VOL. 15, NO. 7, APRIL 2020 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2020 Asian Research Publishing Network (ARPN). All rights reserved.
www.arpnjournals.com

ONBOARD OBSTACLE SENSING MECHANISM FOR DRONES USING MONOCULAR CAMERA

Aswini N.[1] and Uma S. V.[2]
[1]Department of Electronics and Communication Engineering, CMR Institute of Technology, Visvesvaraya Technological University, Karnataka, India
[2]Department of Electronics and Communication Engineering, RNS Institute of Technology, Visvesvaraya Technological University, Karnataka, India
E-Mail: shijiaswini@gmail.com

ABSTRACT

Unmanned Aerial Vehicles (UAV) commonly known as Drones, have become so popular in recent years because of their enormous technical development and demand in both the Military and Civil sectors. “No risk of human life" is the biggest advantage of drones. To become completely autonomous, there is a need for an Onboard Obstacle Sensing Mechanism (OOSM) to detect and avoid a collision. Most of the drones used now in the market have inbuilt cameras that can perform video recording. These videos are sent to the ground control station and are processed using suitable obstacle detection algorithms. The necessary commands are sent back to the drones for further manoeuvres. In a real-time application, obstacle detection becomes difficult when there is a need to process videos from a Monocular camera. By this work, an attempt is made to detect frontal obstacles using computer vision techniques, measure their distance from the monocular camera using a mathematical approach. The direction from which the obstacles are approaching is also estimated by calculating the centroid of each image frame.

Keywords: UAV, obstacle detection, distance calculation, direction estimation.

1. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) can perform various tasks that involve Aerial-mapping, Road/tunnel Inspection, Aerial photography, Search and Rescue, Firefighting, Military services, Mining, Construction site assistance, and Precision farming. The challenging issue faced by UAV is obstacle avoidance in the path of their trajectory. The commonly used sensors are RADARs, LIDARs, Ultrasonic sensors, Cameras, etc. Vision-based sensors like cameras provide more visual information since they mimic the human behavior of detecting obstacles. The constraints of size, weight, and power in mini or micro UAVs make lightweight monocular cameras a better option compared to others. In this paper, vision-based obstacle detection and calculation of the distance between obstacle and camera is proposed. The first section gives a brief survey of the existing vision-based obstacle detection techniques. The second section explains the experimental work and methodology followed by us for calculating distance. The third section briefs the direction estimation by extracting centroid points from image frames. Simulation is done in MATLAB.

2. LITERATURE SURVEY

Unlike stereo vision, obstacle depth estimation is a difficult task using a monocular camera. A single object detection technique may not be enough. A requirement for background subtraction, noise filtering, edge detection along with feature extraction might be a good choice. The presence of approaching obstacles can be estimated from the size expansion ratio as given in [1]. An obstacle is detected, when the size exceeds some threshold value. In the paper, they have briefed the use of Scale Invariant Feature Transform (SIFT) [2] algorithm to extract feature points from consecutive video frames. As this method fails to distinguish between far and near objects, the distance-ratio of matched points is calculated in [3]. The method investigates each point, so that far and near objects can be differentiated properly.

Even though very accurate, due to its computational complexity, SIFT is slow compared to the Speeded Up Robust Feature (SURF) technique [4]. An obstacle avoidance system for low-cost UAVs using SURF feature extraction and matching is proposed in [5]. Images taken from the onboard camera is matched with a database containing obstacles. SURF works with integral images to reduce computation time. In [6], they are utilizing a size expansion cue to detect frontal obstacles using a monocular camera. They also propose SURF feature extraction along with a template matching algorithm. The obstacles need to have enough texture to extract SURF key points.

Using Image segmentation, obstacle detection and avoidance technique are proposed in [7]. A combination of the Lucas Kanade method of optical flow [8] and Harris corner detection technique [9] is used in segmentation. A combination of a monocular sensor and LIDAR is proposed in [10]. They have also combined the SURF algorithm and Harris corner detector to calculate the size of obstacles. In all these aforesaid research works; key point features are extracted using either SIFT or SURF along with corner detectors and template matching. These approaches identify either the expansion in the size of the obstacle as the key method or optical flow is estimated.

In our work, along with feature extraction, an attempt is made to measure the distance between obstacle and camera. It is difficult to exactly locate the position of an obstacle using a monocular camera since the size of the obstacle is unknown. The pixel variation from the previous frame to the current frame is used to estimate the distance

VOL. 15, NO. 7, APRIL 2020 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2020 Asian Research Publishing Network (ARPN). All rights reserved.

covered. Two sets of video data are generated for this work. Section 3 explains the methodology followed.

3. OBSTACLE DETECTION AND DISTANCE CALCULATION

The first set of data is taken using DJI Phantom 4 Pro [11]. The drone is made to fly straight with a constant speed of 1m/s at a height of 5 to 6 feet. Either the drone or the obstacle (a car in this case) is remaining stationary (Figure-1). The video recorded by the frontal camera of the drone is of 47 frames/sec and 1920x1080 resolution. As a pre-processing step, it is reduced to 15 frames/sec and 640x480 resolution, filtered using median filtering. SURF algorithm is used to extract key points from current and previous frames (Figure-2). Using matchFeatures function in Matlab, the strongest SURF features are matched for consecutive frames. After features are matched, a convex hull is created joining the matched key points in both the frames. A bounding box showing the convex hull creation is given in Figure-3.

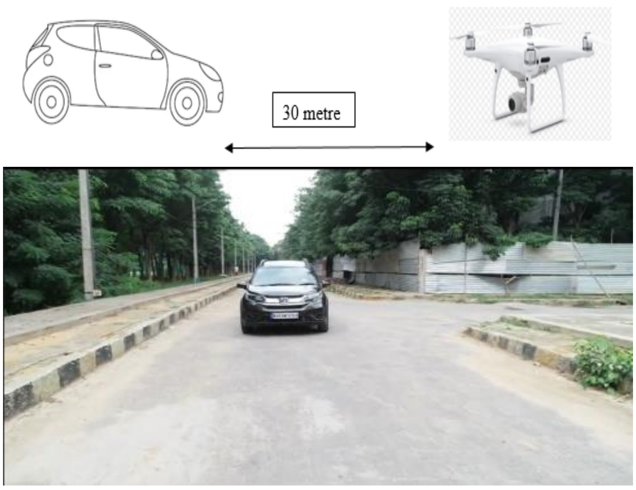

Figure-1. Low flying Drone and Obstacle.

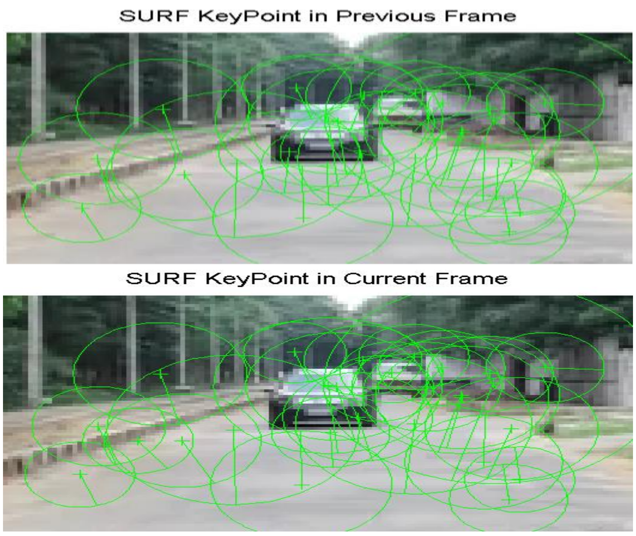

Figure-2. (a) Car coming closer to Drone (b) SURF Key points of ith frame (c) SURF Key points of (i+1)th frame.

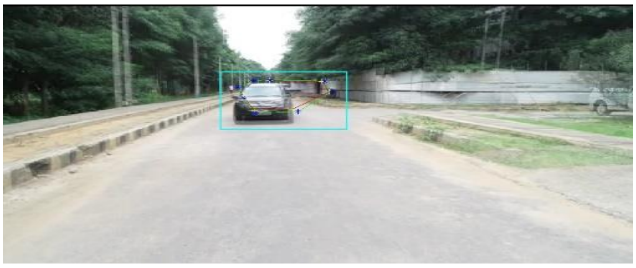

Figure-3. Convex hull created around matched SURF features (red for previous frame and yellow for current frame).

The next step is to calculate the distance (d) moved from the previous frame to the current frame in pixels. Using Euclidean distance formula, d can be calculated as in equation (1).

$$d = \sqrt{(Xc - Xp)^2 + (Yc - Yp)^2} \quad (1)$$

Where (Xc, Yc) is the current frame co-ordinates and (Xp, Yp) is for the previous frame. This distance calculation is done for the entire set of video frames. The number of matched key points for a pair of current and previous frames will be equal to the vector size of d. From camera calibration, for 1-meter change in obstacle state, we obtained 3.31985pixel difference. So, the distance covered by the drone/obstacle can be calculated using the camera calibrated value and the total pixel variation from the starting point as: -

Covered distance in meters (D) = (Distance moved in Pixels X Distance covered in one sec) ÷ 3.31985.

We have kept a safe operating distance of 30 meters to detect the obstacle very early so that a collision can be avoided. Once the covered distance is obtained, it can be subtracted from the total safe distance to get the exact position of the obstacle from the camera. The distance calculated when the car was approximately 22 meters from the camera is given in Figure-4. The entire simulation is done in Matlab 2015b. Table-1 gives a comparison of the actual distance and measured distance. The error obtained is very less to have safe navigation of drones.

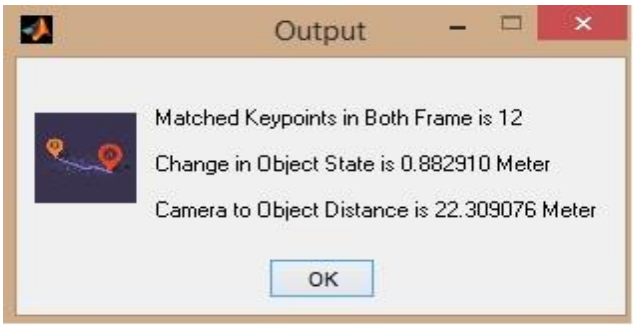

Figure-4. Camera to object distance calculated.

©2006-2020 Asian Research Publishing Network (ARPN). All rights reserved.

Table-1. Comparison of the actual distance versus measured distance.

S. No.	Actual Distance (m)	Measured Distance (m)	Error
1	28	28.78	0.78
2	26	26.60	0.60
3	23	23.10	0.10
4	21	21.50	0.50
5	17	17.40	0.40
6	15	15.03	0.03
7	13	13.52	0.52

Algorithm 1: Onboard Obstacle Sensing Mechanism #1

Input: Input Video from DJI Phantom Pro 4.
Output: Detection of Obstacle.

Step1. Video frames are generated.
Step2. Remove unwanted noise using median filter.
Step3. Key point extraction using SURF descriptor.
Step4. Key point matching using Feature Matching Metric Algorithm.
Step5. Convex hull creation.
Step6. If (Convex hull size) current frame > (Convex hull size) previous frame then Obstacle Detected.
Step7. Euclidian distance calculation.
Step8. Camera Calibration.
Step9. Measure Distance covered in meters.
End of Algorithm

The second set of data is an experimental data obtained by mounting a wireless camera on a toy car (Mimics drone movement). The main intention of this arrangement is to evaluate a mathematical method for distance calculation which will cover three directions of approach of the obstacle. The car with the camera is made to move forward 2 meters on a flat table. Another car is used as a frontal obstacle. In this method, we need to have prior knowledge of the height of the obstacle in pixels at various locations. The experimental arrangement is as shown in Figure-5.

Figure-5. Experimental arrangement.

The area covered can be divide into three different zones as in Figure-6. Once the car reaches the alert zone, a command is generated to change the direction to avoid collision.

Like in the real-time scenario performed previously, either camera or the obstacle can remain stationary. As the obstacle approach towards the camera or vice versa, the pixel size of the obstacle increases. The increase in the size of the obstacle as it approaches the camera is given in Figure-7.

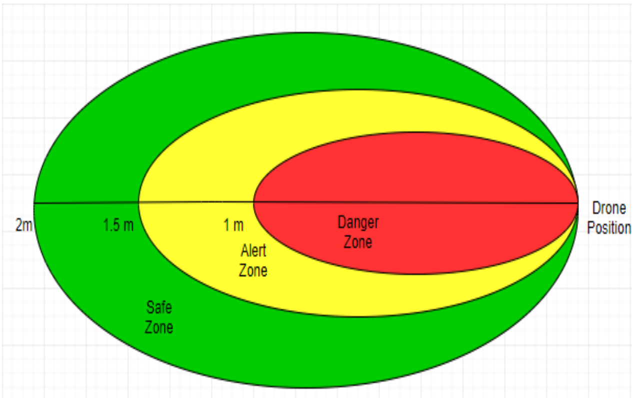

Figure-6. Three zones defined for experimental set up.

Figure-7. Three positions of frontal approaching Obstacle.

VOL. 15, NO. 7, APRIL 2020 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2020 Asian Research Publishing Network (ARPN). All rights reserved.
www.arpnjournals.com

Next, we calculated the pixel size variation for various distances and the pixel values obtained are given in Figure-8.

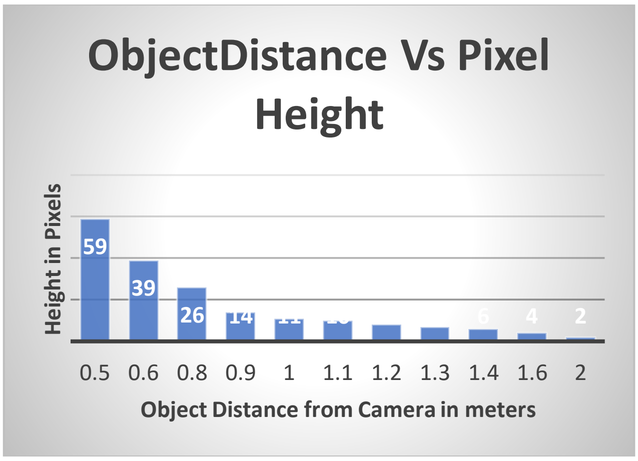

Figure-8. Pixel Variation of Object.

This variation of object height can be approximated as a second-order polynomial as in Equation 2. The coefficient values are very small since the measurement is done only for a two-meter distance.

$$\text{Estimated Physical Distance} = 0.0007148x^2 - 0.06232x + 1.774 \qquad (2)$$

Here, x is the height of the object in pixels. If the relative distance between two consecutive frames is dx and the time taken between the frames is dt, then the speed of the object can be calculated as dx/dt. This principle of distance and speed calculation can be extended to real-time, provided the object height at various distances is known. The object height at various distance points need to be calculated for the safe operating distance defined for that scenario. Here we calculated from 2 meters to 500 cm. The quadratic equation will not change much if similar height objects are being considered.

4. DIRECTION ESTIMATION

To estimate the direction of the object three different scenarios are covered. (1) Object coming straight (2) Object from Left (3) Object from Right as in Figure-9.

Figure-9. Three directions from which obstacle is approaching.

The centroid of the obstacle is tracked for the three sets of movements. A pictorial representation of the path of the obstacle with camera in the center is depicted in Figure-10. The centroid values we obtained for the three cases are as follows:

(x, y) = (300,40), (300,48), (300,58), (300,71), (300,101) for frontal Obstacle.
(x, y) = (132,33), (185,40), (235,53), (276,74), (295,105) for left approaching Obstacle.
(x, y) = (450,32), (410,44), (350, 58), (318,80), (305,100) for right approaching Obstacle.

a) From the x coordinate values of the first case, we can see that for a frontal obstacle, the value remains the same, since there is no movement in the x-direction and the increase in y value shows an approaching obstacle.
b) For a left approaching obstacle, the x value increases when compared to the previous x coordinate value and an increase in y value shows an approaching obstacle.
c) For a right approaching obstacle, the x value decreases when compared to the previous x coordinate value and an increase in y value shows an approaching obstacle.

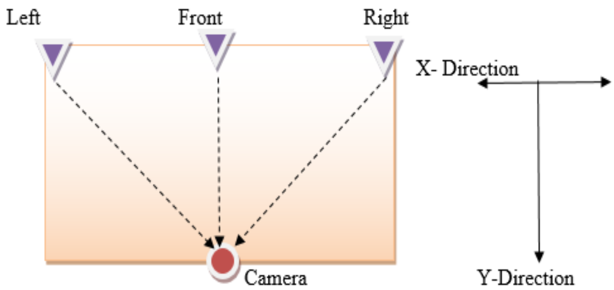

Figure-10. Three Directions of approach considered.

From this study, we were able to estimate the direction of the approaching obstacle as in Figure-11:

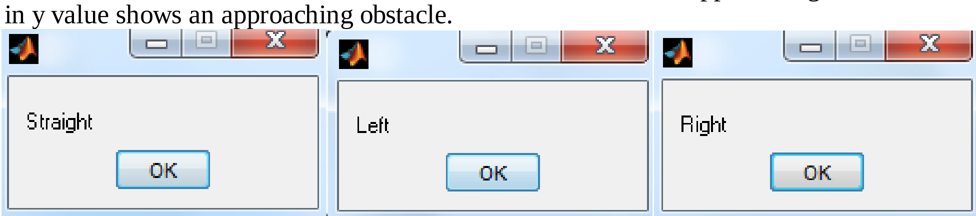

Figure-11. Output obtained for three test cases.

©2006-2020 Asian Research Publishing Network (ARPN). All rights reserved.

Algorithm 2: Onboard Obstacle Sensing Mechanism #2
Input: Input Video from Wireless Camera mounted on a moving car.
Output: Detection of Obstacle.
Step 1 Video frames are generated.
Step 2 Remove unwanted noise using median filter.
Step 3 Key point extraction using SURF descriptor.
Step 4 Key point matching using Feature Matching Metric Algorithm.
Step 5 Convex hull creation.
Step 6 If (Convex hull size) current frame > (Convex hull size) previous frame then Obstacle Detected.
Step 7 Measure Obstacle height in pixels at various locations from camera.
Step 8 Generate a second order polynomial equation for distance calculation.
Step 9 Find the centroid for each frame.
Step 10 Estimate the direction of approach.
End of Algorithm

5. CONCLUSIONS

Distance calculation between object and monocular camera from an image has been an area of research, because of the unknown parameters involved in the measurement. By this experimental work, an attempt is made to calculate the distance and speed of an approaching obstacle towards the monocular camera. In the first method, knowledge of the total scanning distance needs to be known. In the second method, the height of the obstacle in pixels at various distances near to the camera needs to be known. The methodology can be extended for implementation to sense and avoid obstacles in low cost, lightweight micro or mini drones. The drone should maintain a constant speed on a straight path. The principle of expansion of image size and the corresponding pixel variation is being used for distance measurement. Real-time distance and speed computation if done onboard can aid complete autonomous navigation by Drones. By this methodology, the payload of the Drone can be minimized. The incorporation of obstacle direction estimation is an added advantage.

REFERENCES

[1] Abdulla Al-Kaff, Fernando Garcia, David Martín, Arturo De La Escalera and Jose Maria Armingol. 2017. Obstacle detection and avoidance system based on monocular camera and size expansion algorithm for UAVs. Sensors. 17(5): 1061.

[2] David G. Lowe. 2004. Distinctive Image features from scale invariant Key Points. International Journal of Computer Vision. 60(2): 91-110.

[3] S. Badrloo, M. Varshosaz. 2017. Vision-Based Obstacle Detection In UAV Imaging. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLII-2/W6, 2017 International Conference on Unmanned Aerial Vehicles in Geomatics, 4–7 September 2017, Bonn, Germany.

[4] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool. 2008. Speeded-up robust features (SURF). Computer vision and image understanding. 110(3): 346-359.

[5] Wilbert G. Aguilar, Veronica P. Casaliglla and Jose L. Polit. 2017. Obstacle Avoidance for Low-Cost UAVs. IEEE 11thInternational Conference on Semantic Computing (ICSC). pp. 503-508.

[6] Tomoyuki Mori and Sebastian Scherer. 2013. First Results in Detecting and Avoiding Frontal Obstacles from a Monocular Camera for Micro Unmanned Aerial Vehicles. IEEE International Conference on Robotics and Automation, May.

[7] Pooja Agrawal, Ashwini Ratnoo and Debasish Ghose. 2015. Vision-Based Obstacle Detection and Avoidance for UAVs Using Image Segmentation. AIAA Guidance, Navigation, and Control Conference, January.

[8] B. D. Lucas, T. Kanade. 1981. An Image Registration Technique with an Application to Stereo Vision. in Proceedings of Image Understanding Workshop. pp. 121-130.

[9] Chris Harris and Mike Stephens. 1988. A Combined Corner and Edge Detector. Proceedings of the 4th Alvey Vision Conference, Manchester, 31 August-2 September, 147-151.

[10] M. Faiz Bin Ramli et al Object Detection Technique for Small Unmanned Aerial Vehicle, 2017 IOP Conf. Ser.: Mater. Sci. Eng. 260 012040.

[11] https://www.dji.com/phantom-4-pro/info