

ASISTENCIA AL APARCAMIENTO BASADO EN UNA CÁMARA DE TIEMPO DE VUELO

¹Myriam Vaca Recalde, ¹Luis Paarup Peláez, ¹Enrique Martí Muñoz, ¹Ray Lattarulo y ²Xabier Udaeta

¹ División de Industria y Transporte, Tecnalia Research & Innovation, Derio, España,
{myriam.vaca, luis.paarup, enrique.marti, rayalejandro.lattarulo} @tecnalia.com

²Arisens-Ingenet, Orozko, España, xudaeta@ingenet.es

Resumen

En este artículo se propone un asistente de aparcamiento basado en la reconstrucción de objetos alrededor del vehículo, usando información obtenida de una cámara de Tiempo de Vuelo (ToF, del inglés Time-to-Flight). El método propuesto se centra en la detección de las matrículas de los vehículos de alrededor, para estimar su posición y determinar espacios libres de aparcamiento. Para ello, se fusiona una imagen 2D de intensidad de luz recibida, y una nube de puntos 3D creada a partir de la información de profundidad. Esta fusión permite relacionar cada píxel de la imagen con una posición 3D, por lo que se obtiene una referencia 3D del entorno facilitando al conductor la maniobra de aparcamiento que está realizando. El algoritmo propuesto mejora la detección de los vehículos resolviendo dos de los principales problemas de trabajar con cámaras: 1) obtención de sombras y reflejos en las imágenes debido al brillo (principalmente en estacionamientos al aire libre), lo que dificulta su proceso para detectar objetos; y 2) la detección limitada de objetos de baja reflexión, como los coches oscuros. El sistema es evaluado a través de una plataforma Renault Twizy en condiciones reales.

Palabras clave: Cámara de Tiempo de Vuelo (ToF), Sistemas Avanzados de Asistencia a la Conducción (ADAS), Sistemas Inteligentes de Transporte (ITS), Asistencia al aparcamiento.

1. INTRODUCCIÓN

El aumento de la seguridad en las carreteras es una prioridad para el gobierno, empresas y centros de investigación, por lo que se ha incrementado el interés en el desarrollo de Sistemas Avanzados de Asistencia a la Conducción (ADAS, del inglés Advanced Driver Assistance System) dentro de la investigación de los Sistemas de Transporte Inteligente (ITS, del inglés Intelligent Transportation System), para mejorar la eficiencia y comodidad de los conductores, con la reducción del impacto medio ambiental y la tasa de accidentes de tráfico. En estos avances, las aplicaciones de detección ex-

terna en automóviles son herramientas clave ya que permiten analizar el entorno, aligerando la carga del conductor al proporcionar imágenes de lo que ocurre a su alrededor, como en las maniobras de aparcamiento. Las estadísticas de accidentes muestran que se producen miles de choques en estacionamientos y estructuras de garajes anualmente [1], [2] debido a aspectos como el aumento de la población y la mala gestión del espacio de estacionamiento disponible [3]. Cabe señalar que, en los Estados Unidos, es obligatorio que los vehículos nuevos tengan una cámara de visión trasera como método de seguridad [4].

Actualmente, el objetivo principal de las cámaras de visión trasera en los vehículos es confirmar la seguridad de lo que ocurre detrás. Algunos utilizan un mecanismo óptico sofisticado para mostrar una imagen de gran angular [5], sin embargo, no es suficiente para maniobras en tiempo real y, a veces, carece de información relevante como la trayectoria del vehículo en marcha atrás o la distancia a los vehículos cercanos. Además, la apariencia de los objetos varía con el movimiento y depende de las condiciones de iluminación, de las superficies del material, entre otros aspectos. Por otro lado, las cámaras monoculares de gran angular no perciben profundidad, por lo que deben fusionarse con sensores ultrasónicos [6], que se utilizan actualmente en los parachoques (frontal y trasero) de los automóviles.

Este artículo propone una técnica novedosa de detección de vehículos y de espacio libre para sistemas de asistencia al aparcamiento, que se basa en la información obtenida de una cámara de Tiempo de Vuelo (ToF, del inglés Time-of-Flight). Para ello se fusionan los dos tipos de información proporcionados por este sensor, por un lado una imagen 2D de intensidad lumínica, y por otro una nube de puntos 3D a partir de la información de profundidad. Ambas técnicas se han utilizado de forma independiente en tareas de estacionamiento automatizado, como se describe en [7] en el caso de trabajar con información visual y en [8] para el segundo caso. El uso de la nube de puntos también es una característica novedosa, ya que los controladores obtienen información en tiempo real del en-

torno 3D con alertas de colisión, mientras que los sistemas estándares sólo proporcionan estos datos a través de una imagen como en [5].

La aportación de este trabajo reside en la unión de ambas técnicas para mejorar la precisión del sistema, aplicando algoritmos de procesamiento visual a la imagen 2D y, algoritmos geométricos a la nube de puntos 3D, con la limitación del corto alcance en exteriores de los sensores ToF. La propuesta recogida en este trabajo se centra en detectar elementos de alta reflectividad (matrículas) para extender el rango de detección efectivo del sensor. Además, debido a la conocida posición de las matrículas, se puede obtener una estimación completa de la posición del vehículo detectado.

El resto del artículo está distribuido de la siguiente forma: En la sección 2 se expone una descripción general del estado del arte. La sección 3 describe la metodología desarrollada de manera detallada. Los dispositivos experimentales, la plataforma, el software y los escenarios que se utilizaron para validar el enfoque se explican en la sección 4, junto con los resultados obtenidos. Finalmente, las conclusiones y trabajos futuros se enumeran en la sección 5.

2. ESTADO DEL ARTE

Los sistemas de asistencia al estacionamiento se han convertido en uno de los ADAS más demandados debido al desafío que implican las maniobras de aparcamiento, en términos de control y seguridad [3]. Su objetivo principal es asistir al conductor al estacionar, por lo que estos sistemas ayudan en la maniobra verificando que no haya objetos en la parte posterior del vehículo. Un ejemplo de ello es el sistema de asistencia de estacionamiento robusto basado en una cámara de ojo de pez, presentado en [5].

En [9] se describe un sistema que proporciona guía de voz e imagen durante las maniobras de estacionamiento. Un método basado en visión estereoscópica se utiliza en [10] para reconstruir la información 3D comparando imágenes. Sin embargo, los sistemas de visión pura no se pueden utilizar en condiciones de baja/nula iluminación (variaciones dramáticas de iluminación o efecto de sombras). Adicionalmente, la extracción de características en estos métodos de visión estereoscópica es complicada si el color del vehículo es oscuro, debido a la mala reflexión especular [11], [12]. Por otro lado, los sensores ultrasónicos se han utilizado para la detección de objetos o espacios de estacionamiento libres [13], por lo que es común verlos fusionados con cámaras para realizar maniobras de estacionamiento [6].

El RADAR también está presente en algunos sistemas de asistencia de estacionamiento. Más precisamente, se presenta un RADAR de corto alcance (SRR, del inglés Short Range Radar) en [14] para detectar estacionamientos al crear una cuadrícula de ocupación; y en [11] para la detección de esquinas rectangulares y redondas de los vehículos utilizando datos de rango, de tal manera que proporciona una posible posición de estacionamiento en una plaza libre y viable. Otros enfoques proponen el uso de LiDAR, como en [15], donde se segmenta el entorno en la superficie del suelo y otros objetos en función de la distancia euclídea, la orientación y la dimensionalidad del vector normal y, a continuación, clasifica los vehículos utilizando un clasificador de Bosques Aleatorios. En [8] también se propone el uso de LiDAR para la localización y seguimiento de un vehículo en un garaje combinando un enfoque basado en cuadrícula, algoritmo RANSAC y un filtro de Kalman.

Cabe destacar que las tecnologías anteriormente mencionadas tienen partes mecánicas difíciles de mantener (LiDAR) o una resolución angular o temporal limitada (RADAR) que complica la discriminación de objetos. Por lo tanto, se ha propuesto una cámara ToF que mejora el rendimiento y el costo. En [16], se presenta una descripción del sensor de profundidad de ToF, sus problemas y soluciones como el aumento de la resolución del sensor disminuyendo el ruido. Además, en [17] se describen los requisitos, características y parámetros necesarios para el diseño de una cámara ToF aplicada a asistencia de estacionamiento.

3. DESCRIPCIÓN DEL SISTEMA

El sistema de asistencia al aparcamiento que se ha desarrollado consta de dos bloques: detección de matrículas y asistencia visual al conductor. Primero, a la nube de puntos adquirida se le aplica un algoritmo de segmentación para separar objetos (algoritmos de agrupamiento basados en densidad). Éstos se clasifican en dos categorías que son matrícula y no-matrícula. Para ello, se ha utilizado un algoritmo de clasificación supervisado de tipo SVM (Support Vector Machine). En paralelo, la imagen en escala de grises obtenida de la cámara se filtra y procesa usando gradientes de intensidad y descriptores geométricos para detectar las matrículas. Ambos bloques se validan mediante un tercero que combina las salidas mejorando la calidad de detección y minimizar los falsos positivos. Una vez que se han detectado las matrículas, el sistema ejecuta una reconstrucción 3D del entorno usando los datos de la nube de puntos, proyectando la trayectoria estimada del vehículo y prediciendo posibles colisiones (ver Fig.

1). El posicionamiento del vehículo en este trabajo se obtiene de un algoritmo de SLAM basado en RTK-GNSS, IMU y un LiDAR de 16 capas, sin embargo, más adelante se realizaría con odometría de ruedas y giro de volante. Al margen de esto, el sistema desarrollado emplea únicamente la cámara ToF.

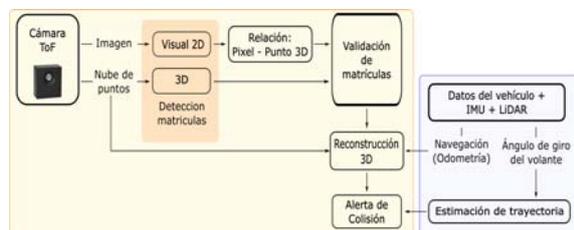


Figura 1: Diagrama general del sistema.

3.1. DETECCIÓN DE MATRÍCULAS

El rango de detección de la cámara depende mucho de las condiciones lumínicas. En esta aplicación se ha considerado el peor caso posible (aparcamiento exterior en día soleado) para demostrar la validez del algoritmo. Como ya se ha comentado, se dispone de tres sistemas de detección: clasificador por procesamiento de imagen, clasificación de nube de puntos y validación y combinación de ambos.

1. **Procesamiento de imagen:** La imagen pre-procesada es altamente ruidosa debido a la contaminación lumínica. El primer paso es filtrarla mediante un filtro Gaussiano 3x3 para obtener una imagen suavizada (ver Fig. 2). Posteriormente se 'binariza' la imagen aplicando un umbral para resaltar los píxeles más luminosos. Una vez binarizada, se aplica un detector de *blobs* para extraer conjuntos de píxeles cuya geometría podría corresponder a una matrícula. Los parámetros geométricos empleados son área mínima (50 píxeles), convexidad mínima (0.75) y máxima (1) y ratio de inercia mínimo (0.01) y máximo (0.1). El principal problema de este detector es el gran número de falsas detecciones en el suelo por su intensidad lumínica. Para evitar las detecciones en el suelo se filtran todos los píxeles correspondientes haciendo uso de la relación directa píxel-punto en la nube de puntos, donde sí es sencillo discernir y aislar el suelo (ver Fig. 3). Este filtrado mejora notablemente la precisión en la detección.
2. **Procesamiento de nube de puntos:** Para el procesamiento de los puntos en el espacio se ha desarrollado un algoritmo que consta de tres partes: filtrado, agrupamiento y clasificación (ver Fig. 3).

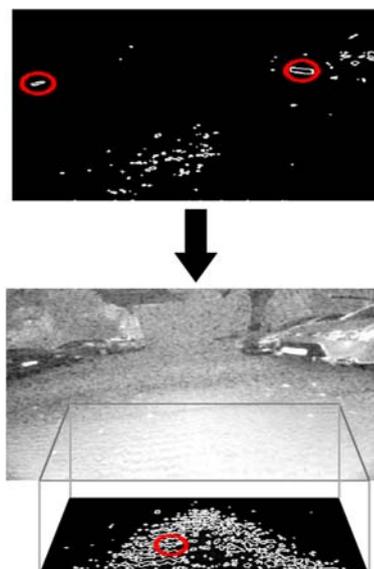


Figura 2: Detección de matrículas. Imagen de entrada de la cámara ToF (superior). Resultados de la detección de matrículas (derecha). Detección del suelo (inferior).

- a) **Filtrado:** En primer lugar, se aplica un filtro volumétrico de celdas 3D (voxel grid [18]) para reducir la información de la nube de puntos a pequeña escala sin perder información general. Posteriormente se aplica un filtro de aislamiento radial para eliminar aquellos puntos que no se encuentran en la vecindad (ver Fig. 3.(a)).
- b) **Agrupamiento:** Dada la gran densidad de puntos que se genera con el reflejo de las matrículas, se aplica un algoritmo de descubrimiento de grupos (DBSCAN) que agrupa puntos basándose en una medida de distancia euclídea y en un número mínimo de puntos (ver Fig. 3.(b)).
- c) **Clasificación:** Se emplea un SVM con un kernel gaussiano (RBF) para clasificar los grupos en matrículas y no-matrículas. Se ha empleado un SVM ya que captura bien las relaciones no lineales entre las características del grupo. También se ha optado por este modelo frente a otros de aprendizaje profundo (Deep Learning) debido a la relativa escasez de datos de entrenamiento y el reducido número de características, en este caso 11 (ver Tabla 1), que capturan bien la geometría rectangular y plana de las matrículas (ver Fig. 3.(c)). El principal inconveniente a la hora de clasificar un clúster espacial es que no es rotacionalmente invariante respecto al sistema de coordenadas de la cámara. Para que la clasificación sea lo más efectiva posible en ejes de cámara, todos los grupos se

rotan con respecto al eje vertical (eje Z) de tal manera que el eje X se corresponde con el autovector con menor autovalor (ver Fig. 4), gracias a la aplicación de Análisis de Componentes Principales (PCA, del inglés Principal Component Analysis). La covarianza del grupo se calcula a posteriori de tal manera que las matrículas tendrán muy baja covarianza XY pero muy alta varianza en YZ.

Cuadro 1: Características propuestas

No.	Descripción	Dimensiones
f1	Número de puntos en el grupo	1
f2	Distancia al centroide del grupo en el plano XY	1
f3	Distancia al centroide del grupo en el eje Z	1
f4	Matriz de covarianza 3D del grupo.	6
f5	Altura del grupo	1
f6	Ancho del grupo	1

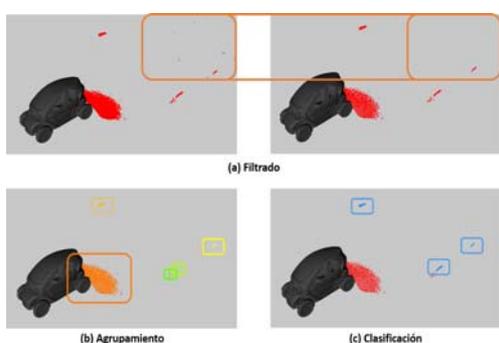


Figura 3: Procesamiento de nube de puntos: (a) Filtrado de puntos aislados, (b) Agrupación DBSCAN y (c) Clasificación SVM.

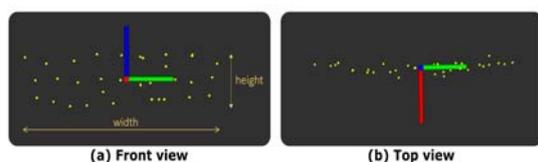


Figura 4: Vistas delantera (izquierda) y superior (derecha) de una nube de puntos de una matrícula después de la rotación (eje x: rojo, eje y: verde, eje z: azul).

Tras detectar las posiciones de los centroides de las matrículas, el siguiente paso es reconstruir el espacio ocupado por el coche en el aparcamiento. Esto se puede hacer conociendo la dirección normal a la matrícula, la cual se ha calculado previamente gracias al PCA. Ya que el tamaño del coche no es extrapolable a partir de la matrícula, se asume una anchura y longitud tal que corresponda a un coche de 7 plazas. La información obtenida de estas reconstrucciones es suficiente para saber si existen (de manera aproximada) espacios libres en el aparcamiento. La fig. 5 muestra

una reconstrucción de un aparcamiento donde los espacios disponibles se calculan en base a la distancia euclídea entre coches consecutivos. Al margen de la reconstrucción de coches (rojo) y espacios disponibles (verde), se puede observar una cartografía del aparcamiento creada en tiempo real mediante SLAM con LiDAR. Nótese que dicha cartografía sólo se ha empleado para localizar el coche y para comprobar las detecciones de manera visual, nunca para ayudar a clasificar ni para estimar las orientaciones de los rectángulos.

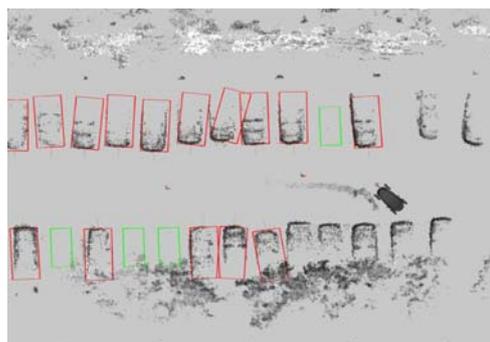


Figura 5: Reconstrucción del aparcamiento.

3. **Validación de matrículas:** Los píxeles asociados a matrículas en la imagen se pueden convertir a puntos 3D y verificar que efectivamente corresponden con una detección de matrícula procesando la nube de puntos. Aquellas matrículas detectadas por ambos sistemas obtienen una mayor confianza de detección que aquellas detectadas únicamente por un sistema.

3.2. Asistencia al aparcamiento

La asistencia al aparcamiento gira en torno a tres sistemas: reconstrucción 3D, estimación de trayectoria y predicción de colisiones. Estos sistemas se activan de manera automática cuando el conductor inicia la maniobra de aproximación a un espacio libre, donde dispone de una representación tridimensional de la plaza de aparcamiento, una proyección de su trayectoria en caso de mantener constante el ángulo de volante, y avisos en caso de que exista posibilidad de colisión.

1. **Reconstrucción 3D:** La reconstrucción hace uso de la localización del coche (obtenida de sistemas como GPS+IMU+SLAM) y de la nube de puntos obtenida de la cámara. A pesar de ser altamente dependiente de las condiciones lumínicas, la nube de puntos gana calidad a medida que se acerca a los obstáculos del entorno. Previamente a ser insertada

en la reconstrucción, la nube de puntos sufre un proceso de filtrado similar al que se describe en la sección 3.1. En primer lugar, se aplica un filtrado del suelo. Posteriormente se eliminan los puntos aislados mediante filtros de asilamiento radial. Por último, para evitar la inserción masiva de puntos en la reconstrucción, se aplican filtros volumétricos de celdas para reducir el tamaño de las nubes de puntos y evitar redundancias innecesarias. La Fig. 6 muestra la reconstrucción de una plaza de aparcamiento donde se puede apreciar las zonas detectadas y reconstruidas de los vehículos laterales en blanco.

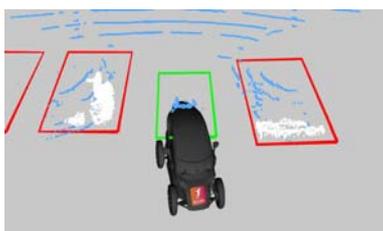


Figura 6: Reconstrucción de una plaza de aparcamiento.

2. **Estimación de la trayectoria:** La trayectoria del vehículo se proyecta combinando el ángulo de volante con el modelo cinemático de bicicleta. La simplificación es suficientemente válida en aparcamientos a baja velocidad, ya que se desprecian los efectos complejos como el derrape. El modelo de movimiento se combina con una representación poligonal del contorno del vehículo en su movimiento, ganando así conocimiento del área que cubriría el coche en caso de mantener el ángulo de volante. Esto permite predecir posibles colisiones con elementos detectados con la cámara.
3. **Predicción de colisiones:** Se emplea un algoritmo de proyección de líneas [19] para comprobar si existe algún punto de la reconstrucción que se encuentra dentro de la trayectoria estimada del coche. Según se puede observar en la Fig. 7, el algoritmo calcula el número de intersecciones entre la línea que une la cámara con cada punto y el contorno de la trayectoria. En función de la paridad del número de intersecciones se determina si el punto se encuentra dentro o fuera de la trayectoria.

4. RESULTADOS EXPERIMENTALES

En esta sección se describirá el hardware, el software y los escenarios utilizados para la realización

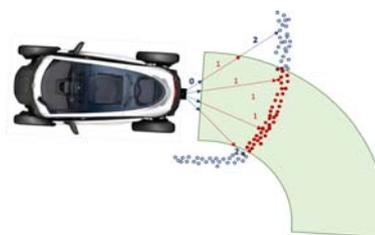


Figura 7: Predicción de colisiones.

de los experimentos, al igual que los resultados obtenidos al evaluar el sistema.

4.1. PLATAFORMA EXPERIMENTAL

La implementación se llevó a cabo en el Sistema Operativo Robótico (ROS) utilizando las librerías `pcl` de C++ y `sklearn` de python. Además, se evaluó con datos reales de una cámara ToF instalada en un Renault Twizy, como se muestra en la Fig. 8. La instrumentación y automatización de esta plataforma se encuentran detalladas en [20]. Los datos utilizados para el entrenamiento, prueba y experimentación en tiempo real se han recopilado durante algunas pruebas en el estacionamiento de las instalaciones de Tecnalía en Derio 700.



Figura 8: Plataforma experimental de Tecnalía: Renault Twizy con cámara ToF.

La cámara utilizada es una *pico monstar Real3TM* desarrollada por Infineon Technologies y PMD-Technologies. El sensor cuenta con un campo de visión de 100° horizontal y 85° vertical, una resolución de 352 × 287 píxeles y frecuencias de modulación de más de 100 MHz. La cámara no solo proporciona datos de nube de puntos 3D, sino también información de amplitud e intensidad para cada píxel en un rango de hasta 6m. Así pues, combinando información de profundidad con datos de imagen, un modelo de geometría del vehículo y su información de estado (posición, velocidad, volante), es posible estimar la distancia al punto de colisión y el Tiempo de Colisión (TTC, del inglés Time-To-Collision) con gran precisión.

4.2. SOFTWARE

ROS es un espacio de trabajo de código abierto basado en Linux, que maneja la comunicación entre programas ejecutables llamados nodos [21]. La comunicación entre nodos se logra mediante un patrón de publicación-suscripción de mensajes a topics. Por lo tanto, los nodos sólo se suscriben y publican en los topics deseados, desacoplando el sistema global en diferentes partes modulares. Los beneficios de este comportamiento desacoplado son la reutilización del código [22] y el funcionamiento global incluso con bloqueos de nodos parciales. Este enfoque se ha llevado a cabo con Ubuntu 16.04 LTS (64 bits), ROS Kinetic, 6th – GenIntel Corei7 y 32Gb de memoria RAM.

4.3. ESCENARIOS

El sistema se probó en el estacionamiento de las instalaciones de Tecnalia en Derio 700, como se ha mencionado anteriormente (ver Fig. 5). Se han realizado varias maniobras de estacionamiento con diferentes condiciones de luz, colores y tamaños de automóviles, ángulos de entrada y direcciones. Los datos de entrenamiento y prueba para el clasificador también se han registrado en estos escenarios de casos reales.

4.4. RESULTADOS

Se ha registrado un total de 22 escenarios para el desarrollo del sistema propuesto, con hasta 20 minutos de conducción en los escenarios presentados anteriormente. La tabla 2 muestra las dimensiones del conjunto de datos utilizados para el entrenamiento y prueba del clasificador SVM.

Cuadro 2: Dimensiones de la base de datos

Descripción	Nº de matrículas	Nº de no matrículas
Total de datos	567	704
Datos de entrenamiento	464	552
Datos de testeo	103	152

Cabe recalcar que el clasificador ha sido entrenado con un 80% de los datos totales usando un kernel RBF con $C = 10$ y $\gamma = 0.01$. El rendimiento del clasificador con los datos de prueba se muestra en la Tabla 3, donde se puede observar que de las 103 matrículas del conjunto de datos de prueba (P), el clasificador reconoce 101 correctamente (TP, del inglés True Positive) y 2 como no-matrícula (FN, del inglés False Negative). **Además, de las 152 casos sin matrícula (N), clasifica 135 detecciones de matrícula correctamente (FP, del inglés False Positive) y falla 17.**

Por otro lado, también se han calculado las estadísticas de procesamiento de imágenes con el

Cuadro 3: Resultados del clasificador durante las pruebas.

P	TP	FN	N	TN	FP
103	101	2	152	135	17
Accuracy = 0.9254			Score = 0.9254		

suelo filtrado. La tabla 4 muestra el número total de vehículos que aparecieron en la imagen de la cámara trasera durante las pruebas, así como el número de matrículas detectadas (TP), matrículas no detectadas (FN) y detecciones incorrectas de matrículas (FP).

Cuadro 4: Resultados del detector de imágenes con el suelo filtrado

Total cars	TP	FN	FP	Accuracy = 0.7778
72	56	16	6	

El clasificador tiene una tasa de acierto del 77.78%, lo cual es aceptable dada la calidad reducida de las imágenes. Por otro lado, resaltar que el número de falsos positivos mejoró de forma significativa una vez filtrado el suelo (se llegaba a obtener 137 falsos positivos de detecciones del suelo).

Por lo tanto, la fusión los datos de la imagen (con filtrado del suelo) y los de la nube de puntos ha demostrado ser eficiente mejorando de manera significativa la detección de los vehículos.

5. CONCLUSIONES

El método propuesto ha demostrado mejorar la fiabilidad en la detección de vehículos circundantes gracias a la doble verificación (procesamiento de imágenes y nubes de puntos), teniendo como principal ventaja la reducción de falsos positivos en la detección de matrículas. Además, al utilizar la distancia entre dos vehículos consecutivos, se detectan los estacionamientos libres al mismo tiempo.

La reconstrucción en 3D del escenario junto a la estimación de la trayectoria y las alertas de colisión brinda una mejor asistencia al conductor que los métodos convencionales, ya que aporta una visualización más real de lo que pasa en el entorno y alivia la carga del conductor en las maniobras de estacionamiento.

Cabe destacar que, al centrar el sistema en la detección de matrículas, que siempre son muy reflectantes, se garantizan resultados aceptables independientemente de las condiciones externas (por ejemplo luz excesiva, oscuridad, lluvia, etc.).

Los trabajos futuros incluyen agregar algo de

automatización a la maniobra de estacionamiento (o automatización completa) o utilizar flujo óptico y ser completamente independiente con la cámara ToF (que no se requieren más sensores).

Agradecimientos

Los autores desean agradecer al Proyecto ESKALA 4.0 (con el número de subvención 063689) por su apoyo en el desarrollo de este trabajo.

English summary

PARKING ASSISTANCE BASED ON TIME-OF-FLIGHT CAMERA

Abstract

This paper proposes a parking assistant based on the reconstruction of objects around the vehicle, using 2D and 3D data obtained from a Time-to-Flight camera (ToF). The proposed method is based on the fusion of the data obtained from the intensity image and the point cloud. It is mainly on the detection of the vehicle's plates to estimate their position and determine free parking lots. This fusion allows to link each pixel of the image with a 3D position, so it gets a 3D reference of the environment and a clear vision during the parking maneuver. In addition, this methodology improves the vehicles detection solving two of the relevant problems found in cameras: 1) obtaining shadows and reflections in the images considering brightness (mainly in outdoor parking lots), which hinders the process to detect objects; and 2) limited detection of low reflection objects, such as dark cars. The system is evaluated in a Renault Twizy platform in real conditions.

Keywords: Time of Flight Camera (ToF), Advanced Driver Assistance Systems (ADAS), Intelligent Transportation Systems (ITS), Parking Assistance.

Referencias

- [1] NSC, "Parking Lot Safety," 2018.
- [2] S. Smith, "500 Deaths Per Year Occur in Parking Lots and Garages — Safety NSC," 2016.
- [3] Faheem, S. Mahmud, G. Khan, M. Rahman, and H. Zafar, "A Survey of Intelligent Car Parking System," *Journal of Applied Research and Technology*, vol. 11, pp. 714–726, oct 2013.
- [4] U.S. Department of Transportation, "Rear Visibility," 2018.
- [5] G. Ma, M. Dwivedi, R. Li, and C. Sun, "Parking guidance system," jan 2011.
- [6] H. Jung, C. Choi, D. Kim, and P. Yoon, "System Configuration of Intelligent Parking Assistant System," *World Congress Intelligent Transportation Systems Services*, vol. 13, pp. 1–8, 2006.
- [7] C. Wang, H. Zhang, M. Yang, X. Wang, L. Ye, and C. Guo, "Automatic Parking Based on a Bird's Eye View Vision System," *Advances in Mechanical Engineering*, vol. 6, 2015.
- [8] A. Ibisch, S. Stumper, H. Altinger, M. Neuhausen, M. Tschentscher, M. Schlipfing, J. Salinen, and A. Knoll, "Towards Autonomous Driving in a Parking Garage: Vehicle Localization and Tracking Using Environment-Embedded LIDAR Sensors," pp. 829–834, jun 2013.
- [9] S. Hiramatsu, A. Hibi, Y. Tanaka, and T. Kakinami, "SAE TECHNICAL Rearview Camera Based Parking Assist System with Voice Guidance," 2002.
- [10] N. Kaempchen, U. Franke, and R. Ott, "Stereo Vision Based Pose Estimation of Parking Lots Using 3D Vehicle Models," vol. 2, pp. 459–464, 2002.
- [11] H. G. Jung, Y. H. Cho, P. J. Yoon, and J. Kim, "Scanning Laser Radar-Based Target Position Designation for Parking Aid System," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, pp. 406–424, sep 2008.
- [12] L. Tong, L. Cheng, M. Li, J. Wang, and P. Du, "Integration of LiDAR Data and Orthophoto for Automatic Extraction of Parking Lot Structure," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, pp. 503–514, feb 2014.

- [13] Y. Shao, P. Chen, and T. Cao, “A Grid Projection Method Based on Ultrasonic Sensor for Parking Space Detection,” pp. 3378–3381, jul 2018.
- [14] A. Loeffler, J. Ronczka, and T. Fechner, “Parking Lot Measurement with 24 GHz Short Range Automotive Radar,” pp. 137–142, jun 2015.
- [15] F. Bock, D. Eggert, and M. Sester, “On-street Parking Statistics Using LiDAR Mobile Mapping,” pp. 2812–2818, sep 2015.
- [16] S. B. Gokturk, H. Yalcin, and C. Bamji, “A Time-of-Flight Depth Sensor - System Description, Issues and Solutions,” no. January, pp. 35–35, 2004.
- [17] S. Acharya, C. Tracey, and A. Rafii, “System Design of Time-of-Flight Range Camera for Car Park Assist and Backup Application,” pp. 1–6, jun 2008.
- [18] C. Moreno and M. Li, “A Comparative Study of Filtering Methods for Point Clouds in Real-Time Video Streaming,” vol. 2225, pp. 389–393, 2016.
- [19] I. S. Kim, W. K. Lee, and Y. D. Hong, “Simple Global Path Planning Algorithm Using a Ray-Casting and Tracking Method,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 90, pp. 101–111, may 2018.
- [20] M. Marcano, J. A. Matute, R. Lattarulo, E. Martí, and J. Pérez, “Low Speed Longitudinal Control Algorithms for Automated Vehicles in Simulation and Real Platforms,” *Complexity*, vol. 2018, 2018.
- [21] R. Mishra and A. Javed, “ROS Based Service Robot Platform,” pp. 55–59, apr 2018.
- [22] S. Cousins, B. Gerkey, K. Conley, and W. Garage, “Sharing Software with ROS,” *IEEE Robotics and Automation Magazine*, vol. 17, pp. 12–14, jun 2010.



© 2019 by the authors.
Submitted for possible
open access publication
under the terms and conditions of the Creative Commons Attribution CC BY-NC-SA 4.0 license (<https://creativecommons.org/licenses/by-nc-sa/4.0/deed.es>).