




Aerial Image Instance Segmentation Through Synthetic Data Using Deep Learning

Felipe X. Viana, Gabriel M. Araujo , Milena F. Pinto , Jefferson Colares, Diego B. Haddad 

Federal Center of Technological Education of Rio de Janeiro - CEFET-RJ

ximenesfel@gmail.com, {gabriel.araujo, milena.pinto, diego.haddad}@cefet-rj.br, jefferson.colares@eic.cefet-rj.br

Abstract – In the last decades, current trends in autonomous navigation have demonstrated an increased use of computational vision over traditional techniques. This relies on the fact that most of the spaces are designed for human navigation. As a result, they are filled with visual cues. In this sense, visual recognition is an essential ability to avoid obstacles when an autonomous vehicle interacts with the real world. Data collection using Unmanned Aerial Vehicles (UAVs) navigating in a real-world scenario is a high-cost and time-expensive activity. For this reason, one of the most valuable assets of technology companies is a database containing locations and interactions. One solution to this problem is the adoption of a photo-realistic 3D simulator as a data source. Using this resource, it is possible to gather a significant amount of data. Therefore, this research creates a dataset for instance segmentation using images from a frontal UAV camera navigating in a 3D simulator. This work applies a state-of-the-art deep learning technique, the Mask-RCNN. The architecture takes an image input and predicts per-pixel instance segmentation. Experimental results showed that Mask RCNN has superior performance in our dataset when refining a model trained using COCO dataset. Besides, the proposed methodology presents a good generalization capability due to the promising results in real-world data.

Keywords – Computer Vision, Deep Learning, Instance Segmentation, 3D Simulator, Aerial Images.

1. INTRODUCTION

Autonomous vehicles are increasingly present in many applications, such as in civil engineering [1] and robotics [2]. A common way for this kind of system to perceive their environment is through vision [3]. Thus, visual recognition is becoming a more demanded application to work in these environments, ensuring a safe and optimized operation. Among visual recognition tasks, the capability of detecting and avoiding obstacles is essential in real environments. However, it may not be sufficient only to identify the obstacles in the mission. The robot also needs to recognize the shape and boundaries from these obstacles to distinguish among viable paths. A simple example would be a bag of plastic in the middle of the road. In this case, the laser scan or stereo odometry could detect it, although those techniques are not able to distinguish the plastic bag from a rock or other types of dangerous objects. As a result, their use may force the car to stop or change lanes, inadvertently exposing human lives to potential danger.

In recent years, the growing interest in autonomous driving, as well as the development of machines able to interact with people, is playing an important role in the improvement of the techniques for vision-based tracking machines [4–7]. However, autonomous systems capable of interacting in not-controlled environments represent a high risk for humans that may be in their range of action. Instance segmentation, which is a computer vision field that combines object detection and semantic segmentation, can aid in avoiding such a risk [8]. Training such a computer vision system requires a considerable amount of data, which can be challenging to acquire in many applications. So, employing synthetic data from a simulator can be an alternative. The main advantage of this approach is that a simulator can provide greater control of the environment. Thus, variations that would be much more complex in the real world can be simply realized by the simulator, e.g., obtaining data representing daytime variations can be done by simulating such conditions. As stated in [9], this data diversification enhances network generalization abilities. This kind of technique introduces new promises for transferring learning acquired in simulators using synthetic data to real-world scenarios. The use of synthetic data can benefit many different methods for computational vision. For instance, conventional methods for object detection are based on features sets, such as Bag-of-Words [10] or shallow networks [11]. The efficiency of these methods is restricted by simple information about the object type and rough estimates of their location and boundaries.

Deep learning represents a strong alternative to new computer vision tasks. This technique is a promising solution to the shortcomings of classic computer vision, which is possible to observe from the recent advances in the applications of deep learning to solve complex real-world problems. Deep learning for computer vision grows popular every year by the fact that Convolutional Neural Networks (CNN) are able to learn descriptors from images for a broad range of tasks, such as classification, segmentation, detection. In relation to these advances in deep learning techniques, additional tools have been proposed to overcome some of the existing limitations in traditional methods. The Region-based Convolutional Neural Networks (R-CNN) is an algorithm that takes advantage of an attention mechanism similar to the mechanism on a human brain [12]. This method works in the following way. First, the method performs a fast scan of the whole scene, and then, performs a detailed analysis of Regions of Interest (ROI). The R-CNN chooses regions that potentially include objects, adopting a selective search algorithm to generate the proposal regions for an input image. As an improvement to the R-CNN architecture, Girshick et al. [13] presented

the Fast R-CNN algorithm. In this architecture, the entire input image is applied to a CNN, and the feature map is extracted from it. An improved method to obtain ROI pooling removes the need for a fixed window from the feature map. By including an additional component to the R-CNN architecture, Ren et al. [14] presented the Faster R-CNN architecture, which includes a Region Proposal Network (RPN). The RPN primary objective is to eliminate selective inference by combining the whole process into the R-CNN. The work of He et al. [15] developed the Mask R-CNN architecture that works towards the instance segmentation problem. Mask R-CNN is an extension of Faster R-CNN that predicts a binary mask for each ROI.

The main contribution of this work is the application of instance segmentation to detect cars and pedestrians in synthetic data from a UAV photo-realistic simulator. The Airsim software [16] was used to create an annotated dataset for instance segmentation task using synthetic data applicable for cars and people detection. An implementation of Mask-RCNN, pre-trained with COCO dataset, was refined to our training set. The resulting model has excellent performance in our dataset and promising results in real-world data, showing the good generalization capability of the proposed strategy.

The remainder of this research work is organized as follows. Section 2 presents the background and related works in the area. Section 3 details the methodology and its foundations. Section 4 shows the proposed experiments with a proper discussion of the results. The concluding remarks and future works are conducted in Section 5.

2. Background and Related Works

Object detection is one of the most important tasks in computer vision research area. This growing interest might be explained by the potential applications that could use the results of object detection (*e.g.*, computer vision has been expected to play a larger role within autonomous systems and in surveillance applications). However, it is typically used along with other sensors due to its history of relatively low precision. For instance, the authors in [17] adopted a Kalman filter to fuse data from multiple sensors, such as LIDAR, radar, and computer vision. However, cameras generate a richer set of features at a fraction of the cost compared to other kinds of sensors, such as sonar and radar.

Deep learning represents a robust approach to computer vision. In the last years, with innovative techniques based on Deep Learning, this technique has been applied in different situations and aims [18–20]. Besides, it is a powerful tool since architectures that make use of deep classifiers, such as CNNs, overcame the necessity of feature extraction steps from the segmented ROIs. Figure 1 depicts the steps needed for object classification using a feature-based model and deep learning strategy. Note that in Figure 1 (a), the process of vision-based object detection is constituted by three fundamental steps, *i.e.*, image acquisition, feature extraction, and classification. However, deep structures allowed the removal of the feature extraction step (Figure 1 (b)). Note that the feature extraction is an automatic procedure performed by the deep classifier that takes images as input and computes features at different layers of abstraction [21].

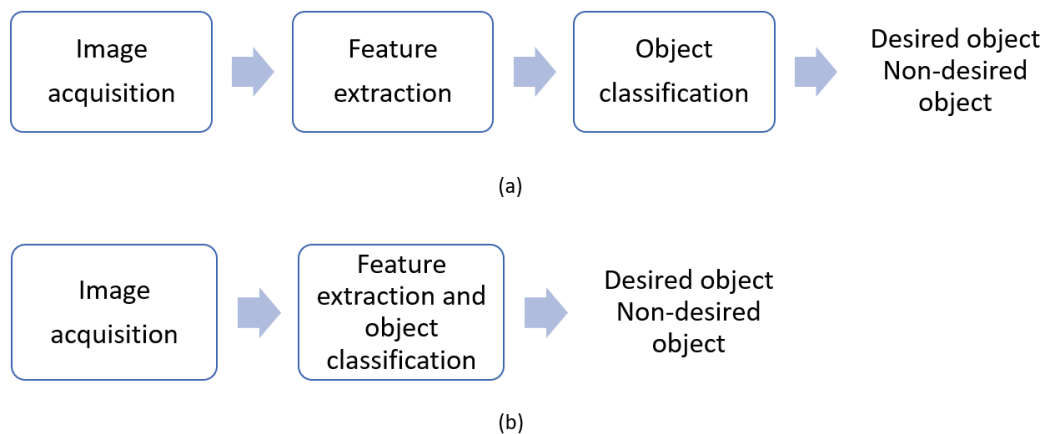


Figure 1: Steps needed for object classification. (a) Features-Based Model. (b) Deep Learning Strategy.

Among the variety of traceable objects, automatic tracking of humans in the video has always been an interesting research topic. However, human detection and segmentation are challenging due to the large variety of conditions and well-known problems related to image segmentation, such as noise [22]. In [23], the authors proposed a robust people detection method in aerial images based on Cluster Boosting Tree (CBT) and Integral Channel Features (ICF). The authors also have proposed a new dataset (GMVRT-v2) to capture a large variety of viewpoints. The work in [24] proposed a Convolutional Neural Network (CNN) to perform people detection in embedded computers in a UAV. The UAV uses two cameras, an RGB and a thermal one. The system was trained using GMVRT-v2 and a set of thermal images acquired through a FLIR Lepton camera. The authors have shown that the combination of thermal and regular cameras has promising results in execution time. Other research fields that present growing interest in automatic tracking of humans in video sequences are the human-computer interaction and augmented reality [25–27].

An essential procedure for image analysis is segmentation. Semantic segmentation associates each pixel of an image with the most likely class label from a finite set of possible labels. For aerial images, it is not a novel subject. Note that in the last years,

much research effort has been applied to the semantic segmentation of satellite and aerial images. The work in [28] describes a method based on classical techniques to perform segmentation in images from Google Earth. The method uses Superpixel [29] to obtain an oversegmented map. An oversegmented map is a segmentation of a figure based on consistent properties such texture, color or intensity as shown in [29]. The Local Binary Pattern Histogram Fourier features (LBP-HF) and color histograms of RGB images are used to get texture and color descriptors, respectively, of each Superpixel. Then each Superpixel is classified into one of five classes using k-NN, which are building, road, grass, tree, or water. In [30], the authors found a method to perform instance segmentation to detect vehicles in aerial images. The authors employed a well known residual CNN, ResNet, to perform detection in ISPRS and IEEE GRSS DFC2015 datasets, They also proposed the “Busy Parking Lot Unmanned Aerial Vehicle Video” dataset.

Bounding boxes are a more straightforward method of scene understanding for determining a different object in an image. Nevertheless, this method has its limits. Some objects, such as a curvy road or sky, can rapidly become problematic or even impossible to convey the exact shape of the object. Using bounding boxes, we only achieve partial scene understanding. Semantic segmentation is a task that assigns meaning to a part of an object. Using this task, we obtain fine-grained inference by inferring labels to every pixel that acquires essential information about each pixel in the image rather than just slicing sections into bounding boxes [31] [32] [33]. Significant improvements of dense classification through semantic segmentation have gone thanks to the use of deep learning [34]. New architectures derived from Fully Convolutional Networks (FCN) have appeared, and they can output dense pixel-wise annotations to achieve fine-grained classification, such as the one presented in [35]. The authors proposed a deep learning approach to the semantic segmentation of very high-resolution aerial images. They designed a that takes as input intensity and range data. Then, through a deconvolution and recycling of early network layers, it converts the inputs into a pixel-wise classification at a full resolution.

Vehicle detection and classification are subjects that have been widely investigated in the computer vision community. The work of [36] proposed a method to detect vehicles in high-resolution aerial images (Toronto dataset). It also used Superpixel to obtain an oversegmented map. Patches centered at each Superpixel are extracted, and grids of HOG descriptors of each patch are computed. In the training stage, these descriptors are used to compose a sparse representation dictionary. In the detection stage, the HOG features are classified using sparse codes and the final dictionary. However, this approach does not have a good performance when the dictionary has a large number of items. Also, HOG features do not have color or contextual information. [37] further improved this work. The feature set was extended to include color through a transformation over RGB color space, texture (HOG), and context (pre-trained small dictionaries using multiple classes of the original samples). The work in [37] also evaluated their method in Toronto dataset but used the ORIDS dataset as well.

The work in [38] describes a method to perform people and cars detection in images taken from a UAV. The system uses a CNN to detect in an NVidia Jetson Tx2 graphic card. It was trained using COCO dataset and has promising results in new images from a UAV taken during a low altitude flight. However, as can be seen, in [39], low altitude aerial image datasets for people detection are few. So, the authors proposed a data augmentation method that transforms general images to have an aerial-like aspect. They have shown that the use of data augmentation can aid in improving the detection results when CNN is employed to perform detection in aerial images.

Another important aspect regards the methodology assessment. Developing and testing algorithms for autonomous vehicles operating in real-world scenarios is an expensive and time-consuming procedure. This process requires an infrastructure to build costly hardware systems, obtain an extensive amount of annotated training data in a range of conditions and environments, and the ability to rapidly test and benchmark the results. An autonomous vehicle simulator can obtain a large amount of data in the simulator to train a deep learning model. After training, the model can be ported to real-world applications.

Airsim, developed by [40], is a photo-realistic simulator with detailed 3D environments. Airsim was designed as a plugin for Unreal Engine, a tool for game development, which means that the autonomous vehicle simulation is decoupled from the environment where it is running. This simulator provides the sensor data from the simulated world to the autonomous vehicle controller. The controller outputs the actuator signals that the autonomous vehicle model in the simulator takes as input. The autonomous vehicle model navigates through the environment capturing data for training a deep learning model. Another relevant data obtained from the simulator is the current RGB image from the camera coupled to the autonomous vehicle. Note that the user can change cameras' parameters and resolution. This flexibility is an essential requirement for creating datasets for deep learning applications. In this paper, the synthetic data is used to train an instance segmentation model using an environment that simulates a city. The used technique is Mask R-CNN, a state of art deep learning model for instance segmentation. After the training, we verify the generalization of this model through this environment.

3. Proposed Methodology for Instance Segmentation

As established in the previous section, the use of synthetic data is a viable tool for data gathering. The software Airsim provides all resources for gathering information to deep learning applications. Through the available Computer Vision mode, it is possible to navigate through the environment collecting training data. The city environment in this simulator is composed of an ample downtown space simulating a daily-basis routine. In this scenario, the cars are moving along the streets while pedestrians are walking on the sidewalk. Due to the complexity of the existing elements, only cars and people are selected as objects in this research. These objects were selected due to their importance in computer vision applications and their dynamic behavior. Airsim API has many types of available images in the simulator. However, for this research proposition, the authors collected two types of images for the city dataset, that is, RGB and semantic segmentation images. Note that the RGB image is the raw input data

for the instance segmentation model, that after training, is used for inference. Semantically segmented images can be useful in pre-processing procedures, in which it is possible to identify each object in the scene using its RGB value and boundary. Figure 2 presents a global overview of the proposed methodology. Initially, the network is trained with COCO dataset [41]. Then, the data is collected from Airsim software and annotated. Finally, the network is retrained. All the code was implemented in Python and used Keras and OpenCV libraries.

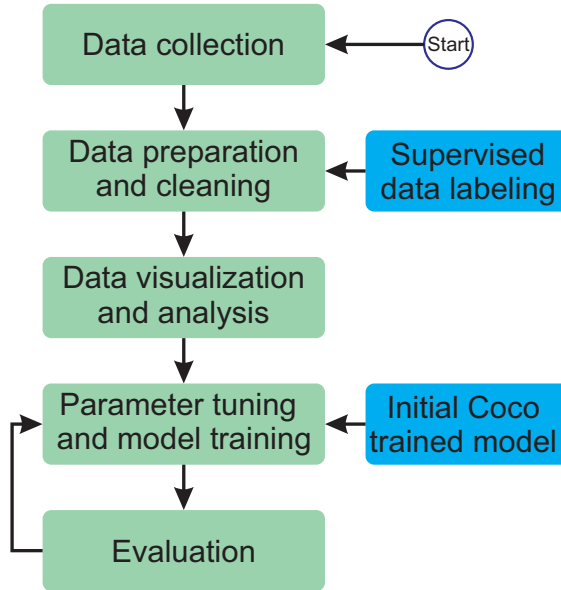


Figure 2: Proposed methodology overview.

The dataset consists of a collection of sequential frames with resolution 1024×1024 pixels from the camera of the virtual UAV. The simulator versatility allows the data collection at different heights, which is also important for the machine learning model since in real-world applications, the height can change during UAV operation. Note that the machine learning model uses weights from COCO dataset. Thus, to obtain a model with this versatility, the dataset was split into images obtained in two different altitudes:

- (1) High: altitudes greater than 10 meters represent applications of supervision and environment analysis;
- (2) Medium: altitudes between 5 and 10 meters represent applications that need more accurate information regarding the environment.

The correct ground truth of the objects and their boundaries are crucial for training a good model to perform instance segmentation. Annotation process for instance segmentation model is a complex task. Unfortunately, automatic labeling is a complicated task due to its complexity. Thus, for this process, manual annotation was performed. The simulator also provides a semantically segmented image for each RGB frame. Since the segmented image has information regarding each object in the image, including the boundaries of this object, this image is more suitable for this research purpose. As each object has a unique color in the segmented image, it is possible to use global color binarization to generate a mask containing only the objects that are useful in the image. So, only the pixels belonging to a car or a pedestrian are white in the resulting binary image. The manual annotation process is easier in the mask image since the annotator only needs to draw the boundaries of the object following the edges of the provided mask as well as to label the object correctly. Figure 3(a) and Figure 3(b) show the semantic segmentation image and a mask generation example, respectively. Figure 4 illustrates the database structure.

The image database is used in the Mask R-CNN training. In the training process, one of the most important parameters of the network is its cost function. Equation (1) describes the cost function used in this work, where p_i is the predicted probability of anchors being positive or negative. An anchor is the proposed region for the object placement. The parameter p_i^* is the ground truth label, that is, 1 for positive and 0 for the negative results. t_i represents a four-parameter coordinate vector of the predicted bounding-box, and t_i^* is the ground-truth bounding box related with a positive anchor. L_{cls} is a classification cost function regarding two classes (i.e., positive or negative). In this work, the authors used $L_{reg}(t_i, t_i^*) = R(t_i, t_i^*)$, where R is the smooth L_1 cost function [13]. The value λ is a training parameter with a default value set to 10. The values N_{cls} and N_{reg} define the number of anchors in a mini-batch and the number of locations, respectively.

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (1)$$

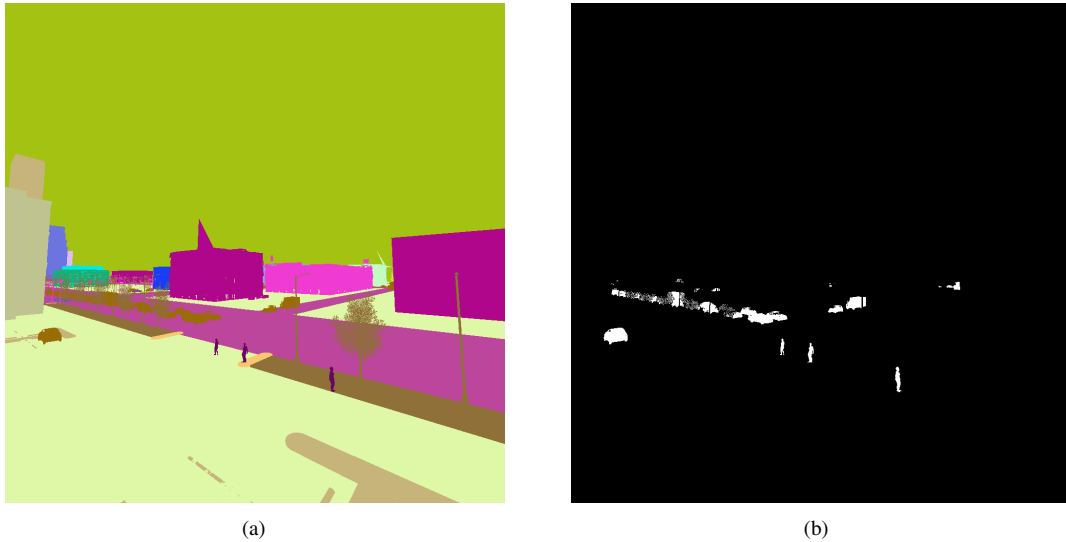


Figure 3: Mask generation. (a) The semantic segmentation image. (b) The mask result after applying color binarization algorithm in the segmentation image.

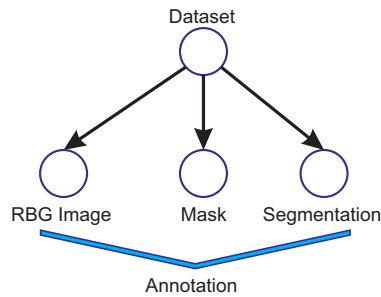


Figure 4: Database contents and its relationship.

4. RESULTS AND DISCUSSION

This work used the Matterport Mask-RCNN implementation proposed by [42] for the experimental evaluation with simplified representation shown in Figure 5. The training was performed in a computer supplied with 1 Tesla k40c GPU, 16GB DDR4 2133MHz of RAM, Intel(R) Core(TM) i7-3770 3.40 GHz processor and Ubuntu 18.04 as the operating system. Besides, a pre-trained Mask R-CNN in COCO dataset was used for instance segmentation. The training for the proposed network was performed using the SGD solver with a momentum of 0.9. The learning rate was set to 0.001.

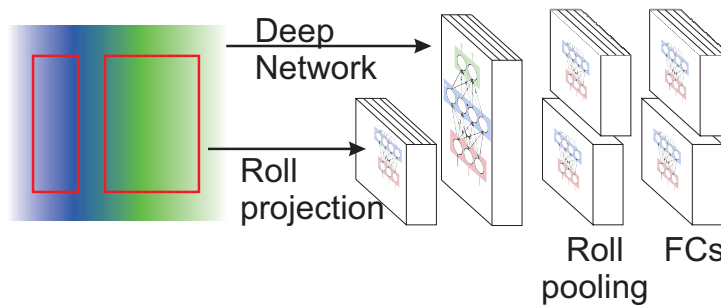


Figure 5: Mask RCNN simplified layout.

The network was evaluated on a city dataset with two performance metrics, namely intersection over union (IoU), and mean average precision (mAP). The metric IoU is a ratio between the area of overlap and the area of the union. The area of overlap corresponds to the area between the predicted bounding box and the ground-truth bounding box. The mAP is a metric that computes the accuracy per class and across all classes in the database at a given IoU value. In this work, we used values between 0.5 and 0.95 of IoU to estimate mAP. Note that both, the ground-truth bounding boxes (a hand-labeled bounding boxes) and the predicted bounding boxes of the model are necessary to compute IoU. Typically, the (x, y) coordinates of the predicted bounding boxes model do not perfectly match with the coordinates of the ground-truth bounding boxes. In this case, the metric needs

to benefit the predicted bounding boxes that heavily overlap the ground-truth. Thus, this approach ensures that the predicted bounding boxes match the ground-truth as close as possible.

The city dataset was constructed using the photo-realistic simulator Airsim, which reproduces a real 3D environment with rich features and complex objects. Among the offered environments, the City environment is constituted of a large downtown environment simulating a daily-basis routine of a city. In this scenario, we have cars moving along the streets while pedestrians were walking on the sidewalk.

The input of our instance segmentation model is an RGB image. Our pre-processing step applied color binarization to segment each frame image and generated a mask containing only objects that we are interested in the image. This conversion is essential to simplify the process of annotation. After that, we use VIA software tool to perform the annotation task (label and boundaries limits). All the annotations are done in the mask image. Thus it is possible to verify if the annotation is correctly related to the original RGB image once it will be the ground-truth data.

To perform visual verification, the authors also developed a script that loads the RGB image and the metadata in JSON, showing a mosaic containing the RGB image and the associated masks for each object detected in the image. This verification is important to analyze if the annotation correctly labeled each object in the image, and the boundaries correspond to the limits of the object in an image. Figure 6(a) and Figure 6(b) show an example of the mosaic output, where it is possible to identify both objects (i.e., car and pedestrian).

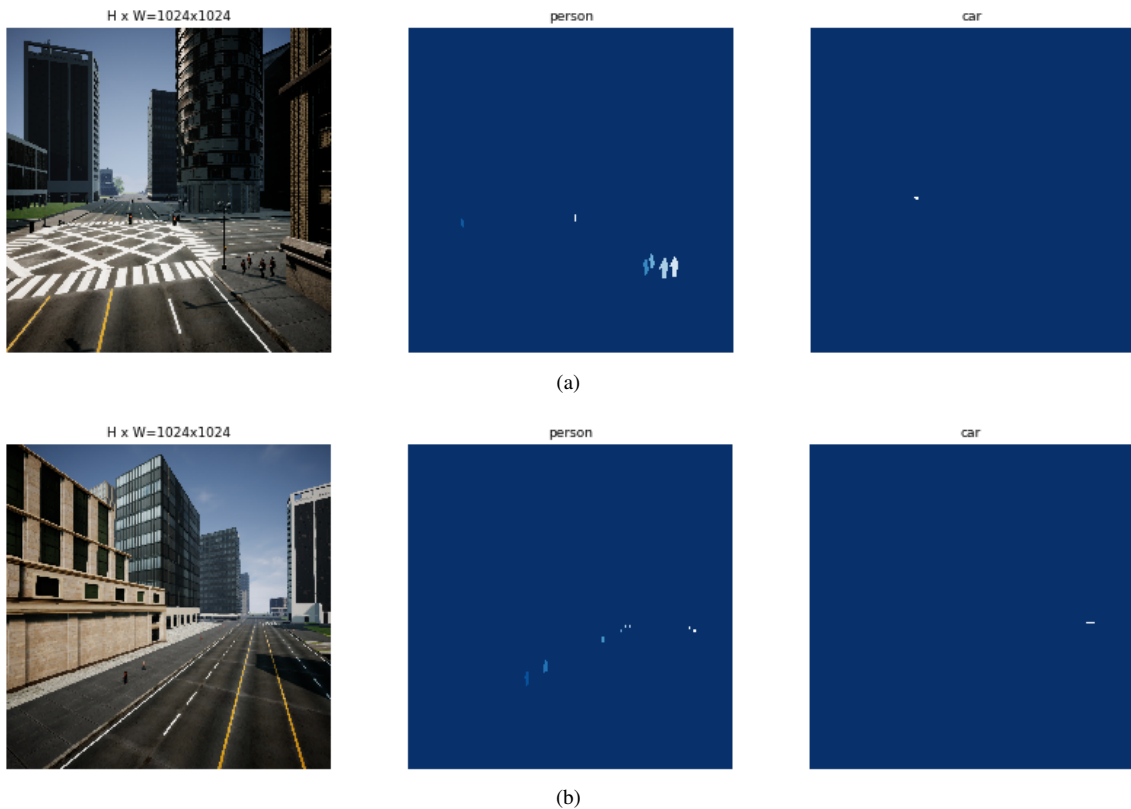


Figure 6: **Mask verification.** Verification that shows if the mask generated in VIA software was correctly identified by Mask R-CNN algorithm.

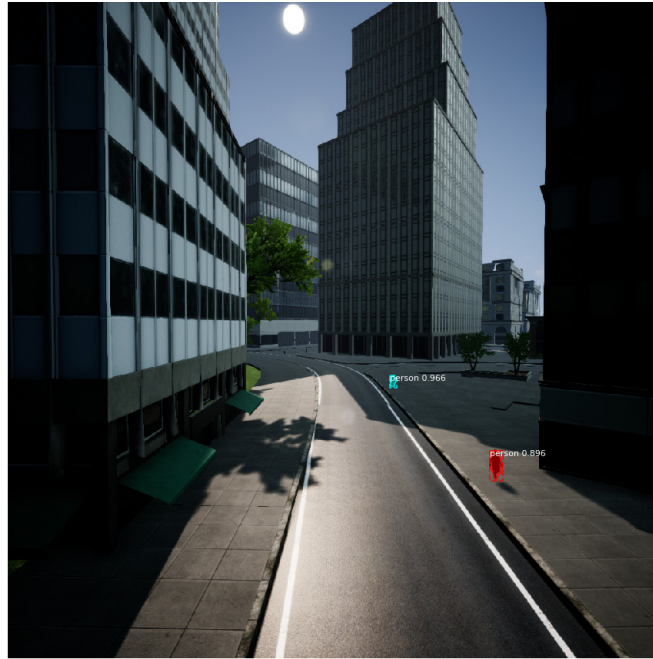
The dataset is available in [43]. Table 1 presents the comparisons for two architectures, that is, Resnet-50 and Resnet-101. Note that the Mask R-CNN with R-101-FPN structure obtained the best result. This result can be explained by the fact that networks with more deep layers are better feature extractors. Figure 7(a) and Figure 7(b) present a few predictions using the proposed model for people and cars, respectively.

Table 1: The results obtained in the city test dataset.

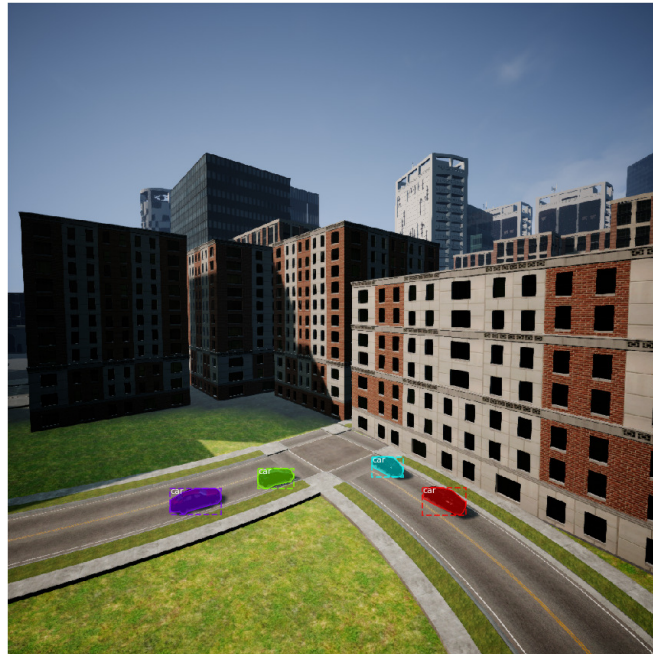
Backbone	mAP	mAP_{50}	mAP_{75}
Resnet-50-FPN	36.25	49.02	41.52
Resnet-101-FPN	39.56	54.36	50.14

Using the dataset and the model trained, it is also possible to compute the confusion matrix. This process was performed for video from Figure 7(b) and the resulting classification metrics can be seen in Table 2. The basic terminology used in the confusion matrix is given by:

- **Condition Positive (P):** The number of real positive cases in the data;



(a)



(b)

Figure 7: Images showing the Mask R-CNN model predictions. (a) People. (b) Cars.

- **Condition Negative (N):** The number of real negative cases in the data;
- **True Positive (TP):** Condition positive detected as positive;
- **True Negative (TN):** Condition negative detected as negative;
- **False Positive (FP):** Equivalent with false alarm;
- **False negative (FN):** Equivalent with miss;

It is possible to define the parameters for the confusion matrix evaluation, as shown in Table 3. Besides, it is possible to build Table 4 using those parameters and the previous values, which allow us to evaluate the algorithm capabilities during the operation.

Another experiment was conducted to verify the model capability of generalizing in a different environment. This result was evaluated using the videos from the dataset presented in [44]. This dataset has several video records taken by DJI Mavic Pro

Table 2: Confusion Matrix.

	Condition positive (CP)	Condition negative (CN)
Predicted condition positive	165 (TP)	54 (FP)
Predicted condition negative	10 (FN)	7 (TN)

Table 3: Confusion Matrix Parameters.

ACC	$\frac{\Sigma True\ Positive + \Sigma True\ Negative}{\Sigma Total\ Population}$	$\frac{TP+TN}{TP+TN+FP+FN}$
TPR	$\frac{\Sigma True\ Positive}{\Sigma Condition\ Positive}$	$\frac{TP}{TP+FN}$
FPR	$\frac{\Sigma False\ Negative}{\Sigma Condition\ Positive}$	$\frac{FP}{TN+FP}$
PPV	$\frac{\Sigma False\ Positive}{\Sigma Positive\ Predicted\ Value}$	$\frac{TP}{TP+FP}$
Prevalence	$\frac{\Sigma Condition\ positive}{\Sigma Total\ Population}$	$\frac{CP}{TP+TN+FN+FP}$

in distinct environments without ground truth data. Besides, it contains people in altitudes that the proposed model was trained for. Figure 8 shows the prediction model. Note that the model was able to perform classification, bounding box regression, and binary class prediction for two people correctly. It is also important to mention the number of frames per second processed by the algorithm. In the tests, we reached about five fps using the computer described in the first paragraph of Section 4. This means that the methods can process data in real surveillance missions in a time-bounded fashion.

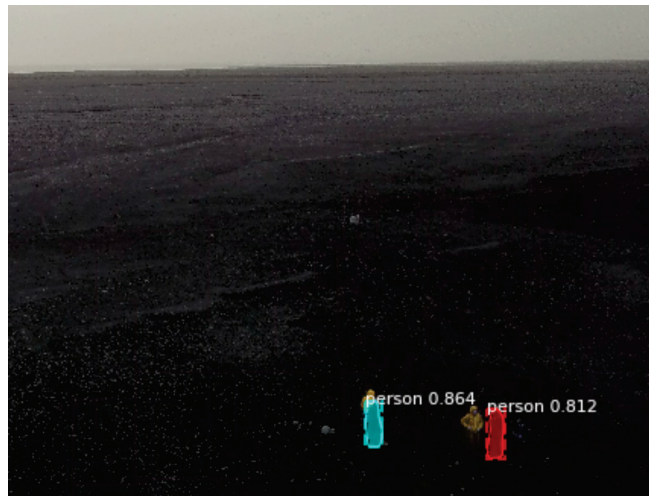


Figure 8: Model detection using real-world image.

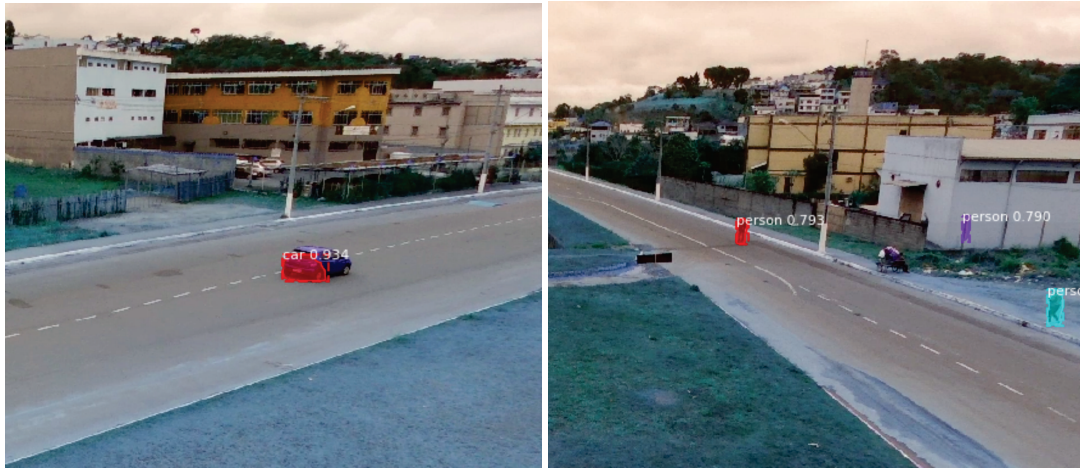
4.1 Tests in a Real Environment

For testing the proposed methodology in a real environment, this work used a Phantom 3 [45] flying in the height of eight meters and moving with low speed to capture the images. Figure 9(a) and Figure 9(b) present the results using the Mask R-CNN model. Note that the predicted results are higher than 0.9, which indicates that the model can generalize the knowledge obtained from the simulated environment.

The built database is composed of only 5000 images, approximately. Despite that, the learning rate dropped significantly, with 30 training iterations. Those results are in line with other recent works in the literature, such as Cai [46] and Kaiming [15].

Table 4: Confusion matrix parameters

True positive rate (TPR)	False positive rate (FPR)	Prevalence	Accuracy (ACC)	Positive predictive value (PPV)
94,29%	5,71%	69,92%	72,88%	75,34%



(a)

(b)

Figure 9: Prediction in a real environment. (a) Car. (b) People.

As described, the work can operate in a single GPU, processing five frames per second, which is a good performance for regular robotic applications. The estimated performance parameters are also a measurement of the proposed database efficiency. Therefore, it is possible to compare the results with other related works. Table 5 shows the results for a few other works, that is, [46], [47] and [15]. The results presented in the mentioned works are comparable to the results presented in Table 1.

Table 5: Comparison with results from the literature.

Backbone	mAP	mAP_{50}	mAP_{75}
RetNet-50 [46]	44.8	78.5	57.1
RetNet-101 [46]	54.2	79.6	59.2
AlexNet [47]	38.9	66.5	40.5
Kaiming ResNet-101-FPN [15]	35.7	58.0	37.8
Resnet-50-FPN	36.25	49.02	41.52

The works in [23, 24] have shown that people detection is an essential task in aerial images, and the works of [30, 36, 37] also showed the relevance of vehicle detection. As in this work, the method in [38] can detect both people and vehicles. None of them but [30], and the method proposed in this research work performs object detection through instance segmentation, which can provide a richer scene understanding. However, the method in [30] can detect the vehicle, and ours can identify both vehicles and people. Besides, as can be seen, in [39], there are few annotated datasets to train and test systems such as the ones in all these works. They also have shown that data augmentation techniques can aid in increasing performance in object detection in aerial images. This work provides an artificial dataset for car and people detection based on instance segmentation. Therefore, this research showed good performance when a deep CNN method is employed in this dataset as well as good generalization capability since the detector trained in this artificial dataset can perform well in real-world scenarios. The result indicates that simulated data can be used to train networks that are latter deployed in real-world conditions. This reduces the burden of deploying real equipment to gather data and allows the generation of a large dataset with low cost and repeatability.

5. CONCLUSIONS AND FUTURE WORKS

Instance segmentation is an important task with a wide range of applications. However, a large amount of data is necessary to train algorithms to perform it. In this sense, developing and testing algorithms for the autonomous systems is expensive and time-consuming to be performed directly in the real-world. Unlike conventional methods that use data from a real image datasets, this work proposes a synthetic dataset from a photo-realistic simulator to train a deep learning architecture. Thus, this research also presented a Mask R-CNN model that predicts per-pixel instance segmentation for a UAV application. The network proposed in this research learns to predict per-pixel instance segmentation for cars and people in a city environment. The proposed methodology was quantitatively evaluated on a city dataset. Experimental results demonstrated that the proposed approach has promising performance w.r.t. intersection over union and mean average precision. Besides, it is comparable to others in the literature. The trained model also has shown good generalization capability, since it could correctly predict real-world images from UAVs.

In terms of evaluation, this work opens the possibility of several future developments. For instance, it is expected that the proposed model can be applied in different situations, such as surveillance and rescue missions. Besides, the authors aim to increase the City dataset by gathering more data from the City environment, including new different areas for improving model

performance. Also, we would run this model in a dataset with real-world images that have ground-truth data to verify the model performance. Furthermore, in the future, it is expected to include more objects (e.g., buildings, trees, etc.) in the pipeline of instance segmentation models.

Acknowledgment

The authors would like to thank the following Brazilian Federal Agencies CEFET-RJ, CAPES, CNPq, and FAPERJ for supporting this research.

REFERENCES

- [1] C. Shinde, R. Lima and K. Das. “Multi-view Geometry and Deep Learning Based Drone Detection and Localization”. In *2019 Fifth Indian Control Conference (ICC)*, pp. 289–294. IEEE, 2019.
- [2] F. O. Coelho, M. F. Pinto, J. P. C. Souza and A. L. Marcato. “Hybrid Methodology for Path Planning and Computational Vision Applied to Autonomous Mission: A New Approach”. *Robotica*, vol. 38, no. 6, pp. 1000–1018, 2020.
- [3] A. Bloor, K. Garimella, X. He, C. Gill, Y. Vorobeychik and X. Zhang. “Attacking vision-based perception in end-to-end autonomous driving models”. *Journal of Systems Architecture*, p. 101766, 2020.
- [4] V. Bevilacqua and A. Di Maio. “A computer vision and control algorithm to follow a human target in a generic environment using a drone”. In *International Conference on Intelligent Computing*, pp. 192–202. Springer, 2016.
- [5] R. Parlange, J. Martinez-Carranza, L. Sucar and B. Ren. “Vision-based Autonomous Navigation for Wind Turbine Inspection using an Unmanned Aerial Vehicle”. 2019.
- [6] R. Miyamoto, Y. Nakamura, M. Adachi, T. Nakajima, H. Ishida, K. Kojima, R. Aoki, T. Oki and S. Kobayashi. “Vision-based road-following using results of semantic segmentation for autonomous navigation”. In *2019 IEEE 9th International Conference on Consumer Electronics (ICCE-Berlin)*, pp. 174–179. IEEE, 2019.
- [7] P. Trsljic, M. Rossi, L. Robinson, C. W. ODonnell, A. Weir, J. Coleman, J. Riordan, E. Omerdic, G. Dooly and D. Toal. “Vision based autonomous docking for work class ROVs”. *Ocean Engineering*, vol. 196, pp. 106840, 2020.
- [8] G. Hägele and D. Söffker. “Risk Areas Determination for Autonomous-and Semi-autonomous Aerial Systems Considering Run-Time Technical Reliability Assessment”. *Journal of Intelligent & Robotic Systems*, vol. 97, no. 3, pp. 511–529, 2020.
- [9] Z. J. D.-Y. Y. Siyi Li, Jiaji Zhou and M. T. Mason. “Learning accurate objectness instance segmentation from photo-realistic rendering for robotic manipulation”. *Proceedings of the 2018 International Symposium on Experimental Robotics (ISER)*, 2018.
- [10] F. Zeng, Y. Ji and M. D. Levine. “Contextual bag-of-words for robust visual tracking”. *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1433–1447, 2017.
- [11] K. Ashraf, B. Wu, F. N. Iandola, M. W. Moskewicz and K. Keutzer. “Shallow networks for high-accuracy road object-detection”. *arXiv preprint arXiv:1606.01561*, 2016.
- [12] R. Girshick, J. Donahue, T. Darrell and J. Malik. “Region-based convolutional networks for accurate object detection and segmentation”. *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 142–158, 2015.
- [13] R. Girshick. “Fast r-cnn”. In *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.
- [14] S. Ren, K. He, R. Girshick and J. Sun. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In *Advances in neural information processing systems*, pp. 91–99, 2015.
- [15] K. He, G. Gkioxari, P. Dollár and R. Girshick. “Mask r-cnn”. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.
- [16] S. Shah, D. Dey, C. Lovett and A. Kapoor. “Airsim: High-fidelity visual and physical simulation for autonomous vehicles”. In *Field and service robotics*, pp. 621–635. Springer, 2018.
- [17] H. Cho, Y.-W. Seo, B. V. Kumar and R. R. Rajkumar. “A multi-sensor fusion system for moving object detection and tracking in urban driving environments”. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1836–1843. IEEE, 2014.
- [18] A. Toshev and C. Szegedy. “DeepPose: Human pose estimation via deep neural networks”. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1653–1660, 2014.

- [19] G. L. Oliveira, A. Valada, C. Bollen, W. Burgard and T. Brox. “Deep learning for human part discovery in images”. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1634–1641. IEEE, 2016.
- [20] H. Xue, Y. Liu, D. Cai and X. He. “Tracking people in RGBD videos using deep learning and motion clues”. *Neurocomputing*, vol. 204, pp. 70–76, 2016.
- [21] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama and T. Darrell. “Caffe: Convolutional architecture for fast feature embedding”. In *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 675–678. ACM, 2014.
- [22] J.-X. Mi, D.-S. Huang, B. Wang and X. Zhu. “The nearest-farthest subspace classification for face recognition”. *Neurocomputing*, vol. 113, pp. 241–250, 2013.
- [23] P. Blondel, A. Potelle, C. Pégard and R. Lozano. “Fast and viewpoint robust human detection in uncluttered environments”. In *2014 IEEE Visual Communications and Image Processing Conference*, pp. 522–525, Dec 2014.
- [24] D. C. D. Oliveira and M. A. Wehrmeister. “Towards Real-Time People Recognition on Aerial Imagery Using Convolutional Neural Networks”. In *2016 IEEE 19th International Symposium on Real-Time Distributed Computing (ISORC)*, pp. 27–34, May 2016.
- [25] M. Billinghamhurst, A. Clark, G. Lee *et al.*. “A survey of augmented reality”. *Foundations and Trends® in Human-Computer Interaction*, vol. 8, no. 2-3, pp. 73–272, 2015.
- [26] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen *et al.*. “Guidelines for human-ai interaction”. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2019.
- [27] M. Younsi, M. Diaf and P. Siarry. “Automatic multiple moving humans detection and tracking in image sequences taken from a stationary thermal infrared camera”. *Expert Systems with Applications*, vol. 146, pp. 113171, 2020.
- [28] M. Ghiasi and R. Amirfattahi. “Fast semantic segmentation of aerial images based on color and texture”. In *2013 8th Iranian Conference on Machine Vision and Image Processing (MVIP)*, pp. 324–327, Sep. 2013.
- [29] X. Ren and J. Malik. “Learning a classification model for segmentation”. volume Vol. 1, pp. 10–17 vol.1, 11 2003.
- [30] L. Mou and X. X. Zhu. “Vehicle Instance Segmentation From Aerial Image and Video Using a Multitask Learning Residual Fully Convolutional Network”. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 11, pp. 6699–6711, Nov 2018.
- [31] C. WANG, L. Mauch, Z. Guo and B. Yang. “On semantic image segmentation using deep convolutional neural network with shortcuts and easy class extension”. In *2016 Sixth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pp. 1–6, Dec 2016.
- [32] H. TAO, W. Li, X. Qin and D. Jia. “Image semantic segmentation based on convolutional neural network and conditional random field”. In *2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*, pp. 568–572, March 2018.
- [33] Q. WANG, C. Yuan and Y. Liu. “Deep conditional neural network for image segmentation”. In *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, July 2016.
- [34] J. Long, E. Shelhamer and T. Darrell. “Fully convolutional networks for semantic segmentation”. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.
- [35] D. Marmanis, J. D. Wegner, S. Galliani, K. Schindler, M. Datcu and U. Stilla. “Semantic segmentation of aerial images with an ensemble of CNNs”. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 3, pp. 473, 2016.
- [36] Z. Chen, C. Wang, C. Wen, X. Teng, Y. Chen, H. Guan, H. Luo, L. Cao and J. Li. “Vehicle Detection in High-Resolution Aerial Images via Sparse Representation and Superpixels”. *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 1, pp. 103–116, Jan 2016.
- [37] Z. Chen, C. Wang, H. Luo, H. Wang, Y. Chen, C. Wen, Y. Yu, L. Cao and J. Li. “Vehicle Detection in High-Resolution Aerial Images Based on Fast Sparse Representation Classification and Multiorder Feature”. *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 8, pp. 2296–2309, Aug 2016.
- [38] H. Eris and U. Cevik. “Implementation of Target Tracking Methods on Images Taken from Unmanned Aerial Vehicles”. In *2019 IEEE 17th World Symposium on Applied Machine Intelligence and Informatics (SAMII)*, pp. 311–316, Jan 2019.

- [39] K. Chen, K. Fu, M. Yan, X. Gao, X. Sun and X. Wei. “Semantic Segmentation of Aerial Images With Shuffling Convolutional Neural Networks”. *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 2, pp. 173–177, Feb 2018.
- [40] S. Shah, D. Dey, C. Lovett and A. Kapoor. “AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles”. In *Field and Service Robotics*, 2017.
- [41] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick. “Microsoft coco: Common objects in context”. In *European conference on computer vision*, pp. 740–755. Springer, 2014.
- [42] Mask_RCNN. “Mask_RCNN”, 2018. Accessed in 20/05/2019.
- [43] F. Ximenes and et al. “CityDatabase”, 2019. Accessed in 01/09/2019.
- [44] Kaggle. “Drone Videos - DJI Mavic Pro Footage in Switzerland”, 2019. Accessed in 25/05/2019.
- [45] DJI. “Phantom 3 Standard Drone”, 2018. Accessed in 20/05/2019.
- [46] Z. Cai and N. Vasconcelos. “Cascade r-cnn: Delving into high quality object detection”. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6154–6162, 2018.
- [47] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal and V. K. Asari. “The history began from alexnet: A comprehensive survey on deep learning approaches”. *arXiv preprint arXiv:1803.01164*, 2018.