Vehicle Detection on Low Altitude Images Based on Edge Density

Antonio J. R. Neves, Manuel Camarneiro and Lucas Cozinheiro

DETI/IEETA, University of Aveiro,
3810-193 Aveiro, Portugal
Email: {an, mcamarneiro, lucas.cozinheiro}@ua.pt

***Abstract*—The popularity of unmanned aerial vehicles, usually denoted as drones, is increasing these days due to various factors. Their capability of capturing images from above, allowing new perspectives of a scene, is for sure one of the most significant. It gets even more interesting when captured images are processed using Computer Vision algorithms, creating a powerful technology combination with appliances in several areas. In this paper, we present algorithms under development to process images captured by drones over parking lots in order to detect parked vehicles and further estimate occupancy rates or cars parked in a wrong place. Another application that we are developing is the monitoring of boats in the Aveiro lagoon. As far as we know, the processing of low altitude images is still an open problem in the computer vision community. The preliminary results presented in this paper show the effectiveness of the approaches under development.**

Keywords–Drones; Image processing; object detection.

I. INTRODUCTION

Aerial videos captured by unmanned aerial vehicles are becoming popular these days. So called drones are fascinating as they are capable of getting images from places where it used to be impossible to put a recording camera.

Introducing interaction between the drone and its camera creates something much more complex and useful: a device that can be the key for solving a wide range of problems. Autonomous flight with a recording device onboard makes it, for instance, a mobile surveillance camera. But as soon as image processing is added, this device will be able to detect suspicious movements around a property or even follow potential intruders.

Solutions using drones and computer vision are not restricted to security. There is a huge number of possible areas where these devices might be useful [2] [4]. Although, only a few commercial drones are able to perform some basic image processing over obtained images. There is still a long way to go through on scientific research about this technology combination. Lately, a few commercial solutions are available for applications in agriculture mainly used to monitor plants growth, watering levels and fruit maturation. Some prototypes are also being tested for save and rescue tasks or fast mail delivery.

Before the proliferation of drones, monitoring vehicles from aerial imagery was already possible, making use of pictures either taken from satellite or from manned aircraft. For this kind of images there are several approaches regarding algorithms to detect and extract cars position. This is often associated with high altitude or satellite imagery [3] [6]. Even though, as this project assumes the usage of drones in lower altitude flights (about 10 meters from the ground), most of the published work does not apply. Thus, the solution was creating algorithms from scratch for parking lots with three different types of pavement, assuming to have drone's altitude and parking zone location regarding the road as program inputs.

Images were previously captured using a Parrot Bebop 2 [1] flying a selected path over some of the University of Aveiro parking lots, sampling parking zones built on tar, block pavement and both. Algorithms were further developed to detect parked vehicles over each type of pavement identified before.

The algorithms were tested on an external computer used for development but were also adapted for further tests in single boards in order to determine the possibility of having image processing onboard as the drone moves over the parking lots.

We present in this paper experimental results showing the effectiveness of the proposed approach, both in terms of detection ration as well as in terms of processing time.

The paper is organised as follows. In Section II, we present the problem studied in this paper. In Section III, we present an algorithm for car detection in three different type of pavement parks, namely blocks, tar and mixed pavements. In Section IV, we present experimental results. Finally, in Section V, we draw some conclusions.

II. CAR DETECTION IN LOW ALTITUDE IMAGES

To evaluate parking lots capacity the first mandatory task is image acquisition. Assuming the drone is correctly positioned regarding the road, a frame should be captured and sent to the image processing unit. Once there, the image might need to be corrected in case of heavy distortion effects. After this, the algorithm should try to detect vehicles, compare them with others detected in previous images to check if they were already counted, and finally update the counter. This repetitive pipeline is presented in a circular graphic in Figure 1.

Vehicle detection is obviously a decisive part of software but a broad range of cars might appear in a parking lot. Features as color, size or shape may vary from one to another making it harder to create a global solution capable of detecting them all based only on these features. At the same time, it is necessary to ensure that detected objects are effectively vehicles, distinguishing them from similar objects that might appear.

 ISBN: 978-1-61208-559-3

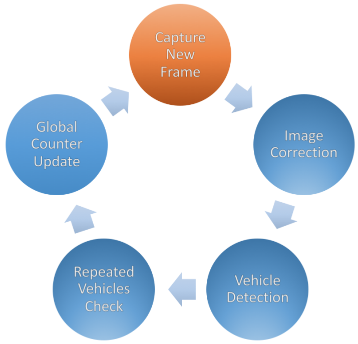

Figure 1. Algorithm Global Pipeline.

Distinguishing pavement from other objects placed above the ground could be a possible solution. Even though, this sets up another challenge. Homogeneous grey concrete or tar might be easily discarded using a color filter threshold. Nevertheless, the same technique will not work in a park built in block pavement.

III. PAVEMENT DETECTION

Distinguishing pavement from other objects placed above the ground could be a possible solution. Even though, this sets up another challenge. Homogeneous grey concrete or tar might be easily discarded using a color filter threshold. Nevertheless, the same technique will not work in a park built in block pavement. In this paper, we developed algorithms for three types of pavement.

A. Block Pavement Detection

Canny Edge Detector algorithm [5] was used as a fast and optimized method to perform gradient computations and retrieve the most important edges for each acquired image. Figure 2 shows a fine mesh, which corresponds to the edges of each small block that composes the pavement. Cars, on the other hand, are found in zones of low edge concentration.

Figure 2. At the top, an image of a Block Pavement Parking. On the bottom the corresponding gradient Image.

It would be possible to simply cluster regions with low edge density and compare their size to the expected car size (which estimation would depend on the drone's altitude). Even though this would give space to detection errors, either by including more than one car in a single cluster or by analyzing uninteresting zones in the surrounding areas. Some gardens or sidewalks, for instance, feature smooth surfaces making it harder to distinguish them from parked vehicles.

The solution was finding the road borders to further estimate parking places position. After applying a color filter and Hough lines detector to locate the grids along the road, it is possible to establish the interest regions, on the left, right, or both sides of the road (Figs. 3 and 4).

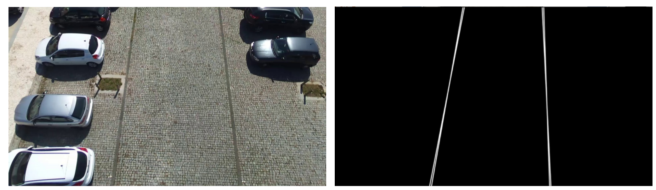

Figure 3. On the left, the image acquired by the drone, on the right the road borders detected.

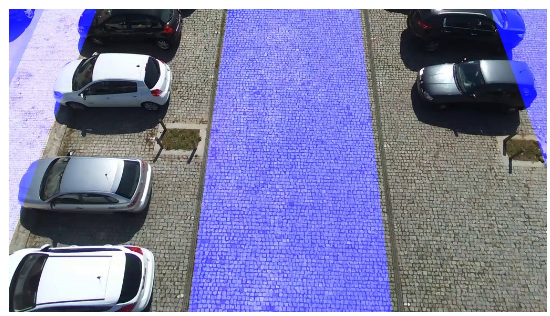

Figure 4. Road is removed as well as unwanted lateral zones. Parking places' length is very close to access road's width, thus interest region is trunked as presented.

Edge density analysis may now be performed for each one of the interest regions identified before. Vehicle's expected size is compared with the size of detected stains according to their position on the image. Objects on the top appear smaller due to the perspective introduced by the Drone used for image acquiring. To minimize errors edge density analysis is performed only on the bottom half of the image.

Despite the drone is moving with an almost constant speed, it is not possible to capture images without any overlay, meaning that the same vehicle might be present in more than one frame. To avoid double counting, it is required storing color, size and position features of vehicles detected in the last frame to compare them with the vehicles detected at the moment (Figure 5).

B. Tar Pavement Detection

Images obtained over tar pavement are smoother and lack of edges when compared to blocks pavement presented before. Despite that fact, it is still possible to reuse the logic from the last algorithm, detecting the road using the limit lines and trunking the interest regions.

As tar is homogeneous either in texture as in colour, checking if a low edge density zone is free or occupied can

 ISBN: 978-1-61208-559-3

Figure 5. New and Repeated Vehicles Detected

Figure 6. On the left, the image acquired by the drone, on the right the road borders detected.

be done using color matching, making sure it is different from the tar found in the road (Figure 7).

Figure 7. On the left, the image acquired by the drone, on the right the vehicle detection on tar parkings.

C. *Mixed Pavement*

Most of the studied parking lots are built on both tar (used in the access road) and blocks with different configurations (used in parking places). The algorithm developed for this type of pavement slightly differs from the others, given the impossibility of detecting road based on color filters.

In this case, road limits are determined using the same notion of edge density. Tar zones are not expected to have high gradient values thus road might be easily detected. Further vehicle detection is performed as explained for the block pavement, as well as vehicle repetition check (see Figure 8 and 9).

IV. EXPERIMENTAL RESULTS

This section presents important values measured and analysed for the studied solutions. These are related to detection accuracy and processing times.

All tests were performed using an Unix distribution (Ubuntu 14.04.3) installed on a computer with an Intel Core i5-3340M CPU @ 2.70GHz 4 processor with 4Gb RAM. Images captured were recorded as video and split into frames considering only one frame each half a second. A splitting

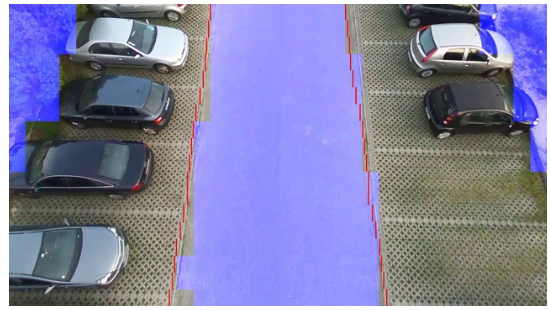

Figure 8. Road detection in mixed pavement parking lots. Raster scan window used to evaluate edge density is variable and affects processing times and road detection accuracy.

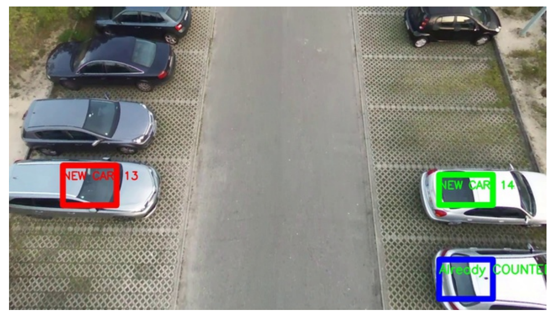

Figure 9. Vehicle detection in mixed pavement parking lots.

tool was also developed to read each video's frame rate and save images every 500 milliseconds in a specific directory previously defined.

Car detection accuracy is evaluated frame by frame comparing manual annotation of the number of cars depicted with detection boxes drawn by the algorithm.

It is crucial to choose a suitable edge detection method since these operations are performed every time a new image is processed. It is important to ensure some points regarding the chosen method:

- Detects low edge concentration over the road pavement (in case the road is made of tar)
- Creates high edge density zones over block pavement, contrasting with uniform surfaces on vehicles.
- Takes a short period of time to compute all the edges in an image.

Choosing the most suitable values enables accurate detection of homogeneous regions as shown in some examples presented in Figure 13.

It is now evident that Canny is an optimised edge detection method, possible to adapt to different situations by conveniently adjusting its parameters. It also features less processing requisites when compared to Sobel making it the best method and the one used for the rest of the algorithm tests.

Finding road limits composes a crucial step in the algorithm's pipeline since this is performed in every park and is essencial to the location of interest zones. It is not relevant to have high accuracy in this procedure as the main goal is to eliminate the major region of the image representing road. It is not decisive to remove every single pixel from the road and

 ISBN: 978-1-61208-559-3

Figure 10. Sliding window influence on road detection.

Figure 11. Cars detected on tar parking lot P1.

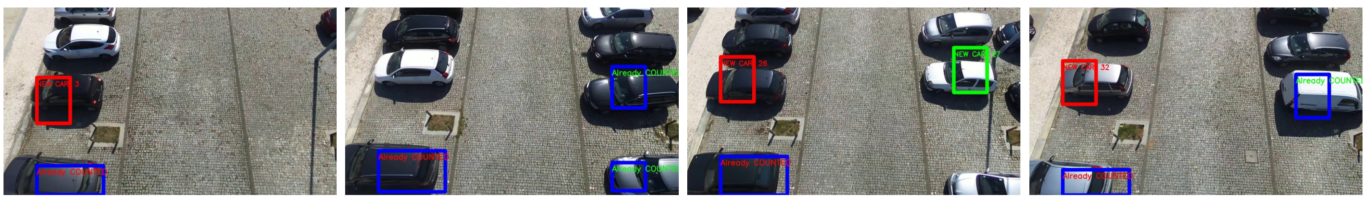

Figure 12. Cars detected on block pavement parking lot P5.

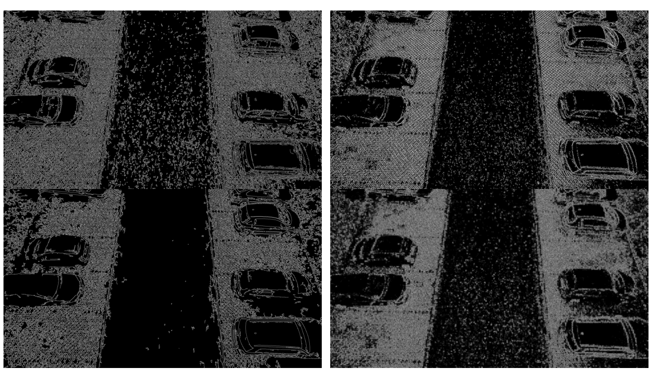

Figure 13. Two different edge detection algorithms tested with different parameters. From left to right, top to bottom: Canny Min=10, Max=50, Sobel X=1, Y=1, Canny Min=2, Max=200, Sobel X=2, Y=2.

preserve all other pixels, what is mandatory is the task to be quickly executed in every frame captured. Even though, road detection methods are distinct for different parking lots and different solutions should be developed for different pavement types.

On the other hand, vehicle detection based on low concentration of edges over the interest regions should be as accurate as possible and is applicable with only a few parameters adjustments to all studied parking lots. The following sections will detail results obtained for each type of parking lot studied.

Parking Lots in homogeneous kind of pavement are perhaps the most simple to deal with. Higher detection rates are then most likely to happen in P1 (Table I and Figure 11).

The block pavement revealed to be a difficult background

TABLE I. RESULTS FOR TAR PARKING LOT.

Parking Lot	Parked Cars	Counted Cars	Repeated Counting	False Positives	Undetected Cars	Detection Rate
P1	17	17	0	0	0	100%

to extract vehicles from. Despite edge density zones concept being easier to imagine in this situation, region segmentation is not easy as there are some homogeneous surfaces in the parking lot borders.

The technique used to detect road borders in the parking lot P5 is based on the grids detection, as referred before. This is very tricky since there is no color differentiation between pavement and grids, all that changes is edges density. Hough Lines detector keeps being used to determine border lines and, as expected, it increases processing time as shown in Table II. Some detection examples are presented in Figure 12.

TABLE II. RESULTS FOR BLOCK PARKING LOT.

Parking Lot	Parked Cars	Counted Cars	Repeated Counting	False Positives	Undetected Cars	Detection Rate
P5	61	61	2	0	2	97%
P5	78	78	1	1	2	97%

In the parking lots 2 and 3, the pavement is mixed. An homogeneous tar surface is found on the road, while the rest is made on block pavement.

Road detection is made based on a sliding window, which runs from the image center to its borders. This window might not move pixel by pixel, it may, for instance, jump 10 pixels every step saving some processing time. On its turn, window

 ISBN: 978-1-61208-559-3

TABLE III. DETAILED PROCESSING TIME FOR BLOCK PARKING LOT USING A RASPBERRY PI 2 MODEL B.

Frame	1	2	3	4	5	6	7	8
Open Time (RGB)	128	125	131	127	130	127	129	127
Gray	36	22	22	22	22	22	22	22
HSV	135	129	131	133	129	127	134	128
Find road	494	478	498	481	491	483	490	481
Car analysis	226	223	234	227	216	224	218	215
Cars found	1	1	2	2	1	1	1	1
Cars on image	1	1	2	2	1	1	1	1
%	100%	100%	100%	100%	100%	100%	100%	100%
TOTAL (ms)	1073	1030	1069	1042	1041	1035	1044	1025
				% AVG	100%			

TABLE IV. DETAILED PROCESSING TIME FOR MIXED PAVEMENT PARKING LOT USING A RASPBERRY PI 2 MODEL B.

Frame	1	2	3	4	5	6	7	8
Open Time (RGB)	141	144	142	146	140	144	137	145
Gray	36	22	23	23	22	22	22	22
HSV	138	128	130	127	128	130	128	129
Find road	431	430	422	428	421	427	422	420
Canny	284	278	272	281	284	280	277	271
Car analysis	226	225	223	218	222	225	245	237
Cars found	1	1	1	1	1	1	2	2
Cars on image	1	1	1	2	1	1	2	2
%	100%	100%	100%	50%	100%	100%	100%	100%
TOTAL (ms)	1309	1280	1265	1276	1269	1280	1284	1276
				% AVG	94%			

size might also be increased, loosing some definition on the road border found but improving algorithm's performance.

To evaluate this, several experiments were made in P2 in order to find a good relation between road borders detection accuracy and processing requisites as shown in Figure 10 and Table V.

TABLE V. RESULTS FOR MIXED PARKING LOTS.

Parking Lot	Parked Cars	Counted Cars	Repeated Counting	False Positives	Undetected Cars	Detection Rate
P2	46	45	0	1	2	96%
P2	20	20	0	0	0	100%
P3	39	39	0	1	1	97%
P3	24	23	0	0	1	96%

Since the goal is the development of fully autonomous drones, we tested the developed algorithms on several single-boards (Raspberry Pi 2 Model B, IGEPv2 DM3730 and EPIA-P910) in order to decide what could be the best hardware solution. Besides the processing time, presented in Tables III and IV we also tested other properties like weight and power consumption. The processing times obviously increase when running the developed algorithms on these single boards. However, we observe that it is possible to reduce the speed of the drone because the images acquired continuously have a considerable repetition of information. With this in mind, and evaluating the experimental results obtained, we consider that Raspberry Pi 2 reaches reasonable values for onboard processing.

V. CONCLUSION

The algorithms presented in this paper showed promising results for the detection of vehicles on low altitude images acquired by Drones, being a solution for parking lots management.

The developed algorithms fulfilled the low processing requirements, which enables the algorithms to process images every second and allows the drone to move at a reasonable speed; the accuracy associated to vehicle detection and counting is also high. Furthermore, results obtained for tests made in three different types of pavement indicated a versatile solution, adaptable to several contexts achieving good performances with slight parameter adjustments from park to park.

As future work, we are developing algorithms for boats detection on water and the preliminaries results were also satisfactory. We think this work can provide an interesting contribution to our future smart cities, as a starting point for monitoring of objects of interest using drones. Moreover, we are optimising the presented algorithms to be used on board of the droned in order to have a fully autonomous solution.

REFERENCES

[1] www.parrot.com/us/Drones/Parrot-Bebop-2, Bebop 2 drone, retrieved April, 2017.

[2] www.amazon.com/b?node=8037720011, Amazon prime air, retrieved April, 2017.

[3] J.-Y Choi and Y.-K. Yang, "Vehicle detection in aerial images", Technical report, College of IT, Kyungwon University, 2009.

[4] dailytech.com, Flying rescue drone prototype to provide lifesaving aid to swimmers, retrieved April, 2017.

[5] R. Maini and H. Aggrwal, "Study and comparison of various image edge detection techniques", Technical report, Punjabi University, India, 2009.

[6] S. Tuermer, J. Leitloff, P. Reinartz and U. Stilla, "Automatic vehicle detection in aerial image sequences of urban areas using 3D HOG features", Technical report, Technische Universitaet Muenchen, German Aerospace Center, 2010.