UAV World 2013

Autonomous Quadrocopter for Search, Count and Localization of Objects

Nils Gageik[1*], Christian Reul[1] and Sergio Montenegro[1]

1 Chair of Computer Science 8: Aerospace Information Technology, University of Würzburg, Germany
* gageik@informatik.uni-wuerzburg.de

Abstract. The present paper describes and evaluates the design and implementation of a fully autonomous quadrocopter, which is capable of self-reliant search, count and localization of a predefined object on the ground inside a room. In a preliminary scan the parameters of the object are defined: As an example object a red ball is used. The scan determines the color and radius of the ball. After determining the scanning parameters, the autonomous search can be executed. This is done autonomously by the quadrocopter, which uses inertial, infrared, ultrasonic, pressure sensors and an optical flow sensor to determine and control its orientation and position in 6 DOF (degree of freedom). Another camera attached to the quadrocopter and directed to the ground is used to find the searched objects and to determine its positions during the autonomous flight. Hence, objects which fulfill the scanning parameters can be found in different positions. Based on its own known position and the position of the object in the picture of the camera, the position of the found objects can be determined. Thus repeated detections of objects can be excluded. Consequently, objects can be counted and localized autonomously. The position of the object is transferred to the ground station and compared with the true position to evaluate the system. Two different search situations and two different strategies, breadth first search (BFS) and depth first search (DFS), are investigated and their results are compared. The evaluation shows the potential, constraints and drawbacks of this approach just as the effect of the search strategy, and the most important parameters and indicators such as field of vision, masking area and minimal object distance as well as accuracy, performance and completeness of the search. The entire system is composed of low-cost components and constructed from scratch. This allows an easy and cheap adaption, multiplication as well as simultaneous research and development of sub-systems with a great flexibility, understanding and documentation.

Keywords. Autonomous UAV, Quadrocopter, Quadrotor, Search and Rescue, Count, Object Localization

1. Introduction

Equipping UAVs (unmanned aerial vehicles) like quadrocopters with more and more autonomous abilities is an interesting field of research. Furthermore it is a requirement for challenging autonomous search and rescue missions, which are still a field of interest [1-14]. Especially, fully autonomous systems are challenging since they cannot rely on external systems like GPS or optical tracking for accurate positioning. State of the Art is the usage of a laser scanner for obstacle detection, collision avoidance and via a SLAM-algorithm (simultaneous localization and mapping) for positioning [15-16]. But laser scanners are heavy, expensive and fail in some situations like a smoking environment. Other approaches are vision-based, but the high computational burden often requires an external computer for computation [17-19].

Therefore, we present a solution for a fully autonomous system using a new hardware design combining optical and pmd (photo mixing device) cameras with infrared and ultrasonic distance holders for a reliable system capable of search and rescue missions. The present paper focuses on the concept, implementation and evaluation of the search, count and localization of red balls (example search targets) with an autonomous system based on the mentioned new hardware design.

This research is part of the AQopterI8 Project of the Chair Aerospace Information.

2. Terms and Background

To clarify different terms, parameters and algorithms, which will be used later, they are defined in this chapter. For reasons of simplification, the entire search is performed in a rectangular room free of obstacles. The main idea of the here presented search implementation is, that the quadrocopter uses a camera directed to the ground and by flying through the search area it scans all possible locations on the floor for a target (red ball). If a target is detected, it is added to the list of found targets, unless a target has already been detected at this position. Thus the whole area can be searched for targets and the amount of targets as well as their positions can be determined.

The most significant parameters for the performance of the search, the virtual field of vision (VFOV), the masking area (MA) and the search strategy are investigated, and therefore need to be defined.

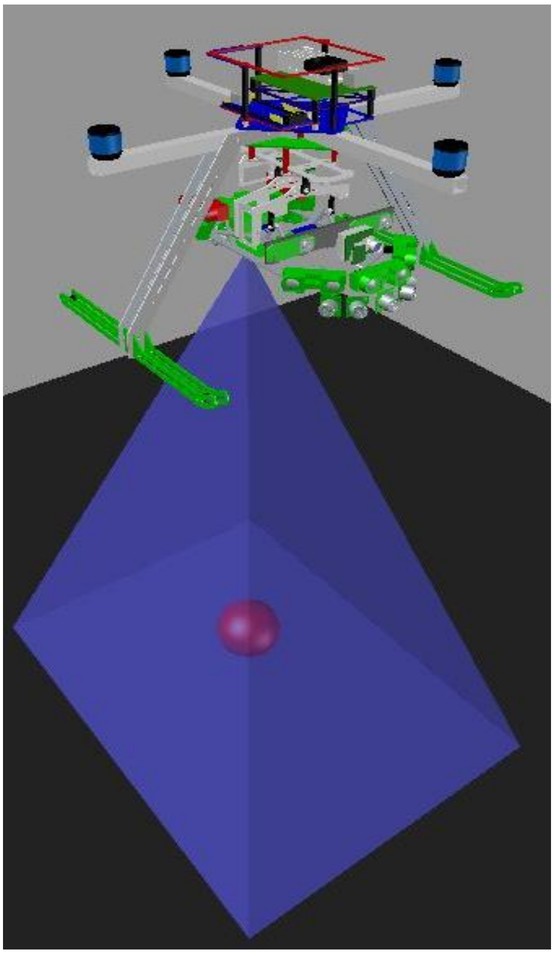

Figure 1: Field of View (FOV)

The field of view (FOV) is the area on the floor, which the camera used for object detection covers (Fig. 1). Using computer vision object detection a target can be found on this single picture of the floor. The field of view is specified by the camera (hardware), whereas the virtual field of view is the area which the search strategy uses in order to cover the whole search area at least once. For VFOV a smaller value than the true FOV may be used to leave room for inaccuracy. Detection might fail if the quadrocopter does not fly exactly as expected by the search strategy or if the target is located between two pathways and cannot be seen completely. A smaller VFOV leads to a higher coverage and a longer search pathway (compare Fig. 2 and Fig. 3).

Two different search strategies, which are later referred to as BFS (breadth-first-search) and DFS (depth-first-search), are investigated. They correspond to the original algorithms, which are used to search nodes in a graph. For reasons of simplification, all search algorithms start in the bottom left corner of the search area.
The idea of the BFS strategy is shown in figure 2. This strategy follows the general rule which says that closer positions are reached before farther ones. In general the used algorithm follows the iterative rule Up-Right-Down-Right-Up-Left.

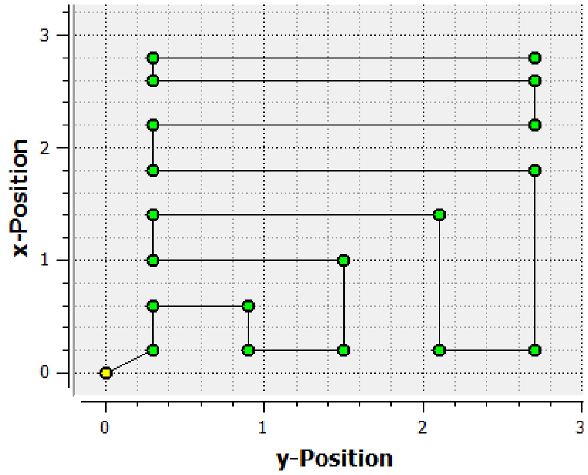

Figure 2: BFS Waypoint List (VFOV 40x60)

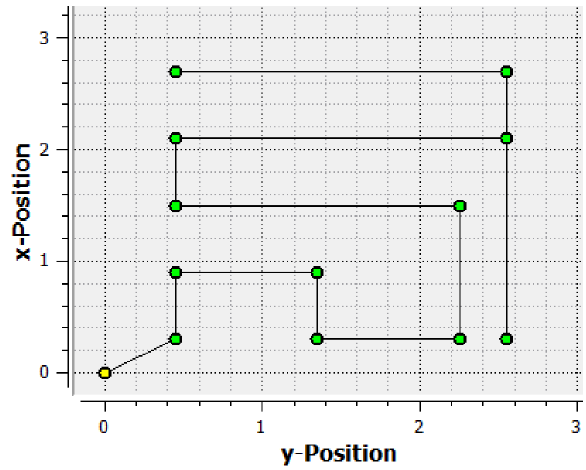

Figure 3: BFS Waypoint List (VFOV 60x90)

In contrast to the BFS, the DFS does not search nearer but farther positions first. At first the algorithm covers the sides of the search area and proceeds with smaller iterations until the complete search area is covered (compare Fig. 4 and Fig. 5).

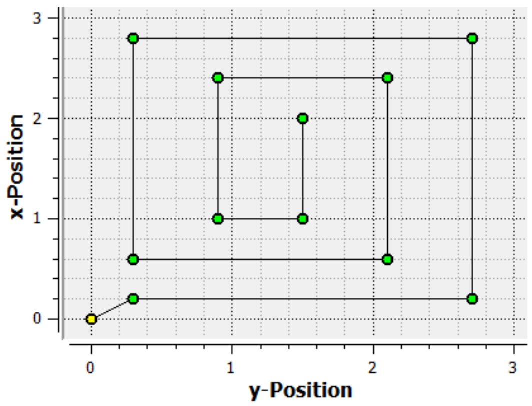

Figure 4: DFS Waypoint List (VFOV 40x60)

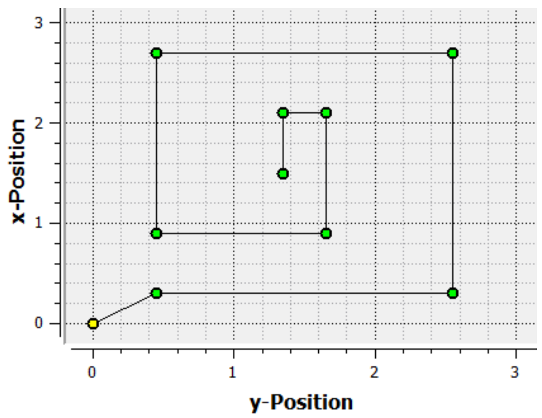

Figure 5: DFS Waypoint List (VFOV 60x90)

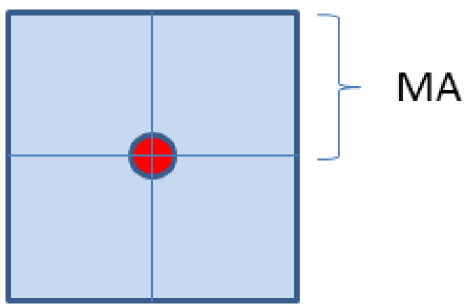

Figure 6: Masking area around an accepted target (red dot)

The masking area (Fig. 6) determines a square which is set around a detected object to avoid multiple detections of the same target. It is determined by a distance named MA. During the search, a target might be seen several times from different positions. Because of errors and noise, the target is never detected exactly at the same position again, and therefore would be considered as a new object multiple times. The masking area is subtracted and added to the X-coordinate and Y-coordinate of every accepted target and it is proved, if the newly found target is located within one of these coordinates. If so, the newly found target is discarded, otherwise it is accepted. Instead of a circle a square masking area was chosen because the FOV is also a square.

3. Concept

The concept of the overall system can be separated into two parts: The object or target search and the flight search.

3.1 Object Search

The task of the object search is to determine the amount and positions of the targets by fusing the results of the object detection with the current position of the quadrocopter (Fig.7). It manages the list of found objects and adds new ones if necessary.

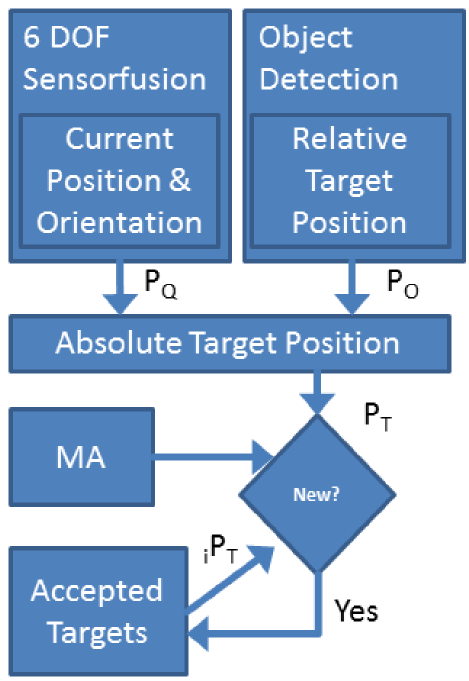

Figure 7: Object Search Concept

Whenever the object detection has a hit, the absolute position of this new target is computed by formula 1,

$$P_T = P_O + P_Q + C_{Offset} \qquad (1)$$

where C_{Offset} is the offset between the camera and the center of the quadrocopter or its position sensor, P_Q is the current position of the quadrocopter and P_o is the relative

position of the found object determined by formula 2.
M_x and M_y in formula 2 are the coordinates of the objects center point determined by the object detection, C_{wx} and C_{wy} are the calibration width in X and Y respectively at a height of C_h, h is the current height and Z is a constant. C_w and C_h are determined by the true FOV of the camera. R_x and R_y are the resolution of the camera in X- and Y-direction, respectively.

$$P'_O = \frac{h}{C_h} \cdot (\boldsymbol{M} - \boldsymbol{Z}) \cdot \boldsymbol{C_w}$$

$$\boldsymbol{M} = \begin{bmatrix} \frac{M_x}{R_x} & \frac{M_y}{R_y} \end{bmatrix} \quad (2)$$

$$\boldsymbol{Z} = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}$$

$$\boldsymbol{C_w} = \begin{bmatrix} \boldsymbol{C_{wx}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{C_{wy}} \end{bmatrix}$$

Next, the position P_T is compared with all positions ${}_iP_T$, with i indicating the index of the already accepted position. If the new position P_T occurs within the masking area of any target ${}_iP_T$ it is discarded, otherwise it is accepted.

3.2 Flight Search

The task of the flight search is to ensure that the quadrocopter with a determined VFOV covers the whole search area at least once. Waypoints are not generated next to another iteratively in small steps because of the bad flight performance of this approach [20], but with maximal distance according to the search strategy.
For simplification purposes, the flight search is executed statically. That means the waypoint list is generated once at the beginning and it is not changed during the flight. The waypoint list is determined by the search strategy, the search area and the VFOV.

---

4. Implementation

4.1 Hardware Design

The overall hardware-design of the quadrocopter is shown in figure 8. The total price of all 51 hardware components is about 1800€ with already 800€ for the PMD Camera Nano [21] and the onboard Pico-ITX PC LP-180 [22].
The system fuses ultrasonic, infrared, pressure and inertial sensors for height estimation [23]. For 2D positioning the optical flow sensor ADNS-3080 [24] is used. The height sensors, the inertial sensor IMU-3000 and the optical flow sensor are connected to the AVR32 UC3A0512 microcontroller, where 6 DOF (degree of freedom) position and orientation estimation is executed. The AVR32 is connected to the LP-180 via USART/RS232. A simple and cheap webcam, the Logitech C270, is used for object detection and also connected to the LP-180 via USB. To focus on the problem of search, count and localization the obstacle detection sensors are not used in the following evaluation, and therefore disconnected.

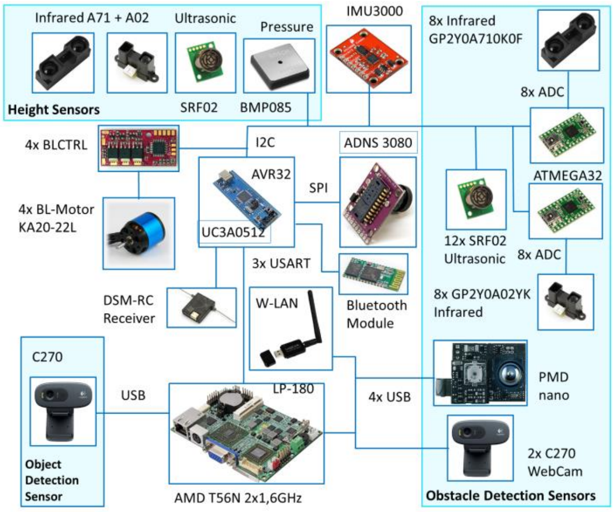

Figure 8: Hardware Design

4.2 Software Design

The overall software is separated into the software for the AVR32 and the LP-180. The AVR32 performs the 6 DOF computation and control, and is therefore executing the flight search, which means the processing of the waypoint list. This list is sent from the LP-180 to the AVR32. The LP-180 is mainly responsible for the object search. It receives the current position with a sample time of 10ms from the AVR32 and continuously performs the object detection with a sample rate of about 35ms. Depending on the chosen settings for masking area, search strategy, search area and VFOV the waypoint list is generated on the LP-180 and sent to the AVR32.
The software for the LP-180 is programmed in C++ with Qt [25]. It provides a GUI for steering the autonomous functionality and the object detection as well as to control, debug and evaluate the entire system. The same GUI software can be executed on the onboard computer LP-180 as well as on a ground station PC. It can be used to communicate with the quadrocopter and to perform the functionality as previously mentioned. Hence, the quadrocopter (AVR) can be connected via Bluetooth as well as via a W-LAN (LP-180). Through remote login it can be investigated, what is happening on the LP-180.
Figure 9 shows a screenshot of a video stream recorded

via remote login from the quadrocopter's onboard PC while performing autonomous search and count. On the bottom right corner the video stream of the object's detection web cam and the results of the current detection are displayed. There are also buttons and boxes for starting and setting up the parameters of interest as well as the waypoint list, the positions of found targets and the current position are shown.

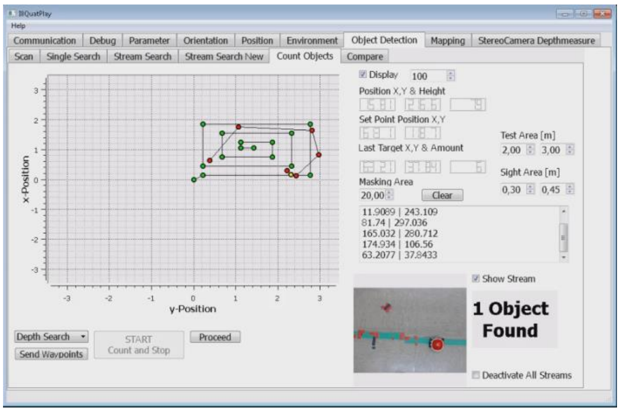

Figure 9: Qt Control-Software

4.3 Object Detection

The implemented object detection is based on OpenCV [26] and uses the Hough transformation for circle detection [27]. In a preliminary scan, which is performed manually before the autonomous flight, the parameters of the object, here the color and radius of the ball, are determined. This is done by taking a picture of the target. These parameters are automatically saved, and therefore a scan is only necessary if the target changes its color or size. Therefore, the system depends on the lighting conditions and greater changes require a new scan.

5. Evaluation

5.1 Overview Evaluation

To generally investigate the performance, accuracy and limitations of the system and to compare both search strategies (DFS, BFS) as well as to find the concrete optimal parameters for the masking area and VFOV, in total 72 experiments were executed in two setups. The first setup contained 51 experiments, of which 9 failed for unrelated reasons. The issue was an orientation drift because of inertial sensor failure, vibrations and electromagnetic disturbances on the I2C bus preventing the quadrocopter from finishing the search. After replacing the sensor and changing the mechanical and electrical setup the second setup with 21 runs could be executed without any failed experiment. In the following evaluation only the data of the 63 successfully completed experiments are discussed.

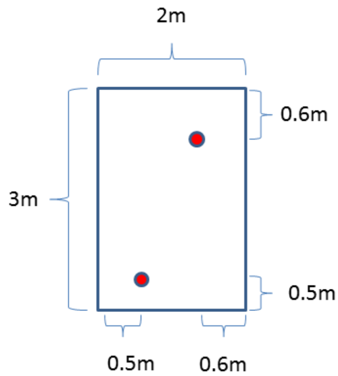

Figure 10: First Setup

5.2 First Setup

In the first setup the search area consisted of a 3m x 2m square with two randomly placed balls at the positions (50, 50) and (240, 140), according to figure 10.
In this setup the experiment was repeated for both search strategies, BFS and DFS, for four different masking areas with MA = 0.1m, MA = 0.15m, MA = 0.2m and MA = 0.3m and with three different VFOV: 0.3m x 0.45m, 0.40m x 0.60m and 0.60m x 0.90m. Then the computed position of the target was compared with the manually measured one, supposed to be the true position. For every single parameter setting the average error dx in X- and dy in Y-direction was computed, first over both targets and then over the entire run together (table 1).
Also the number of double detections D (fail positive) and misses M (fail negative) has been counted (table 2). In the second run the experiments for MA = 0.3m have been skipped, because MA = 0.2m showed no problem in this setup.

Average Error [cm]		**DFS**				**BFS**			
		Run 1		Run 2		Run 1		Run 2	
MA	**VFOV**	dx	dy	dx	dy	dx	dy	dx	dy
10	**30-45**	-8	16	-15	19	-24	20	-8	9
	40-60	-11	8	-19	8	-12	26	-19	19
	60-90	M		-10	11	-17	15	-14	13
15	**30-45**	-18	18	-27	7	-15	20	-19	20
	40-60	-28	8	-9	14	-4	19	-14	18
	60-90	-13	15	-3	12	-22	30	-16	12
20	**30-45**	-15	12	-14	23	-11	15	-12	20
	40-60	-39	4	-4	7	-35	34	-10	17
	60-90	-10	14	-10	17	-8	22	-11	9
30	**30-45**	-10	12	Skipped		-21	29	Skipped	
	40-60	-15	11	Skipped		-6	17	Skipped	
	60-90	M		Skipped		-12	13	Skipped	
Total Average		**-17**	**12**	**-12**	**13**	**-16**	**22**	**-14**	**15**

Table 1. Average Errors for first setup

From these data no clear difference in accuracy between DFS and BFS or between the different parameter settings could be identified, but it could be concluded that the average error in one axis is less than 15cm. This setup of randomly placed balls is predominantly affected by coincidence. It might be that one setting leading to one flight path fits well to the placement of the balls.
By taking a look at the detection failures (table 2), clear conclusions can be made: The real FOV is about 65cm x 45cm and it can clearly be seen that a VFOV of 40cm x 60cm or higher leads to misses. The bigger the VFOV is, the more misses occur, as expected. A proper VFOV of 30cm x 45cm leads to no misses for both search strategies. The data shows that a lower MA can lead to double detections. This is the case because a target might be seen several times. As the position error in one direction is about 15cm, MA should be at least in the same range. Conclusively, it can be seen that the DFS performed better and also that there is still a dominating systematical error.

Detection Failures		DFS		BFS	
		M	D	M	D
VFOV	30-45	0	0	0	3
	40-60	0	1	1	0
	60-90	4	0	2	1
MA	10	2	1	1	3
	15	0	0	1	1
	20	0	0	1	0
	30	2	0	0	0

Table 2. Detection Errors First Setup:
M missing and D double detections

5.3 Second Setup

Based on the outcome of the first setup, in the second setup more balls were placed to reduce the effect of coincidence. In addition, the search area was changed to a 2m x 3m square (Fig. 10 and 11), which aimed to equalize the results between the two search strategies and to improve the results of the BFS. Table 3 shows the positions of the 8 targets. This time positions were selected which could cause trouble for all settings.

Target Positions [cm]					
	X	Y		X	Y
Position 1	60	20	**Position 5**	100	180
Position 2	180	280	**Position 6**	20	240
Position 3	120	60	**Position 7**	40	220
Position 4	180	100	**Position 8**	100	300

Table 3. Target Positions Second Setup

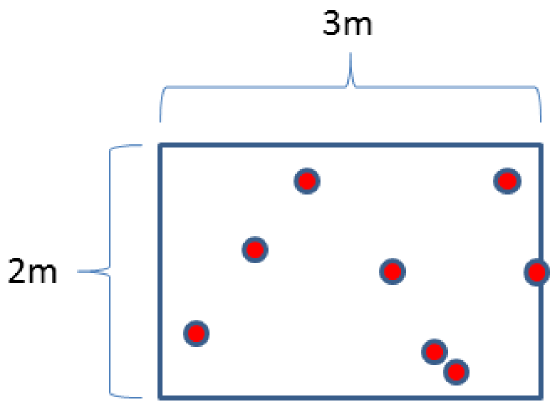

Figure 10: Second Setup

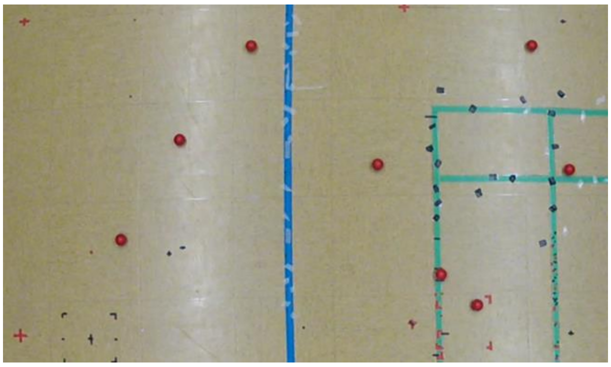

Figure 11: Picture of Second Setup from above

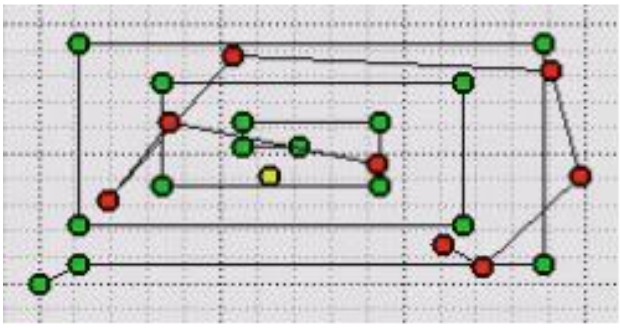

Figure 12: GUI Picture of Search Result (20-30-45)
Red: Found Targets; Green: Waypoints; Yellow: Position

Figure 12 depicts the results shown in the QT Control-Software for a run with the settings MA = 20 cm and 30 x 45 cm for VFOV. It demonstrates that for these settings all targets have been detected properly.

The second setup showed more clearly the effect of each parameter or setting and underlined the already expected results. More targets reduced the effect of coincidence, and therefore one run was seen to be enough.
Table 4 shows the position errors of the system in the second setup. The average position error for the DFS was 16cm and for the BFS it was about 20cm. According to this data the DFS can already be concluded as more accurate, but a clearer distinction between both search strategies can be made by taking the detection failures

into account (table 5).

Errors [cm]		DFS			BFS				
		Average			Average			Max	
MA	VFOV	dx	dy	E	X	Y	E	X	Y
15	25-35	Skipped			-19	8	21	32	24
	30-45	-17	16	24	-22	17	30	23	31
	40-60	-9	5	12	-13	9	18	23	17
	60-90	-6	9	11	-12	13	19	18	26
20	25-35	Skipped			-17	2	22	26	23
	30-45	-7	6	13	-17	17	25	28	32
	40-60	-23	0	24	-23	19	31	35	31
	60-90	-4	5	8	-7	4	10	13	10
30	25-35	Skipped			-14	7	17	27	12
	30-45	-12	-5	13	-15	9	18	20	19
	40-60	-6	13	17	-17	11	21	24	21
	60-90	-16	15	22	-9	5	13	16	13
Average:		**-11**	**7**	**16**	**-15**	**10**	**20**		

Table 4. Positions Errors of Second Setup: dx, dy and total Euclidean Error E as well as the Maximum Errors of BFS

Detection Failures		DFS		BFS	
MA	VFOV	M	D	M	D
15	25-35	Skipped		2	4
	30-45	0	2	2	0
	40-60	0	0	1	0
	60-90	2	0	4	0
20	25-35	Skipped		3	2
	30-45	0	0	1	0
	40-60	0	0	2	0
	60-90	2	0	4	0
30	25-35	Skipped		2	0
	30-45	0	0	1	0
	40-60	1	0	1	0
	60-90	3	0	4	0

Table 5. Detection Errors Second Setup: M missing and D double detections

For the DFS there are 10 detection errors in 9 experiments compared to 20 detections errors of the BFS in the same setup. Considering these bad results a value of 25cm x 35cm for VFOV was tested with the BFS, but this led to even worse results. There is no setting for the BFS without detection error, but there are four settings with no detection error for the DFS.

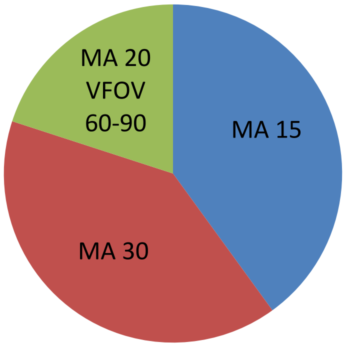

Figure 13: DFS Detection Failures (Distribution)

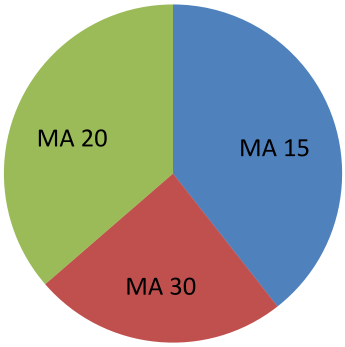

Figure 14: BFS Detection Failures (Distribution after MA)

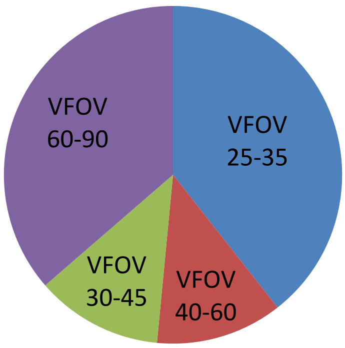

Figure 15: BFS Detection Failures (Distribution after VFOV)

5.4 Summary Evaluation

To sum up, it can be said that all settings, the search strategy, the masking area and the VFOV have a significant effect on the performance of the search. Although still other, partly random parameters and circumstances have an important influence on the result, optimal values of these parameters are required. This is underlined by figures 13-15, which show, that the DFS with MA = 20cm and a VFOV of 30cm x 45cm or 40cm x 60cm detected completely the 8 balls and nothing mistakenly else. This means there exist settings, which solved this challenging setup. It shall be mentioned that a MA of 30cm led to a miss in one of two cases, because then one of the closest balls, which are 20cm away, was not accepted. The BFS showed no good results here and only a VFOV of 30cm x 45cm and of 40cm x 60 cm led to acceptable results. Altogether these results also made the BFS look worse than it was. Some balls were positioned in such way, that the BFS failed them by few centimeters.

6. Conclusion and Discussion

The evaluation demonstrated that the system is capable to autonomously detect, count and localize objects with an accuracy of about 15-20cm. It was proven that an optimal value for MA (20cm) has to be a bit higher than the accuracy of the system and that objects with distance of 20cm (MA) in each axis can still be distinguished. Also the coherence of the parameters MA and VFOV on the performance of the search and the detection errors was demonstrated. A smaller VFOV with a smaller MA leads to more double detections, while a too high MA leads to misses of nearby objects. As a general rule too high VFOV leads to misses, because some areas are not searched properly. In this context the acceptance tolerance, which was set to 25cm in setup 2, is a parameter, which comes into effect. A waypoint is already marked as reached, if the current position of the quadrocopter is within this tolerance. This can result in an incomplete cover of the search area and it explains why the BFS misses some targets at the side of the search area.

The best parameter for VFOV was 30cm x 45cm. This setting together with the best value for MA showed no detection error even in a challenging room with 8 objects.

Furthermore, the evaluation proved that the DFS performed better than the BFS. The reason for that is the fact that more small waypoint steps are less accurate than less big ones, because of the set point jumps and the jump effect as well as the control and sensor system. This could be already demonstrated in previous experiments [20].

Although the system was proven capable of performing autonomous and challenging search, count and localization missions, the evaluation of the system did not show a very high accuracy according to the determined positions and the fact, that optical sensors were used, which generally can reach higher accuracies. There are multiple sources for accuracy errors, which start from the manually measured and placed target positions in a region of several centimeters. The next major source of error is the starting error, which means the wrong position measured by the optical flow during lift off and the wrong initial position and orientation or placement error of the quadrocopter on the starting position. An initial orientation error for yaw of only 1° leads to a position error of 5cm after 3m. It is most likely that the initial yaw orientation error was sometimes in the range of a few degrees. These are good explanations for the high systematical error, which can be seen in the data. A proof of this fact is given by a closer look to some raw data (Appendix: Table A). They demonstrate that the accuracy for the closer object is much better than for the farer object, even if the closer object is detected later in some cases. This is most likely because of an initial yaw orientation error.

In general it can be concluded that for this setup proof of high accuracy is challenging and the accuracy of the system might be better than the data show, but at the same time this is not the focus of this paper.

Other sources of error are wrong calibration values for the relative position of the found object P_o (Formula 2) and simplifications of Formula 2, a wrongly measured height, a wrong scaling factor for the optical flow, bad lighting and surface conditions, which lead to position errors measured by the optical flow sensor.

The current orientation of the quadrocopter was not considered in the computation of the position P_T. This was intended, because the effect of orientation drift should be excluded from the evaluation and orientation drift is still a problem in this system. In some cases not considering the orientation of the quadrocopter led to double detection errors.

7. Perspective

Although the system performed quite well in general, there are many things, which can be improved and should be investigated next. The effect of the already mentioned acceptance tolerance and an improved procedure for the waypoint navigation would allow higher values for VFOV. A waypoint control has already been implemented, which improved the waypoint flight significantly. But the additional control led to more orientation drift and therefore was not used. This in combination with a more accurate orientation computation would be the next step. It is already planned to switch to better sensors with SPI interface.

Furthermore, the procedure was supposed to be improved by using two phases. However, this was not evaluated because of the insufficient time. In the first phase the object search just tries to find something with a low resolution reducing the computational burden and increasing the possible sample time. The focus of the first phase is not to overlook something. If it has a hit, the quadrocopter suspends the waypoint search and flies to the position of the hit. Then the second phase is executed using a high resolution and accuracy and only in this phase the accepted position is determined. Computational burden is unimportant in the second phase because the quadrocopter is on position hold.

A different approach with a moving camera and flexible height could also be investigated. In this case the

quadrocopter would possibly not need to search the whole area or at least the waypoint list could be much smaller. In our setup the quadrocopter could simply fly 4 meters up and could see the complete search area. But that is not possible in every situation as usually rooms are not that high. However, it would be interesting to compare which accuracies and detection performance could be achieved then. Taking obstacles and unknown limitations into account as well as the fact, that objects might not be detected properly from a distance and at an angle, this approach is much more sophisticated, but offers as well more potential and might save flight time, and therefore could reduce the energy consumption.
Another interesting improvement would be to add the obstacle detection sensors and to search and count objects in a room with obstacles. The extra sensors should improve the position computation, and therefore the accuracy of the localizations. A challenging part is a reasonable distribution of the limited resources of the LP-180 to the different demanding tasks.

8. Acknowledgments

This work was funded by the Universitätsbund Würzburg as the Project "Lebensretter mit Propellern" (life-saver with propellers). We would like to thank Sascha Dechend for the design of figure 1. Furthermore we would like to thank Simone Bayer and Barbara Tabisz for their help.

9. Appendix

DFS		P1 [50, 50]			P2 [240, 140]		
MA	**VFOV**	**X**	**Y**	**E**	**X**	**Y**	**E**
10	**30-45**	47	64	14	227	159	23
	40-60	50	51	1	218	155	27
	60-90	Fail			Fail		
15	**30-45**	43	67	18	211	158	34
	40-60	30	53	21	204	153	39
	60-90	44	64	16	220	155	25
20	**30-45**	47	61	11	214	153	29
	40-60	31	52	20	182	146	58
	60-90	44	62	14	225	156	22
30	**30-45**	46	63	14	225	152	19
	40-60	45	59	11	214	153	29
	60-90	Fail			Fail		
Average		**43**	**60**	**14**	**214**	**154**	**30**

Table A: Raw Data of Setup 1 (DFS Run 1), E total Euclidean Error

10. References

[1] Nonami K., Autonomous Flying Robots, Springer 2010, ISBN-10: 4431538550

[2] Aibotix GmbH, www.aibotix.de

[3] Microdrones GmbH, www.microdrones.com

[4] ArduCopter, http://code.google.com/p/arducopter

[5] HiSystems GmbH, www.mikrokopter.de

[6] Mellinger D. et al, Trajectory Generation and Control for Precise Aggressive Maneuvers with Quadrotors, The International Journal of Robotics Research, April 2012, Vol. 31 No.5

[7] Gageik N., Mueller T., Montenegro S., Obstacle Detection and Collision Avoidance Using Ultrasonic Distance Sensors for an Autonomous Quadrocopter, UAVveek 2012

[8] Autonomous Flying Robots, University Tübingen, Cognitive Systems, www.ra.cs.uni-tuebingen.de

[9] Gronzka S., Mapping, State Estimation, and Navigation for Quadrotors and Human-Worn Sensor Systems, PhD Thesis, Uni Freiburg, 2011

[10] Lange S., Sünderhauf N. Neubert P., Drews S., Protzel P., Autonomous Corridor Flight of a UAV Using a Low-Cost and Light-Weight RGB-D Camera, Advances in Autonomous Mini Robots, 2012, ISBN: 978-3-642-27481-7

[11] Ding W. et al, Adding Optical Flow into GPS/INS Integration for UAV navigation, International Global Navigation Satellite Systems Society, IGNSS 2009

[12] Wang J. et al, Integration of GPS/INS/VISION Sensor to Navigate Unmanned Aerial Vehicle, The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII Part B1, Beijing 2008

[13] Blösch M.et al, Vision Based MAV Navigation in Unknown and Unstructured Environments, Robotics and Automation (ICRA), 2010 IEEE International Conference, 21-28

[14] Corke P., An Inertial and Visual Sensing System for a Small Autonomous Helicopter, Journal of Robotic Systems 21(2), 43-51, 2004

[15] Dryanovski I., Valenti R., Xiao J., Pose Estimation and Control of Micro-Air Vehicles, August 2012

[16] Grzonka S. et al, A Fully Autonomous Indoor Quadrotor, IEEE Transactions on Robotics, Vol. 28 No. 1, Februrary 2012

[17] Kendoul F. et al, Optical-flow based vision system for autonomous 3D localization and control of small aerial vehicles, Robotics and Autonomous Systems 2009, Elsevier

[18] Herisse B. et al, Hovering flight and vertical landing control of a VTOL Unmanned Aerial Vehicle using Optical Flow, 2008 IEEE International Conference on Intelligent Robots and Systems

[19] Zing S. et al, MAV Navigation through Indoor Corridors Using Optical Flow, 2010 IEEE International Conference on Robotics and Automation

[20] Gageik N., Strohmeier M., Montenegro S., Waypoint Flight Parameter of an Autonomous UAV, IJAIA, Vol. 4 No.3, May 2013

[21] pmdtechnologies GmbH, www.pmdtec.com

[22] Commell, www.commell.com.tw

[23] Gageik N., Rothe J., Montenegro S., Data Fusion Principles for Height Control and Autonomous Landing of a Quadrocopter, UAVveek 2012

[24] ADNS-3080 High-Performance Optical Mouse Sensor, Data Sheet, Avago Technologies, http://www.avagotech.com

[25] Qt Project, http://qt.digia.com

[26] OpenCV Library, Open Source Computer Vision, http://opencv.org/

[27] Christian Reul, Implementierung und Evaluierung einer Objekterkennung für einen Quadrocopter, BA Thesis, April 2013