Exploring Gridmap-based Interfaces for the Remote Control of UAVs under Bandwidth Limitations

Maik Riestock, Frank Engelhardt, Sebastian Zug*
Faculty of Computer Science
Otto-von-Guericke University
39106 Magdeburg, Germany
riestock, fengelha, zug@ovgu.de

Nico Hochgeschwender
Department of Computer Science
Bonn-Rhein-Sieg University
Sankt Augustin, Germany
nico.hochgeschwender@h-brs.de

ABSTRACT

The successes of teleoperation scenarios for mobile robots depends on a stable and reliable communication link. The environment information collected by the robot – represented by 2D or 3D images – has to be provided with a high resolution and a low delay to ensure a fast and precise system response. But in most realistic applications, the communication parameters fluctuate strongly over time. It is necessary to monitor the communication link continuously to react in case of reduced bandwidth and increased delay. But which environment information and correspondingly which bandwidth is necessary to control a robot safely? Due to a missing reliable rule set we investigated this question for a UAV scenario based on two different environment representations (camera images, gridmaps). We designed a simulator based study and evaluated the capability of the participants to control a robot in case of delayed or coarsely rasterized information.

Keywords

UAV teleoperation; remote-controlled robots; user interface design

1. MOTIVATION

Remotely controlled robots are widely used in industry, disaster management or in medical scenarios. Each remote robot scenario represents a closed loop control application. Sensors attached to a robot generate the input information for the operator. He reacts on the received environment data and provides commands via different interfaces. For this task, the operator needs to receive all relevant information in an appropriate format. In many applications the environment situation is reflected by either using (multiple) video streams, or by transmitting whole (3D) maps to the operator's desk, representing a cumulated sensor information over time. Depending on the dynamics of the controlled system and its environment, we have to consider constraints within the control loop that are induced by the communication channel. In case of a long *delay* the robot is not able to move with high velocity, to avoid occurring obstacles or to interact precisely with the surroundings. In case of limited *bandwidth* the video streams have to be downsampled or compressed. The aspects of communication bandwidth and delay are strongly connected with the communication configuration. Especially in ad-hoc scenarios (e.g. disaster robotics [4]) the communication bandwidth is limited, varies over time and does not ensure a specific response time. This is especially true for the operation of Unmanned Aerial Vehicles (UAV) which neccessarily depend on error-prone wireless communication links. Special middleware concepts exist for mobile robotics in the industrial domain [2] that manage Quality of Service (QoS) constraints as non-functional parameters. Individual applications can register their needs for communication resources, e.g. bandwidth, to an admission manager which decides wether there are enough resources available within the network. It is up to the application to assess its QoS needs very precisely and to not overuse once granted resources. When we assume a teleoperated UAV scenario, one has to transmit the information from onboard cameras and the collected 3D environment map in a continous flow to the operator, avoiding any communication losses and delay, to enable flawless remote control. s

*This work was partially supported by the German Research Foundation (DFG) research project "MoCoRo Plattform für mobile kooperative Robotik"

HRI '17 Companion March 06-09, 2017, Vienna, Austria

ACM ISBN 978-1-4503-4885-0/17/03.
DOI: http://dx.doi.org/10.1145/3029798.3038350

2. METHODS

We develop a study where participants have to guide a teleoperated quadrotor UAV through a desaster scenario within a simulation. For the pilots interface we focused on two common environment representations: A camera image and a grid map. The study followed the within-subject design with 9 conditions for each interface. These conditions origin from 3 different level of delay and resolution and their combinations, which represent the bandwidth limitation.

2.1 Procedure

The procedure can be separated in two parts. The first part contains the subjects preparation on the conditional runs, which includes an introducing video, experience questionnaires, a trainings phase and a test-flight. The second part are the conditional runs and the perception questionnaires in between. The order of conditions where chosen randomly.

2.2 Conditions

In order to generate a model representing the influence of delay and resolution on performance three level of delay and resolution were developed. These level differ between the interfaces, due to different representations. The idea was to cover the worst, medium and best case of bandwidth limitations and current approaches e.g. video streaming in naval missions using UMV.[3] Grid map resolutions are located below the physical size of the UAV. The size in bytes of the gridmap messages is determined by the largest possible point cloud (`pcl::PointXYZRGB`) needed to transfer the centroids of all occupied grid cells. We assume this to be the case when our Kinect-like sensor model looks at a plain wall at its maximum distance of 5 m.

Camera				
Resolution			Delay	
category	pixel	Size [KB]	category	[s]
low	160x120	57.6	low	0.01
mid	320x240	230.4	mid	0.1
high	480x320	460.8	high	0.5

Table 1: Overview of used settings (delay, resolution) for camera interface in different categories

Gridmap					
Resolution				Delay	
category	voxel size [m]	pixel	Size [KB]	category	[s]
low	0.350	14x10	4.4	low	0.1
mid	0.225	22x15	10.6	mid	1.0
high	0.100	50x36	57.6	high	2.0

Table 2: Overview of used settings (delay, resolution) for gridmap interface in different categories

2.3 Data Collection

The data to collect strongly depends on the goal of the teleoperation, which is in our case the safe navigation to a goal point. The word save stands for keeping the chance of any collision low during the flight and thus always holding a safe distance to near obstacles. As a numeric value to rate the performance, we measure the distance to near objects. In addition to performance we also measured the pilot‘s perception on their own performance using questionnaire after each conditional run.

2.4 Participants

31 participants were invited and successfully finished the test-flight. Participants are mostly computer science students or researchers in the field of computer science($M = 24.10, SD = 4.86$).

3. DATA ANALYSIS & PRELIMINARY RESULTS

Each of the 279 data series contains the complete trajectory of the flight, the interface configuration and the participant ID. For each participant we collected personal data and a self-assessment related to the reached performance.

For the exemplary evaluation in this paper we focus on the criticality of a trajectory point based on the distance of the UAV to an obstacle as described in Sec. 2.3. This distance value is used to calculate the time of violating the safety area of the UAV defined by the smallest ellipsoid covering the UAV plus a safety distance (side 0.2 m, top/bottom distance 0.1 m).

For the data analysis we used linear mixed model, where the collision number as the dependent variable and the configuration parameters of the interface – delay and resolution – as fixed effects and the subject as random effect. In order to calculate an adequate p-value for each fixed effect a likelihood ratio test were made comparing the previously described model with a reduced model, where the respective fixed effect were excluded.

Starting with the analysis of performance using the collision-time. We observe that pilot‘s had more difficulties keeping a save distance using the gridmap-based interface($M = 2.63, SD = 3.21$) as using the camera-based interface($M = 1.33, SD = 2.53$). Based on the interviews with the participants afterwards, this problem originates mainly from missing trust into the gridmap and a bug where some voxels appeared suddenly for a short moment. These problems need to be addressed in further investigations e.g. gain operators confidence by additionally training.

Coming to the results of the linear mixed models. The results on the camera-based interface indicates that delay($\chi^2(1) = 21.654, p = 3.265 \times 10^{-6}$) has a significant effect on the collision-time whereas resolution($\chi^2(1) = 1.771, p = 0.1833$) has no significant impact. Comparing the results from the camera-based interface to the results of the gridmap-based interface we can observe a similar pattern. Here delay($\chi^2(1) = 5.7127, p = 0.01684$) has a significant effect on the collision-time whereas resolution($\chi^2(1) = 0.0049, p = 0.9442$) has no significant impact. The difference between both interfaces lies in the influence of delay($p = 3.265 \times 10^{-6} vs. p = 0.01684$). These results indicate that there is a non-trivial level of delay where pilot‘s using gridmap-based interface perform better comparing with the camera-based interface.

4. FUTURE WORK

The contribution of this paper is focused on an study for UAV remote control applications. We investigated two interfaces for drone pilots and simulated a limited communication bandwidth. The first step is to completing the data analysis of the remaining data like completion time using the same approach with linear mixed models and the performance perception and experience data using linear models.

The next step further investigation in gridmap-based interfaces and their fields of applicability, including additional modes of control (e.g. supervisory) and other kind of robotic types like unmanned ground vehicles or high velocity vehicles. For this purpose the development of a standalone interface is necessary, which gives the possibility to integrate different kind of enhancements to support the operators in theirs specific tasks.

5. REFERENCES

[1] S. Blumenthal, N. Hochgeschwender, E. Prassler, H. Voos, and H. Bruyninckx. An approach for a distributed world model with QoS-based perception algorithm adaptation. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 1806–1811, Sept 2015.

[2] T. Lindhorst and E. Nett. Dependable communication for mobile robots in industrial wireless mesh networks. In *Cooperative Robots and Sensor Networks 2015*, pages 207–227. Springer, 2015.

[3] J.-M. Moreno-Roldán, M.-Á. Luque-Nieto, J. Poncela, V. Díaz-del Río, and P. Otero. Subjective Quality Assessment of Underwater Video for Scientific Applications. *Sensors*, 15(12):31723–31737, 2015.

[4] R. R. Murphy. *Disaster Robotics*. The MIT Press, 2014.