Human-Drone Teaming: Use Case Bookshelf Inventory

Alexander Freistetter
Johannes Kepler University
Linz, Austria
alexander.freistetter@gmx.at

Karin Anna Hummel
Johannes Kepler University
Linz, Austria
karin_anna.hummel@jku.at

ABSTRACT

Drones are flying machines that can sense in 3D, which makes them well suited for tasks that require sensing in different altitudes. Besides stable autonomous indoor operation of drones, drones also need to be prepared to interact with humans on dedicated tasks. We study the use of custom drones in a sample 3D scanning task, namely bookshelf inventory. We build a prototype distributed system consisting of a drone with an onboard camera, autopilot, and Wi-Fi connectivity, and a smartphone app that performs book recognition and navigation control, while also interacting with the human teammate. We report first findings in book detection success and discuss interaction options for humans to team up with drones.

CCS CONCEPTS

• **Networks**; • **Computer systems organization** → **Distributed architectures**; **Robotics**;

KEYWORDS

Human-drone teaming; unmanned aerial vehicles; distributed architecture

ACM Reference Format:
Alexander Freistetter and Karin Anna Hummel. 2019. Human-Drone Teaming: Use Case Bookshelf Inventory. In *Proceedings of ACM Internet of Things (IoT 2019)*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.475/123_4

IoT 2019, October 2019, Bilbao, Spain

Figure 1: Use case bookshelf inventory: University library.

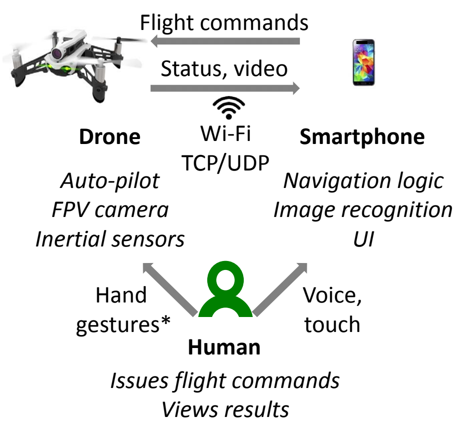

Figure 2: Schematic view of the proposed distributed architecture for camera-based scanning. The prototype system consists of the human, the drone, and a smartphone. Touch and voice human input is implemented, hand gestures are under development (marked with ∗).

INTRODUCTION

Employing small drones in civilian applications has become popular in entertainment, cartography, surveillance, etc., in particular for taking pictures and streaming video. Important system design decisions target navigation, communication, human interfaces, and the system architecture. In indoor scenarios, vision-based navigation is often chosen [5]. Concerning transmission of video content, 4G/5G and aerial Wi-Fi networks provide sufficient bandwidth [1]. Human-drone interfaces can be traditional screen-based, realized by drone gestures [4], and any kind of human input modalities including touch, gestures, and voice. Finally, a major design decision relates to whether control is implemented locally on the drone or remotely on a more powerful computer [2, 3].

Focus and Contribution. We focus on indoor scanning tasks for off-the-shelf drones, exemplified by the use case bookshelf inventory in libraries such as depicted in Figure 1. We propose a distributed system, where the drone is equipped with a camera for scanning, but only basic flight control is run on the drone's microcontroller. Image processing, vision-based navigation, and the human interface, are implemented on a remote computer. While the use case and image recognition is specific, the distributed approach can be generalized for drone systems. In this paper, we extend our own previous work [3] by reporting recent experiments and advances in book recognition, and provide suggestions on how to integrate humans in such a scanning task.

Use Case Bookshelf Inventory. The drone operates indoors and mostly autonomously, i.e., after the human has directed the drone towards the bookshelf, it positions itself close to the bookshelf at a starting position to perform the scanning task thereafter. Among possible scanning options, vision has been selected as tagging of books is not required. Scanning starts at the upper left corner of the bookshelf and is implemented as a vertical zigzag movement pattern, scanning one layer of the shelf after the other. Discovered books are stored on the smartphone and can be uploaded to the library database. In the current implementation we assume that there are no multiples of book titles, yet, with filtering this can be relaxed. Scanning is finished when there are no more layers left.

SYSTEM DESIGN AND IMPLEMENTATION

Figure 2 visualizes a concrete case of a distributed architecture for a camera-based human-drone scanning system. The drone captures and sends video data using the onboard FPV (first person view) camera, performs flight maneuvers, and sends status information about the current directional speed. The smartphone app receives image information from the drone, performs image recognition, and sends back flight commands. The smartphone provides also voice recognition and a touch interface to interact with a human; note that human interventions always overrule the autonomous navigation logic. The smartphone and the drone are connected through Wi-Fi IEEE 802.11ac; standard TCP

Table 1: Parrot Mambo quadcopter HW.

Size, Weight	CPU
$180 \times 180 \times 40$ mm, 70 g (with battery)	800 MHz ARM processor
Sensors	**Lifetime**
3-axis accelerometer pressure, gyroscope, vertical camera	10 min, 660 mAh

Table 2: Overview of used app libraries.

Name (purpose), URL
Parrot developers' classes (messaging) https://github.com/Parrot-Developers/Samples/blob/master/Android/
OpenCV 3.4.1 (rectangle detection) http://opencv.org/
Tesseract tess-two 5.4.1 (text recognition used in [3]) https://github.com/rmtheis/tess-two/
Google ML toolkit (new text recognition) https://developers.google.com/ml-kit/

and UDP messages are used. The drone provides the Wi-Fi network and acts as an access point and DHCP server, while the smartphone connects to the Wi-Fi network as a client device. In addition, we foresee (but have not implemented yet) hand gestures that command the drone to navigate to a specific location. Note that the concept of the distributed architecture is general and is not limited to smartphones, camera input, or to the sample use case.

Hardware and Software. Table 1 summarizes the hardware capabilities of Parrot Mambo. This light-weight quadcopter comes with an autopilot and flight command interface. The smartphone app is a Java-based Android app implementing the user interface for the bookshelf scanning task (i.e., visualization of detected books and voice and touch-based manual navigation), image and text recognition, and navigation control. The app was successfully run on four different smartphone platforms, among them a Samsung Galaxy S5. Table 2 summarizes the libraries used.

Image Recognition. Obstacles that have the contour of a rectangle are identified as books (i.e., the angles have to be about 90° and four edges need to be detected). Further, the ratio between the book's width and height is evaluated and a minimum number of books needs to be detected to recognize a bookshelf. At the same time, the app scans the book title whenever a book is in the center of the view. The output of the app are the number of books and an estimate of book titles.

Drone Navigation Control. Based on camera images, the drone's position is determined and a 3D flight command is issued to reposition the drone (i.e., moving up/down, forward/backward, or left/right). These high-level flight commands are based on Parrot's messaging API and packed into a TCP message. A flight command is basically a four-tuple consisting of the vertical speed (gaz) and the three angles of movement along the aircraft axes: pitch, yaw, and roll. The values are expressed as percentages of the maximum possible parameter value.

EXPERIMENTS AND INSIGHTS

Communication Performance. Our first results [3] show a mean latency of about 10 ms and zero packet loss on the Wi-Fi link, and a mean command response time of about 376.1 ms (in rare cases the response time exceeded 1 s). This is encouraging for the distributed control approach, yet worst-case fail-operational behavior has to be implemented by hovering.

Book and Text Recognition Experiments. We mimic the bookshelf by a printed poster consisting of three layers (see Figure 3). Compared to our previous work [3], we stabilized flight behavior by reducing the speed to 6 – 8% of the maximum speed, pre-flight trimming of the drone to minimize drifting, and forcing small hovering pauses. We run ten scanning experiments performing book and text recognition. As detailed in Table 3, with navigation improvements the achieved true positive rate increases up to 0.83 for book detection (mean value). Concerning text recognition, book titles detected

Figure 3: Experiment setup: poster-bookshelf (with floor markers for improved basic drone stability when used on plain colored floors).

Table 3: Mean number of books found (true positive rate) when scanning 66 books at a distance of 1m running basic and advanced navigation (ten experiments each).

	Number of books found	True positive rate
Basic nav. [3]	21	0.32
Advanced nav.	55	0.83

by Google ML toolkit are compared with the actual book title list. Spelling mistakes are allowed up to a Levenshtein distance of about 20% of the detected text length, which allows to derive the book title during post-processing. With this setting, the true positive rate of detected titles is about 0.74.

Human Interactions. Human manual navigation is mainly needed for (i) finding the start position for scanning, (ii) recovering from a misinterpretation of the end of one layer in the bookshelf, and (iii) improving bad recognition results by rescanning a layer. In our current implementation, voice and touch interfaces are provided to issue manual flight commands through the smartphone. Touch-based navigation is reliable, while voice commands are often not identified correctly due to ambient copter noise. In ongoing work we integrate direct interaction with the drone by hand gestures of the human.

CONCLUSIONS

We described a distributed architecture for human-drone teams that cooperate on a scanning task, namely bookshelf inventory. The system consists of an off-the-shelf drone with camera and Wi-Fi connectivity and a smartphone app, which provides interfaces for the human, book recognition, and vision-based navigation control. We summarized communication performance results and described calibration options of the drone's flight behavior. These enhancements improve scanning performance in terms of number of detected books (0.83 true positive rate) and number of recognized book titles (0.74 true positive rate). In future work, we plan to integrate gesture-based interaction of humans and drones and to conduct studies on user experience and user acceptance.

ACKNOWLEDGMENTS

We thank Johannes Krahofer and Simon Karlicher for initial implementations and testing.

REFERENCES

[1] M. Asadpour, B. Van den Bergh, D. Giustiniano, K. A. Hummel, S. Pollin, and B. Plattner. 2014. Micro Aerial Vehicle Networks: An Experimental Analysis of Challenges and Opportunities. *IEEE Communications Magazine* 52, 7 (July 2014), 141–149. https://doi.org/10.1109/MCOM.2014.6852096

[2] S. Chaumette and F. Guinand. 2017. Control of a Remote Swarm of Drones/Robots Through a Local (Possibly Model) Swarm: Qualitative and Quantitative Issues. In *14th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN '17)*. ACM, New York, NY, USA, 41–45. https://doi.org/10.1145/3134829.3134840

[3] K.A. Hummel, M. Pollak, and J. Krahofer. 2019. A Distributed Architecture for Human-Drone Teaming: Timing Challenges and Interaction Opportunities. *Sensors* 19, 6 (2019), 13. https://doi.org/10.3390/s19061379

[4] W. Jensen, S. Hansen, and H. Knoche. 2018. Knowing You, Seeing Me: Investigating User Preferences in Drone-Human Acknowledgement. In *2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 365, 12 pages. https://doi.org/10.1145/3173574.3173939

[5] K. McGuire, M. Coppola, Ch. de Wagter, and G. de Croon. 2017. Towards Autonomous Navigation of Multiple Pocket-drones in Real-world Environments. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, New York, NY, USA, 244–249. https://doi.org/10.1109/IROS.2017.8202164