



Javier Sánchez-Soriano

Doctor en Inteligencia Artificial, Profesor Contratado Doctor, Escuela Politécnica Superior, Universidad Francisco de Vitoria

Gonzalo de Las Heras de Matías

Grado Ingeniería Informática UFV, Máster Universitario Big Data UEM, Doctor Tecnologías Información Aplicadas UEM.

Enrique Puertas Sanz

Universidad Europea de Madrid. Ingeniería en Informática por la UEM, Doctor Tecnologías Información Aplicadas por la UEM.

Javier Fernández Andrés

Universidad Europea Madrid. Ingeniería técnica Industrial ICAI, Ingeniería Industrial UPM. Doctor Ingeniero Industrial UPM.

**SISTEMA AVANZADO DE AYUDA A LA
CONDUCCIÓN (ADAS) EN ROTONDAS
/GLORIETAS USANDO IMÁGENES AÉREAS
Y TÉCNICAS DE INTELIGENCIA
ARTIFICIAL PARA LA MEJORA DE LA
SEGURIDAD VIAL**

SISTEMA AVANZADO DE AYUDA A LA CONDUCCIÓN (ADAS) EN ROTONDAS/GLORIETAS USANDO IMÁGENES AÉREAS Y TÉCNICAS DE INTELIGENCIA ARTIFICIAL PARA LA MEJORA DE LA SEGURIDAD VIAL

Sumario: 1.- INTRODUCCIÓN. 1.1.- Motivación. 1.2.- Capacidad de las rotondas/glorietas. 1.3.- Detección de objetos. 1.4.- Detección de objetos en imágenes aéreas. 2.- METODOLOGÍA. 2.1.- Fase de instalación y calibración. 2.2.- Fase de ejecución. 3.- RECONOCIMIENTO DE LA ROTONDA. 3.1.- Preprocesado. 3.2.- Cálculo de circunferencias. 3.3.- Resultados. 4.- DETECCIÓN DE VEHÍCULOS. 4.1.- Dataset. 4.2.- Métricas. 4.3.- Experimentos. 4.4.- Resultados. 5.- EXTRACCIÓN DE INFORMACIÓN. 6.- CONCLUSIONES. 7.- REFERENCIAS.

Resumen: Las rotondas son un tipo de construcción vial en el que confluyen varios caminos que se comunican a través de un anillo mediante una circulación rotatoria. Estas han traído un aumento en la seguridad, sin embargo, su correcta circulación no es tarea fácil, tanto para vehículos convencionales como autónomos. Existen publicaciones sobre ADAS (Sistema Avanzado de Ayuda a la Conducción) que toman a las rotondas como un objeto por el que transitar, guiando a estos últimos en la circulación. Este trabajo toma la propia rotonda como fuente de información la cual podría transmitirse a estos vehículos para mejorar su toma de decisiones. Para ello, se detalla la creación de un prototipo para la monitorización de rotondas españolas mediante imágenes aéreas y aprendizaje automático. Este sistema requiere de una fase de instalación por la cual se calibra mediante técnicas de tratamiento de imágenes para reconocer las circunferencias de la isleta principal y los distintos carriles. Seguidamente, usando un modelo de RetinaNET basado en resnet50, se localiza cada vehículo. Con esta información se extrae información útil tanto para monitorización de la rotonda, como para los vehículos que transitan por ella (autónomos y convencionales). Todo ello, con el fin de mejorar la seguridad haciendo uso de técnicas de inteligencia artificial.

Abstract: Roundabouts are a type of traffic construction in which several roads converge and communicate through a rotating circulation around a central island. Although, they have increased safety, driving around them properly is not an easy task for conventional and autonomous vehicles. There are publications about ADAS that considers roundabouts as objects to be transited, guiding these autonomous ones in the circulation. This work present takes roundabouts as a source of information that could be transmitted to these vehicles to improve their decision making. To this end, it details the creation of a prototype for monitoring Spanish roundabouts using aerial imagery and machine learning. This system requires an installation phase in which it is calibrated using image processing techniques to recognize the circumferences of the central island and the different lanes. Then, using a RetinaNET model based on a resnet50 backbone, each vehicle and its type is detected. Combining both subsystems, the prototype extract information about the position of vehicles (exact location and lane), entering/exiting ones

and those that affect the entering (key aspect for capacity calculation). The aim is to improve security by using artificial intelligence techniques.

Palabras clave: Rotondas/Glorietas, Aprendizaje Automático, imágenes aéreas, Detección de Objetos, Aprendizaje profundo, Visión por Computador.

Keywords: Roundabouts, Machine Learning, Aerial Imagery, Object Detection, Deep Learning, Computer Vision.

1.- INTRODUCCIÓN

1.1.- Motivación

La mortalidad en carretera es un problema de la sociedad actual. Es la primera causa de muerte entre niños y jóvenes entre 5 a 29 años y cada año causan 1.3 millones de muertes en el mundo (World Health Organization, 2021). Mirándolo desde un punto de vista económico, a cada país le supone a un 3% de su PIB (Producto Interior Bruto) (World Health Organization, 2021). La Organización Mundial de la Salud (OMS) en su informe (World Health Organization, s.f.) identificó diversas áreas de actuación para mejorarla: carreteras más seguras, más concienciación/formación y vehículos más seguros, entre otros. Dentro de esta última, el proyecto GIDS (Generic Intelligent Driver Support) (Michon, 1993) sentó las bases de los denominados ADAS (Advanced Driver Assistance Systems), sistemas avanzados que asisten al conductor para mejorar su seguridad y comodidad en la conducción (aspectos muy relacionados (Dirección General de Tráfico - Ministerio del Interior., 2018)). Los ADAS son una evolución de los sistemas seguridad clásicos como ABS (Antilock Braking System), ESP (Electronic Stability Program), airbag o el control de crucero y han sido claves en la creación de los vehículos autónomos (BCG, 2015). Estos ADAS se pueden subdividir dependiendo del tipo de sensores que emplean (Kukkala et al., 2018): cámaras, LiDAR, radares, ultrasonidos, etc.

Los vehículos autónomos son aquellos que, idealmente, conducen como un humano sin requerir de la intervención de ninguna persona. Funcionan simplemente con ayuda de distintos ADAS para percibir el entorno y tomar decisiones. Existen 6 niveles de autonomía que van desde dependencia total del conductor (nivel 0), hasta autonomía total (nivel 5) (BCG, 2015). Hoy en día, hay diversas empresas como Telsa, Waymo o Baidu produciendo vehículos autónomos (Badue et al., 2019). No obstante, se siguen sucediendo noticias de accidentes (BBC, 2021; The New York Times, 2021) (que incluso plantean interrogantes legales sobre quien es el responsable (BBC, 2020)) y de errores descubiertos que requieren de revisiones (Popular Science, 2021). El nivel de autonomía más alto alcanzado es el 4. La conducción autónoma no es una tarea sencilla, existen demasiados condicionantes y situaciones que el vehículo debe percibir (velocidades, estado de la carretera, otros vehículos) y manejar (Naranjo et al., 2007). Algunas de estas ya están resueltas con algunos ADAS, como el control de crucero, la asistencia a cambio/mantenimiento de carril o aparcado automático (Kukkala et al., 2018).

Una de las situaciones complicadas de manejar para los vehículos autónomos son las rotondas o glorietas (Rodrigues et al., 2017). Según la DGT, “las rotondas son un tipo especial de intersección caracterizado por que los tramos que en ella confluyen se

comunican a través de un anillo en el que se establece una circulación rotatoria alrededor de una isleta central” (Rodríguez, 2014). En cierto modo, se pueden considerar como varios T-junctions seguidos (Brilon W. , 1991). La normativa española indica los siguientes puntos (Rodríguez, 2014; Dirección General de Tráfico, 2021):

1. Los vehículos dentro de la glorieta tienen prioridad sobre aquellos que desean incorporarse. No es como en una intersección en el que la prioridad la tienen los vehículos situados a la derecha.
2. La circulación se mueve siempre en el sentido contrario a las agujas del reloj.
3. El conductor puede elegir el carril que más se ajuste a su trayectoria. No obstante, siempre se debe salir de la glorieta por el carril exterior. En caso de estar ocupado el carril exterior, se deberá realizar otra vuelta.
4. Los indicadores de dirección se deben utilizar para cambiar de carril dentro de la rotonda y para salir de ella. En este último caso, no deberá ser antes de haber sobrepasado la anterior salida (ilustrado en el vehículo verde en la Figura 1).

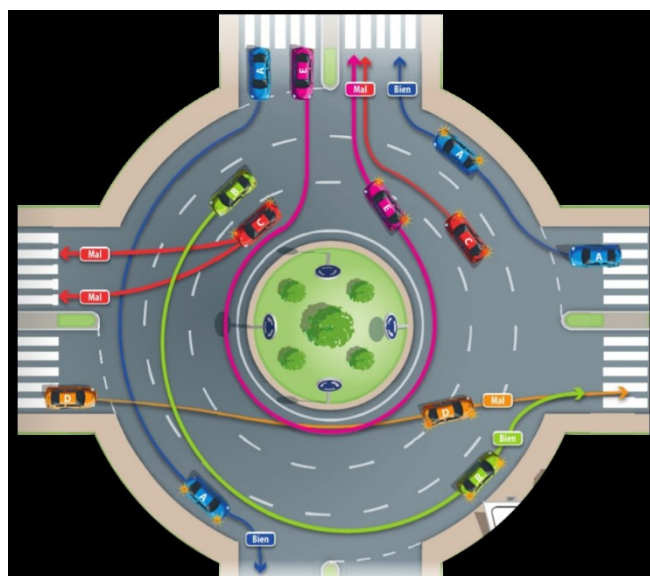


Figura 1. Representación de trayectorias correctas (vehículos azul y verde) y erróneas (naranja, rosa y rojo) (Rodríguez, 2014).

Existen diversas tipologías de rotondas (García Cuenca et al., 2019) las cuales se pueden clasificar dependiendo el aspecto a destacar (Gallardo, 2005):

- **Modo de funcionamiento:** Según el funcionamiento existen: rotondas que funcionan mediante incorporaciones por trenzado (Figura 2-A), rotadas con prioridad a la derecha, es decir, los entrantes tienen prioridad sobre los que se encuentran ya dentro, rotondas con prioridad en el anillo (Figura 2-B) y rotondas y rotondas con semáforos en ella (Figura 2-C).
- **Geometría:** Las rotondas pueden adoptar distintas formas geométricas. Se pueden encontrar formas circulares, ovaladas, dobles rotondas (Figura 2-D) o incluso partidas en las que dos carriles cruzan la isleta principal (Figura 2-E).
- **Según diámetros:** Grandes rotondas (con un diámetro interior de más de 20 metros), compactas (entre 4 y 20 metros) y mini-rotondas (menores de 20 metros). Estas últimas se encuentran normalmente en zonas en las que no se dispone de demasiado espacio.
- **Ubicación:** Existen rotondas urbanas, en las que las vías que desembocan en ellas no tienen una velocidad mayor de 50km/h al contrario de las semiurbanas, en las que estas

vías pueden alcanzar grandes velocidades. También se encuentran rotondas fuera de poblado que regulan vías interurbanas y rotondas cuyas vías la cruzan a varios niveles (Figura 2-F).

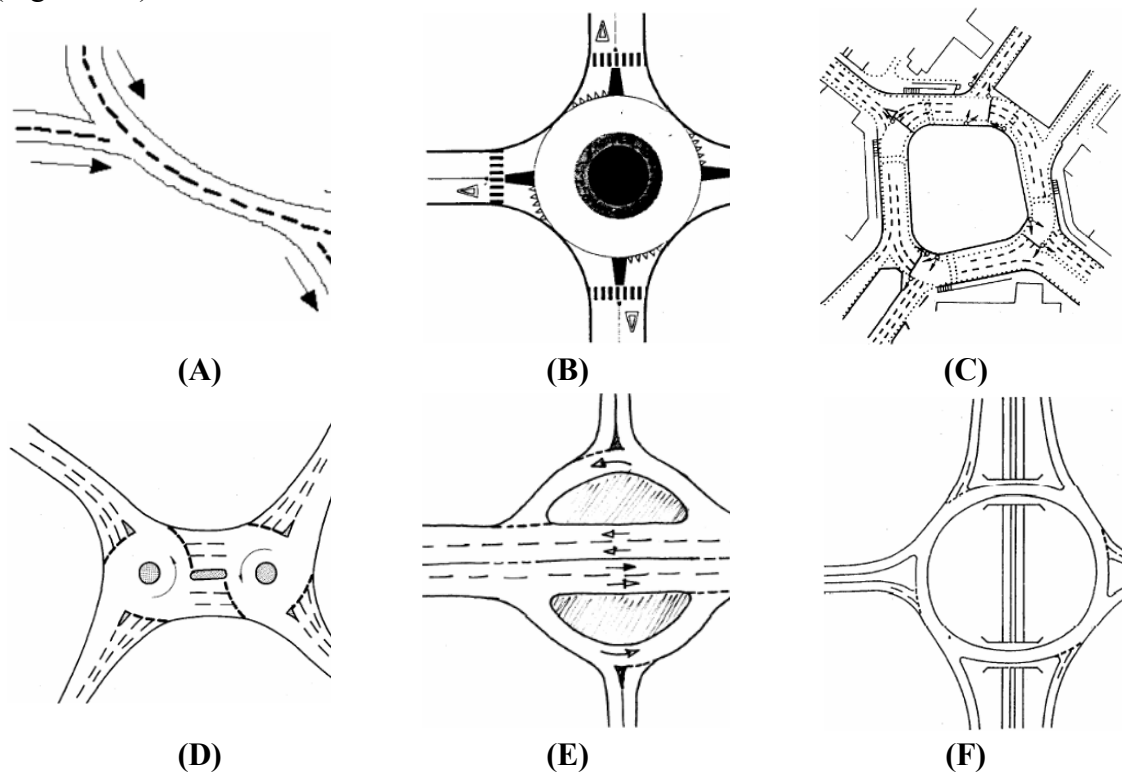


Figura 2. Ejemplos de diferentes topologías de rotondas: (A) por trenzado, (B) con prioridad en el anillo, (C) con semáforos, (D) rotonda doble, (E) rotonda partida, (F) rotonda con distintos niveles (Gallardo, 2005).

Con relación a los vehículos autónomos, actualmente las rotondas son objeto de estudio en diversas publicaciones. Estas principalmente, las consideran un escenario por los que los coches autónomos deben aprender a maniobrar (García Cuenca et al., 2019; García Cuenca et al., 2019). El objetivo de esta publicación es crear un prototipo de un sistema capaz de extraer información de la rotonda (acotado a las rotondas circulares con prioridad en anillo ya que son las más estándar (Gallardo, 2005)) mediante imágenes aéreas y técnicas de aprendizaje automático, no solo para proporcionar información a los vehículos autónomos, sino también a los conductores de vehículos convencionales. La información considerada de relevancia es la localización de los vehículos en la rotonda (posición y carril) con la cual se pueda extraer información útil relativa al estado de la rotonda, que otros sistemas las entreguen a los vehículos para que les faciliten su tránsito.

1.2.- Capacidad de las rotondas/glorietas

Uno de los aspectos clave a la hora de evaluar el desempeño de rotondas es calcular la capacidad de la rotonda (Bernhard & Portmann, 2000). Esta puede considerarse como una función dependiente de la geometría, de la capacidad de entrada/salida de la rotonda y de las características de conductores y vehículos (Gallardo, 2005). La capacidad de entrada se define como el máximo flujo entrante a la rotonda desde una entrada, por vehículos esperando a incorporarse (Gibson & Waterson, 2013). Esta depende de varios factores como son la geometría de las rotondas (Mauro & Guerrieri, 2012; Tollazzi et al., 2011; Silva et al., 2014; Yap et al., 2015), los diferentes factores ambientales que

comprenden tanto las condiciones o niveles de luz e iluminación así como la meteorología) (Yap et al., 2013; Montgomery et al., 2010), la demanda de orígenes y destinos de los conductores (Yap et al., 2013; Akçelik, 2004; Hagring, 2000), los peatones en las inmediaciones e interior de la rotonda así como las posibles obstrucciones dentro de la rotonda (Yap et al., 2013; Marlow & Maycock, 1982). Debido a las complicaciones de la recogida de datos se han desarrollado múltiples métodos de estimación que se clasifican en: métodos empíricos basados en la geometría y mediciones de la capacidad, “*gap acceptance models*” que tienen en cuenta el comportamiento de los conductores y “*microscopic simulation models*” que modelan kinemáticas de los vehículos e interacciones (Bernhard & Portmann, 2000; Yap et al., 2013; Macioszek, 2020). No obstante, a pesar de la similitud de los diseños de rotondas en todo el mundo, puede haber grandes diferencias a la hora de aplicar los distintos modelos para el cálculo de la capacidad (Yap et al., 2015).

Los modelos empíricos se basan en el ajuste de parámetros que relacionan la geometría de la rotonda y mediciones de la capacidad. Estos modelos se crean mediante regresiones multivariantes mediante la capacidad de entrada (Q_e), flujo de tráfico (Q_c) y otras variables que afectan a la capacidad. La relación entre Q_e y Q_c se presume que es lineal (1) o exponencial (2) (Yap et al., 2013).

$$Q_e = \alpha - \beta Q_c \quad (1)$$

$$Q_e = \alpha e^{-\beta Q_c} \quad (2)$$

Algunos ejemplos son LR942 (Kimber, 1980) o French Girabase (Guichet, 1997). También el aprendizaje automático resulta de utilidad en este campo. Publicaciones como (Anagnostopoulos et al., 2021) relatan el cálculo de capacidad mediante redes neuronales, y que además pueden servir para descubrir qué variables realmente afectan a la capacidad.

Los “*Gap Acceptance*” se basan en modelos teóricos que se emplean parámetros obtenidos de mediciones entre los vehículos que circulan por la rotonda y aquellos que se van a incorporar (Yap et al., 2013). La capacidad de entrada es calculada mediante 3 parámetros: el “*critical gap*” (t_c), “*follow-on headway*” (t_f) y la distribución de los huecos en el tráfico de la rotonda. Algunos ejemplos son German Brilon-Wu (Brilon et al., 1997) o SR45/SIDRA (Troutbeck, 1989).

Las “*microscopic simulations*” modelan los movimientos e interacciones de los vehículos. La gran ventaja de estos es su capacidad de simular distintos escenarios para su correcto estudio (Yap et al., 2013). Se pueden encontrar tanto modelos comerciales entre los que se encuentran por ejemplo S-Paramics (Microsimulation, 2011), Aimsun (Systems, 2011) o SUMO (Behrisch et al., 2011), así como otros modelos de corte más teórico y dedicados al estudio de las diferentes situaciones (Chin, 1985; Chung et al., 1992).

1.3.- Detección de objetos

La percepción del entorno es muy importante para los ADAS. Existen distintos sensores capaces de proveer información, pero en cuanto a reconocimiento de objetos, las cámaras o el LiDAR son los más importantes y actualmente existe un debate sobre cual es mejor solución (O'Kane, 2018). Antes de la irrupción de las redes neurales y del *Deep learning*, estos reconocedores de objetos operaban bajo pocos recursos computacionales y se basaban en características hechos a mano (Zou et al., 2019). Algunos ejemplos de estos son el reconocedor de caras Viola Jones (Viola & Jones, 2004; Viola & Jones, 2001), el reconocedor de peatones HOG (*Histogram of Oriented Gradients*) (Zou et al., 2019; Dalal & Triggs, 2005), o el modelo DPM (*Deformable Part-based Model*) (Zou et al., 2019; Felzenszwalb et al., 2008).

El aprendizaje automático es una rama de la inteligencia artificial que se distingue por la capacidad del sistema de entrenarse a sí mismo. Esta se subdivide a su vez en dos grandes grupos: aprendizaje supervisado y no supervisado. La diferencia entre estos radica en la forma que tiene el sistema de aprender. Mientras que el aprendizaje supervisado requiere de una serie de ejemplos previamente etiquetados el no supervisado no depende de ellos.

Dentro del grupo del aprendizaje supervisado destacan las redes neuronales, entre las que se encuentran las redes neuronales convolucionales, que alcanzaron una gran popularidad en 2012 (Krizhevsky et al., 2017). Su principal cualidad es que son un tipo de red muy concreto que pueden extraer características presentes en una imagen, lo cual las convierte en un algoritmo muy adecuado para la clasificación de imágenes (Krizhevsky et al., 2017; Girshick, 2015) y la detección de objetos (Girshick, 2015; Girshick et al., 2013; Sermanet et al., 2013). Estas últimas se subdividen a su vez en redes de una o de dos etapas. Ejemplos de redes de dos etapas son R-CNN (Girshick et al., 2013), Fast R-CNN (Girshick, 2015) o Faster R-CNN (Zou et al., 2019; Ren et al., 2017). En el caso de una etapa están las redes tipo YOLO (You Only Look Once) en sus diferentes versiones v1 (Redmon et al., 2016), v2/9000 (Redmon & Farhadi, 2017), v3 (Redmon & Farhadi, 2018), v4 (Bochkovskiy et al., 2020) y también existen otras alternativas como son SSD (Single Shot Detector) (Liu et al., 2016) o RetinaNet (Lin et al., 2020). Cada tipo tiene una serie de ventajas, por ejemplo, las de un paso son más rápidas, pero carecen de tanta precisión (Soviany & Ionescu, 2018).

Para este trabajo se ha tomado la iniciativa de RetinaNet ya que tiene un buen balance entre precisión y velocidad y es competente comparativamente con otros modelos (Tan et al., 2021), además de haberse utilizado con éxito en (De-Las-Heras et al., 2021; Stuparu et al., 2020).

1.4.- Detección de objetos en imágenes aéreas

La detección de objetos en imágenes aéreas ha demostrado su utilidad en múltiples aplicaciones como mapeado 3D (Elkhrachy, 2021), la agricultura (Dijkstra et al., 2019) o la evaluación de desastres (Gupta et al., 2021). Esta tecnología también ha resultado ser eficaz en tareas relacionadas con los vehículos y sus infraestructuras. Existen numerosos trabajos acerca del reconocimiento de vehículos en imágenes aéreas (Stuparu et al., 2020; Shen et al., 2021; Liu & Mattyus, 2015; Zhong et al., 2017; Tang et al., 2020; Deng et al., 2017; Yu et al., 2019; Chen et al., 2015; Kembhavi et al., 2010) o de infraestructuras

(Cheng et al., 2017) e incluso el reconocimiento de peatones en este tipo de imágenes, siempre más complejo debido al reducido tamaño de estos en comparación con el de los vehículos (Chang et al., 2018; Soleimani & Nasrabadi, 2018).

Como se ha visto en la sección anterior, los modelos de *Deep learning* empleados en el reconocimiento de objetos requieren para su entrenamiento de una serie de ejemplos previamente etiquetados. Actualmente existen multitud de datasets de imágenes aéreas (Krajewski et al., 2018; Bock et al., 2020; Robicquet et al., 2016; Du et al., 2018; Bock et al., s.f.; Krajewski et al., s.f.), incluso algunos más específicos y relativos a rotondas (Puertas et al., 2022) así como de trayectorias de vehículos en ellas (Krajewski et al., 2020; Breuer et al., 2020). Estos resultan de mucha utilidad, pero normalmente no son de fácil y rápido acceso debido a las aplicaciones comerciales que tienen.

Para la captación de imágenes aéreas, además del uso de satélites o aeronaves con un alto coste de operación han proliferado los UAS (“Sistema de aeronaves no tripuladas”) o drones. Se trata de una tecnología muy prometedora para acceder a lugares de difícil acceso y enviar imágenes de alta resolución en tiempo real con un coste asequible. Estos se complementan muy bien con la detección de objetos comentada en los párrafos previos, ya que estos drones envían las instantáneas al sistema de procesamiento y este extrae información. Esta sería una arquitectura típica, ya que el procesamiento de imágenes requiere de una alta capacidad de procesamiento, el cual se realiza mediante GPUs (Graphics Processing Unit).

2.- METODOLOGÍA

En el desarrollo del presente trabajo de investigación se ha seguido la metodología de trabajo que puede observarse en la Figura 3. Se trata de un proceso que tiene dos fases: (1) Instalación y Calibración y (2) Ejecución. Dichas fases y sus pasos se describen en los siguientes subapartados.

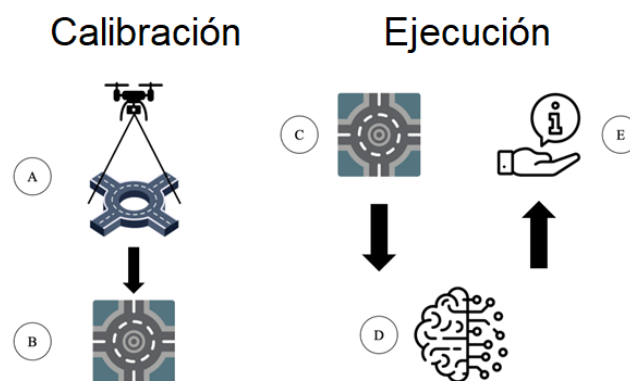


Figura 3. Funcionamiento del prototipo con sus dos fases: Calibración y Ejecución.

2.1.- Fase de instalación y calibración

Es necesaria una fase de instalación previa a la captación y procesamiento de las imágenes aéreas. Para ello, se sitúa el dispositivo con cámara en el centro de la rotonda a una altura conocida (Figura 3-A), de modo que la rotonda esté lo menos deformada posible. En este caso y con el fin de validar el presente trabajo se utilizó un UAS o dron. En concreto se

utilizó una aeronave civil modelo Mavic Mini 2 del fabricante DJI (DJI, s.f.) equipado con una cámara cuyas características se indican en la Tabla 1. La autonomía de vuelo del aparato es de 30 minutos con la batería a máxima carga y en condiciones meteorológicas favorables (sin viento y con temperatura ambiental media o baja).

Aunque una vez el sistema se encuentre en producción, la opción más simple sería utilizar un poste en el centro de la rotonda con un sistema de cámaras 360°, de forma que se genere la vista cenital de la rotonda mediante la composición de imágenes. Muchas rotondas ya disponen de estos postes para su alumbrado, por lo que se podría reutilizar para la instalación conjunta del sistema de cámaras.

Componente	Especificación
Resolución	1920x1080 px
Ángulo FOV	83°
Focal (35 mm)	24 mm
Apertura	f/2.8
Relación de aspecto	19:9
Sensor	1/2.3" CMOS

Tabla 1. Especificaciones de la cámara del UAS

El siguiente paso, independientemente del sistema de captación utilizado (UAS o cámara 360° instalada en un poste), sería el calibrado de las imágenes aéreas mediante un proceso semiautomatizado con el fin de reconocer y delimitar los carriles (Figura 3-BC). Este proceso se detalla más adelante en el apartado 3

2.2.- Fase de ejecución

Una vez preparado el sistema tras la fase de instalación y calibración, y mediante el uso de una red neural convolucional previamente entrenada y puesta a punto (Figura 3-D), se reconoce cada vehículo y se extrae información de la rotonda relativa a su posicionamiento en la misma (Figura 3-E). El hardware empleado en el prototipo para el procesamiento de las imágenes aéreas se indica en la Tabla 2.

Componente	Especificación
Procesador	Intel i7 9800K 3.6 GHz
Placa base	MPG Z390 Gaming Pro Carbon
Memoria	32 GBs
Gráficos	Nvidia RTX 2080 Ti
Almacenamiento	500 Tb SSD M2
Sistema Operativo	Ubuntu 18.04.4 LTS

Tabla 2. Especificaciones del prototipo usado para las pruebas

3.- RECONOCIMIENTO DE LA ROTONDA

Del pipeline de procesamiento, el primer paso calcular el centro y radio de las circunferencias que conforman la isleta central y cada uno de los carriles, de manera que

se pueda focalizar correctamente el reconocimiento de los vehículos y su posición dentro de las distintas secciones de la rotonda. Esto es de importancia debido a la existencia de distintos tipos de rotondas (García Cuenca et al., 2019) y de las distintas alturas sobre las que se podría colocar la cámara, el sistema requiere de una calibración previa.

3.1.- Preprocesado

Antes reconocer las circunferencias que conforman la rotonda (la isleta central y cada uno de los carriles) es necesario realizar un procesamiento previo. Este consiste en aplicar una serie de transformaciones a la imagen que faciliten el reconocimiento y que la ajuste a lo que los algoritmos requieren. Los pasos son los siguientes:

Región de interés. Dado que se puede elegir la posición de la cámara, se asume que la isleta central se va a encontrar dentro de unas coordenadas determinadas. Esto permite borrar ciertas partes de la imagen en las cuales se sabe que no va a estar y que por tanto solo van a aportar ruido en el reconocimiento. Esto es una técnica utilizada en *Deep learning* (Kim et al., 2018). La región de interés (RoI) establecida sobre la imagen original (Figura 4) es un recuadro en las coordenadas (300, 100) y (1620, 980) tal y como muestra la Figura 5.



Figura 4. Imagen capturada del dron sin procesar.

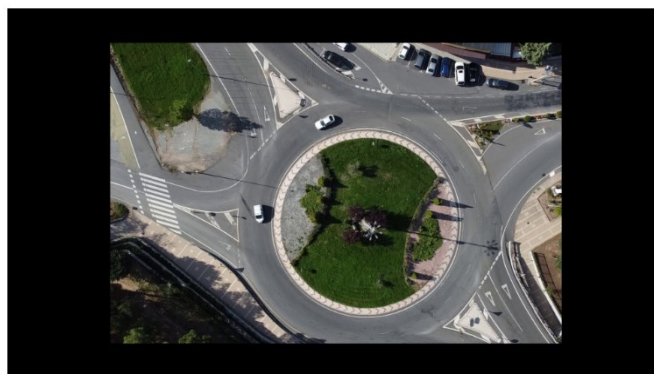


Figura 5. RoI aplicado a la imagen.

Eliminado de ruido y suavizado. Dentro de la imagen se encuentran muchas siluetas que aportan muchos trazados dentro de la imagen. Mediante la operación morfología de dilatado (OpenCV, s.f.; Gonzalez et al., 2009; Sreedhar & Panlal, 2012) y un filtro gaussiano (OpenCV, s.f.; Gonzalez et al., 2009) (ambos con un kernel de (9, 9)) se cierran

aquellos que presenten alguna pequeña discontinuidad y se suaviza la imagen. Seguidamente se calcula la resta entre una imagen blanca diferencia absoluta (OpenCV, s.f.) entre la imagen original con la RoI y la obtenida tras el filtro gaussiano anterior, para conseguir una imagen de fondo blanco y líneas grises (Figura 6). Finalmente, se aplica una función umbral cuyos parámetros experimentalmente calculados son $thresh=230$, $maxval=0$ y del tipo THRESH_TRUNC (1) (OpenCV, s.f.).

$$dst(x,y) = \begin{cases} threshold, & \text{if } src(x,y) > thresh \\ src(x,y), & \text{otherwise} \end{cases} \quad (3)$$

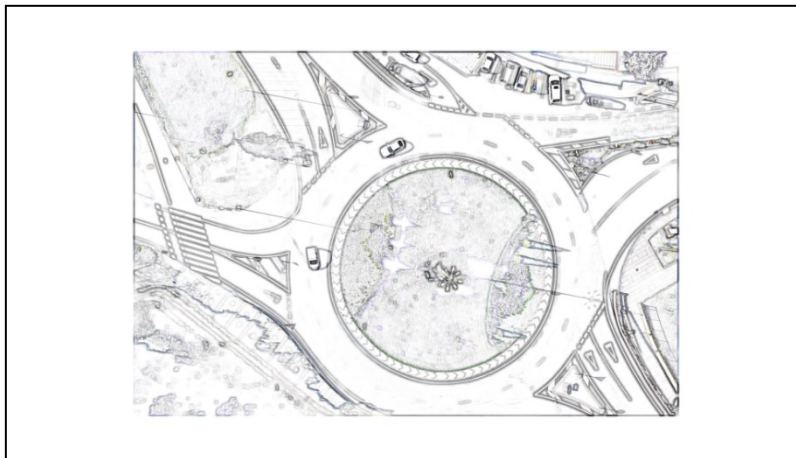


Figura 6. Imagen preprocesada.

Cálculo de bordes. El último paso es obtener una imagen que coste solamente de los bordes de los objetos en la imagen (Figura 7), puesto que es el verdadero requerimiento para la detección de las circunferencias. Esto se logra mediante el algoritmo de Canny (Canny, 1986; Gonzalez et al., 2009; Rebaza, 2007; OpenCV, s.f.) al que previamente se ha aplicado un filtro gaussiano de (7, 7). La elección de los parámetros de Canny se realiza automáticamente mediante (Rosebrock, 2015), utilizado y probado en otros trabajos como (De-Las-Heras et al., 2021). Por último, se aplica la operación morfológica de dilatado con un kernel (2, 2) para remarcar los trazos.

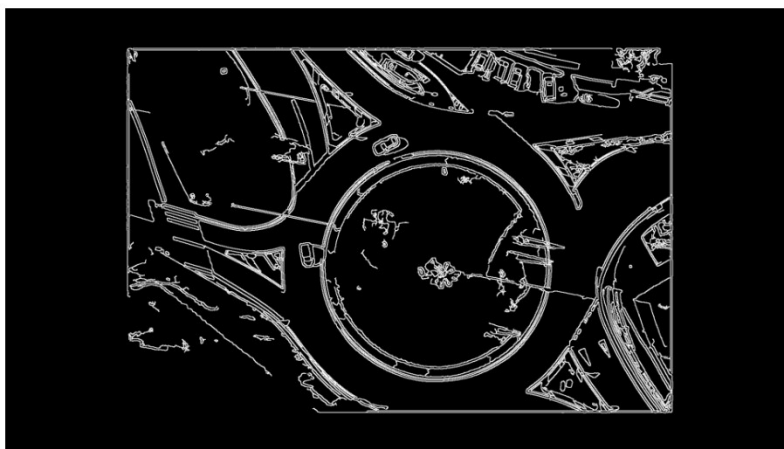


Figura 7. Imagen tras aplicar el algoritmo de Canny.

3.2.- Cálculo de circunferencias

Gracias a la imagen obtenida en el paso anterior, esta se encuentra en disposición de ser tratada por el algoritmo que calcula el centro y el radio de la isleta central de la rotonda, así como sus carriles. La transformada de Hough (Gonzalez et al., 2009; OpenCV, s.f.; Shehata et al., 1962; Ballard & Brown, 1982) es un algoritmo que reconoce líneas rectas dentro de imágenes. Resulta de mucha utilidad ya que provee como salida las ecuaciones de estas. Existe una versión para calcular circunferencias y elipses (OpenCV, s.f.; ImageJ, s.f.; scikit-image, s.f.; Xie & Ji, 2002). Gracias a esta, se obtiene la ecuación que representa la isleta principal y junto a datos previamente obtenidos como el número de carriles, ancho de carriles se obtiene el resto de las circunferencias (Figura 8).



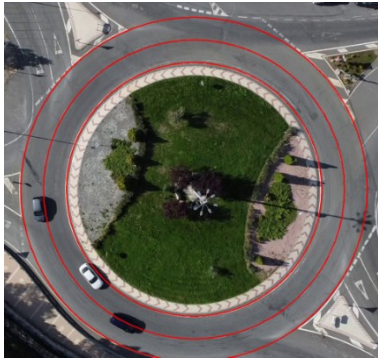
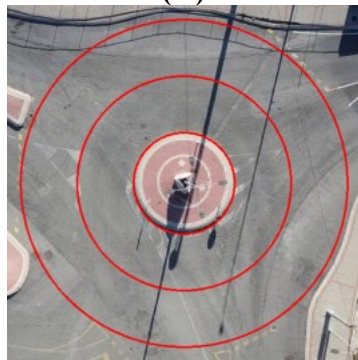
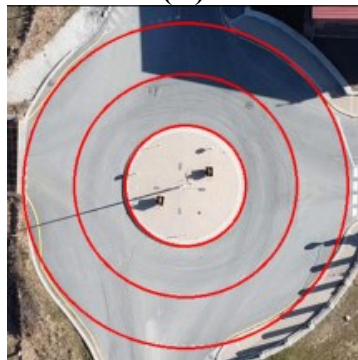
Figura 8. Funcionamiento del prototipo.

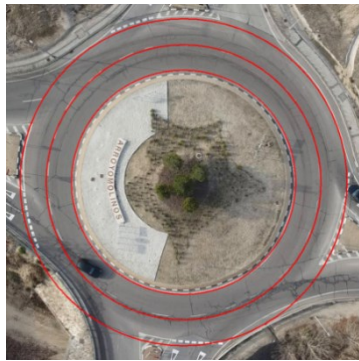
3.3.- Resultados

Cualitativamente, el pipeline reconoce la isleta central de manera bastante precisa. Sin embargo, experimentalmente se han detectado una serie de factores que afectan al reconocimiento:

- Tamaño de la rotonda. Aquellas con un tamaño mayor resultan en un tiempo de procesamiento mayor ya que hay un mayor número de píxeles que tratar.
- Vegetación en la imagen. A pesar de que el pipeline de procesamiento logra eliminar gran parte del ruido de la imagen, ciertas siluetas pertenecientes a la vegetación permanecen. Estas producen errores a la hora de reconocer la circunferencia central, de manera que, en ocasiones, esta se identifica en la vegetación.
- Aceras en la rotonda. Algunas rotondas tienen una pequeña acera que se muestra como una pequeña circunferencia concéntrica que puede confundir al reconocedor.
- Colores en la rotonda muy similares al asfalto. Esto produce que la circunferencia no sea lo suficientemente distinguible del asfalto por lo que no se reconoce la isleta correctamente.
- Tamaño de los carriles irregulares. Algunas rotondas no tienen un tamaño de carril fijo. Algunas tienen el carril más exterior tiene un ancho mayor, como por ejemplo la rotonda en la Figura 10-D, que tiene las esquinas de las entradas/salidas un poco más grandes que el carril. Esto ocasiona un pequeño error a la hora de delimitar los distintos carriles.

Todos estos factores se ven atenuados y la calidad del reconocimiento de la rotonda mejora si se realiza una calibración manual del RoI de manera que solo quede la isleta principal rodeada de asfalto. Para el prototipo esto no supone un gran contratiempo puesto que se trata de una fase de calibración que se realiza la primera vez que se implanta el sistema. Los resultados de cada rotonda se muestran en la Figura 9.

**(A)****(B)****(C)****(D)****(E)****(F)**



(G)



(H)

Figura 9. Resultados obtenidos con un ROI personalizado a cada rotonda.

4.- DETECCIÓN DE VEHÍCULOS

Una vez calculada las circunferencias y la ROI de la rotonda. El siguiente paso es reconocer cada uno de los vehículos que se encuentran en ella. Debido a que el objetivo de esta publicación no es encontrar el mejor modelo de reconocimiento per se, gracias a la revisión del estado del arte, se ha seleccionado un modelo de RetinaNet (Lin et al., 2020) construido sobre una resnet50 ya que, como se ha indicado anteriormente, el aprendizaje automático ha resultado ser un método muy eficaz en la localización de objetos. Además, esta CNN ha demostrado un buen desempeño en tareas de reconocimiento de objetos (Tan et al., 2021; De-Las-Heras et al., 2021) y de vehículos en imágenes aéreas (Stuparu et al., 2020). Debido a que este prototipo es un sistema que tiene que trabajar en tiempo real, es decir, que su información es válida durante un corto espacio de tiempo, se ha establecido el balance entre la calidad del modelo y la velocidad de reconocimiento como valores a optimizar.

4.1.- Dataset

Como requisito inicial de los algoritmos de aprendizaje supervisado, se requiere de un dataset con ejemplos previamente anotados. En este caso, se ha empleado (Puertas et al., 2022) que cuenta con 947,400 coches, 19,596 bicicletas, 2,262 camiones y 7,008 autobuses.

El data-augmentation consiste en aplicar una serie de transformaciones a las imágenes de entrenamiento, con el fin de crear sintéticamente nuevos ejemplos con los que los modelos de machine learning puedan generalizar mejor (Zoph et al., 2020). Esto ha sido probado en varios proyectos con buenos resultados (De-Las-Heras et al., 2021; Stuparu et al., 2020). Sobre este dataset ya se han aplicado estas técnicas, concretamente varios flípeos. No obstante, se han aplicado las siguientes transformaciones y ajustes del color para incrementar las posibilidades del dataset y permitir al modelo generalizar mejor:

- Rotaciones sobre la imagen original.
- Traslaciones, desplazar el contenido de la imagen.
- Shear (Figura 10), recortar los bordes de la imagen resultando el contenido real como un paralelogramo (Kathuria, s.f.).
- Aumentar o disminuir el tamaño de la imagen (escalado).

- Aumentos o disminuciones de propiedades de la imagen como: el brillo, el contraste, el hue y la saturación.



Figura 10. Ejemplo de shearing (Kathuria, s.f.).

Estas transformaciones se aplican de manera aleatoria sobre cada imagen individualmente durante la etapa de entrenamiento bajo los parámetros indicado en la Tabla 3 utilizando la librería (fizyr, s.f.).

Transformación	Valor mínimo	Valor máximo	
Rotación	-0.5	0.5	Radianes
Traslación	-0.5	0.5	Vector de traslación
Shearing	-0.5	0.5	Radianes
Escalado	0.9	1.1	Vector de escalado
Brillo	-0.1	0.1	Cantidad añadida a cada píxel.
Contraste	0.9	1.1	Factor de contaste.
Hue	-0.05	0.05	Cantidad añadida al canal de hue
Saturacion	0.95	1.05	Factor de multiplicación

Tabla 3. Parámetros del data-augmentation.

Finalmente, se ha dividido el dataset en varias porciones (Tabla 4). Se ha apartado el 10% del dataset para la evaluación del modelo después del entrenamiento. Del 90% restante, otro 90% se ha dedicado al propio entrenamiento, mientras que el otro 10% se ha empleado para evaluar el modelo después de cada época.

81%	9%	10%
Entrenamiento	Validación	Evaluación

Tabla 4. División del conjunto de datos.

La selección de a qué conjunto pertenece cada imagen se ha realizado de manera aleatoria teniendo en cuenta que el dataset presenta un desbalanceo entre clases. De esta manera, aproximadamente, el 81% de los ejemplos de cada clase se encuentran en el set de entrenamiento, el 9% en el de validación y el 10% en el de evaluación. De esta manera se asegura la suficiente representación de cada clase en cada conjunto.

4.2.- Métricas

Como se ha comentado anteriormente, las métricas interesantes para el prototipo son el tiempo de inferencia y la calidad del modelo. Para esta última, se ha seleccionado el mAP

(Mean Average Precision), utilizada en otras publicaciones para demostrar la bondad del modelo (De-Las-Heras et al., 2021; Stuparu et al., 2020; Everingham et al., 2010). Este se define como la media de los AP de cada clase (4), siendo el AP el área bajo la curva cobertura-precisión (5), calculada mediante (5) y (6). Para calcular tanto la cobertura como la precisión, se ha establecido un IoU (Intersection over Union) de 0.5. Esto significa que se requiere al menos un 50% de solapamiento entre el cuadro delimitador real y el predicho, para considerar una detección como positiva.

$$mAP = \frac{1}{n} \sum_{i=0}^n AP_i \quad (4)$$

$$AP_{class} = \int_0^1 p(r) dr \quad (5)$$

$$Precision = \frac{TP}{TP + FP} \quad (6)$$

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

4.3.- Experimentos

Para los experimentos acerca del entrenamiento de la red neuronal, se han realizado los siguientes escenarios teniendo en cuenta los resultados de (Stuparu et al., 2020):

1. Modo normal.

El primer experimento ha sido simplemente entrenar el modelo de Retinanet con el dataset y condiciones detalladas en la sección anterior. El mejor modelo se ha seleccionado mediante el mAP más alto encontrado para el conjunto de evaluación. Los resultados se indican en la Tabla 5.

Clase	Minimum Score = 0.05		Minimum Score = 0.5	
	AP@50	AP@75	AP@50	AP@75
Coche	0.9976	0.9409	0.9967	0.9401
Bicicleta	0.8496	0.4695	0.8345	0.4683
Camión	0.9624	0.7886	0.9562	0.7858
Autobus	0.9654	0.5337	0.9234	0.5125
mAP	0.9438	0.6832	0.9277	0.6767

Tabla 5. Resultados modelo 2000x2000px (0.07 s. de inferencia por imagen).

2. Escalados en el tamaño de la imagen.

Este experimento se ha propuesto gracias a lo observado en (Stuparu et al., 2020) donde, al incrementar el tamaño de las imágenes, se consiguieron mejores resultados. Para ello, originalmente se multiplicó por 2 el tamaño la imagen original (1920x1080px), resultando imágenes de 3840x2160px. Sin embargo, este tamaño ocasionaba continuos

errores en el entrenamiento debido a que el hardware disponible disponía de insuficiente potencia de cálculo para manejar imágenes tan grandes. De esta manera, se cambió el factor de escalado de 2 a 1.5, resultando imágenes de (2880x1620px) que sí que funcionan con el hardware disponible. Con esta configuración se ha logrado un tiempo de inferencia por cada imagen de 0.1817 segundos. No se continuó con este entrenamiento puesto que este tiempo se ha considerado demasiado elevado.

Tras este experimento se realizó otro más reduciendo cada dimensión de la imagen a la mitad. También se descartó debido a que los cuadros de la clase Cycle eran tan pequeños que el modelo no reconocía ninguno (su AP era de 0).

3. Extraer las rotondas generando un nuevo dataset.

Para el último experimento, se ha tomado el trabajo de reconocimiento de la rotonda (explicado en la sección) para generar un nuevo dataset formado únicamente por imágenes cuadradas que contienen la rotonda misma. Esto lleva la ventaja de que se elimina parte de la imagen que no interesa. Sin embargo, requiere de rescalar todas las imágenes para que tengan el mismo tamaño, lo que conlleva a que a las imágenes más pequeñas se les tenga que aplicar un escalado mucho mayor. Esto produce que estas pierdan calidad. Este experimento se ha realizado con imágenes de 1000x1000px y 2000x2000px, cuyos resultados se encuentran en las Tablas 6 y 7:

Clase	Mínimo Score = 0.05		Mínimo Score = 0.5	
	AP@50	AP@75	AP@50	AP@75
Coche	0.9892	0.9720	0.9776	0.9658
Bicicleta	0.6234	0.1651	0.5057	0.1430
Camión	0.9881	0.9699	0.9714	0.9556
Autobus	0.9739	0.9261	0.9242	0.8978
mAP	0.8937	0.7583	0.8447	0.7406

Tabla 6. Resultados modelo 1000x1000px (0.04 s. de inferencia por imagen).

Clase	Mínimo Score = 0.05		Mínimo Score = 0.5	
	AP@50	AP@75	AP@50	AP@75
Coche	0.6899	0.6529	0.6576	0.6404
Bicicleta	0.0414	0.0067	0.0000	0.0000
Camión	0.4317	0.4071	0.2978	0.2893
Autobus	0.4639	0.3885	0.2443	0.2261
mAP	0.4067	0.3638	0.6899	0.2889

Tabla 7. Resultados modelo 2000x2000px (0.17 s. de inferencia por imagen).

4.4.- Resultados

Tras los resultados anteriores, los modelos obtenidos del primer experimento, medidos con AP@50 con el mínimo score de 0.05 y 0.5 (0.9438 y 0.9277), son los más prometedores. Para realizar la mejor decisión, se ha calculado el F1-score (8) de ambos. Los resultados son: 0.483 y 0.983. Tras este resultado, se ha optado por el segundo modelo, debido a que esta última métrica es muy superior a la del primero, aunque el primero tenga un mAP mejor que el segundo. En general, el modelo funciona

correctamente (Figuras 11 y 12). No obstante, se han observado algunos errores de detección con elementos urbanos (porciones de isletas, marquesinas, contenedores, etc.), con vehículos negros, coches que se encuentran sobre un paso de cebra o zonas oscuras y vehículos.

$$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{8}$$

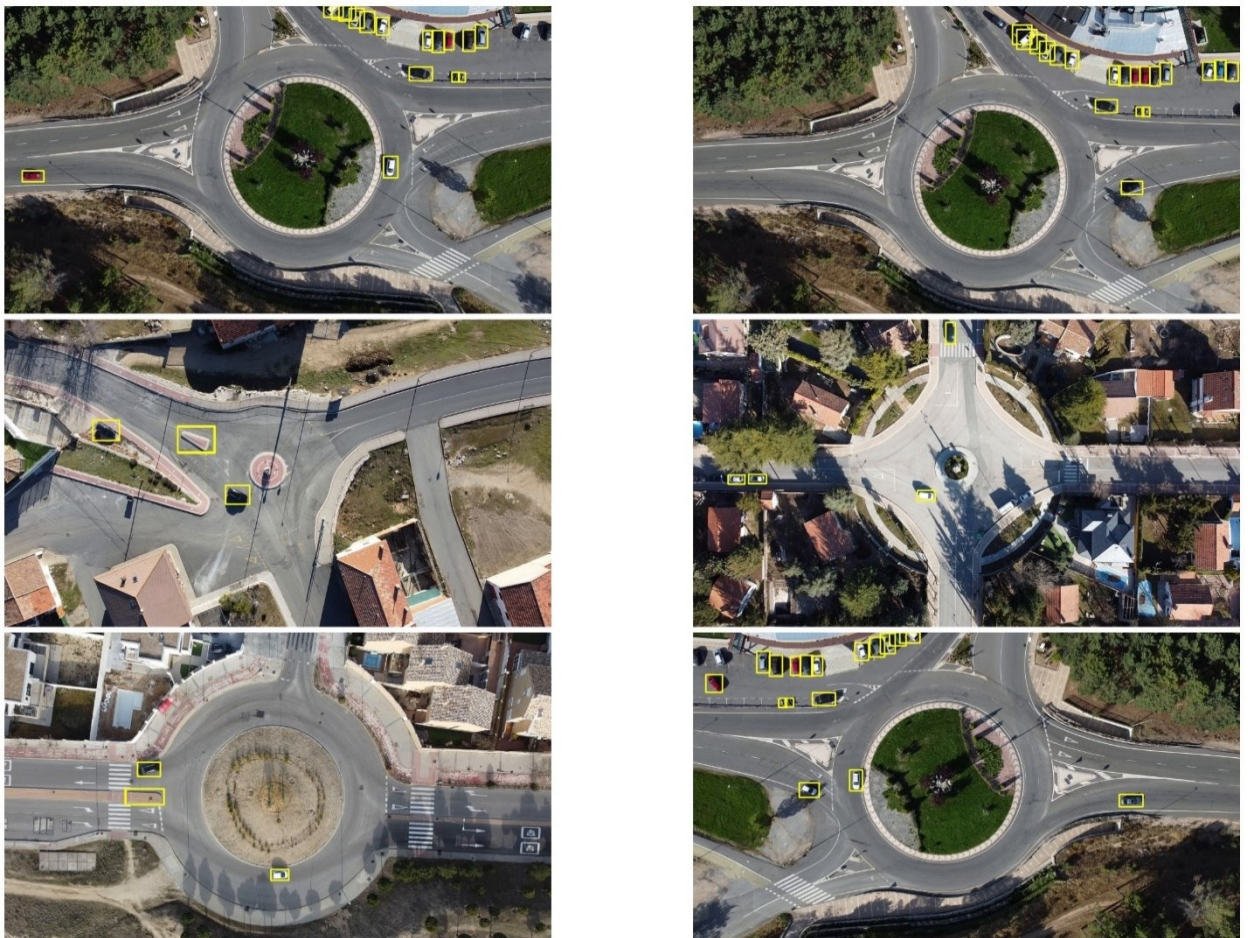


Figura 11. Resultados del modelo de detección de vehículos (parte 1).



Figura 12. Resultados del modelo de detección de vehículos (parte 2).

5.- EXTRACCIÓN DE INFORMACIÓN

Una vez obtenido el detector de vehículos, el siguiente paso es la extracción de información propia de la rotonda. Gracias al estudio del cálculo de la capacidad de dichas rotondas (sección 1.2), se ha identificado que mucha información de interés se encuentra localizada en partes concretas de la rotonda. De esta manera, el sistema provee la siguiente información:

1. Posición y tipo de vehículo dentro de la rotonda. Esta información es inmediata al ser la salida del modelo de Deep Learning logrado en el paso anterior.
2. Tráfico por cada carril. Gracias a la información de las circunferencias obtenidas en la sección 3 y la salida del detector de vehículos, se calcula la distancia euclídea entre ellas para descubrir en qué carril se encuentra. Una medida muy común para expresar la cantidad tráfico es el PCU (Passager Car Unities) (Mauro, 2010), que pondera el tipo de cada vehículo asignando una cantidad determinada por el espacio físico que ocupa (Tabla 8).

Tipo de vehículo	PCU
Coche	1 pcu
Camión/Bus	1.5 pcu
Camión con trailer	2 pcu
Moto/Bicicleta	0.5 pcu

Tabla 8. Unidades PCU por tipo de vehículo.

3. Tráfico que disturba a una nueva incorporación (tráfico entrante, saliente y circulante). Una de las claves del cálculo de la capacidad de entrada en una rotonda, es el tráfico que afecta de algún modo a la incorporación (Figura 13). Publicaciones lo definen como flujo perturbador (Q_d) el cual, según el método de cálculo empleado, tienen encuentra simplemente el tráfico circulante en frente de la rotonda (Q_c) o incluyen el tráfico saliente (Q_u) y entrante (Q_e) (Mauro, 2010). La Figura 15 ilustra cada una de estas medidas. Para hallar todas estas situaciones, es necesario delimitar esta área de influencia (Figuras 14 y 15) y simplemente comprobar si un vehículo se encuentra en ellas.

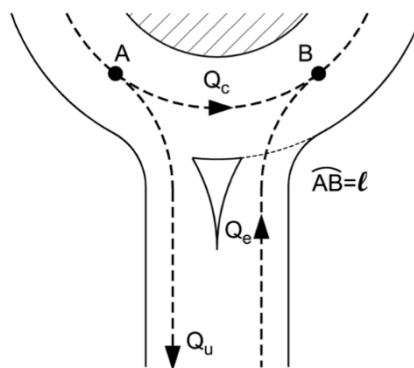


Figura 13. Diagrama del tráfico perturbador en la incorporación a una rotonda.

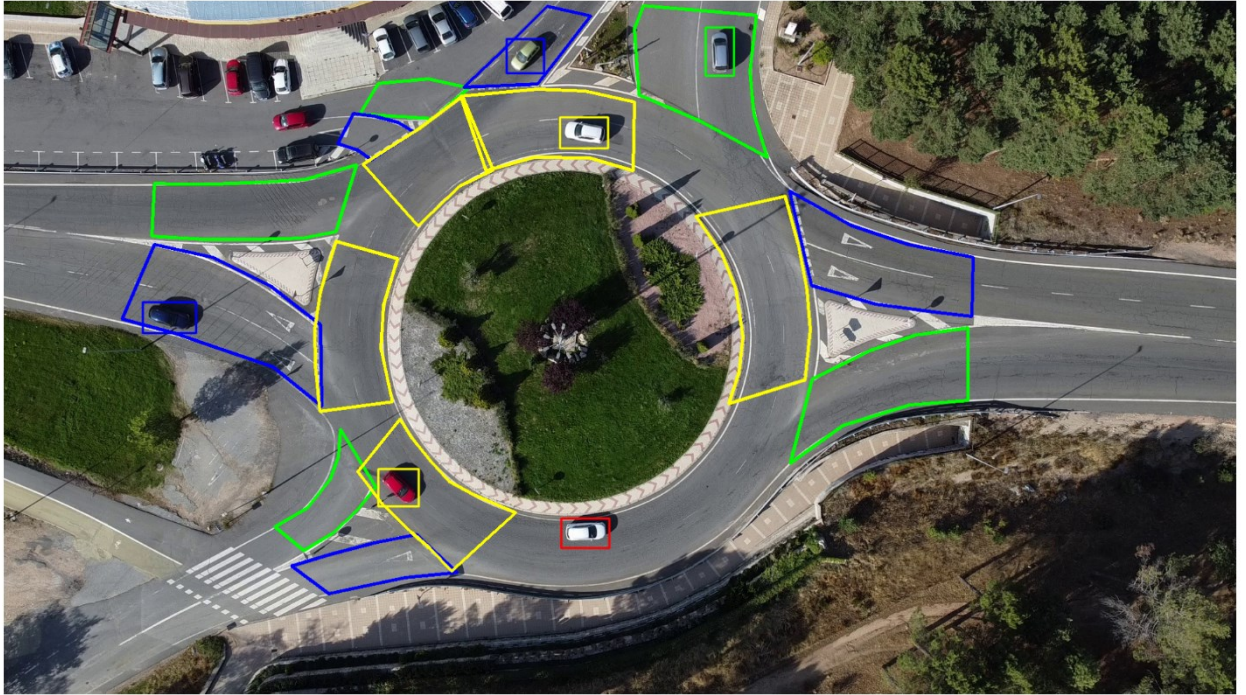


Figura 14. Delimitación de las distintas áreas de una rotonda. (azul: entrada, verde: salida, amarillo: tráfico circulante, rojo: resto).



Figura 15. Vehículos pertenecientes a cada área (azul: entrada, verde: salida, amarillo: tráfico circulante, rojo: resto).

6.- CONCLUSIONES

La principal motivación de este trabajo ha sido aportar un punto de vista distinto a las publicaciones actuales acerca de vehículos autónomos y rotondas. Estas las abordan como

un escenario complicado por el que estos vehículos deben aprender a transitar. Sin embargo, este trabajo las considera como una fuente de información consistente en imágenes aéreas de las propias rotondas que pueden ser tomadas desde UAS (drones) o bien desde un sistema de cámaras fijas instaladas en altura con un mástil en el centro de la rotonda.

A lo largo de este trabajo se ha detallado el proceso de desarrollo de un prototipo de ADAS de extracción de información acerca de rotondas españolas a partir de imágenes aéreas de dichas rotondas. Este se divide en dos partes. Una primera que consiste en reconocer las circunferencias de la isleta central y los carriles. Esto se ha logrado con técnicas de visión por computador utilizadas anteriormente en otros aspectos, pero no específicamente en rotondas. La segunda, mediante un modelo de Deep Learning, con un mAP de 0.9277 (más alto que otras publicaciones similares), se detectan los vehículos. Seguidamente, se extrae información del estado de la rotonda: vehículos en entradas/salidas, aquellos que perturban en las incorporaciones y la localización cada uno de ellos en los carriles.

El uso de imágenes aéreas tomadas desde UAS (drones) ha permitido realizar el prototipo de ADAS que sirve para monitorizar el estado de las rotondas sin necesidad de instalar cámaras 360° en postes en el centro de las propias rotondas (como se explicaba en la sección 2), lo que ha agilizado este estudio. A pesar de la autonomía limitada de estas aeronaves, su uso seguiría siendo una opción interesante en situaciones concretas, como por ejemplo en zonas con una gran concentración de rotondas. Estas podrían ser controladas con uno o varios UAS en ciertos episodios y bajo demanda (horas punta, accidentes, etc.). Estos UAS también podrían utilizarse en diferentes tipos de misiones de control además de proporcionar este servicio de ADAS.

Además del propósito de proveedor de información a los vehículos, este sistema también tiene una utilidad en el estudio de las rotondas. La información que extrae de ella se puede almacenar para posteriores investigaciones acerca de ellas, siendo algunas aplicaciones concretas el cálculo de la capacidad global de la entrada, horas punta y combinado, con un sistema de identificación unívoca de vehículos, también trayectorias y comportamientos de los conductores.

Con ello se logra mejorar la seguridad en la circulación en rotondas o glorietas mediante el uso de Inteligencia artificial.

7.- REFERENCIAS

- Akçelik, R. (2004). Roundabouts with unbalanced flow patterns. *Paper presented at the Institute of Transportation Engineers 2004 Annual Meeting*.
- Anagnostopoulos, A., Kehagia, F., Damaskou, E., Mouratidis, A., & Aretoulis, G. (2021). Predicting Roundabout Lane Capacity using Artificial Neural Networks. *Journal of Engineering Science and Technology Review*, 14, 210-215.
- Badue, C., Guidolini, R., Carneiro, R., Azevedo, P., Cardoso, V., Forechi, A., . . . De Souza, A. (2019). Self-Driving Cars: A Survey,.
- Ballard, D., & Brown, C. (1982). Computer vision. Englewood Cliffs, NJ: Prentice Hall.
- BBC. (2020, 09 16). *Uber's self-driving operator charged over fatal crash*. Retrieved 12 31, 2022, from <https://www.bbc.com/news/technology-54175359>

- BBC. (2021, 12 15). *Tesla Model 3: Paris' largest taxi firm suspends cars after fatal crash*. Retrieved 12 31, 2022, from <https://www.bbc.com/news/world-europe-59647069>
- BCG. (2015). *A Roadmap to Safer Driving through Advanced Driver Assistance Systems*. Retrieved 12 30, 2022, from https://image-src.bcg.com/Images/MEMA-BCG-A-Roadmap-to-Safer-Driving-Sep-2015_tcm9-63787.pdf
- Behrisch, M., Bieker, L., Erdmann, J., & Krajzewicz, D. (2011). SUMO - Simulation of Urban MObility.
- Bernhard, W., & Portmann, P. (2000). Traffic simulation of roundabouts in Switzerland. *2000 Winter Simulation Conference Proceedings (Cat. No.00CH37165)*, 2, 1148-1153.
- Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. (2020). Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.
- Bock, J., Krajewski, R., Moers, T., & Vater, L. (n.d.). *UniD. The University Drone Dataset. Naturalistic Trajectories of Vehicles and Vulnerable Road Users Recorded at the RWTH Aachen University Campus*. Retrieved 03 26, 2022, from <https://www.unid-dataset.com/>
- Bock, J., Krajewski, R., Moers, T., Runde, S., Vater, L., & Eckstein, L. (2020). The inD Dataset: A Drone Dataset of Naturalistic Road User Trajectories at German Intersections. *2020 IEEE Intelligent Vehicles Symposium (IV)*, 1929-1934.
- Breuer, A., Termöhlen, J.-A., Homoceanu, S., & Fingscheidt, T. (2020). OpenDD: A Large-Scale Roundabout Drone Dataset. *Proceedings of International Conference on Intelligent Transportation Systems*.
- Brilon, W. (1991). *Intersections Without Traffic Signals II: Proceedings of an International Workshop*. Springer-Verlag.
- Brilon, W., Wu, N., & Bondzio, L. (1997). Unsignalized intersections in Germany — a state of the art 1997. *Third international symposium on intersections without traffic signals*, 61–70.
- Canny, J. (1986). A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8(6)*, 679-698.
- Chang, Y.-C., Huang, C., Chuang, J.-H., & Liao, I.-C. (2018). Pedestrian Detection in Aerial Images Using Vanishing Point Transformation and Deep Learning. *2018 25th IEEE International Conference on Image Processing (ICIP)*, 1917-1921.
- Chen, Z., Wang, C., Wen, C., Teng, X., Chen, Y., Guan, H., . . . Li, J. (2015). Vehicle detection in high-resolution aerial images via sparse representation and superpixels. *IEEE Transactions on Geoscience and Remote Sensing*, 54(1), 103-116.
- Cheng, G., Wang, Y., Xu, S., Wang, H., Xiang, S., & Pan, C. (2017). Automatic Road Detection and Centerline Extraction via Cascaded End-to-End Convolutional Neural Network. *IEEE Transactions on Geoscience and Remote Sensing*, 55(5), 3322-3337.
- Chin, H. C. (1985). SIMRO: A model to simulate traffic at roundabouts. *Traffic Engineering and Control*, 26, 109–113.
- Chung, E., Young, W., & Akçelik, R. (1992). Comparison of roundabout capacity and delay estimates from analytical and simulation models. *16th ARRB Conference*, 16, 369–385.
- Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. *In Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1, 886-893.

- De-Las-Heras, G., Sánchez-Soriano, J., & Puertas, E. (2021). Advanced Driver Assistance Systems (ADAS) Based on Machine Learning Techniques for the Detection and Transcription of Variable Message Signs on Roads. *Sensors*, 21(17).
- Deng, Z., Sun, H., Zhou, S., Zhao, J., & Zou, H. (2017). Toward fast and accurate vehicle detection in aerial images using coupled region-based convolutional neural networks. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(8), 3652--3664.
- Dijkstra, K., van de Loosdrecht, J., Schomaker, L. R., & Wiering, M. A. (2019). Hyperspectral demosaicking and crosstalk correction using deep learning. *Machine Vision and Applications*, 30, 1-21.
- Dirección General de Tráfico - Ministerio del Interior. (2021, 12 23). *Código de Tráfico y Seguridad Vial*. Retrieved 01 01, 2022, from https://www.boe.es/biblioteca_juridica/codigos/abrir_pdf.php?fich=020_Codigo_de_Trafico_y_Seguridad_Vial.pdf
- Dirección General de Tráfico - Ministerio del Interior. (2018). *Cuestiones De Seguridad Vial*. Retrieved 12 30, 2021, from <http://www.dgt.es/Galerias/seguridad-vial/formacion-vial/cursos-para-profesores-y-directores-de-autoescuelas/XXI-Cuso-Profesores/Manual-II-Cuestiones-de-Seguridad-Vial-2018.pdf>
- DJI. (n.d.). *DJI Mini 2*. Retrieved 03 26, 2022, from <https://www.dji.com/ca/mini-2/specs>
- Du, D., Qi, Y., Yu, H., Yang, Y., Duan, K., Li, G., . . . Tian, Q. (2018). The Unmanned Aerial Vehicle Benchmark: Object Detection and Tracking. *European Conference on Computer Vision (ECCV)*.
- Elkhachy, I. (2021). Accuracy Assessment of Low-Cost Unmanned Aerial Vehicle (UAV) Photogrammetry. *Alexandria Engineering Journal*, 60(6), 5579-5590.
- Everingham, M., Van Gool, L., & Williams, C. (2010). The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88, 303-338.
- Felzenszwalb, P., McAllester, D., & Ramanan, D. (2008). A discriminatively trained, multiscale, deformable part model, computer vision and pattern recognition. *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 1-8.
- fizyr. (n.d.). *Keras RetinaNet*. Retrieved 04 05, 2022, from <https://github.com/fizyr/keras-retinanet>
- Gallardo, V. (2005). Funciones de las rotondas urbanas y requerimientos urbanísticos de organización.
- García Cuenca, L., Puertas, E., Fernández Andrés, J., & Aliane, N. (2019). Autonomous Driving in Roundabout Maneuvers Using Reinforcement Learning with Q-Learning. *Electronics*, 8(12).
- García Cuenca, L., Sánchez-Soriano, J., Puertas, E., Fernández Andrés, J., & Aliane, N. (2019). Machine Learning Techniques for Undertaking Roundabouts in Autonomous Driving. *Sensors*, 19(10).
- Girshick, R. (2015). Fast R-CNN. *2015 IEEE International Conference on Computer Vision (ICCV)*, 1440-1448.
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2013). Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*.
- Gonzalez, R., Woods, R., & Masters, B. (2009). *Digital image processing*.
- Guichet, B. (1997). Roundabouts in France: Development, safety, design, and capacity. *Proceedings of the third international symposium on intersections without traffic signals*, 100-105.

- Gupta, A., Watson, S., & Yin, H. (2021). Deep learning-based aerial image segmentation with open data for disaster impact assessment. *Neurocomputing*, 439, 22-33.
- Haging, O. (2000). Effects of OD Flows on Roundabout Entry Capacity. *Fourth international symposium on highway capacity*(434–445).
- Hough, P. (1962, 12 18). *U.S. Patent No. US3069654A*.
- ImageJ. (n.d.). *Hough Circle Transform*. Retrieved 01 16, 2022, from <https://imagej.net/plugins/hough-circle-transform>
- Kathuria, A. (n.d.). *Data Augmentation for Bounding Boxes: Rotation and Shearing*. Retrieved 04 05, 2022, from <https://blog.paperspace.com/data-augmentation-for-object-detection-rotation-and-shearing/>
- Kembhavi, A., Harwood, D., & Davis, L. S. (2010). Vehicle detection using partial least squares. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(6), 1250-1265.
- Kim, J., Candemir, S., Chew, E. Y., & Thoma, G. R. (2018). Region of Interest Detection in Fundus Images Using Deep Learning and Blood Vessel Information. *2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS)*, 357-362.
- Kimber, R. M. (1980). LR942 The traffic capacity of roundabouts. *Transport and Road Research Laboratory*.
- Krajewski, R., Bock, J., Kloeker, L., & Eckstein, L. (2018). The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems. *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2118-2125.
- Krajewski, R., Bock, J., Moers, T., & Vater, L. (n.d.). *ExiD. The Exits and Entries Drone Dataset. Naturalistic Trajectories of Vehicles Recorded at Exits and Entries of German Highways*. Retrieved 03 26, 2022, from <https://www.exid-dataset.com/>
- Krajewski, R., Moers, T., Bock, J., Vater, L., & Eckstein, L. (2020). The roundD Dataset: A Drone Dataset of Road User Trajectories at Roundabouts in Germany. *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, 1-6, 2020.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM*, 60(6), 84–90.
- Kukkala, V. K., Tunnell, J., Pasricha, S., & Bradley, T. (2018). Advanced Driver-Assistance Systems: A Path Toward Autonomous Vehicles. *IEEE Consumer Electronics Magazine*, 18-25.
- Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2020). Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2), 318-327.
- Liu, K., & Mattyus, G. (2015). Fast Multiclass Vehicle Detection on Aerial Images. *IEEE Geoscience and Remote Sensing Letters*, 12(9), 1938-1942.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. (2016). Ssd: Single shot multibox detector. *European conference on computer vision*.
- Macioszek, E. (2020). Roundabout Entry Capacity Calculation—A Case Study Based on Roundabouts in Tokyo, Japan, and Tokyo Surroundings. *Sustainability*, 12, 1533.
- Marlow, M., & Maycock, G. (1982). SR724 The effect of Zebra crossings on junction entry capacities. *Crowthorne: Transport and Road Research Laboratory*.
- Mauro, R., & Guerrieri, M. (2012). Right-turn Bypass Lanes at Roundabouts: Geometric Schemes and Functional Analysis. *Modern Applied Science*, 7.

- Michon, J. (1993). Generic Intelligent Driver Support: A Comprehensive Report on GIDS.
- Microsimulation, P. (2011). S-Paramics (Parallel Microscopic Traffic Simulator).
- Montgomery, F., Tenekeci, G., & Wainaina, S. (2010). Roundabout capacity in adverse weather and light conditions. *Proceedings of The Institution of Civil Engineers-transport - PROC INST CIVIL ENG-TRANSPORT*, 163, 29-39.
- Naranjo, J., Sotelo, M.-A., Gonzalez, C., Garcia, R., & Pedro, T. (2007). Using Fuzzy Logic in Automated Vehicle Control. *Intelligent Systems, IEEE*, 22, 36-45.
- O'Kane, S. (2018, 04 19). *How Tesla and Waymo are tackling a major problem for self-driving cars: data*. Retrieved 01 01, 2022, from <https://www.theverge.com/transportation/2018/4/19/17204044/tesla-waymo-self-driving-car-data-simulation>
- OpenCV. (n.d.). *Canny Edge Detector* . Retrieved 01 16, 2023, from https://docs.opencv.org/4.x/da/d5c/tutorial_canny_detector.html
- OpenCV. (n.d.). *Image Thresholding* . Retrieved 01 15, 2023, from https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html
- OpenCV. (n.d.). *Morphological Transformations*. Retrieved 01 15, 2023, from https://docs.opencv.org/3.4/d9/d61/tutorial_py_morphological_ops.html
- OpenCV. (n.d.). *Operations on arrays*. Retrieved 01 15, 2023, from https://docs.opencv.org/3.4/d2/de8/group_core_array.html#ga6fef31bc8c4071cbc14a758a2b79c14
- OpenCV. (n.d.). *Smoothing Images*. Retrieved 01 15, 2023, from https://docs.opencv.org/4.x/d4/d13/tutorial_py_filtering.html
- OpenCV. Open Source Computer Vision. (n.d.). *Hough Circle Transform* . Retrieved 01 16, 2023, from https://docs.opencv.org/3.4/d4/d70/tutorial_hough_circle.html
- OpenCV. Open Source Computer Vision. (n.d.). *Hough Line Transform* . Retrieved 01 16, 2023, from https://docs.opencv.org/4.x/d9/db0/tutorial_hough_lines.html
- Popular Science. (2021, 12 30). *Tesla caps off a tumultuous year with a massive recall*. Retrieved 12 31, 2022, from <https://www.popsci.com/technology/tesla-recalls-vehicles-safety-issues/>
- Puertas, E., De-Las-Heras, G., Fernández-Andrés, J., & Sánchez-Soriano, J. (2022). Dataset: Roundabout Aerial Images for Vehicle Detection. *Data*, 7(4).
- Rebaza, J. (2007). *Detección de Bordos Mediante el Algoritmo de Canny*. Master's Thesis. Trujillo, Peru: Universidad Nacional de Trujillo.
- Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, 779–788.
- Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137-1149.
- Robicquet, A., Sadeghian, A., Alahi, A., & Savarese, S. (2016). Learning Social Etiquette: Human Trajectory Prediction In Crowded Scenes. *European Conference on Computer Vision (ECCV)*.

- Rodrigues, M., Gest, G., McGordon, A., & Marco, J. (2017). Adaptive behaviour selection for autonomous vehicle through naturalistic speed planning. *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 1-7.
- Rodríguez, J. I. (2014). Cómo circular por una glorieta. *Tráfico y seguridad Vial, Septiembre-Octubre(228)*, 28-30.
- Rosebrock, A. (2015). *Zero-Parameter, Automatic Canny Edge Detection with Python and OpenCV*. Retrieved 01 16, 2022, from <https://www.pyimagesearch.com/2015/04/06/zero-parameter-automatic-canny-edge-detection-with-python-and-opencv/>
- scikit-image development team. (n.d.). *Circular and Elliptical Hough Transforms*. Retrieved 01 16, 2022, from https://scikit-image.org/docs/stable/auto_examples/edges/plot_circular_elliptical_hough_transform.html
- Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & Lecun, Y. (2013). OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. *arXiv preprint arXiv:1312.6229*.
- Shehata, A., Mohammad, S., Abdallah, M., & Ragab, M. (n.d.). A survey on Hough transform, theory, techniques and applications. *arXiv 2015, arXiv:1502.02160*.
- Shen, J., Liu, N., & Sun, H. (2021). Vehicle detection in aerial images based on lightweight deep convolutional network. *IET Image Processing, 15(2)*, 479-491.
- Silva, A., Vasconcelos, A., & Santos, S. (2014). Moving from Conventional Roundabouts to Turbo-Roundabouts. *Procedia - Social and Behavioral Sciences, 111*, 137-146.
- Soleimani, A., & Nasrabadi, N. M. (2018). Convolutional Neural Networks for Aerial Multi-Label Pedestrian Detection. *2018 21st International Conference on Information Fusion (FUSION)*, 1005-1010.
- Soviany, P., & Ionescu, R. (2018). Optimizing the Trade-off between Single-Stage and Two-Stage Object Detectors using Image Difficulty Prediction. *In Proceedings of the 2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*.
- Sreedhar, K., & Panlal, B. (2012). Enhancement of images using morphological transformations. *Int. J. Comput. Sci. Inf. Technol(4)*, 33–50.
- Stuparu, D.-G., Ciobanu, R.-I., & Dobre, C. (2020). Vehicle Detection in Overhead Satellite Images Using a One-Stage Object Detection Model. *Sensors, 20(22)*.
- Systems, T.-T. S. (2011). Aimsun (Advanced Interactive Microscopic Simulator for Urban and Non-Urban Networks).
- Tan, L., Huangfu, T., & Wu, L. (2021). Comparison of RetinaNet, SSD, and YOLO v3 for real-time pill identification. *BMC Medical Informatics and Decision Making, 21*.
- Tang, T., Zhou, S., Deng, Z., Zou, H., & Lei, L. (2020). Vehicle detection in aerial images based on region convolutional neural networks and hard negative example mining. *Sensors, 20(22)*.
- The New York Times. (2021, 04 18). *2 Killed in Driverless Tesla Car Crash, Officials Say*. Retrieved 12 31, 2022, from <https://www.nytimes.com/2021/04/18/business/tesla-fatal-crash-texas.html>
- Tollazzi, T., Rencelj, M., & Turnsek, S. (2011). New Type of Roundabout: Roundabout with “Depressed” Lanes for Right Turning – “Flower Roundabout”. *PROMET - Traffic&Transportation, 23*.
- Troutbeck, R. J. (1989). SR45 evaluating the performance of a roundabout.

- Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *In Proceedings of the 2001 IEEE Computer Society Conference, 1*.
- Viola, P., & Jones, M. (2004). Robust real-time face detection. *Int. J. Comput. Vis.*, 57, 137–154.
- World Health Organization. (2021, 06 21). *Road traffic injuries*. Retrieved 12 31, 2022, from <https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>
- World Health Organization. (n.d.). *Decade of Action for Road Safety 2011–2020*. Retrieved 12 30, 2022, from <https://www.who.int/publications/i/item/decade-of-action-for-road-safety-2011-2020>
- Xie, Y., & Ji, Q. (2002). A new efficient ellipse detection method. *Pattern Recognition, 2002. Proceedings. 16th International Conference on, 2*.
- Yap, Y. H., Gibson, H. M., & Waterson, B. J. (2013). An International Review of Roundabout Capacity Modelling. *Transport Reviews*, 33(5), 593-616.
- Yap, Y. H., Gibson, H., & Waterson, B. (2015). Models of Roundabout Lane Capacity. *Transportation engineering journal of ASCE*, 141.
- Yu, Y., Gu, T., Guan, H., Li, D., & Jin, S. (2019). Vehicle detection from high-resolution remote sensing imagery using convolutional capsule networks. *IEEE Geoscience and Remote Sensing Letters*, 16(12), 1894-1898.
- Zhong, J., Lei, T., & Yao, G. (2017). Robust Vehicle Detection in Aerial Images Based on Cascaded Convolutional Neural Networks. *Sensors*, 17(12).
- Zoph, B., Cubuk, E., Ghiasi, G., Lin, T.-Y., Shlens, J., & Le, Q. (2020). Learning Data Augmentation Strategies for Object Detection. *Computer Vision - ECCV 2020*, 566-583.
- Zou, Z., Shi, Z., Guo, Y., & Ye, J. (2019). Object Detection in 20 Years: A Survey. *arXiv:1905.05055v2*.

