Object-Centric Camera Drone Control for Unconstrained Telepresence

Jiannan Li
University of Toronto
40 St George St, Toronto, ON,
Canada
jiannanli@dgp.toronto.edu

Ravin Balakrishnan
University of Toronto
40 St George St, Toronto, ON,
Canada
jiannanli@dgp.toronto.edu

Tovi Grossman
University of Toronto
40 St George St, Toronto, ON,
Canada
tovi@dgp.toronto.edu

Interdisciplinary Workshop on Human-Drone Interaction (iHDI 2020)
CHI '20 Extended Abstracts, 26 April 2020, Honolulu, HI, US

Abstract
Camera drones, a rapidly emerging technology, offer people the ability to remotely inspect an environment with a high degree of mobility and agility. However, manual remote piloting of a drone is prone to errors. In contrast, autopilot systems are not necessarily designed to support flexible visual inspections. We propose the object-centric control paradigm for efficient camera drone navigation, in which a user directly specifies the navigation of a drone camera relative to a specified object of interest. We demonstrated the strengths of this approach through our first prototype, StarHopper, and discuss future research opportunities.

Author Keywords
Drone; telepresence; object-centric

Introduction
Researchers in telepresence have long envisioned 'beyond being there' [1]. Replicating all relevant local experiences, while remote, should not be the only goal of telepresence; rather, we should also strive to create telepresence systems which can enable benefits that are not possible when the person is physically present. As such, telepresence goes from replication to augmentation. One particular instance of this vision is enabled by camera drones: our local bodies can only walk on the ground, but our remote bodies can fly.

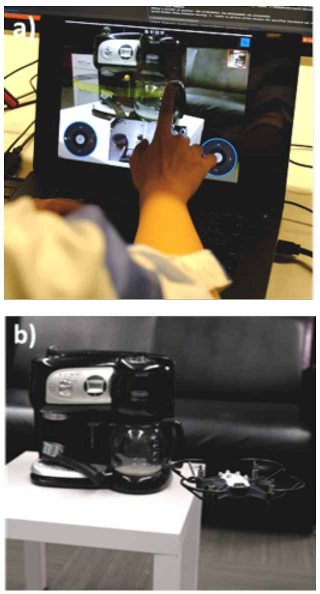

Figure 1: Operating a camera drone remotely to inspect an apartment. (a) The user specifies a desired view of the coffee machine by dragging on the drone's camera view (b) the drone flies towards to the specified viewpoint.

Researchers have noted a number of social and functional issues due to the insufficient mobility of current remote robotic presence platforms [15]. With drones becoming more affordable and reliable, they hold the potential for enabling more flexible remote presence and visual inspection experiences (e.g. [2]) for the general population.

While drones offer promise for such telepresence applications, they are challenging to manually control remotely, due to numerous factors including high degrees of freedom, narrow camera field-of-views, and network delays [7]. Their control interfaces - virtual or physical joysticks for consumer drones - are also unfamiliar for many users and take extended training time to master [9].

To relieve the burden of manual piloting, autopilot techniques have been applied to drone control. Most existing drone autopilot interfaces are based on specifying a series of planned waypoints in a 2D or 3D global map (e.g. [9]). However, in a situation where a user wishes to perform a real-time inspection, setting waypoints a priori may not be efficient for producing the viewer's desired viewpoints. Some autonomous systems avoid the use of waypoints and execute higher-level plans, such as following a subject to form canonical shots [3], but they typically do not offer the flexibility for exploring remote environments.

The difficulty of drone piloting poses a significant barrier for the widespread adoption of free-flying robots. The goal of this research is to design a camera drone control interface to support efficient and flexible telepresence experience. Our work is inspired by decades of research in interactive graphics, for which many camera navigation techniques have been established (e.g. [5]).

Most relevant, we build upon object-centric techniques, where zooming, panning, and orbiting occurs relative to

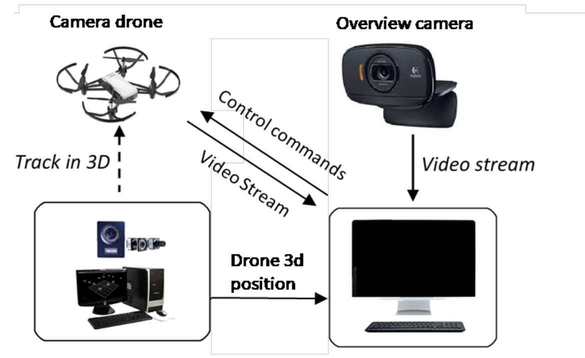

Figure 2: StarHopper system components.

the location of a 3D object of interest. We demonstrated the potential of this approach through our first prototype, StarHopper [6], and illustrate future research opportunities.

Previous Work: An Object-Centric Interface for Remote Inspection

StarHopper is a remote object-centric camera drone navigation interface that is operated through familiar touch interactions and relies on minimal geometric information of the environment (Figure 1). It consists of an overhead camera view for context and a 3D-tracked drone's first-person view for focus (Figure 2). New objects of interest can be specified through simple touch gestures on both camera views. We combine automatic and manual control via four navigation mechanisms that can complement each other with unique strengths, to support efficient and flexible visual inspection. The system focuses on indoor environments, representative of tasks such as remote warehouse inspection [9] and museum visits [11], and where positional tracking

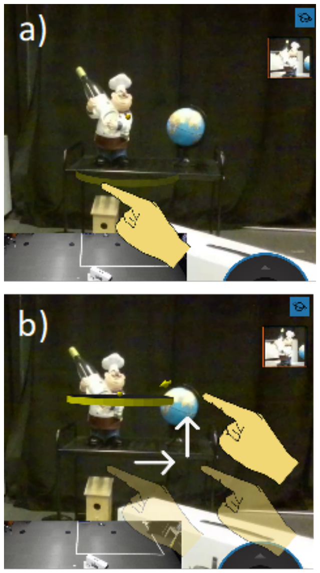

Figure 4: Interaction with the 360 viewpoint widget. (a) The user touches the area around the ring to activate the widget. (b) The user drags the finger to adjust the viewing angle and camera height. Upon releasing the drag, the drone navigates to the specified viewpoint.

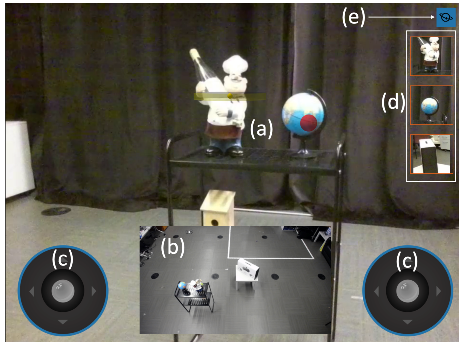

Figure 3: The StarHopper user interface. (a) Remote drone camera view. (b) Overview camera view. (c) Virtual joysticks. (d) Object-of-interest list. (e) Icon for object-centric mode.

technology is more reliable.

Design Guidelines
We base our design for remote object-centric drone navigation on a set of guidelines grounded by our review of prior literature. These guidelines are: (1) support situation awareness (2) minimize reliance on environmental information (3) combine automated and manual control (4) Support simple touch interactions (5) Respect physical constraints.

User Interface and Navigation Mechanisms
StarHopper provides a touch screen interface for the users to view the drone's live stream video and to perform drone navigations (Figure 3). The drone camera feed fills the screen. The overview camera video and two virtual joysticks are at the bottom of the interface.

The user can obtain the approximate position and dimensions of an object through a simple two-step procedure, without using pre-built maps or expensive real-time 3D reconstruction methods. She selects the object of interest through a drag gesture first in the overview camera view and then in the drone camera view. A computer vision algorithm triangulates the position of the object from these two regions and estimates the dimensions of a bounding cylinder of the object (see [6] for more technical details).

Inspired by camera control mechanisms in interactive graphics, we have designed three object-centric physical camera navigation mechanisms for viewing an object of focus: 360 viewpoint widget, delayed through-the-lens control, and object-centric joysticks.

360 Viewing widget
The 360 viewpoint widget is a widget for quickly navigating to and focusing on an object of interest, from a user-specified viewing angle. The widget takes the shape of a semi-transparent 3D ring, surrounding the focus object (Figure 4a). A 3D arrow aimed at the ring appears upon touch, indicating the desired viewing direction. The user can drag the finger on the ring to set the desired viewpoint position (Figure 4b). Once the user releases the finger, the autopilot system moves the drone to the calculated viewpoint. The algorithm determines a reasonable default viewing distance, based on the size of the bounding cylinder.

Delayed through-the-lens control
To use this technique, the user first rests two fingers on the drone camera view to freeze the current frame (Figure 5a, next page). The user then performs a two-finger pinch-and-pan gesture to transform the current frame to the desired viewpoint (Figure 5b, next page). The system then calculates a new drone position that can produce the desired viewpoint which the drone navigates towards.

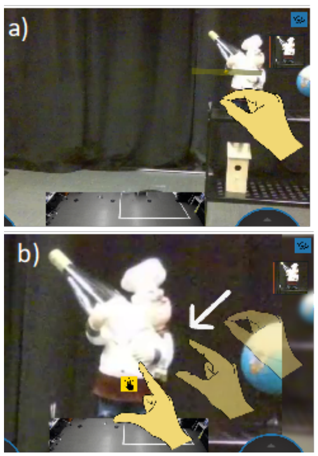

Figure 5: Adjusting the camera view using delayed through-the-lens control. (a) The user rests two fingers on the screen to freeze the current view. (b) A pan and zoom gesture on the frozen frame specifies the desired view.

Object-centric joysticks
We remap the axis of traditional drone control joysticks to object-centric commands and add constraints to prevent manipulation errors. More specifically, under the object-centric constraints, the drone keeps the object of interest in its field-of-view during the pan movements (Figure 7a, next page). In object-centric zoom, the drone aims its camera at the object of interest and moves closer or further away from it (Figure 7b, next page). In response to the orbiting commands, the drone orbits around the object while aiming at its center (Figure 7c, next page).

Manual joysticks
In addition to the three object-centric navigation mechanisms, StarHopper also supports fully manual controls. This could be useful in cases where the user wishes to make slight adjustments to a viewpoint that the auto-pilot system navigated to.

Managing objects of interest
The object-of-interest list on the right of the interface (3d) records the thumbnails of all previously registered objects of interest. The user can tap on the thumbnail to set it as the object-of-interest, and the drone will turn towards it. A double-tap on the thumbnail will trigger the drone to approach that object.

Navigation mechanism properties
StarHopper consists of a set of four navigation mechanisms, ranging from fully automated to fully manual. This suite of techniques allows users to perform both flexible and efficient scene inspections by leveraging their contrasting capabilities (Table 1, page 6).

We recognize the trend that a higher automation level increases efficiency but reduces flexibility. Taken together, the system offers the user both efficient and flexible navigation mechanisms (Table 1). The 360 viewpoint widget, despite its high efficiency, lacks in flexibility and can be complemented by delayed through-the-lens control, object-centric joysticks and manual controls.

User Study
To evaluate the navigation mechanisms of StarHopper, we conducted a user study consisting of a remote object inspection task with 12 volunteers (7 female, $M_{age} = 26.3, SD_{age} = 4.4$). A Ryze Tello drone was used in the study. We compared StarHopper to a baseline, consisting of conventional manual joystick controls. In each trial, the participant was instructed to fly the drone from the starting position to inspect one of the four sides (Left, Right, Front, Back) of an item (Figure 8) using one of the two control interfaces, StarHopper or manual joysticks (Manual). We recorded the completion time of each trial.

A repeated measures analysis of variance showed that it was significantly faster to complete the task with StarHopper than with egocentric manual control ($F_{1,11} = 23.8, p < 0.001$). Overall StarHopper was 35.4% faster (StarHopper: 20.33s, Manual: 31.45s), demonstrating a substantial gain in efficiency (Figure 6).

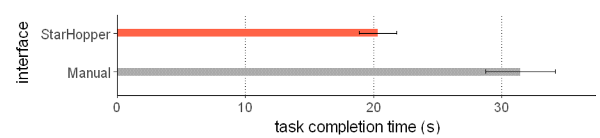

Figure 6: Mean task completion time of manual control and StarHopper. Error bars represent 95% CI.

Future Research Opportunities
The StarHopper prototype demonstrated the potential efficiency advantage for object-centric camera drone con-

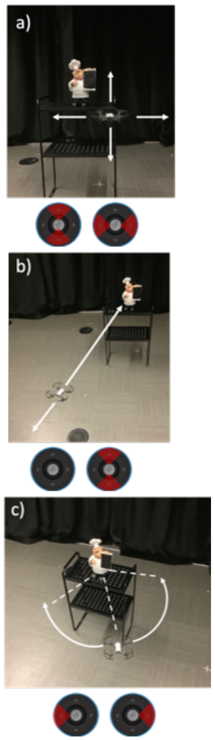

Figure 7: The object-centric joystick controls. Red areas indicate the joystick axes used. (a) Pan. (b) Zoom. (c) Orbit.

trol. More importantly, it revealed several future challenges and opportunities for better leveraging the object-centric paradigm for unconstrained telepresence.

Increasing Situation Awareness for Leveraging Greater Mobility

Supporting situation awareness has long been a key theme in robot teleoperation research ([14]). With greater mobility, drone operators face greater risk of getting lost in space ([7]). Prior research has shown the effectiveness of a live exocentric overview for enhancing situation awareness in teleoperation (e.g. [8]). StarHopper incorporated a static live overview camera, but this setup reduced the area where the drone could fly. Future research can explore awareness mechanisms that do not sacrifice mobility. For example, a second, spatially coupled camera drone serving as the overhead camera [10].

Richer Interaction Using Objects-of-Interest Semantic Information

Interactions with objects-of-interest in StarHopper were limited to specifying desirable viewpoints, as StarHopper only exploited simple geometric information. With recent advancements in image understanding, a natural next step would be enabling richer and more meaningful interactions using semantic information about objects of interest. For instance, instead of following the single rule of placing the object at the center of the camera frame by default, the system could choose a more appropriate camera framing and trajectory depending on the object and relevant context. The drone could focus on the upper body of a person in conversation, or zooming in on the console of an instrument for key readings.

Design for Local Users

While tele-operated robots give the ability to control viewpoints back to remote users, they raise challenges of accurately interpreting remote users' actions and intentions for local users. Such challenges are exacerbated on drones as their movements and form factors can be very different from humans. Recent research proposed signaling drone motion intent with augmented reality [12]. However, future flight paths and waypoints can be insufficient for a remote user who operates the drone to establish common ground with a local user, for example, when they want to make sure that they are discussing about the same object among a number of candidates in the environment. Visualizing objects of interest can complement the above signaling method and facilitate communication.

Privacy Considerations

A free-roaming viewpoint, such as a drone, raises privacy concerns about remote users intentionally or unintentionally seeing private visual information of local users. Prior research in video-mediated communication has looked extensively into privacy issues, but largely for fixed cameras (e.g. [4]). Privacy research on drones mostly studied perceptions about drones operated by strangers (e.g. [13]). Drones for telepresence, especially drones that work closely with humans, call for new privacy mechanisms. Local users can define sensitive objects or zones, which remotely operated drones should always avert.

Conclusion

Remotely operated camera drones hold potential for unconstrained telepresence 'beyond being there' but require careful control interface designs to realize such potential. Through prototyping and evaluating StarHopper, we showed the advantage of the object-centric control paradigm for camera drone teleoperation. We further invite the research community to consider future opportunities in applying the object-centric paradigm to develop more useful and usable camera drone control interfaces for unconstrained

StarHopper Navigation Mechanisms	Automation Level	Efficiency	Flexibility
360 Viewpoint Widget	Automated	High	Low
Delayed Through-the-Lens	Semi-Automated	Medium	Medium
Object-Centric Joysticks	Semi-Automated	Medium	Medium
Manual Joysticks	Manual	Low	High

Table 1: Properties of the four control mechanisms.

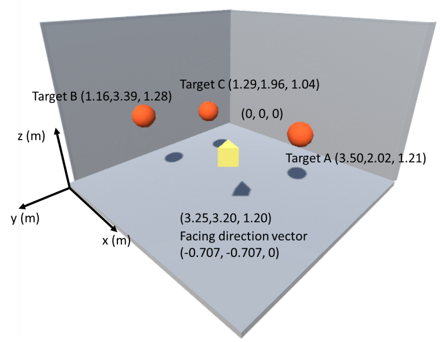

Figure 8: Mean task completion time of manual control and StarHopper. Error bars represent 95% CI.

telepresence.

REFERENCES

[1] Jim Hollan and Scott Stornetta. 1992. Beyond Being There. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (1992), 119–125. DOI: http://dx.doi.org/10.1145/142750.142769 event-place: Monterey, California, USA.

[2] Brennan Jones, Kody Dillman, Richard Tang, Anthony Tang, Ehud Sharlin, Lora Oehlberg, Carman Neustaedter, and Scott Bateman. 2016. Elevating Communication, Collaboration, and Shared Experiences in Mobile Video Through Drones. *Proceedings of the 2016 ACM Conference on Designing Interactive Systems* (2016), 1123–1135. DOI:http://dx.doi.org/10.1145/2901790.2901847 event-place: Brisbane, QLD, Australia.

[3] Niels Joubert, Jane L. E, Dan B. Goldman, Floraine Berthouzoz, Mike Roberts, James A. Landay, and Pat Hanrahan. 2016. Towards a Drone Cinematographer: Guiding Quadrotor Cameras using Visual Composition Principles. *arXiv:1610.01691 [cs]* (5 10 2016). http://arxiv.org/abs/1610.01691 arXiv: 1610.01691.

[4] Tejinder K. Judge, Carman Neustaedter, and Andrew F. Kurtz. 2010. The family window: the design and evaluation of a domestic media space. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (10 4 2010), 2361–2370. DOI: http://dx.doi.org/10.1145/1753326.1753682 [Online; accessed 2020-02-10].

[5] Azam Khan, Ben Komalo, Jos Stam, George Fitzmaurice, and Gordon Kurtenbach. 2005. HoverCam: Interactive 3D Navigation for Proximal Object Inspection. *Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games* (2005), 73–80. DOI: http://dx.doi.org/10.1145/1053427.1053439 event-place: Washington, District of Columbia.

[6] Jiannan Li, Ravin Balakrishnan, and Tovi Grossman. 2020. StarHopper: A Touch Interface for Remote Object-Centric Drone Navigation. *Proceedings of Graphical Interface 2020* (2020).

[7] David Pitman and Mary L. Cummings. 2012. Collaborative Exploration with a Micro Aerial Vehicle: A Novel Interaction Method for Controlling a Mav with a Hand-held Device. *Adv. in Hum.-Comp. Int.* 2012 (1 2012), 18:18–18:18. DOI: http://dx.doi.org/10.1155/2012/768180

[8] D. Saakes, V. Choudhary, D. Sakamoto, M. Inami, and T. Lgarashi. 2013. A teleoperating interface for ground vehicles using autonomous flying cameras. *2013 23rd International Conference on Artificial Reality and Telexistence (ICAT)* (12 2013), 13–19. DOI: http://dx.doi.org/10.1109/ICAT.2013.6728900

[9] Daniel Szafir, Bilge Mutlu, and Terrence Fong. 2017. Designing planning and control interfaces to support user collaboration with flying robots. *The International Journal of Robotics Research* 36, 5-7 (1 6 2017), 514–542. DOI: http://dx.doi.org/10.1177/0278364916688256

[10] Ryotaro Temma, Kazuki Takashima, Kazuyuki Fujita, Koh Sueda, and Yoshifumi Kitamura. 2019. Third-Person Piloting: Increasing Situational Awareness Using a Spatially Coupled Second Drone. *Proceedings of the 32Nd Annual ACM Symposium on User Interface Software and Technology* (2019), 507–519. DOI: http://dx.doi.org/10.1145/3332165.3347953 event-place: New Orleans, LA, USA.

[11] S. Thrun, M. Beetz, M. Bennewitz, W. Burgard, A. B. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. 2000. Probabilistic Algorithms and the Interactive Museum Tour-Guide Robot Minerva. *The International Journal of Robotics Research* 19, 11 (1 11 2000), 972–999. DOI: http://dx.doi.org/10.1177/02783640022067922

[12] Michael Walker, Hooman Hedayati, Jennifer Lee, and Daniel Szafir. 2018. Communicating Robot Motion Intent with Augmented Reality. *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction* (2018), 316–324. DOI: http://dx.doi.org/10.1145/3171221.3171253 event-place: Chicago, IL, USA.

[13] Yang Wang, Huichuan Xia, Yaxing Yao, and Yun Huang. 2016. Flying Eyes and Hidden Controllers: A Qualitative Study of People's Privacy Perceptions of Civilian Drones in The US. *Proceedings on Privacy Enhancing Technologies* 2016, 3 (1 7 2016), 172–190. DOI:http://dx.doi.org/10.1515/popets-2016-0022

[14] H. A. Yanco and J. Drury. 2004. "Where am I?" Acquiring situation awareness using a remote robot platform. *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)* 3 (10 2004), 2835–2840 vol.3. DOI: http://dx.doi.org/10.1109/ICSMC.2004.1400762

[15] Lillian Yang, Brennan Jones, Carman Neustaedter, and Samarth Singhal. 2018. Shopping Over Distance Through a Telepresence Robot. *Proc. ACM Hum.-Comput. Interact.* 2, CSCW (11 2018), 191:1–191:18. DOI: http://dx.doi.org/10.1145/3274460