Vision based detection and tracking of dynamic objects using UAV

Santosh Muthireddy[1] and Chinmayee Shinde[2] and Kaushik Das[3]

Abstract— **In this paper, we present an algorithm to combine depth information derived from stereo camera and RGB information for detection of drone and ball within range of 1 to 20 meters. We utilized Connected component Labeling algorithm on disparity map got from a stereo camera (ZED Camera) to get information of different floating objects in the scene. We also implemented a reduced search area based on the previous detection and full image search is carried out every 10 frames to get the information of any added or deducted floating objects. This reduced search area implementation resulted in improved Frames Per Second (FPS) making algorithm usable in real-time detection. This implementation overcomes the challenges faced in algorithm implemented in Onboard Stereo Vision for Drone Pursuit or Sense and Avoid [3] likes lower FPS, detection, and tracking of multiple floating objects and effectively surpass the false detection of floating objects.**

I. INTRODUCTION

The use of micro aerial vehicles (MAVs) / unmanned aerial vehicles (UAVs) in the commercial sector, such as food delivery, air taxi, etc., has now increased due to it's advantages such as smaller size, dexterity and ability to navigate through any situation. This in turn increases the security threat posed by airborne MAVs / UAVs. To get rid of a potential intruder drone we need MAVs / UAVs with counter-UAV capabilities. These applications require the detection, tracking and maneuvering of each MAV / UAV relative to other nearby MAVs / UAVs. This has previously been demonstrated to be a self-contained system that meets all the acquisition and computational requirements for operating any on-board MAV / UAV in [3]

This paper addresses the issues and improvements over the above mentioned paper. Detection is performed by segmentation of disparity map, both on board platform and intruder MAV/UAV is moving which inserts motion blur effect in the disparity map looks like aura around the drone which will result in lose of detection or false blob detection instead of drone.So we are introducing another technique which uses edge information along with disparity map which will keep up the accuracy of detection and will be faster than YOLOv2. To overcome this problem edge information from RGB image is merged with disparity map to clearly define and hold the shape of MAV/UAV. Apart from this improvements are made to increase the performance of algorithm, reduced search area algorithm is implemented to decrease the computation time is described in section V - OPTIMIZATION.

II. RELATED WORK

Drone detection has been proposed using image sensors mainly in visible spectrum [6], [8], [9], [10]. Monocular vision has the possibility to fail because uncertainty in texture and the rapid manoeuvring of the MAV/UAV, especially against the confusing background that is usually happening in the outdoor. Thermal image based drone detection has also been proposed [1] but thermal images have lower resolution compared to that of RGB images. Several other techniques are also proposed based on acoustic sensors [7], radar [5], LIDAR [4]. However this techniques cannot be used due to size, weight and power constraints which becomes a limitation to integrate on board a drone.

At this point of time vision based detection and tracking is good considering they are light weight and low in power consumption but monocular vision has certain drawbacks which are discussed above. To overcome that stereo vision can be used as they provide 3D space information. Disparity maps are used to detect flying drones in [3], [2] using the key concept that flying vehicles can be separated from background, as flying objects appears in depth contrast with background. [3] uses depth information and classical vision techniques to detect drone and [2] uses depth information and deep neural network YOLOv2 for detection of drone in disparity map. [2] will be slower compared to classical vision techniques and [3] is failing to detect the drone when there is motion blur in the disparity map.

III. PROBLEM FORMULATION

In this experiment we are trying to detect and track dynamic objects using the fact that flying objects can be differentiated from surroundings in 3D space taking "Onboard Stereo Vision for Drone Pursuit or Sense and Avoid" as reference but this fails when there is motion blur involved in disparity image which will change the shape and size characteristics of the blob. This is being addressed in our algorithm which uses the edge information from the RGB image to hold the characteristics of the blob. The flow chart of algorithm is shown in Fig 1.

Fig 1 shows the pipeline used for whole algorithm, In the first stage images are captured using stereo camera and desired information is made using canny edge detection on left RGB image and stereo matching algorithm to get disparity and depth information. Next blocks consists of processing of gathered images using connected components labelling and filters mentioned in the Fig 1. Later blocks explain about the reduced search area implementation and so on.

Where in this algorithm overcomes the detection of false blobs when there is motion blur effect in disparity map.

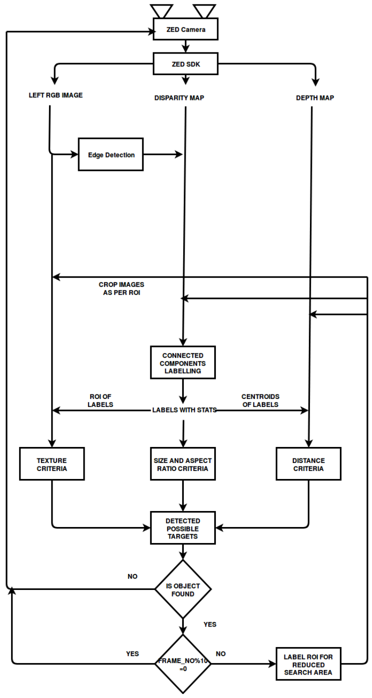


Fig. 1: Block Diagram

IV. IMPLEMENTATION

In this experiment disparity information, depth information and left RGB image are used for detection of dynamic moving object.ZED stereo camera is coupled with ZED SDK to retrieve the required information. Once the required data is gathered, left RGB to is used get the edge information of the objects present in the scene. Then preprocessed the edge information to close any opened contours using morphological operations. This edge information is then combined with disparity map to define the object edges in disparity map for further processing. Fig 2 demonstrates the merging of the edges information with disparity information and corresponding results.

We used the above disparity map with which edges information is merged for the further process, this helped us to get rid of extended disparity of dynamic object due to motion blur in the RGB images used for stereo matching.

To detect the dynamic moving objects we are using disparity map with edge information and Connected Components technique is used to separate out the background, noise and dynamic objects. disparity map is passed to Connected components Labeling algorithm to generate the labels and other statistics of the labels like centroid, top-left corner and height, width of bounding box.

After getting the labels from Connected Components Labeling(CCL) we start filtering of labels based on size, aspect ratio, texture and depth corresponding to centroid of the label. First filter is based on size i.e, number of pixels associated with the label, this filter has upper threshold to remove background and lower threshold to remove noise. Then we used aspect ratio of bounding box to eliminate the

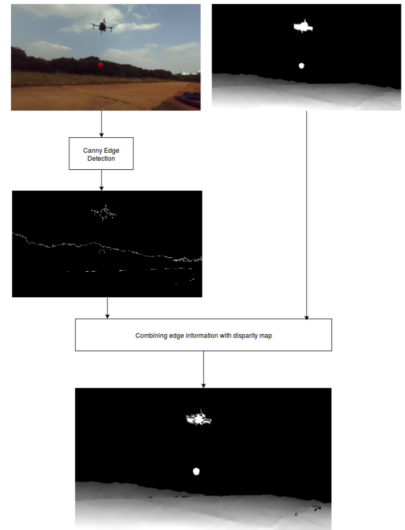


Fig. 2: Merging Data

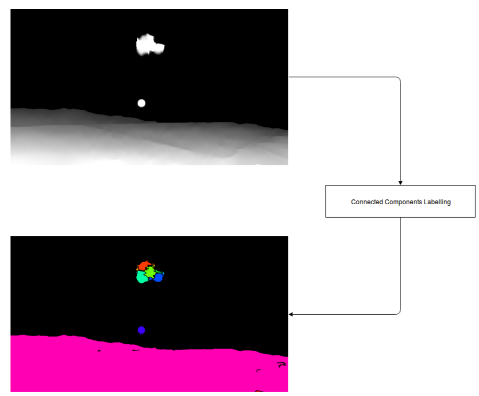


Fig. 3: Connected Components Labeling

blobs having lower aspect ratio in vertical and horizontal directions. Next filter is based on distance at which blob is present blobs above N meters(N is based on the target that need to be detected) is considered as noise and eliminated and blobs with distance values of "inf", "-inf" and "Nan" are removed as they are in blind zone of camera. Final filter is based on texture associated with the blob in RGB image. Below equation is used for calculating quantitative value of the texture.

$$B_{texture} = \sqrt{\sum_{i \in B} S_x^2(i) + S_y^2(i)} \tag{1}$$

where S_x is Sobel gradient along horizontal axis and S_y is Sobel gradient along vertical axis

ROI in the RGB image is defined from the bounding box of the blob obtained from CCL. Above equation [1] is applied on ROI in RGB image to get the absolute value of the texture and then it is normalized to comparable value based on the size of the ROI.

$$B_{norm-texture} = \frac{B_{texture} \quad (1)}{Size\ of\ the\ Blob} \tag{2}$$

After the application of filters which are fine tuned based on target to be detected on the disparity map the blobs left out are our possible targets. Centroids of the detected blobs are preserved to get the pose estimate of the dynamic moving objects.

V. OPTIMIZATION

Above implementation gives the detection of possible dynamic moving objects but it was the full pass of image through the algorithm which increases the computation time results in reduced frame rate. To overcome this issue reduced search area have been opted based on the previous detection. This results in lower computation time and higher frame rate. ROI for search area is defined based on number of detections and there corresponding positions in previous frame.

$$C_{ROI_x} = \frac{\sum_{i \in C} C_x(i)}{Length\ of\ C} \tag{3}$$

$$C_{ROI_y} = \frac{\sum_{i \in C} C_y(i)}{Length\ of\ C} \tag{4}$$

$$C_{ROI} = [C_{ROI_x}, C_{ROI_y}] \tag{5}$$

$$h_{ROI} = 100 * Length\ of\ C \tag{6}$$

$$w_{ROI} = 100 * Length\ of\ C \tag{7}$$

where C is centroids list of detections in previous frame, C_{ROI} is the centroid of the defined search area, h_{ROI} is the height of the defined search area and w_{ROI} is the width of the defined search area.

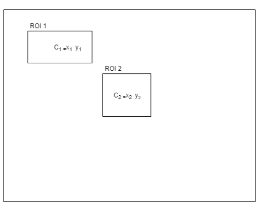


Fig. 4: Image with multiple bounding boxes

In Fig 4 image shows 2 bounding boxes named ROI 1 and ROI 2 with centroids $C_1 = (x_1, y_1)$ and $C_2 = (x_2, y_2)$. Centroid list C is given as $C = [C_1, C_2]$. So ROI for reduced search area can be defined from above equations 3,4,5,6,7

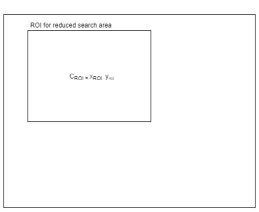


Fig. 5: Image with multiple bounding boxes

In Fig 5 C_ROI is calcuated using equations 3 and 4. Height and width of the reduced search ROI is calulated using equations 6 and 7.

Now disparity map and RGB image are cropped to defined search area and passed to CCL then followed by filters to get the dynamic moving objects. Every 10^{th} frame is processed fully passed through CCL to detect other dynamic moving objects if any. If the detection is lost in current search area then whole image is passed through CCL to get the current location then reduced search area is used for next 10 frames.

Yolo is trained on the custom dataset of drones which was used to detect the drone in most complex situations. It is implemented so that accuracy of the whole model to detect dynamic objects gets increased without reducing the efficiency and speed.

Ask chinmay to give graphs from yolo and place it here.

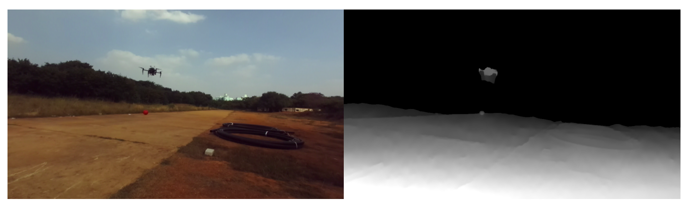
As I discussed previously using yolo alone to detect drones will give lower frequency of processing, below are the frame rates using yolo alone and yolo combined with our algorithm and our algorithm alone.

Method	Video	FPS			
		Trail 1	Trail 2	Trail 3	Avg
Classical vision	Video 1	17.60565	18.02837	18.16403	17.93268
Classical vision	Video 2	18.54529	18.3618	19.01601	18.64103
Classical vision + YOLO	Video 1	17.48395	17.16329	17.67139	17.43954
Classical vision + YOLO	Video 2	18.65345	18.87798	18.88768	18.80637
YOLO	Video 1	9.107232	9.135821	9.266668	9.169907
YOLO	Video 2	9.215426	9.249313	9.340665	9.268468

Fig. 6: FPS results

next section accuracy results should be placed

VI. RESULTS



(a) Left RGB Image (b) Disparity Image

Fig. 7: Retrieved images from ZED SDK

Fig 4 shows the image data acquired from ZED camera, Fig 4a is the image acquired from left camera and fig 4b is the disparity image aligned with left RGB image.

(a) Edge information derived from RGB Image using canny edge detector

(b) Edge information combined with disparity map

Fig. 8: Preprocessing on images

Fig 5a shows the edges detected from left RGB image using canny edge detector and Fig 5b shows disparity image combined with edge information from left RGB image.

(a) CCL applied without edge information

(b) CCL applied with edge information

Fig. 9: Connected Components Labeling application

Fig 6 shows the difference between results from application of CCL on normal disparity image and disparity image with edge information. Fig 6a shows the CCL map with deformed information of drone due to motion blur where as Fig 6b have drone information retained drone information using edges imposed on disparity.

Fig. 10: Detected dynamic objects

Fig 7 is the final detected drone from the above mentioned algorithm with displayed distance from the camera on top.

REFERENCES

[1] Petar Andraši, Tomislav Radišić, Mario Muštra, and Jurica Ivošević. Night-time detection of uavs using thermal infrared camera. In *INAIR 2017*, 2017.

[2] Adrian Carrio, Sai Vemprala, Andres Ripoll, Srikanth Saripalli, and Pascual Campoy. Drone detection using depth maps. *CoRR*, abs/1808.00259, 2018.

[3] Cevahir Cigla, Rohan Thakker, and Larry Matthies. Onboard stereo vision for drone pursuit or sense and avoid. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[4] Maarten Uijt de Haag, Chris G Bartone, and Michael S Braasch. Flight-test evaluation of small form-factor lidar and radar sensors for suas detect-and-avoid applications. In *Digital Avionics Systems Conference (DASC), 2016 IEEE/AIAA 35th*, pages 1–11. IEEE, 2016.

[5] Jedrzej Drozdowicz, Maciej Wielgo, Piotr Samczynski, Krzysztof Kulpa, Jaroslaw Krzonkalla, Maj Mordzonek, Marcin Bryl, and Zbigniew Jakielaszek. 35 ghz fmcw drone detection system. In *Radar Symposium (IRS), 2016 17th International*, pages 1–4. IEEE, 2016.

[6] Changhong Fu, Adrian Carrio, Miguel A Olivares-Mendez, Ramon Suarez-Fernandez, and Pascual Campoy. Robust real-time vision-based aircraft tracking from unmanned aerial vehicles. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 5441–5446. IEEE, 2014.

[7] József Mezei, Viktor Fiaska, and András Molnár. Drone sound detection. In *Computational Intelligence and Informatics (CINTI), 2015 16th IEEE International Symposium on*, pages 333–338. IEEE, 2015.

[8] Artem Rozantsev, Vincent Lepetit, and Pascal Fua. Flying objects detection from a single moving camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4128–4136, 2015.

[9] Yuanwei Wu, Yao Sui, and Guanghui Wang. Vision-based real-time aerial object localization and tracking for uav sensing system. *IEEE Access*, 5:23969–23978, 2017.

[10] Tamás Zsedrovits, Ákos Zarándy, Bálint Vanek, Tamás Péni, József Bokor, and Tamás Roska. Collision avoidance for uav using visual detection. In *ISCAS*, pages 2173–2176, 2011.