2017 26th IEEE International Symposium on
Robot and Human Interactive Communication (RO-MAN)
Lisbon, Portugal, Aug 28 - Sept 1, 2017.

Monocle: Interactive Detail-in-Context Using Two Pan-and-Tilt Cameras to Improve Teleoperation Effectiveness

Stela H. Seo[1] Daniel J. Rea[1] Joel Wiebe[2] James E. Young[1]
Computer Science
University of Manitoba
[1]{stela.seo, daniel.rea, young}@cs.umanitoba.ca, [2]joel.p.wiebe@gmail.com

***Abstract*—Robot teleoperation, such as for search and rescue, uses multiple specialized cameras (e.g., wide environmental and sharp narrow views) to aid in task awareness. Simple display techniques, such as tiling, require ongoing mental mapping between the views; cameras that pan or tilt exacerbate the problem as the inter-view relationship changes. The detail-in-context technique bypasses this mental mapping requirement by providing a single integrated feed showing all cameras, with detail overlaid within the context. However, how this can be adapted to for robot teleoperation with multiple pan-and-tilt cameras has not yet been demonstrated. We present Monocle, an interactive detail-in-context teleoperation interface that integrates a pan-and-tilt narrow-angle first-person view into a wide-angle behind-robot view; operators can move the Monocle around a scene to obtain more resolution when and where needed. Evaluation results demonstrate Monocle's feasibility and show that it can help operators complete search and rescue tasks more effectively in comparison to simple solutions.**

Keywords—Human-robot interaction; Teleoperation; Video-centric interface; Urban search and rescue; Detail-in-context

I. Introduction

Teleoperation robots for professional tasks are continuing to emerge into fields including industrial inspection, urban search and rescue, and military operations. These robots enable people to remotely investigate dangerous and hard to reach places, such as industrial complexes (water pipes, nuclear reactors, etc.) or disaster sites for search and rescue. Improving teleoperation interfaces for increased effectiveness and lower operator effort is an ongoing research challenge in human-robot interaction [1], [2]; improving operator performance may result in more saved lives, reduced operator stress, and reduced costs.

Teleoperators control robots through unknown and dangerous terrain while monitoring multiple camera feeds into the remote environment (e.g., [3], [4]). Multiple, specialized views improve task effectiveness [5], e.g., wide angle, behind-robot views show the robot within the environment (e.g., to help navigate holes and obstacles) and help with visual search (e.g., showing large areas at once) [4], [6]. Narrow views are useful for close inspection, for instance, of equipment or potential survivors [1], [2], [4]. However, providing multiple feeds to an operator increases the required mental demand in comparison to a single feed, as they need to monitor both feeds, and when they change focus, perform a mental mapping between the feeds to maintain awareness of how they relate; this mental overhead impacts task effectiveness [2], [7], [8].

Our solution is to leverage the information visualization paradigm *detail-in-context* [9] (also known as *focus-plus-context* [10]) to have a single view with the appropriate detail

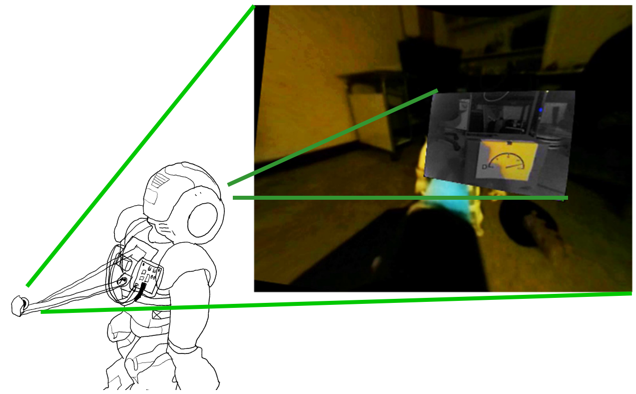

Fig. 1. Monocle, a fully interactive detail-in-context teleoperation interface with wide pannable (behind robot) and narrow pan-and-tilt (in robot head) views. An operator can move the monocle around to gain detail where needed. (Low light, image quality due to search and rescue scenario).

(narrow, high resolution camera feed) interactively embedded in context (wider camera feed), removing the need for users to mentally map between related views. Rather than the typical static picture-in-picture (e.g., screen corner), side-by-side, or static image-stitching mosaics, our novel teleoperation interface *Monocle* integrates two interactive and steerable cameras into a single real-time, interactive, mixed resolution view (Fig. 1). Monocle enables users to move both the wide environment and the high-resolution cameras around a scene as needed, while maintaining a single integrated view that does not require mental mapping.

We present results from a formal study indicating that Monocle helps operators to accomplish teleoperation tasks with fewer critical incidents (mistakes) than simpler interface solutions. Further, our detailed analysis provides insight into the trade-offs between interface approaches, beneficial for ongoing robot teleoperation research.

II. Related Work

Improving the usability of robot teleoperation is an ongoing challenge for UAV control [11], inspection [2], domestic robots [7], [12], or medical consulting [13]. A primary challenge is to improve operator performance such as task completion time [2], [4], [7], [12], [14] and to reduce errors [4]. Techniques include novel control schemes (e.g., [4], [6], [15]) that abstract away low-level control problems to provide high-level control of the robot [2], [14], [15], or improved data presentation [6], [8], [11], [12]. A common goal of these techniques is to increase the information available to the operator while minimizing cognitive load. Monocle focuses on mitigating these problems in the case of a robot with two complementary cameras.

While multiple views can provide more information and thus increase situational awareness, it can increase operator cognitive load [3], [8], [16]. Providing additional context can mitigate this problem, such as SLAM-based or pre-built 3D maps [12], [17], or sonar [18]. Cameras external to the robot, such as third person views on a boom or flying robot [4], or elsewhere in the room [15], have been used to provide a contextual, exocentric view to improve navigation. Monocle relies on on-robot 2D camera feeds without pre-built maps or environmental cameras, and tackles the problem of how to effectively provide these simultaneous views in real-time while minimizing cognitive load.

Software solutions use rendered transformations and mark-up, such as projected views or virtual camera models, to indicate the physical relationship between camera views [19]. Monocle bypasses this problem by combining the feeds, removing the need for mental mapping. While image stitching techniques are similarly used to fuse multiple camera images into one [1], [20], [21], this work has important differences: 1) Monocle uses a mixed-resolution view instead of aiming for one homogenous result image to maintain important detail, 2) Monocle is real-time interactive with the both views being direct-able, and 3) provides the results from a formal study of the use of this technique in a search and rescue scenario.

Ecological interface design has been used to support the mapping and situational awareness needs of particular tasks, e.g., fusing sensor data into single task-oriented displays and widgets [22]. Camera-specific work has embedded a single video feed within virtual environments to show a robot on a 2D map or within a 3D environment, and projecting the camera's view at the correct angle and location within the scene [4], [12], [17]. Monocle builds on this work and uses two camera feeds instead of virtual 3D environments with one feed, taking a lighter weight (no SLAM processing, additional sensors or environmental camera), high fidelity (camera feed instead of constructed environment), and portable (no map required a priori) approach.

Much of detail-in-context work deals with how to visually transition between scales, e.g., between a large scale map and close-up region; a challenge is to use distortions that highlight the transition in a legible fashion [23]. Monocle is single scale, but multi-resolution (sharp view under the monocle), bypassing this requirement; while others have explored mixed-resolution data sources [10], Monocle extends this to teleoperation in a search and rescue scenario. Monocle is the first system to integrate and evaluate two direct-able live camera feeds into a single mixed-resolution, real-time interactive display, providing detail-in-context interface that enables operators to simultaneously view the context of their robot operation while gaining finer detail where needed, as needed (Fig. 2).

III. Monocle Interface

Monocle presents a single video feed to the operator that embeds a sharp narrow view (first-person, in-robot head), into a wide environment view (behind the robot); the sharp view is overlaid onto the wide view at the corresponding location, forming an integrated mixed-resolution feed. This integration removes the requirement for the operator to mentally map between two image feeds, with respect to the robot and the remote environment (Fig. 1 and Fig. 2).

The operator can steer both cameras to look and get detail where needed, with the sharp view movable similar to a monocle on a map. The wide camera can pan, while the sharp camera can pan and tilt. The integration automatically updates as the cameras move, with the sharp overlay calibrated within the wide view. If the sharp image is moved over the robot (thus obstructing it, Fig. 2), it can be toggled off to see through the robot under the operator's desire.

A. Robot Platform

We use the Aldebaran NAO H25 humanoid, controlled using the NaoQi API and in-house remote control software. While not a search and robot, it provides stable locomotion and camera operation in our environment and so is appropriate for the implementation and evaluation.

We use the NAO's built-in first-person camera for our sharp view (pan and tilt control moves the robot's head), and Raspberry Pi's camera for wide view. We mount a lightweight boom (60 cm) and servo (120° pan) on the NAO's back to hold a 180° fish eye lens; the boom is pointed slightly downward to show the robot in the environment (Fig. 3). We use OpenCV 2.4.1.

B. Implementation

Common techniques for stitching multiple images together [24] use feature matching and transforms to warp one image onto another. More recently, parallax-tolerant methods (e.g., [24]) address issues of cameras being at different locations, as in our case. Through extensive testing, we found current packaged image stitching methods to be unstable and

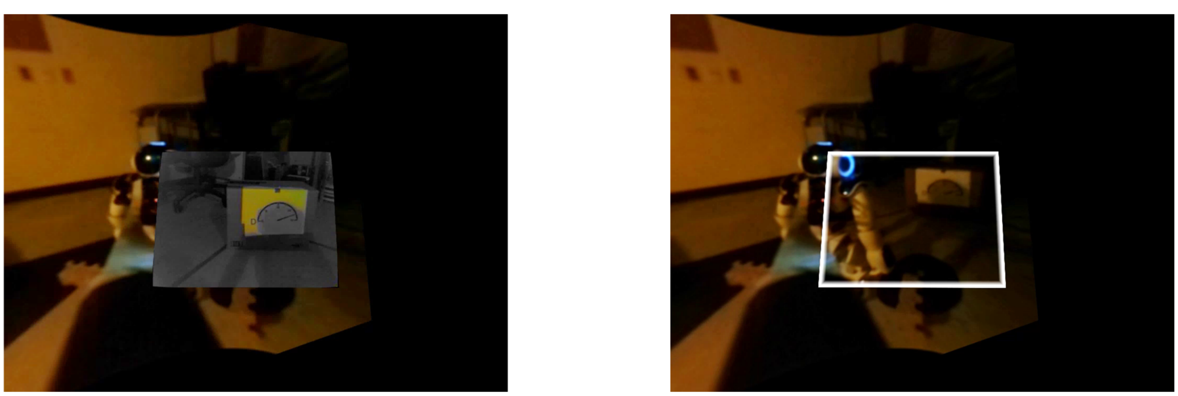

Fig. 2. Monocle detail-in-context interface. the sharp-view (left) integrates into the wide view; (right) the monocle toggled off by the operator. Third-person view provides wide context but at a reduced clarity due to limited resolution. The first-person view has a narrow field-of-view to provides more clarity.

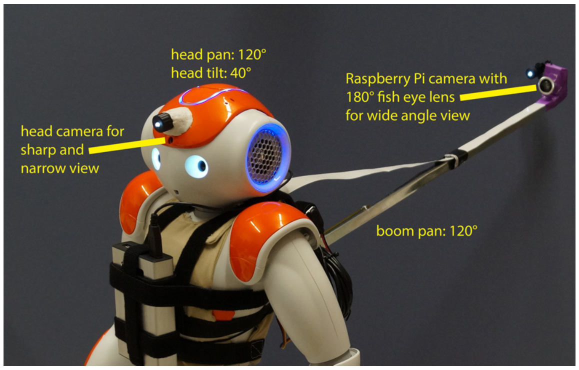

Fig. 3. Robot with boom; boom camera is above the robot, pointed downward to capture the full body. Both cameras have LED lights attached to aid in dark environments.

not robust enough for use in our highly-dynamic real-time streaming scenario with unstable lighting, a moving robot, and two steerable cameras. Even temporary or sporadic loss of calibration is unacceptable in search and rescue and would hinder our evaluation results. For our purposes, we implemented simple stitching using camera odometry and linear interpolation between calibrated transforms at fixed places in the scene. We leave advanced image processing stitching techniques as future work.

IV. Evaluation

We conducted an experiment comparing Monocle to two simple methods of combining multiple feeds, *toggle* and *side-by-side*. Toggle provides one full-screen view at a time (one of each in Fig. 4) selectable by the operator, and side-by-side provides two views simultaneously (as in Fig. 4).

A. Apparatus

Participants used a PC with 27-inch monitor and a dual-axis Xbox 360 controller. The chair and monitor locations were fixed although participants were free to lean in or back to get comfortable. Participants used the left controller joystick to move the robot, the right stick to steer the sharp view (robot head), and buttons to pan the wide view (boom) left and right, to swap between sharp and wide view (toggle interface), or to toggle the sharp view on and off in Monocle. A print-out legend of the controls was provided.

B. Comparison Interfaces

The toggle interface (one camera feed at a time, with a button to go between them) was designed to maximize screen real-estate (full-screen feeds) and reduce distraction by having only a single view at a time. The side-by-side interface (based on [2], two views side by side, sharing screen real estate) was designed to provide more simultaneous information than toggle, while reducing individual feed size.

We incorporated a compass-like feature to the toggle and side-by-side interfaces to indicate to the operator which way the robot will walk if moved (bottom of left image, Fig. 4), as pilot studies indicated that this information was critical for operation; this was not necessary in Monocle interface.

C. Task

We implemented a simulated disaster environment (Fig. 5) with holes (black cutouts on the floor), debris (cardboard blocks to trip the robot), survivors (plush toys, red tape to indicate blood), and equipment gauges (boxes); all boxes had a paper attached, inspection is required to tell if a gauge is present. The room had low ambient lighting to improve realism, and the robot was equipped with two LED lights next to the cameras.

Participants have to navigate through the environment from start to finish (Fig. 5) and find as many gauges and victims as possible. While the route was fairly linear, participants have to wander to inspect boxes, look behind obstacles, and to avoid debris and holes. When the robot hit an obstacle or hole, the screen would pulse red to inform them of the problem; this was implemented as pilot study results suggested that participants would often ignore obstacles and holes, tripping the robot. The warning light was operated by a researcher monitoring the interaction, unbeknownst to the participant.

Participants need to record the reading and label of found gauges, requiring them to get close, and need to inform the researcher when they find a victim. The robot is not required to physically move obstacles or debris to find these, although some are more hidden than others.

D. Measures

Our aim in this experiment was to investigate the overall impact of the interfaces on operator effectiveness when performing the search and rescue task. We did not feel it was mean-

Fig. 4. Sharp narrow view (left), wide angle view (right). Green arrow indicates robot facing direction (in contrast to look direction), blue border appears indicating limitations of camera movement. Both shown at once on screen for the side-by-side interface, and one at a time for the toggle interface. Note the victim, with red simulated blood, is easier to see in the sharp view.

ingful or appropriate to focus on the specific targeted task outcomes in isolation, such as number of victims found: such results are heavily impacted by our specific implementation, e.g., perhaps our victim design favors one interface. Instead, given our interest in overall task effectiveness, our primary measure was an aggregate performance score that considers the various components of the task, including how well they completed the goals balanced with errors they made. While we do provide the detailed breakdown in our results for post-hoc discussion, we emphasize the importance of taking a more holistic view on the operator performance. We calculated our aggregate score as follows:

$$\begin{aligned} score = {} & correct\ victims\ + correct\ gauges \\ & - accidents - misreports \\ & - missed\ items \end{aligned}$$

Accidents included colliding with an obstacle or stepping into a hole, misreports were either falsely identified victims or incorrect gauge readings, and missed items included gauges and victims.

In piloting operator strategies varied between participants, for example, some lingered around to double check, while others aimed to finish quickly. As such, given the open-ended nature of the task without a clear number of victims or gauges, we did not measure task completion time.

Finally, we administered a NASA TLX questionnaire per interface, with additional questions pertaining to perception of efficiency, awareness, nausea, and enjoyment.

Post-study, participants completed a written questionnaire on the three interfaces and ranked them in terms of preference.

E. Procedure

This study was approved by our university ethics board. Participants completed an informed consent form and demographics questionnaire before starting.

Participants completed the primary task three times, once with each interface (toggle, side-by-side, and Monocle). There were three gauges and three victims placed in the environment (the number unknown to participants), with the locations of the victims and gauges changed between interface runs; the locations as well as interface order were counterbalanced between participants. Before starting with each interface, participants completed minor training it, and immediately after each interface completed the post task questionnaire.

Each task ended when the participant reached the course end and announced they are finished (they were allowed to go back), or a 12-minute time limit elapses; participants were told that time is a factor and the building may collapse, but they were not told how long they had.

After the tasks were complete, participants completed a post-test questionnaire, and we conducted a semi-structured interview to inquire about the overall experience. Finally, participants were debriefed on the experiment.

V. Results

We recruited 13 participants (3 female) from our university population, and paid them $10 for their one-hour participation.

Fig. 6 (left) shows the results of the performance scores by interface. The left shows the grand means and 95% confidence intervals across participants. We analyzed our results with repeated-measures one-way ANOVAs, with planned contrasts comparing Monocle to the two alternatives.

We found a medium-sized main effect of interface on performance score ($F_{2,24}$=3.57, p<.05, η^2=0.15) with (planned contrasts) Monocle (M=-0.1) scoring higher than toggle (M=-3.6, p<.05) and side-by-side (M=-2.5, p<.05). Fig. 6 (right) illustrates more deeply participants' relative performance with each interface.

We found no effect of interface on reported post-condition questionnaire data, NASA TLX reports, and no dominant preference for interface.

A. Participant Feedback Analysis

We conducted qualitative analysis on the written responses and semi-structured interviews to better understand the pros and cons of each interface. We extracted dominant themes from the data using iterative open coding, with initial codes inspired by our pilots. We used a single coder given our exploratory and descriptive focus (no inter-coder reliability tests).

1) Interface Simplicity and Usability

Participants commented heavily on the importance of simplicity of the interface, which was one of the primary benefits noted of side-by-side (e.g., *"both views are available without switching,"* 8/13 participants). While participants also commented on the simplicity of toggle (e.g., *"one thing to focus on at one time,"* 7/13), the requirement to toggle back and forth was seen as an issue that impacted performance (e.g. *"have to keep flipping back and forth," "the toggle was not so convenient and time consuming,"* 7/13).

For Monocle, some disliked the way that sharp view occluded the view, even when it was just an outline (e.g., *"the front screen gets in the way,"* and *"distracting view because*

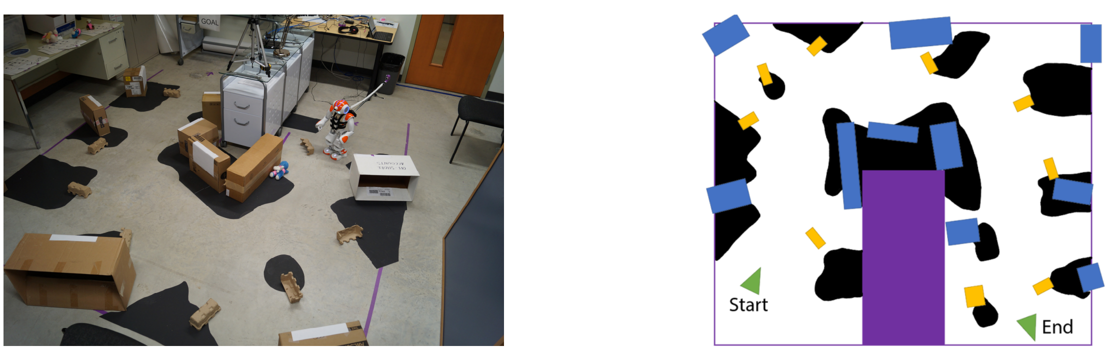

Fig. 5. USAR evaluation environment picture and map. The robot is at the start position. Blue: equipment. Yellow: rubble. Black: (fake) holes. Purple: wall.

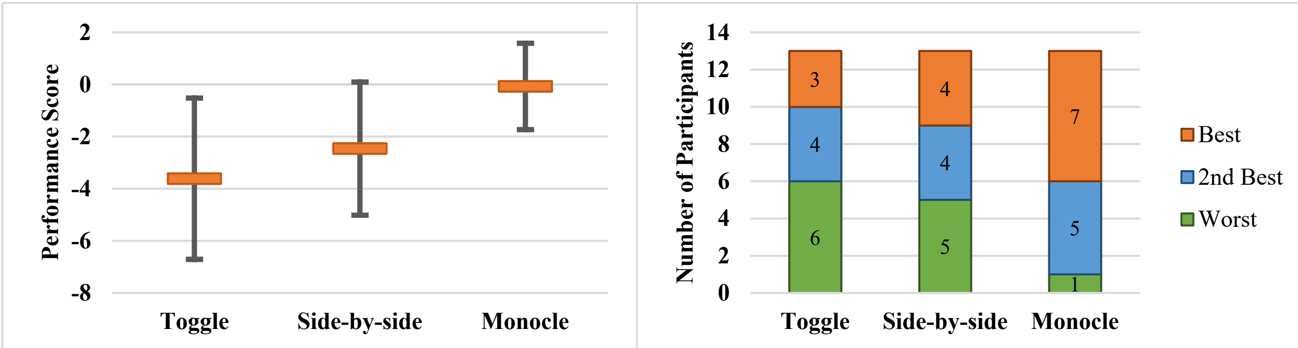

Fig. 6. (left) Overall average performance scores with 95% CI. (right) How many participants did best, second best, or worst with each interface.

of the outline of the merge camera," 6/13). Only discussed for Monocle, a few participants thought that they could perform better with Monocle if they had some time to practice (e.g., *"it takes a while to get used to overall [maneuverability] using the merged interface [Monocle]," "felt easy to use after getting the hang of it,"* 3/13).

2) Supporting Awareness

Participants described how the interfaces supported their awareness of the environment, such as the two views in side-by-side (e.g., *"You can see more around you," "can view the environment as first and third person at the same time, easier to avoid obstacles,"* 9/13 participants). This was also a cited advantage of Monocle (e.g., *"still have third person view when using other camera makes looking closer at things easier,"* 5/13), which included being able to see through the robot via the lens (e.g., *"you can see in front of the robot and know where it is in relation to the back → easier maneuvering"*).

Participants also noted the awareness gains of toggle's full screen view (e.g., *"larger visible area – easier to see," "it's good to have a bigger picture when inspecting," "it was more comfortable to see the view,"* 8/13).

3) Hindering Awareness

There were noted awareness-related issues with each interface. For toggle, participants noted that being unable to look at both views at once was a disadvantage (e.g., *"It's could be ignored one side of environment [sic.]"* and *"easier to trip on something if you aren't careful to go back-and-forth,"* 7/13 participants). Similarly, while both views were available in side-by-side, participants either found it distracting to have the two (e.g., *"Deciding which screen to look at was slightly more mentally tasking,"* 7/13), or conversely, that they'd end up focusing on one screen only (e.g., *"concentrate too much as one view at a time and forget about the other,"* 7/13).

For Monocle, some participants complained of the small size of the sharp view (e.g., *"much smaller if you want to look at detail,"* 4/13), which even impacted perception of robot capability (e.g., *"felt like the front screen was smaller [than other interfaces] in terms of motion and viewing side to side"*).

VI. Discussion

Our results highlight that Monocle improved overall USAR task performance compared to simpler interface solutions. While the results were inconclusive on the NASA TLX and self-report measures, this is a clear indication that the detail-in-context view helped our participants complete their task effectively.

To gain a better sense of what impacted the overall score difference between the interfaces, Fig. 7 shows the breakdown of the average scores. As can be seen, the number of accidents appears to be the largest change between interfaces. Intuitively, this matches the design goals of Monocle in providing clear environmental context while monitoring detail, without mental mapping, but further study is required to confirm this.

As importantly as the overall task effectiveness result, the participant feedback analysis provides insight into possible reasons behind the performance of the interfaces. We have summarized these tradeoffs into Table 1, which can be referenced with respect to the benefits and problems with each approach, and for future exploration on teleoperation interfaces.

Feedback on Monocle highlighted that, even with a 27" monitor and close seating, the smaller screen size of the sharp view may be a problem for operators. We emphasize that our source image was only 320x240, due to limitations in network bandwidth and robot capability, and that even in the smallest case the image was scaled up to match the overlap, and no resolution was lost. Despite this, participants felt perhaps that they could see more if the overlay was larger. This remains an important point of future work.

Although our results indicate that Monocle improved overall operator performance, it did not have a lower cognitive load (as per NASA TLX), and no preference emerged in participant feedback. However, as some participants mentioned, the interface is quite unique in comparison to the other two

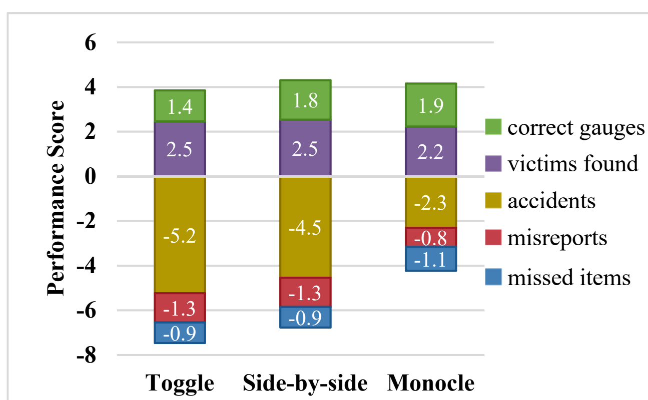

Fig. 7. Average performance scores with breakdown of positive and negative factors. As shown, the number of accidents was the largest difference between interfaces.

Interface	Advantages	Disadvantages
toggle	full screen view for detailed search low effort to focus on one view	context switching (toggle to other view) less information (one view at a time)
side-by-side	two views simultaneously support awareness simple to operate (no toggling)	two views can be distracting may over-focus on one view
Monocle	improved task effectiveness no mental mapping between views	small sharp narrow view occlusion of robot

Table 1. Summary of trade-offs emerging from the participant feedback analysis.

and may require more longitudinal study with more operator experience, before other gains are realized.

VII. Future Work

Our method for combining two camera feeds using detail-in-context is just a starting point, and there remains many open questions.

Further improvements should be made to Monocle interface to address raised concerns. For example, perhaps the small sharp-view could dynamically get larger if needed to enable an operator to look closer, perhaps using scale-transition techniques from detail-in-context. Further, techniques should be investigated to mitigate the occlusion issue, such as providing a robot wireframe through the sharp view, or improved image stitching and alpha-blending merge.

Our interface only dealt with two cameras. Robots often have more, including cameras in robot arms, panoramic cameras, etc. Extensions of Monocle to a generic space with more cameras is a non-trivial problem, particularly as cameras may be looking in completely different directions.

VIII. Conclusion

We presented Monocle, a novel interface that leverages the detail-in-context paradigm to integrate two teleoperation cameras into a single mixed-resolution view, removing the need for an operator to mentally map between the views. Our study results indicate that this technique can improve task effectiveness in a USAR scenario, and provides qualitative insight into how Monocle compares to other simple solutions.

References

[1] K. Kruckel, F. Nolden, A. Ferrein, and I. Scholl, “Intuitive visual teleoperation for UGVs using free-look augmented reality displays,” *IEEE Int. Conf. Robot. Autom.*, pp. 4412–4417, 2015.

[2] A. Singh, S. H. Seo, Y. Hashish, M. Nakane, J. E. Young, and A. Bunt, “An interface for remote robotic manipulator control that reduces task load and fatigue,” in *IEEE RO-MAN*, 2013, pp. 738–743.

[3] A. Kelly *et al.*, “Real-time photorealistic virtualized reality interface for remote mobile robot control,” *Int. J. Rob. Res.*, vol. 30, no. 3, pp. 384–404, Oct. 2010.

[4] D. Saakes, V. Choudhary, D. Sakamoto, M. Inami, T. Lgarashi, and T. Igarashi, “A teleoperating interface for ground vehicles using autonomous flying cameras,” in *Int. Conf. on Artificial Reality and Telexistence (ICAT)*, 2013, pp. 13–19.

[5] S. Hughes and M. Lewis, “Robotic camera control for remote exploration,” *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, vol. 6, no. 1, pp. 511–517, 2004.

[6] S. Hashimoto, A. Ishida, M. Inami, and T. Igarash, “TouchMe: an augmented reality based remote robot manipulation,” in *Proc. of Int. Conf. on Artificial Reality and Telexistence (ICAT)*, 2011, pp. 1–6.

[7] M. Mast *et al.*, “Semi-autonomous domestic service robots: evaluation of a user interface for remote manipulation and navigation with focus on effects of stereoscopic display,” *Int. J. Soc. Robot.*, vol. 7, no. 2, pp. 183–202, 2014.

[8] J. L. Drury, J. Scholtz, and H. a. H. A. Yanco, “Awareness in human-robot interactions,” *IEEE Int. Conf. Syst. Man Cybern.*, vol. 1, no. October, pp. 912–918, 2003.

[9] T. A. Keahey, “The generalized detail in-context problem,” in *Proc. of the IEEE Symposium on Information Visualization*, 1998, pp. 44–51.

[10] P. Baudisch, N. Good, V. Bellotti, and P. Schraedley, “Keeping things in context: a comparative evaluation of focus plus context screens, overviews, and zooming,” in *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems (CHI)*, 2002, pp. 259–266.

[11] J. Richer and J. L. Drury, “A Video Game-based Framework for Analyzing Human-robot Interaction: Characterizing Interface Design in Real-time Interactive Multimedia Applications,” in *Proc. of the 1st ACM SIGCHI/SIGART Conf. on Human-robot Interaction*, 2006, pp. 266–273.

[12] D. Labonte, P. Boissy, and F. Michaud, “Comparative analysis of 3-D robot teleoperation interfaces with novice users,” *IEEE Int. Conf. Syst. Man Cybern.*, vol. 40, no. 5, pp. 1331–1342, 2010.

[13] R. Agarwal, A. W. Levinson, M. Allaf, D. V. Makarov, A. Nason, and L.-M. Su, “The RoboConsultant: telementoring and remote presence in the operating room during minimally invasive urologic surgeries using a novel mobile robotic interface,” *Urology*, vol. 70, no. 5, pp. 970–974, 2007.

[14] A. E. Leeper, K. Hsiao, M. Ciocarlie, L. Takayama, and D. Gossow, “Strategies for human-in-the-loop robotic grasping,” in *Proc. of the 7th annual ACM/IEEE Int. Conf. on Human-Robot Interaction (HRI)*, 2012, pp. 1–8.

[15] D. Sakamoto, K. Honda, M. Inami, and T. Igarashi, “Sketch and run: a stroke-based interface for home robots,” in *Proc. of the SIGCHI Conf. on Human Factors in Computing Systems (CHI)*, 2009, pp. 197–200.

[16] H. A. Yanco and J. Drury, “‘Where am I?’ Acquiring situation awareness using a remote robot platform,” in *IEEE Int. Conf. on Systems, Man and Cybernetics*, 2004, vol. 3, pp. 2835–2840.

[17] Y. Ochiai, K. Takemura, A. Ikeda, J. Takamatsu, and T. Ogasawara, “Remote control system for multiple mobile robots using touch panel interface and autonomous mobility,” *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, no. Iros, pp. 3272–3277, 2014.

[18] S. Salmanipour and S. Sirouspour, “Teleoperation of a mobile robot with model-predictive obstacle avoidance control,” in *Annual Conf. of the IEEE Industrial Electronics Society (IECON)*, 2013, pp. 4270–4275.

[19] D. Ribeiro *et al.*, “Analysis of choreographed human movements using depth cameras: a systematic review,” in *Human-Computer Interaction. Interaction Platforms and Techniques*, 2016, pp. 82–92.

[20] M. Uyttendaele, A. Criminisi, S. B. Kang, S. Winder, R. Szeliski, and R. Hartley, “Image-based interactive exploration of real-world environments,” *IEEE Comput. Graph. Appl.*, vol. 24, no. 3, pp. 52–63, 2004.

[21] L. Lee, R. Romano, and G. Stein, “Monitoring activities from multiple video streams: Establishing a common coordinate frame,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 758–767, 2000.

[22] C. Borst, H. C. H. Suijkerbuijk, M. Mulder, and M. M. van Paassen, “Ecological interface design for terrain awareness,” *Int. J. Aviat. Psychol.*, vol. 16, no. 4, pp. 375–400, 2006.

[23] J. Böttger, M. Preiser, M. Balzer, and O. Deussen, “Detail-in-context visualization for satellite imagery,” *Comput. Graph. Forum*, vol. 27, no. 2, pp. 587–596, Apr. 2008.

[24] F. Zhang and F. Liu, “Parallax-tolerant image stitching,” in *The IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 3262–3269.