 1 el vehículo sigue siendo controlado por el conductor pero ciertas características de ayuda a la conducción son añadidas, en el siguiente nivel el vehículo es capaz de acelerar, frenar y hasta dirigir el vehículo pero con el conductor siempre atento, en el nivel 3 el conductor es necesario pero no es requerimiento la atención al entorno, pero debe de estar listo para tomar en control en todo momento, el nivel 4 permite un nivel de autonomía donde el vehículo no requiere de atención pero únicamente en ciertos escenarios y el último nivel es el que habilita la conducción autónoma completa [2].

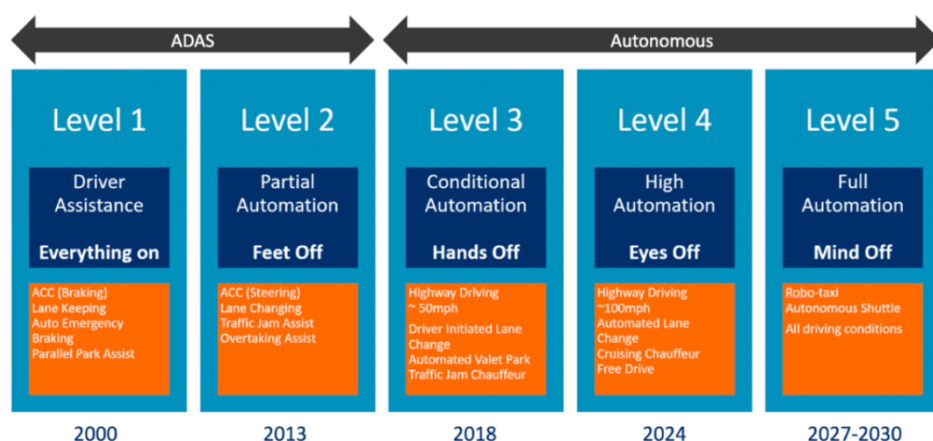


Figura 1.2: Niveles de autonomía.

El desarrollo de este tipo de sistemas no es una tarea sencilla, múltiples empresas involucradas en el desarrollo de vehículos autónomos pretendían tener vehículos en el nivel 4 de autonomía en poco tiempo, pero como se ha visto esto no es posible, actualmente nos encontramos en el mercado con sistemas que se encuentran entre el nivel 2 y el 3, por lo que aún queda un largo camino antes de llegar a conducción autónoma completa.

1.2 Sistemas de percepción

Los sistemas de ADS/ADAS requieren de un entendimiento del entorno para poder funcionar correctamente, para ello es necesario añadir diversos sensores a los largo del vehículo que nos permitan obtener la mayor información del exterior posible. A partir de este conocimiento es posible la toma de decisiones y la planificación, por lo que en este apartado se va a explicar de que manera se puede configurar un sistema de percepción, cuales son los principales sensores y que información se puede obtener de cada uno de ellos.

1.2.1 Principales sensores para la percepción en vehículos autónomos

Para la creación un sistema de percepción robusto es necesario el uso de diversos tipos de sensores que ofrezcan una información de relevancia de manera diferente al resto, por ello se utilizan sensores como cámaras, Radar, LiDAR, sensores de ultrasonidos, GPS, GNSS, IMU etc.

Estos ofrecen información de localización, velocidad, distancia de objetos en el entorno, e incluso información del propio vehículo, como su propia localización, la velocidad lineal y angular que este tiene. También es necesario tener en cuenta que no todos los sensores funcionan de la misma manera en distintos escenarios por lo que en situaciones donde un sensor es incapaz de obtener buenos datos otro sensor puede suplir esta carencia, por lo que la redundancia de sensores aporta otro nivel de seguridad al vehículo ya no solo un nivel mayor de detección del entorno.

1.2.1.1 Cámara

Uno de los sensores más utilizados es la cámara, este es el más extendido debido a la gran riqueza de información que ofrece del entorno. Actualmente se pueden encontrar cámaras que generen imágenes a una gran resolución y a una alta tasa de FPS por un precio bastante asequible, este es uno de los sensores más baratos.



Figura 1.3: Ejemplo de cámara utilizado en vehículos autónomos.

El problema de este sensor recae en el compute que es necesario para obtener información a partir de las imágenes, ya que estas no son más que píxeles en escala de grises o con un sistema de colores como el RGB. Por ello no solo es necesario tener en cuenta el coste del sensor, sino que también hay que aumentar la capacidad de compute del ordenador de a bordo para que pueda analizar en tiempo real las imágenes. Por último es necesario conocer las limitaciones de la cámara, esta funciona de forma correcta en situaciones de buena luminosidad y sin reflejos, por lo que en situaciones con lluvia 1.4, niebla, durante la noche y otros escenarios climatológicos adversos, no es capaz de obtener toda la información que esta obtendría en situaciones más favorables, lo cual hace que otros sensores sean usados en estas condiciones adversas para lidiar con estas limitaciones.



Figura 1.4: Uso de cámara con lluvia.

Aún conociendo las desventajas de estos sensores la gran mayoría de enfoques incluyen cámaras para la obtención de los objetos del entorno tanto en 2D como en 3D, pudiendo utilizar para lo segundo un sistema de cámaras estéreo que obtienen también información de la profundidad.

1.2.1.2 Radar

Los [Radar](#) son utilizados en múltiples aplicaciones como la previsión meteorológica, la astronomía, las comunicaciones, la navegación oceánica y la conducción autónoma entre otras.

Este sensor emite ondas de radio, las cuales son reflejadas devuelta a este, lo cual da una información de donde se hayan los objetos en el espacio tridimensional lo cual implica la distancia a estos junto con los dos ángulos necesarios, además gracias al efecto Doppler se puede inferir la velocidad de los objetos a partir de un fenómeno que hace variar la frecuencia de la onda enviada si hay algún tipo de movimiento local relativo respecto del propio [Radar](#) [3,4].



Figura 1.5: Ejemplo de Radar utilizado en vehículos autónomos.

Como se ha visto este sensor al contrario que la cámara obtiene directamente información utilizable para el entendimiento del entorno de forma directa, el problema radica en la escasa cantidad de datos que provee. Aunque se obtenga información de localización 3D y de velocidad, la cantidad de la nube de puntos producida es muy pequeña, por lo que es necesario de otros sensores para obtener una información completa del entorno.

Por otra parte, una de las principales ventajas del [Radar](#) es que se puede utilizar en cualquier situación meteorológica, unicamente podría verse afectado por lluvias muy intensas. Por lo que es un sensor muy completo y una gran adición para obtener información adicional de posición 3D y velocidad a un precio inferior a un [LiDAR](#) y por ello es adoptado por gran cantidad de sistemas [ADAS](#) en conjunto con sistemas de cámaras 360 alrededor del vehículo.

1.2.1.3 LiDAR

De forma similar al [Radar](#), los sistemas [LiDAR](#) basan su funcionamiento en el escaneo del entorno a partir de el envío de láseres y el cálculo del tiempo desde su envío hasta su retorno. Con esta información de distancia y el ángulo de inclinación del haz que envió esa señal, se construye una nube de puntos que consta de valores x, y, z de posición y otro valor que es el coeficiente de reflectividad del rayo de luz con el objeto incidido [3].

Actualmente los [LiDAR](#) más utilizados son de 64 canales, lo cual indica que se tienen 64 láseres funcionando al mismo tiempo lo que da una gran resolución del entorno, además la nube de puntos generada es de hasta 120 metros alrededor del vehículo lo cual permite detectar objetos a una distancia considerable

y saber de manera casi perfecta su distancia en un entorno tridimensional gracias a los alrededor de 2.000.000 de puntos que se generan por segundo del entorno [5].



Figura 1.6: Ejemplo de LiDAR utilizado en vehículos autónomos.

Los sistemas **LiDAR** tienen la ventaja de ser un sensor que aporta mucha información del entorno, pero tienen el mismo problema que las cámaras, en condiciones de lluvia, nieve, granizo o niebla, la efectividad de este sensor decae aunque se trata de minimizar ajustando la longitud de onda del láser utilizado [6].

Aún siendo un sensor muy útil, que puede aumentar el nivel de redundancia del sistema además de la seguridad, múltiples compañías como Tesla tratan de evitar su uso utilizando únicamente cámaras y **Radar**, esto es debido a que un **LiDAR** suele costar entre 8.000 y 100.000 dólares si se requiere de un resolución similar al estado del arte entre 16 y 128 haces [1].

1.2.2 Sistemas de detección

Unicamente con un sistema de sensores no es posible la comprensión del entorno, también es necesario de un procesamiento de los datos, mientras que la cámara no da ninguna información de forma directa, el **LiDAR** y el **Radar** son capaces de obtener la posición de obstáculos alrededor del vehículo, y además el **Radar** es capaz de inferir la velocidad de los objetos sin necesidad de un seguimiento.

Principalmente en los sistemas de detección para conducción autónoma se trata de obtener las posiciones de los diferentes objetos de interés del entorno. Estas detecciones pueden ser tanto en 2D como en 3D, pero el problema radica en como obtener un rectángulo u ortoedro que identifique donde se encuentran dicho objetos del entorno.

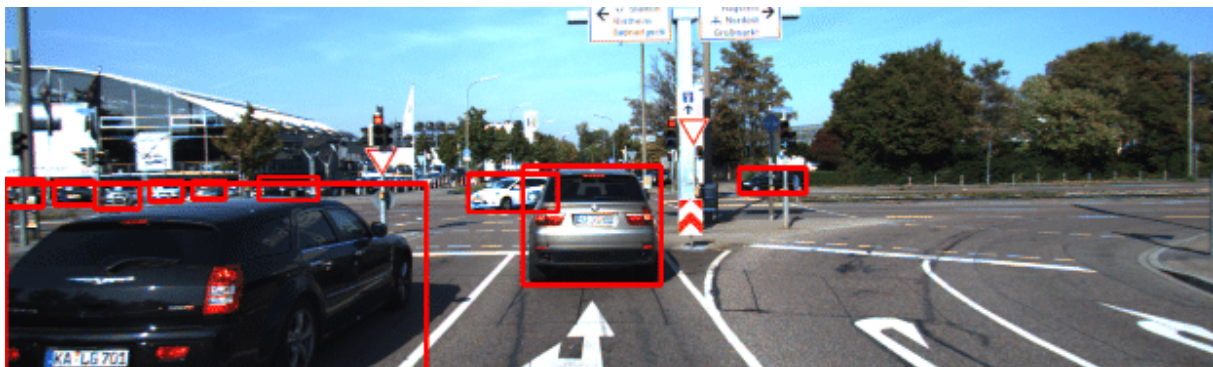


Figura 1.7: Ejemplo de detecciones 2D utilizando cámara.

Una de las formas de detección del entorno es a partir de cámaras, esto puede ser conseguido con un sistema de una cámara o de un sistema multicámara que abarque los 360 grados alrededor del coche, con esto instalado en el coche se puede generar rectángulos sobre las imágenes de los objetos del entorno como coches, peatones, bicicletas, motocicletas... Con esto se obtendría un listado de objetos detectados en 2D, lo cual es obtenible con un modelo de basado en redes neuronales como YOLO [7] que es capaz de hacerlo en tiempo real tal y como se ve en 1.7.

1.2.3 Sistemas de seguimiento

1.2.4 Fusión sensorial

1.3 Deep Learning

Capítulo 2

Propuesta de trabajo

La educación científica de los jóvenes es al menos tan importante, quizá incluso más, que la propia investigación.

Glenn Theodore Seaborg

Capítulo 3

Sistemas clásicos de percepción con LiDAR

El placer más noble es el júbilo de comprender.

Leonardo da Vinci

3.1 Voxelización

3.2 Ransac-3D

3.3 KD-tree

3.4 Filtrado posterior a la detección

Capítulo 4

Sistemas de percepción con LiDAR basados en Deep Learning

Si no conozco una cosa, la investigaré.

Louis Pasteur

4.1 Principales datasets

4.1.1 Kitti

4.1.2 Waymo

4.1.3 nuScenes

4.1.3.1 nuScenes-devkit

4.1.4 Comparativa entre los diferentes datasets

4.2 Estado del arte en detección utilizando LiDAR

4.2.1 PointPillars

4.2.2 SECOND

4.2.3 PointRCNN

4.2.4 PV-RCNN

4.2.5 CBGS

4.3 OpenPCDet

Capítulo 5

Desarrollo realizado

*La persistencia es muy importante. No debes renunciar
al menos que te veas obligado a renunciar.*

Elon Musk

5.1 Estado del proyecto T4AC

5.1.1 ROS

5.1.2 Docker

5.2 Implementación en CARLA

5.2.1 CARLA

5.2.2 Funcionamiento del LiDAR en CARLA

5.2.3 Tratado de datos

5.3 Fusión sensorial

5.4 Vehículo del proyecto T4AC

5.5 Implementación sobre el vehículo T4AC

Capítulo 6

ad-devkit

*La inspiración existe, pero tiene que encontrarte
trabajando.*

Picasso

6.1 Estado del arte en evaluación de vehículos autónomos

6.2 Obtención del ground truth

6.3 Evaluación de los modelos

Capítulo 7

Resultados obtenidos

Ninguna investigación humana puede ser llamada ciencia real si no puede demostrarse matemáticamente.

Leonardo da Vinci

7.1 Análisis cuantitativo en Kitti

7.2 Análisis cuantitativo en nuScenes

7.3 Análisis cualitativo del modelo clásico en CARLA

7.4 Análisis cualitativo de CBGS en CARLA

7.4.1 Comparativa con PointPillars en CARLA

7.5 Análisis cuantitativo de CBGS en CARLA

7.6 Análisis cualitativo de CBGS sobre el vehículo T4AC

7.6.1 Comparativa con PointPillars sobre el vehículo T4AC

Métrica	Resultado
mAP	0.4474
mATE	0.3379
mASE	0.2598
mAOE	0.3156
mAVE	0.2886
mAAE	0.2025
NDS	0.5832

Tabla 7.1: Rendimiento medio de CBGS PointPillars Multihead en nuScenes.

Tipo de objeto	AP	ATE	ASE	AOE	AVE	AAE
car	0.812	0.189	0.154	0.123	0.664	0.269
truck	0.500	0.354	0.189	0.093	0.415	0.277
bus	0.634	0.367	0.183	0.048	0.869	0.380
trailer	0.352	0.606	0.208	0.396	0.281	0.183
construction_vehicle	0.121	0.761	0.453	0.785	0.123	0.332
pedestrian	0.723	0.167	0.277	0.394	0.440	0.270
motorcycle	0.300	0.229	0.243	0.454	0.988	0.324
bicycle	0.064	0.189	0.273	0.506	0.494	0.093
traffic_cone	0.472	0.182	0.328	nan	nan	nan
barrier	0.499	0.341	0.288	0.071	nan	nan

Tabla 7.2: Análisis por clase de CBGS PointPillars Multihead en nuScenes.

Tipo de objeto	AP	IoU	AVE
Unknown	0.0	0.0	0.0
Unknown_Small	0.0	0.0	0.0
Unknown_Medium	0.0	0.0	0.0
Unknown_Big	0.0	0.0	0.0
Pedestrian	0.0	0.0	0.0
Bike	0.0	0.0	0.0
Car	0.731	0.491	0.715
Truck	0.0	0.0	0.0
Motorcycle	0.0	0.0	0.0
Other_Vehicle	0.0	0.0	0.0
Barrier	0.0	0.0	0.0
Sign	0.0	0.0	0.0

Tabla 7.3: Análisis por clase de CBGS PointPillars Multihead en CARLA.

Capítulo 8

Conclusiones

La verdadera felicidad radica en la finalización del trabajo utilizando tu propio cerebro y habilidades.

Soichiro Honda

8.1 Modelos estudiados

8.2 Comparativas adicionales

8.2.1 Ajuste de modelos basados en Kitty a nuScenes

8.2.2 Número de PCL de entrada en modelos evaluados sobre nuScenes

8.2.3 Tamaño del voxel en modelos basados en redes neuronales

8.3 Futuros trabajos

Bibliografía

- [1] L. Liu, S. Lu, R. Zhong, B. Wu, Y. Yao, Q. Zhang, and W. Shi, “Computing systems for autonomous driving: State of the art and challenges,” *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6469–6486, 2021.
- [2] “Automated vehicles for safety,” Tech. Rep., 2018. [Online]. Available: <https://www.nhtsa.gov/technology-innovation/automated-vehicles-safety>
- [3] “How self-driving cars work: Sensor systems,” Tech. Rep., 2021. [Online]. Available: <https://www.udacity.com/blog/2021/03/how-self-driving-cars-work-sensor-systems.html>
- [4] Y. Zhang, J. Ran, X. Chen, K. Fang, and H. Chen, “Observation of the inverse, zero and normal doppler effect in configurable transmission lines,” in *2015 IEEE 4th Asia-Pacific Conference on Antennas and Propagation (APCAP)*, 2015, pp. 229–230.
- [5] *HDL-64E, High Definition Real-Time 3D Lidar*, Velodyne Lidar. [Online]. Available: <https://velodynelidar.com/products/hdl-64e/>
- [6] A. M. Wallace, A. Halimi, and G. S. Buller, “Full waveform lidar for adverse weather conditions,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7064–7077, 2020.
- [7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.
- [8] “Información sobre gnu/linux en wikipedia,” <http://es.wikipedia.org/wiki/GNU/Linux> [Último acceso 1/noviembre/2013].
- [9] “Página de la aplicación emacs,” <http://savannah.gnu.org/projects/emacs/> [Último acceso 1/noviembre/2013].
- [10] “Página de la aplicación kdevelop,” <http://www.kdevelop.org> [Último acceso 1/noviembre/2013].
- [11] L. Lamport, *LaTeX: A Document Preparation System, 2nd edition*. Addison Wesley Professional, 1994.
- [12] “Página de la aplicación octave,” <http://www.octave.org> [Último acceso 1/noviembre/2013].
- [13] “Página de la aplicación cvs,” <http://savannah.nongnu.org/projects/cvs/> [Último acceso 1/noviembre/2013].
- [14] “Página de la aplicación gcc,” <http://savannah.gnu.org/projects/gcc/> [Último acceso 1/noviembre/2013].
- [15] “Página de la aplicación make,” <http://savannah.gnu.org/projects/make/> [Último acceso 1/noviembre/2013].

Apéndice A

Herramientas y recursos

Las herramientas necesarias para la elaboración del proyecto han sido:

- PC compatible
- Sistema operativo GNU/Linux [8]
- Entorno de desarrollo Emacs [9]
- Entorno de desarrollo KDevelop [10]
- Procesador de textos L^AT_EX[11]
- Lenguaje de procesamiento matemático Octave [12]
- Control de versiones CVS [13]
- Compilador C/C++ gcc [14]
- Gestor de compilaciones make [15]

Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR



Universidad
de Alcalá