

# Dataset: Roundabout Aerial Images for Vehicle Detection

Enrique Puertas <sup>1</sup>, Gonzalo De-Las-Heras <sup>2</sup>, Javier Fernández-Andrés <sup>3</sup> and Javier Sánchez-Soriano <sup>4,\*</sup>

<sup>1</sup> Department of Science, Computing and Technology, Universidad Europea de Madrid, Calle Tajo s/n, Villaviciosa de Odón, 28670 Madrid, Spain; enrique.puertas@universidadeuropea.es

<sup>2</sup> SICE Canada Inc., Toronto, ON M4P 1G8, Canada; gdelasheras@sice.com

<sup>3</sup> Department of Engineering, Universidad Europea de Madrid, Calle Tajo s/n, Villaviciosa de Odón, 28670 Madrid, Spain; javier.fernandez@universidadeuropea.es

<sup>4</sup> Escuela Politécnica Superior, Universidad Francisco de Vitoria, 28223 Pozuelo de Alarcón, Spain

\* Correspondence: javier.sanchez@ufv.es

**Abstract:** This publication presents a dataset of Spanish roundabouts aerial images taken from a UAV, along with annotations in PASCAL VOC XML files that indicate the position of vehicles within them. Additionally, a CSV file is attached containing information related to the location and characteristics of the captured roundabouts. This work details the process followed to obtain them: image capture, processing, and labeling. The dataset consists of 985,260 total instances: 947,400 cars, 19,596 cycles, 2262 trucks, 7008 buses, and 2208 empty roundabouts in 61,896 1920 × 1080 px JPG images. These are divided into 15,474 extracted images from 8 roundabouts with different traffic flows and 46,422 images created using data augmentation techniques. The purpose of this dataset is to help research into computer vision on the road, as such labeled images are not abundant. It can be used to train supervised learning models, such as convolutional neural networks, which are very popular in object detection.



**Citation:** Puertas, E.; De-Las-Heras, G.; Fernández-Andrés, J.; Sánchez-Soriano, J. Dataset: Roundabout Aerial Images for Vehicle Detection. *Data* **2022**, *7*, 47. <https://doi.org/10.3390/data7040047>

Academic Editor:  
Joaquín Torres-Sospedra

Received: 17 March 2022

Accepted: 10 April 2022

Published: 12 April 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Dataset:** <https://doi.org/10.5281/zenodo.6362360>.

**Dataset License:** Creative Commons Attribution 4.0 International.

**Keywords:** roundabouts; aerial; dataset; UAV; object detection; machine learning; ADAS; PASCAL VOC; autonomous driving; deep learning; neural networks; RetinaNet

## 1. Introduction

UAVs (unmanned aerial vehicles) are motorized vehicles capable of accessing hard-to-reach places and sending high-resolution images in real time at an affordable cost. These are complemented by processing centers that receive and extract information from the images through object detection. This consists of recognizing the object and being able to locate it in the image. This way, the input would be an entire image, and the output would be a series of names and locations. Deep learning models, particularly object detection CNNs (convolutional neural networks), have shown great performance in this task. These are machine learning algorithms that require previously labeled examples for training (supervised learning). They are divided into two groups: one and two-stage. One-stage models treat the detection as a regression problem, learning the probabilities of a class and the location. Two-stage models, first, send a series of regions of interest to the class classifier and then, the second step, to the coordinate delimiter. One-stage ones are faster but have less accuracy than two-stage ones [1]. Some examples are: one-stage—YOLO (You Only Look Once, v1 [2], v2/9000 [3], v3 [4], v4 [5]), SSD (Single Shot Detector) [6], and RetinaNet [7]; two-stage—R-CNN [8], Fast R-CNN [9], and Faster R-CNN [10]. These models have proven to be useful in a variety of fields [11–13], including traffic and its infrastructures. Some examples are vehicle [14–22], road [23], or pedestrian detection [24,25].

According to the Spanish Traffic Department (DGT) [26], “roundabouts are a special type of intersection in which they are connected by a ring that establishes rotating traffic flow around a central island.” These are the subject of study since they are already complex maneuvers [27,28] for autonomous vehicles. Furthermore, this type of traffic infrastructure offers a lot of information that can be extracted from images, such as vehicle trajectories or positions. There are several datasets available to support these studies [29–32]. However, although they are very useful, they are not abundant and are not easily accessible.

This publication presents an open-access dataset containing images of eight roundabouts along with the location of the vehicles. This has been accomplished using a methodology that simplifies the task of labeling, as it is a mainly manual and time-consuming task.

## 2. Data Description

### 2.1. Dataset Summary

The dataset is constituted of 985,260 instances in 61,896 color images (15,474 real images and 46,422 created using data augmentation techniques) in JPG format, each of which is complemented by an XML (Extensible Markup Language) file, according to the PASCAL VOC (Visual Object Classes) format, with the annotations of the location of the vehicles within them. The images have been taken from eight different roundabouts with different traffic flow conditions. Table 1 shows a breakdown of vehicles obtained from each of them, and Figures 1–3 show some examples.

**Table 1.** Training params and result (the first column indicates the number of the roundabout and the initial text of each file corresponding to that one).

Roundabout (Videos Names)	Number of Images	Car	Truck	Cycle	Bus	Empty
1 (00001)	1996	34,558	0	4229	0	0
2 (00002)	514	743	0	0	0	157
3 (00003–00017)	1795	4822	58	0	0	0
4 (00018–00033)	1027	6615	0	0	0	0
5 (00034–00049)	1261	2248	0	550	0	81
6 (00050–00052)	5501	180,342	1420	120	1376	0
7 (00053)	2036	5789	562	0	226	92
8 (00054)	1344	1733	222	0	150	222
Total	15,474	236,850	2262	4899	1752	552
Data augmentation	×4	×4	×4	×4	×4	×4
Total after augmentation	61,896	947,400	9048	19,596	7008	2208

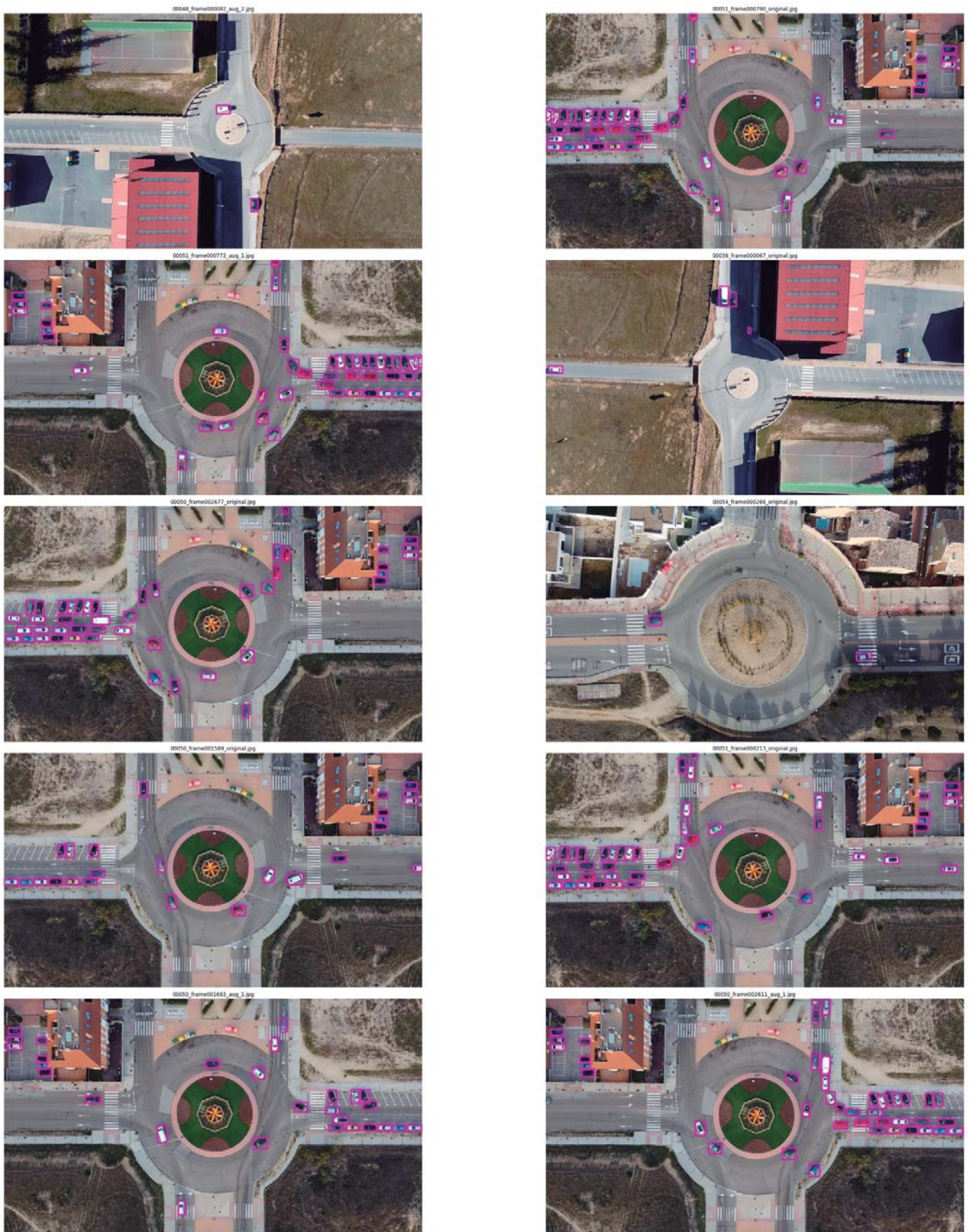


Figure 1. Dataset examples (part 1).

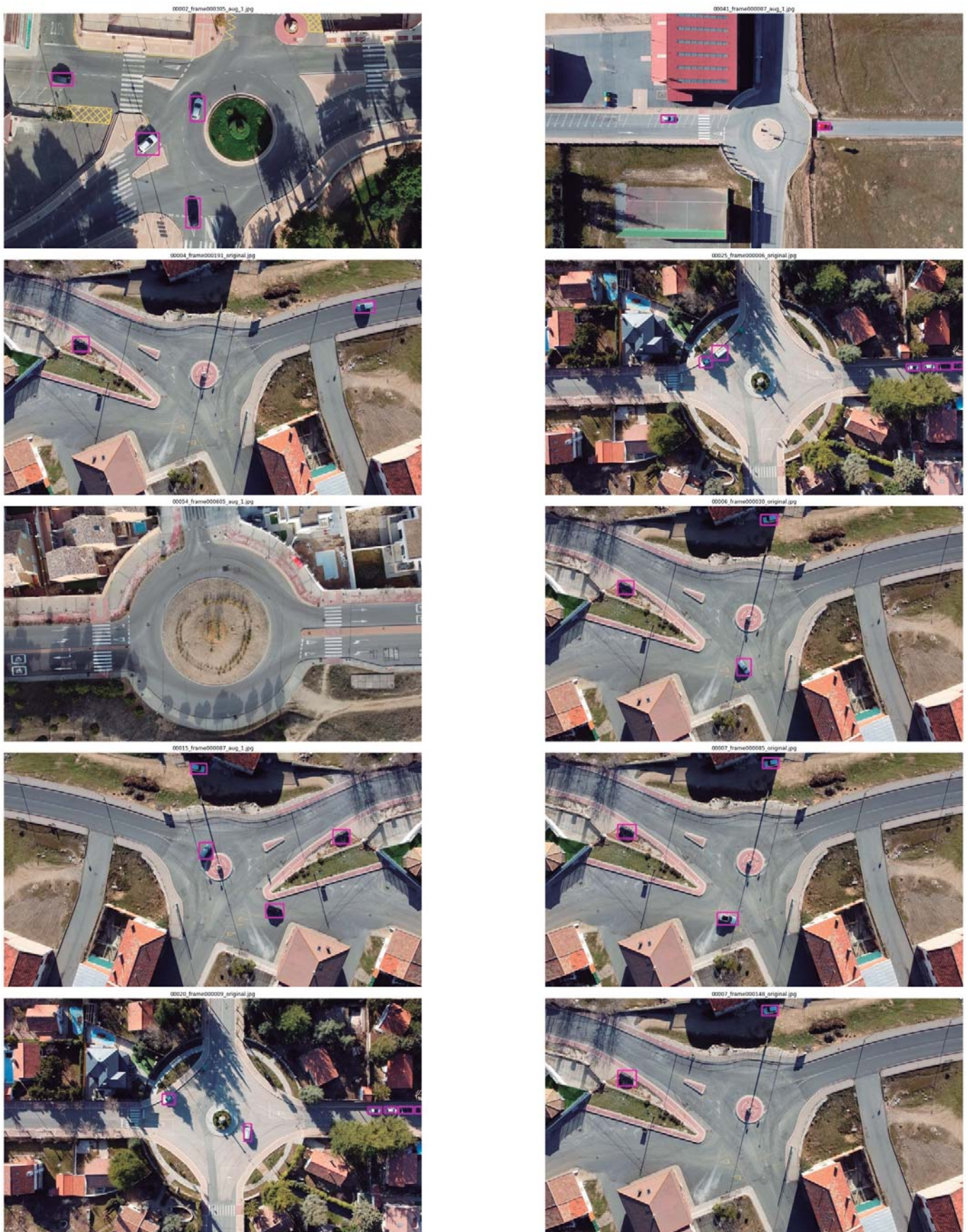


Figure 2. Dataset examples (part 2).



Figure 3. Dataset examples (part 3).

## 2.2. Annotations File

The structure of this file is as follows:

```
<annotation>
  <folder></folder>
  <filename></filename>
  <path></path>
  <source>
    <database></database>
  </source>
  <size>
    <width></width>
    <height></height>
    <depth></depth>
  </size>
  <segmented></segmented>
  <object>
    <name></name>
    <pose></pose>
    <truncated></truncated>
    <difficult></difficult>
    <bndbox>
      <xmin></xmin>
      <ymin></ymin>
      <xmax></xmax>
      <ymax></ymax>
    </bndbox>
  </object>
</annotation>
```

Although PASCAL VOC offers many fields, the ones used in the dataset are the following:

- **folder:** Indicates the folder where the images are located.
- **filename:** Name and extension of the image file to which the annotation file refers.
- **path:** Absolute path of the image file after annotation.
- **size:** Size in pixels and number of channels. Color images have three channels, while black and white images have one channel.
- **object:** It contains the data related to the object located in the image. This label and its contents are repeated for every single object located.
  - **name:** Object class name.
  - **bndbox:**
    - **xmin:** x-coordinate top left corner.
    - **ymin:** y-coordinate top left corner.
    - **xmax:** x-coordinate bottom right corner.
    - **ymax:** y-coordinate bottom right corner.

## 2.3. Folder Contents

The folder structure of the dataset is as follows:

- roundabout\_dataset/
  - data.csv
  - roundabouts.csv
  - original/
    - imgs/
    - annotations/

- o aug\_0/
  - imgs/
  - annotations/
- o aug\_1/
  - imgs/
  - annotations/
- o aug\_2/
  - imgs/
  - annotations/

In which:

- **data.csv:** Each row contains the following information separated by commas (,): image\_name, x\_min, y\_min, x\_max, y\_max, class\_name.
- **roundabouts.csv:** scene, id\_roundabout, lat, long, height\_(meters), with\_zoom\_height\_(meters).
- **imgs:** Image files in .jpg format.
- **annotations:** Annotation files in .xml format.

The name of the image and annotation files (original and data augmentation ones) follow the following patterns:

```
<scene>_frame<num_frame>_original.xml
<scene>_frame<num_frame>_original.jpg
<scene>_frame<num_frame>_aug_<num_aug>.xml
<scene>_frame<num_frame>_aug_<num_aug>.jpg
```

### 3. Methodology

The annotation of images is a tedious task, which is why a methodology that saves part of the task has been chosen. Figure 4 summarizes the process. It consists of annotating the minimum number of images to train CNN models to auto-annotate as many cases as possible. Although these require revisions, this avoids a lot of manual annotations. In addition, to increase the number of instances without having to annotate any, data augmentation techniques are applied to create apparently new images.

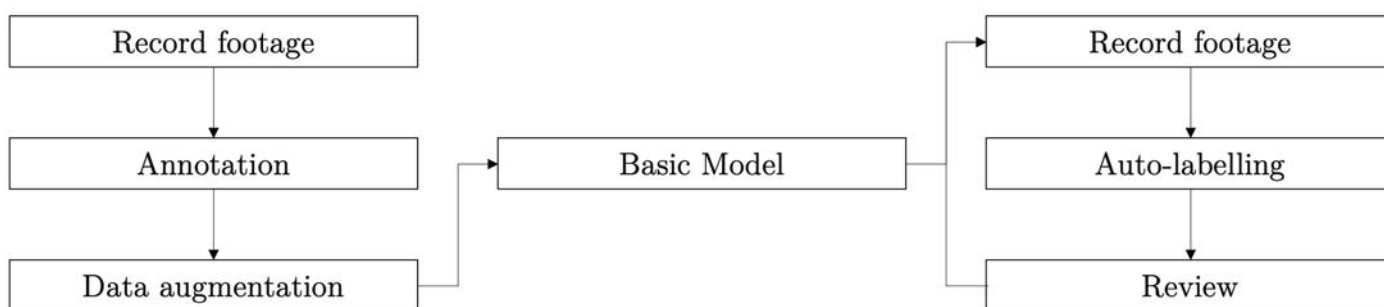


Figure 4. Methodology followed to obtain the dataset.

**Record road footage.** The first task is to collect some aerial videos of roundabouts and remove those with poor quality. These were taken during daylight, at different heights (indicated in the file roundabouts.csv) with sunny and cloudy conditions, using a DJI Mavic Mini 2 drone whose specifications can be found in [33] and in Table 2. The heights used to keep the roundabout in the center of the image are between 100 and 120 m so that it can be clearly seen with its entrances and exits. For that range of heights, the camera obtained a resolution (ground sampling distance—GSD) with values between 6.67 and 8 cm per image pixel, as also shown in Table 2. The footages were recorded in compliance with civilian regulations for the use of remotely piloted aircraft [34].

**Table 2.** DJI Mavic Mini 2 specifications.

Component	Specification
Size	1920 × 1080 px
GSD (100 m height)	6.67 cm/px
GSD (110 m height)	7.27 cm/px
GSD (120 m height)	8.00 cm/px
FOV angle	83°
Focal (35 mm)	24 mm
Aperture	f/2.8
Aspect Ratio	19:9
Sensor	1/2.3" CMOS

**Annotation.** Once the first videos are recorded, several frames are extracted using a Python script and manually annotated using software [35] (experimentally, every 10 frames, the roundabout image is different enough to be considered a new instance). This generates an XML file in PASCAL VOC format for each image.

**Data augmentation.** Once the images are annotated using a Python script and the OpenCV library [36], synthetic images are created by applying different flips (horizontal, vertical, and both at the same time). This is a technique widely used to create seemingly new examples with the least amount of work [37].

**Basic model.** The next step is to create the basic model, which is trained using [38]. The selected model is a RetinaNet [7], a one-step CNN that has already proven its effectiveness [39] for this task [40], with a Resnet 50 backbone pretrained with the COCO dataset. The mean average precision (mAP) has been established as the parameter to be optimized. This metric is very suitable, as it considers the entire precision-recall graph, unlike others such as the f1-score. The mAP is the mean AP of all classes and is defined as the area under the precision-accuracy graph (1), where precision and recall scores are calculated using (2) and (3), respectively. To obtain TP (true positive), FP (false positive), and FN (false negative), 0.5 has been set as the IoU (intersection over union), which is the minimum overlap between the true and predicted bounding box to consider a positive detection.

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

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

Tables 3 and 4 show, respectively, the hardware used, the training parameters, and the result obtained.

**Table 3.** Hardware used for training.

Component	Name
Processor	Intel i7 9800K 3.6 GHz
Motherboard	MPG Z390 Gaming Pro Carbon
RAM	32 GBs
Graphics card	Nvidia RTX 2080 Ti
Hard disk	500 Tb SSD M2
OS	Ubuntu 18.04.4 LTS



**Table 4.** Training params and results.

Metric	Value
Num. images	2020
Batch size	2
Steps	Auto
Backbone	Resnet 50
Learning rate	$10^{-5}$
Train/validation split	80–20%
IoU	0.5
Freeze backbone	True
Total epochs	6
Final mAP	0.9879

**More instances.** Using the model, new images are annotated by another iterative process that involves even less manual work. This would be: (1) record new videos, extract frames, and remove those with poor quality, (2) predict the location of the vehicles using the model, (3) review and confirm the images and predictions, and (4) use the data augmentation script to increase the size of the dataset.

#### 4. Data Quality

For data quality assurance, the same RetinaNet model used to auto-label new instances has been trained, with the difference that the entire created dataset has been used. This has been divided into a training (70%) and validation (20%) set, plus an evaluation set (10%), to test the model once trained. Table 5 shows the results obtained from the training, and Table 6 shows the results obtained from the evaluation split of the dataset.

**Table 5.** Model validation training params.

Metric	Value
Num. images	61,896
Batch size	2
Steps	Auto
Backbone	Resnet 50
Learning rate	$10^{-5}$
Train/validation/test split	70%/20%/10%
IoU	0.5
Freeze backbone	True
Total epochs	4
Final mAP	0.9622

**Table 6.** Model validation training results divided by minimum score and IoU (AP@.50 and AP@.75 means AP with .50 and .75 IoU).

Class	Minimum Score = 0.05		Minimum Score = 0.5	
	AP@.50	AP@.75	AP@.50	AP@.75
Car	0.9992	0.9920	0.9987	0.9916
Cycle	0.8791	0.7816	0.8485	0.7721
Truck	0.9991	0.9856	0.9960	0.9836
Bus	0.9955	0.9720	0.9866	0.9645
Weighted AP	0.9969	0.9879	0.9957	0.9872
mAP	0.9682	0.9328	0.9574	0.9280

#### 5. Conclusions

As shown in Table 6, the dataset is good enough to train a model that generalizes correctly. Among all the classes, motorcycles have the least AP; this is explained by the fact

that their size is much smaller than the rest of the vehicles. Publications such as [15,41] show how increasing the image resolution causes a better generalization of all the classes. Rescaling the images could be a solution.

As future work, it would be interesting to record footage in poor visibility conditions, such as at night, heavy rain, or snowfall. However, this dataset offers the necessary tools to train models of vehicle recognition in roundabouts. In addition, these images could even be used to generate other datasets with annotations of other objects.

**Author Contributions:** Conceptualization, G.D.-L.-H., J.S.-S. and E.P.; methodology, J.S.-S. and G.D.-L.-H.; software, G.D.-L.-H.; validation, G.D.-L.-H., J.S.-S. and E.P.; formal analysis, G.D.-L.-H., J.S.-S. and E.P.; investigation, G.D.-L.-H. and J.S.-S.; resources, G.D.-L.-H., J.S.-S. and E.P.; data curation, G.D.-L.-H.; writing—original draft preparation, G.D.-L.-H., J.S.-S., and E.P.; writing—review and editing, G.D.-L.-H., J.S.-S., E.P. and J.F.-A.; visualization, G.D.-L.-H.; supervision, J.S.-S. and E.P.; project administration, J.S.-S.; funding acquisition, J.F.-A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This publication is part of the I+D+i projects with reference PID2019-104793RB-C32, PIDC2021-121517-C33, funded by MCIN/AEI/10.13039/501100011033/, S2018/EMT-4362/“SEGVA-UTO4.0-CM” funded by Regional Government of Madrid and “ESF and ERDF A way of making Europe”.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are openly available at <https://doi.org/10.5281/zenodo.6362360> with DOI: 10.5281/zenodo.6362360.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Soviany, P.; Ionescu, R.T. Optimizing the Trade-Off between Single-Stage and Two-Stage Deep Object Detectors using Image Difficulty Prediction. In Proceedings of the 2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNAS), Timisoara, Romania, 20–23 September 2018; pp. 209–214.
2. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
3. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. *arXiv* **2017**, arXiv:1612.08242v1.
4. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
5. Bochkovskiy, A.; Wang, C.-Y.; Liao, H.-Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
6. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
7. Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327. [[CrossRef](#)] [[PubMed](#)]
8. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013.
9. Girshick, R. Fast R-CNN. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
10. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1137–1149. [[CrossRef](#)] [[PubMed](#)]
11. Elkhachy, I. Accuracy Assessment of Low-Cost Unmanned Aerial Vehicle (UAV) Photogrammetry. *Alex. Eng. J.* **2021**, *60*, 5579–5590. [[CrossRef](#)]
12. Dijkstra, K.; van de Loosdrecht, J.; Schomaker, L.R.B.; Wiering, M.A. Hyperspectral demosaicking and crosstalk correction using deep learning. *Mach. Vis. Appl.* **2019**, *30*, 1–21. [[CrossRef](#)]
13. Gupta, A.; Watson, S.; Yin, H. Deep learning-based aerial image segmentation with open data for disaster impact assessment. *Neurocomputing* **2021**, *439*, 22–33. [[CrossRef](#)]
14. Shen, J.; Liu, N.; Sun, H. Vehicle detection in aerial images based on lightweight deep convolutional network. *IET Image Processing* **2021**, *15*, 479–491. [[CrossRef](#)]
15. Stuparu, D.-G.; Ciobanu, R.-I.; Dobre, C. Vehicle Detection in Overhead Satellite Images Using a One-Stage Object Detection Model. *Sensors* **2020**, *20*, 6485. [[CrossRef](#)]
16. Liu, K.; Mattyus, G. Fast Multiclass Vehicle Detection on Aerial Images. *IEEE Geosci. Remote Sens. Lett.* **2015**, *12*, 1938–1942.

17. Zhong, J.; Lei, T.; Yao, G. Robust Vehicle Detection in Aerial Images Based on Cascaded Convolutional Neural Networks. *Sensors* **2017**, *17*, 2720. [[CrossRef](#)]
18. Tang, T.; Zhou, S.; Deng, Z.; Zou, H.; Lei, L. Vehicle detection in aerial images based on region convolutional neural networks and hard negative example mining. *Sensors* **2020**, *20*, 336. [[CrossRef](#)] [[PubMed](#)]
19. Deng, Z.; Sun, H.; Zhou, S.; Zhao, J.; Zou, H. Toward fast and accurate vehicle detection in aerial images using coupled region-based convolutional neural networks. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2017**, *10*, 3652–3664. [[CrossRef](#)]
20. Yu, Y.; Gu, T.; Guan, H.; Li, D.; Jin, S. Vehicle detection from high-resolution remote sensing imagery using convolutional capsule networks. *IEEE Geosci. Remote Sens.* **2019**, *16*, 1894–1898. [[CrossRef](#)]
21. Chen, Z.; Wang, C.; Wen, C.; Teng, X.; Chen, Y.; Guan, H.; Luo, H.; Cao, L.; Li, J. Vehicle detection in high-resolution aerial images via sparse representation and superpixels. *IEEE Trans. Geosci. Remote Sens.* **2015**, *54*, 103–116. [[CrossRef](#)]
22. Kembhavi, A.; Harwood, D.; Davis, L.S. Vehicle detection using partial least squares. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *33*, 1250–1265. [[CrossRef](#)]
23. Cheng, G.; Wang, Y.; Xu, S.; Wang, H.; Xiang, S.; Pan, C. Automatic Road Detection and Centerline Extraction via Cascaded End-to-End Convolutional Neural Network. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3322–3337. [[CrossRef](#)]
24. Chang, Y.-C.; Huang, C.; Chuang, J.-H.; Liao, I.-C. Pedestrian Detection in Aerial Images Using Vanishing Point Transformation and Deep Learning. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 1917–1921.
25. Soleimani, A.; Nasrabadi, N.M. Convolutional Neural Networks for Aerial Multi-Label Pedestrian Detection. In Proceedings of the 2018 21st International Conference on Information Fusion (FUSION), Cambridge, UK, 10–13 July 2018; pp. 1005–1010.
26. Rodríguez, J.I. Cómo circular por una glorieta. *Tráfico Segur. Vial* **2014**, *228*, 28–30.
27. Cuenca, L.G.; Sanchez-Soriano, J.; Puertas, E.; Andrés, J.F.; Aliane, N. Machine Learning Techniques for Undertaking Roundabouts in Autonomous Driving. *Sensors* **2019**, *19*, 2386. [[CrossRef](#)]
28. Cuenca, L.G.; Puertas, E.; Andrés, J.F.; Aliane, N. Autonomous Driving in Roundabout Maneuvers Using Reinforcement Learning with Q-Learning. *Electronics* **2019**, *8*, 1536. [[CrossRef](#)]
29. Breuer, A.; Termöhlen, J.-A.; Homoceanu, S.; Fingscheidt, T. OpenDD: A Large-Scale Roundabout Drone Dataset. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–6.
30. Bock, J.; Krajewski, R.; Moers, T.; Runde, S.; Vater, L.; Eckstein, L. The inD Dataset: A Drone Dataset of Naturalistic Road User Trajectories at German Intersections. In Proceedings of the 2020 IEEE Intelligent Vehicles Symposium (IV), Las Vegas, NV, USA, 19 October–13 November 2020; pp. 1929–1934.
31. Krajewski, R.; Bock, J.; Kloeker, L.; Eckstein, L. The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 2118–2125.
32. Krajewski, R.; Moers, T.; Bock, J.; Vater, L.; Eckstein, L. The roundD Dataset: A Drone Dataset of Road User Trajectories at Roundabouts in Germany. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–6.
33. DJI. DJI Mini 2. Available online: <https://www.dji.com/ca/mini-2/specs> (accessed on 8 February 2022).
34. Ministerio De La Presidencia, Para Las Administraciones Territoriales. Boletín Oficial Del Estado. 29 December 2017. Available online: <https://www.boe.es/boe/dias/2017/12/29/pdfs/BOE-A-2017--15721.pdf> (accessed on 14 March 2022).
35. Tzotalin. LabelImg. GitHub. Available online: <https://github.com/tzotalin/labelImg> (accessed on 11 January 2021).
36. OpenCV. OpenCV. Available online: <https://docs.opencv.org/4.x/index.html> (accessed on 11 January 2022).
37. Zoph, B.; Cubuk, E.; Ghiasi, G.; Lin, T.; Shlens, J.; Le, Q. Learning Data Augmentation Strategies for Object Detection. In Proceedings of the Computer Vision–ECCV 2020, Glasgow, UK, 23–28 August 2020; pp. 566–583.
38. Fizyr. Keras RetinaNet. Available online: <https://github.com/fizyr/keras-retinanet> (accessed on 23 January 2022).
39. Hui, J. Object Detection: Speed and Accuracy Comparison (Faster R-CNN, R-FCN, SSD, FPN, RetinaNet and YOLOv3). Available online: <https://jonathan-hui.medium.com/object-detection-speed-and-accuracy-comparison-faster-r-cnn-r-fcn-ssd-and-yolo-5425656ae359> (accessed on 11 January 2022).
40. De-Las-Heras, G.; Sánchez-Soriano, J.; Puertas, E. Advanced Driver Assistance Systems (ADAS) Based on Machine Learning Techniques for the Detection and Transcription of Variable Message Signs on Roads. *Sensors* **2021**, *21*, 5866. [[CrossRef](#)] [[PubMed](#)]
41. Shermeyer, J.; van Etten, A. The Effects of Super-Resolution on Object Detection Performance in Satellite Imagery. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Long Beach, CA, USA, 16–17 June 2019; pp. 1432–1441.