Virtual 360° Panorama for Remote Inspection

Mattias Seeman, Mathias Broxvall, Alessandro Saffiotti
AASS Mobile Robotics Lab
Department of Technology, Örebro University
SE-70182 Örebro, Sweden
{mattias.seeman,mbl,asaffio}@aass.oru.se

Abstract — The use of remotely operated robotic systems in security related applications is becoming increasingly popular. However, the direct teleoperation interfaces commonly used today put a large amount of cognitive burden on the operators, thus seriously reducing the efficiency and reliability of these systems. In the context of an adjustable autonomy control architecture meant to relieve operators from unnecessary low-level tasks, we present an user interface technique for 360° virtual panorama video as a perception aid to increase the situation awareness in tele-operation tasks, and as a block in the overall adjustable autonomy control architecture. At the hardware level, we rely on the intrinsic autonomy and robustness provided by the spherical morphology of our GroundBot robot. The work presented here is a step towards the overall goal of increasing the effectiveness of the GroundBot robot for remote inspection tasks.

I. Introduction

The use of remotely operated robots for inspection is desirable whenever the tasks are *dirty, dull, and dangerous*, otherwise known as fulfilling the *three D's*. The use of a robotic system allows efficient remote sensing and actuation, and keeps the human operator out of harm's way. However, the utility of the system is contingent on keeping the operator's situation awareness at an adequate level. It is therefore of interest to investigate techniques that make user interfaces intuitive to use and let operators maintain a high degree of situation awareness, thereby improving the robustness and reliability of teleoperated robotics.

Here we present a high-resolution panoramic video technique, meant to increase the operator's awareness of the robot's surroundings. The panorama is a feature of the user interface to our spherical (ball-shaped) robot seen in Figure 1, the GroundBot. Ball-shaped robots have a number of favourable advantages, such as stability and robustness, also noted by Suomela and Ylikorpi [1], that make them especially suitable for the types of applications we are considering [2], [3]:

- Security surveillance of airports, industrial plants, etc.
- Disaster area inspection.
- Human search and rescue.

The tasks above are characterized by the presence of two conflicting requirements regarding the modalities of human-robot interaction.

On the one hand, one would like to delegate as much responsibility as possible to the remote robot, e.g., to navigate an area, negotiate obstacles, identify possible threats or survivors. In other words, we would like to endow the remote robot with a *high degree of autonomy*. The would ease the operator's cognitive burden and allow simultaneous control of multiple robots.

Fig. 1. The spherical security robot GroundBot; a pan/tilt zoom camera is housed in each of the transparent bulbs on either side of the shell.

On the other hand, there are good reasons why most current surveillance and inspection robots present a *low degree of autonomy*, in which the operator directly teleoperates the robot. For instance, the operator's attention might be required for careful maneuvers around obstacles or in difficult terrain.

As described earlier [4], we plan to tackle the above trade-off by implementing an *adjustable autonomy* control architecture. *Adjustable or variable autonomy* (AA) refers to the ability to have the level of autonomy changed during operation, not only by a human user but by another system, or the autonomous system itself. While in most instances, autonomy levels are discrete, as exemplified by Goodrich et al. [5], some work on continuously adjustable autonomy is available [6]. The general problem of AA, as formally defined by Scerri et al. [7], is to choose the degree of autonomy that maximizes the overall utility of the team (where the team might consist of a constellation of multiple robots and operators, or just a single robot with an operator). In our control architecture, the robot can exhibit different types and degrees of autonomy, and dynamically switch between them depending on the context.

In this paper, we present a low-level block of this architecture: a high-resolution 360° virtual panorama, efficiently constructed using two pan/tilt cameras. Without higher levels

Proc of the EEE International Workshop on Safety, Security, and Rescue Robotics (SSRR-07)
Rome, Italy, September 2007

of autonomy available, this panorama is a way to convey the maximum amount of information about the robot's surroundings to an operator. As higher levels of autonomy are implemented, they may use the information from lower level blocks to fulfill their tasks.

Next, we review some previous work related to the concept of situation awareness and panoramic video in human-robot interaction, In the succeeding sections, we present the concept of our adjustable autonomy control architecture, the hardware platform used in this work, and finally, the design of our user interface and the virtual panorama technique in particular.

II. Related Work

A properly designed human-robot interface is vital to achieve situation awareness, which is key to efficient utilization of the system at hand. For instance, experiments set in a search and rescue scenario have shown that a majority of the operator time was spent trying to perceive and comprehend the situation, and only a minority of the time was spent on planning, projecting, and problem-solving activities [8]. In response to these kinds of issues, Baker et al. [9] have formulated the following set of general design guidelines for improved human-robot interaction.

- *Enhance awareness*; provide more spatial information about the robot in the environment.
- *Lower cognitive load*; fuse sensor information to allow an operator to focus on a single area of the interface, etc.
- *Increase efficiency*; support interaction with multiple robots in a single window; minimize the use of multiple windows.
- *Provide help in choosing robot modality.* Assist the user in determining the appropriate level of autonomy.

One way to enhance situation awareness is to provide an overhead view that includes part of the robot, as shown by Keyes et al. [10]. The overhead view comes from a camera mounted on a vertically extended arm, but there are also several examples [11]–[14] of interfaces that use virtual reality (VR) or augmented reality displays to achieve the goals of the guidelines. These interfaces have a natural way to convey spatial information in a single view, and they can be used in several modes of interaction. A robot-centric view-point provides a traditional direct teleoperation interface, while a bird's eye view can provide a suitable interface for high-level tasks, such as mission planning. With a free view-point, VR interfaces also provide natural ways to interact at different degrees of robot autonomy.

Concerning panoramic images in the context of mobile robotics, their use has been considered at different levels of autonomy. For example, Gaussier et al. take 24 image strips of 30×288 pixels, separated by 7.5 degrees using a single servo-controlled camera, to construct (1066×288 pixels) 270° panoramas, The panoramas are used for autonomous navigation in open environments without maps [15], and their system reportedly works even though it uses an image merging technique that introduces some discontinuities in the panoramic images.

Nielsen et al. on the other hand use snapshots to implement external memory (to the operator) in their semantics maps [16]. In one example, they show a panoramic view of the robot's surroundings; however, the single camera is fixed to the robot and the user has to actively turn the robot and take snapshots to get a panoramic image. Each snapshot is displayed the corresponding location and orientation of the robot when the snapshot was taken.

This is similar in intent to previous efforts by Maxwell et al., where an autonomous urban search and rescue (USAR) robot generates 360° panoramas, annotated with detected motion and skin color, and connects them to their corresponding positions in a constructed map. The panoramas are constructed as 180° field of view images by concatenating eight frames from the robots pan/tilt zoom camera [17]. In subsequent work [18], Maxwell et al. have their USAR robot generate stereo panoramas from its single camera; however, it is unclear whether this feature is actually used to enhance situation awareness.

Kadous et al. mention "virtual panning" as possible future work on their user interface for the teleoperated USAR robot, CASTER [19]. Their design is a traditional direct interface in which they propose to use an omnidirectional vision system to increase the responsiveness of panning commands from the operator by virtually panning the displayed video. This is a way to alleviate the effect of inevitable video lag by minimizing the time between command generation and when the resulting action can be seen. Of course, the resolution of this panorama is limited to that of a single frame from the camera used in the omnidirectional vision system.

The examples above require that the images be projected on a single pre-determined manifold, as opposed to the technique, described by Peleg et al. [20], where strips of images are projected on manifolds determined dynamically based on the motion of the camera. This enables the construction of panoramas under unrestricted camera motion; however, it is unclear whether this has ever been performed in realtime as would have to be the case in a mobile robot user interface.

III. Adjustable Autonomy Control Architecture

We have previously introduced our long-term vision of a robotic remote surveillance system using the GroundBot, possibly involving multiple robots and/or one or more operators [4]. A robot in this system might, for example, navigate in an autonomous mode when it detects a possible intruder, and hence call on the attention of a human operator. In the reverse direction, a robot being tele-operated by a human might detect a loss of contact (possibly caused by entering radio shadow or something diverting the operator's attention) and hence jump to a higher degree of autonomy until contact with the operator can be re-established. Consequently, we need to enable both tasks that require a high degree of autonomy, as well as tasks that need to be performed under careful teleoperated control of an operator.

We use the notion of a *triple tower of autonomy* [4] dealing with degrees of autonomy in three different respects: control,

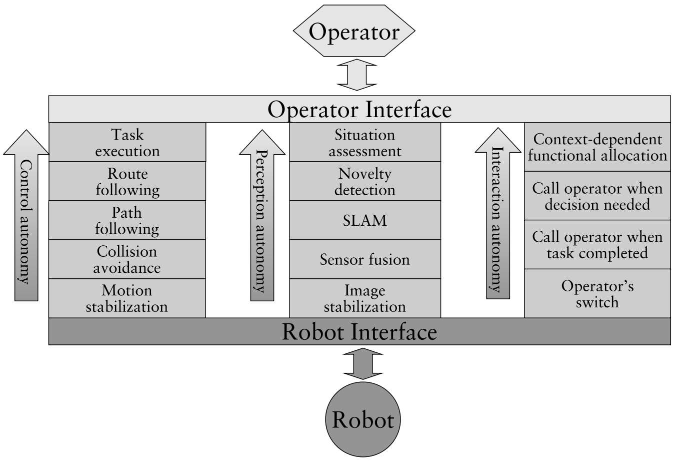


Fig. 2. Illustration of the three towers of autonomy.

perception, and interacton. An example of this concept is outlined in Figure 2.

Control autonomy is the type of autonomy most often considered in robotics. This typically entails routines for (semi-)automatic navigation, giving goals that range from velocity set-points, position way-points, or specifications of full navigation tasks, much like the four different autonomy modes distinguished by Goodrich et al. [5]: intelligent teleoperation, way-points and heuristics, goal-biased autonomy, and full autonomy.

Perceptual autonomy deals with the processing and presentation of sensor data. In its simplest form, sensor data is presented in raw form, e.g. images from a remote camera. To alleviate some of the interpretation effort, the operator interface can incorporate more advanced processing; from registering and visualizing a set of camera images taken over a time interval, through simultaneous localization and mapping, to novelty detection, for example.

The final type of autonomy considered is *interaction autonomy*, that is, the ability to dynamically adapt the other two types of autonomy (and hence the type of operator-robot interaction) to the current situation. From its simplest form, where this is done manually by an operator who makes an assessment of the situation and selects the desired level of autonomy, this type ranges over returning control to the operator when a task is completed, switching to a higher or lower degree when an abnormal situation occurs, and full situation assessment and automatic distribution of tasks between robot and operator.

This paper describes a small step towards the overall goal of our control architecture. In Section V, we describe the implementation of a virtual panorama imaging technique, a perception aid to increase situation awareness, and a block in the lower levels of the perception autonomy tower.

IV. Hardware

This project currently uses the spherical security robot GroundBot from Rotundus AB [21], which is a ball-shaped robot with a diameter of 60 cm. This robot is capable of navigating rough outdoor terrain at speeds approaching 3 m/s (just under 7 mph). As a spherical robot, both locomotion and steering of the GroundBot is accomplished by displacement of its center of mass. Almost all of the robot's weight is suspended on a rigid axle mounted through the shell. The distribution of this weight is managed by two perpendicular motors able to rotate the weight about the robot's center. The robot is equipped with a PC/104+ format computer with an 800 MHz Crusoe CPU, a long-range 802.11a/g wireless network card, two motor controllers, loudspeaker, and a Lithium ion battery pack able to provide power for up to 12 hours of runtime. The current sensor outfit includes a (differential) Global Positioning System unit, microphones, a Microstrain 3DM-GX1 gyro enhanced orientation sensor, and two pan/tilt zoom network video cameras.

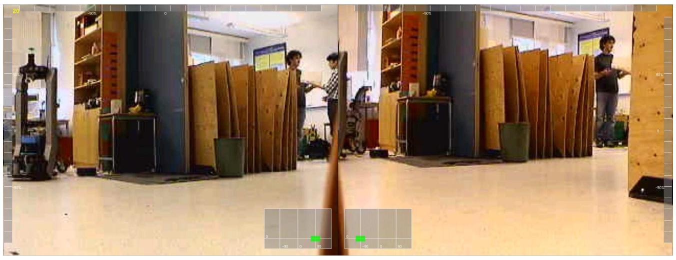


Fig. 3. Screenshot of the user interface in 2D mode, as displayed on two screens.

What can be considered the intrinsic autonomy provided by a spherical morphology allows the robot to naturally negotiate rough terrain and sufficiently small obstacles, simply by rolling over them. However, there are also some technical difficulties with spherical robots; they present a number of challenges in terms of control and perception, for example camera image stabilization. Any change in motion induces unwanted oscillations that are hard to rectify; Suomela and Ylikorpi [1] mention the robot Rollo that controls oscillation only around the rolling axis, i.e. pitch control. They also mention Rotundus AB [21], which has a control algorithm, developed at the Swedish Defence Research Agency (FOI), that controls both the pitch and roll of the GroundBot. At the time of writing, this algorithm is implemented and tested in simulation, but has yet to be tested on the robot. Spherical robots offer limited freedom in placement of sensors; for example, the most natural place for video cameras is at the points where the main axle meets the shell. Together with any oscillation, this placement easily disorientates operators, as established by Johansson and Seeman [22].

V. GUI & Virtual Panorama

The user interface to our GroundBot is a hybrid between *direct* and *multi-sensor/multi-modal interface* [23]. It can be used in a 2D mode where the displayed is comprised of video images from the robot's cameras overlaid with non-intrusive vertical bar speed indicators and widgets indicating the pose of the cameras, as seen in Figure 3. Alternatively, it can be used as a 3D virtual reality interface, which gives us new ways to address challenges introduced by the GroundBot's spherical shape, such as representing sensor readings and exercising control. An immediate advantage of using a 3D

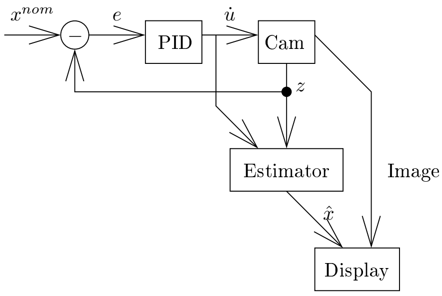
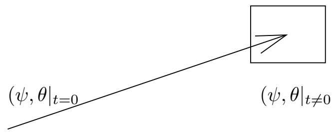
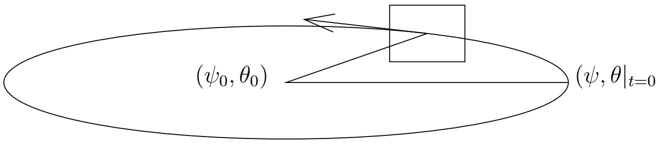


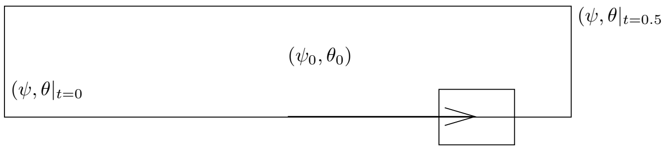
Fig. 4. Camera subsystem of the GroundBot and its user interface in parametric mode. x^{nom} is the nominal trajectory generated by the active pan/tilt pattern.



(a) Step pattern.



(b) Ellipse pattern.



(c) Rectangle pattern.

Fig. 5. The selectable parametric pan/tilt patterns. $t \in [0, 1]$ describes a full period of each pattern. For the step pattern, only a long, first period is used for system tuning.

interface with a controllable point of view is the ability to seemlessly move between different control paradigms. On one end, we have a traditional first-person teleoperation view of the environment, suited for manually exploring an uncharted area. On the other end, we have a supervisory, birds-eye view of the world, desirable for tasks involving surveillance of known areas and controlling several robots exhibiting some higher form of autonomy.

Figure 4 gives a schematic view of the camera subsystem of the GroundBot and its user interface. Using the highest resolution, the cameras produce images at a rate of 15 Hz; whereas the update rate of pan/tilt information is limited to 2.5 Hz. Therefore, we use an estimator for the pose of each camera at the time of each video frame. The cameras can either be controlled directly by joystick input from the operator, or be set to follow a periodic parametric pan/tilt pattern. When in parametric mode, a PID controller is responsible for generating commands to the camera in order for it to follow the selected pan/tilt trajectory, $x^{nom} = (\psi, \theta)^{nom}$, which is given by one of the patterns *ellipse, rectangle,* and *step* showed in Figure 5.

The step pattern is mainly useful for tuning the PID controller parameters. The elliptic pattern was implemented with two things in mind; firstly, given that the tilt range is less than the pan range, the image density will be highest when the cameras are facing backwards and forwards respectively, i.e. the directions in which the robot can be driven and where the most information is supposed to be required; secondly, the sinusoidal nature of the ellipse pattern will minimize abrupt acceleration of the camera rotation, thereby minimizing the chance to cause both unwanted oscillations of the robot and camera vibrations. However, since the rotating parts of the cameras constitute a small fraction of their total weight of 1.3 kg, even sudden accelerations of the pan/tilt units have no detectable effect on the robot's pose. The rectangle pattern, in which cameras rotational velocity along each side of the rectangle and, thereby, the angle separating subsequent images is constant, is favourable since the piecewise linear motion of the cameras can be fairly accurately estimated, even with a naive approach.

The user selects the time interval between each image in the panorama, and the maximum number of images from each camera used for the panorama at any time. Figure 6 shows two panoramas generated by saving 32 images from each camera at 0.5 second intervals. The data in these cases come from two runs using the rectangle and ellipse patterns respectively; in both cases the height of the pattern is zero. The estimator used here is a linear interpolator between subsequent pan/tilt measurements which, as can be seen in Figure 6, works well for constant angular rate motion patterns such as the rectangle. However, it does slightly worse on the ellipse pattern, where the pan/tilt angular rate is actually sinusoidal.

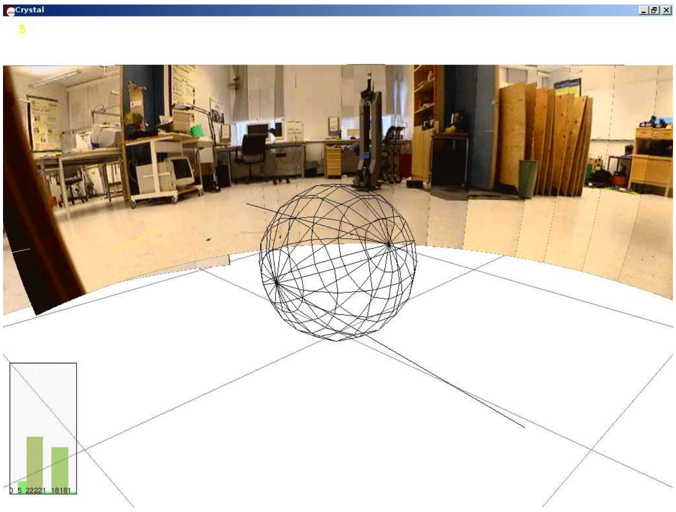
The images are projected onto spherical manifolds centered at each cameras position, simply by texture mapping the inside of a sphere segment with each video frame.

VI. Conclusions

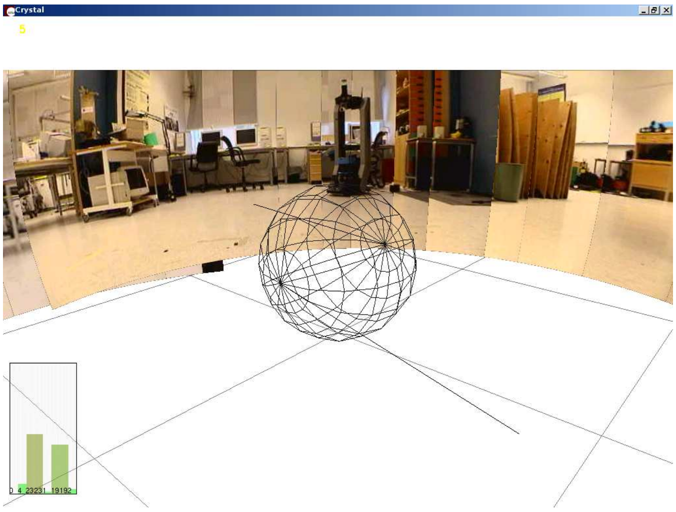
The work presented here constitutes a small part of a larger goal, namely the implementation of a 360° virtual panorama as a block in the perception tower of our adjustable autonomy control architecture.

While we consider the panoramic imaging successfully implemented, there is without doubt room for improvement; for example, the system still needs to undergo thorough evaluation and validation. The naive camera pose estimation in use could be substituted by a Kalman filter to take camera control input into consideration, in order to get more accurate pose estimates, also for other types of camera pan/tilt motion. Autocorrelation between subsequent frames in the video streams could also be used for more accurate image registration. By projecting the images on spherical manifolds, the robot is restricted to standing still while constructing panoramas. By incorporating the use of dynamic manifolds, as mentioned in Section II, panorama imaging could be in effect at all times during robot operation. This requires, of course, that the underlying problem of estimating the cameras' paths as the robot moves is solved.

As a step towards developing adjustable autonomy features, this technique brings us closer to freeing guards from



(a) Rectangle pattern.



(b) Ellipse pattern.

Fig. 6. Screenshots of the user interface in 3D mode. The robot is represented by the wireframe sphere at the center of the images. The view-point has been rotated to the left of the robot and slightly pulled back to give a wider view of the panorama.

monotonous patrolling, increasing security and lowering costs. In addition to security, the GroundBot is also expected to have applications within other areas such as search and rescue, disaster mitigation and general surveillance.

Acknowledgement

This work is supported by Robotdalen (Robot Valley) and Vetenskapsrådet (the Swedish Research Council).

References

[1] T. Ylikorpi and J. Suomela, "Ball-shaped Robots: A historical Overview and recent Developments at TKK," in *International Conference on Field and Service Robotics*, July 2005.

[2] S. H. Kenyon, D. Creary, D. Thi, and J. Maynard, "A small, cheap, and portable reconnaissance robot," in *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IV*, E. M. Carapezza, Ed., vol. 5778, no. 1. Society of Photo-Optical Instrumentation Engineers, 2005, pp. 434–443. [Online]. Available: http://link.aip.org/link/?PSI/5778/434/1

[3] B. Chemel, E. Mutschler, and H. Schempf, "Cyclops: miniature robotic reconnaissance system," in *IEEE Int. Conf. on Robotics and Automation (ICRA '99)*, vol. 3, May 1999, pp. 2298–2302.

[4] M. Seeman, M. Broxvall, A. Saffiotti, and P. Wide, "An Autonomous Spherical Robot for Security Tasks," in *IEEE International Conference on Computational Intelligence for Homeland Security and Personal Safety*, 2006.

[5] M. A. Goodrich, D. R. Olsen, J. W. Crandall, and T. J. Palmer, "Experiments in Adjustable Autonomy," in *Proc. of IJCAI Workshop on Autonomy, Delegation and Control*, 2001.

[6] M. Desai and H. A. Yanco, "Blending Human and Robot Inputs for Sliding Scale Autonomy," in *Proc. of the 14th IEEE International Workshop on Robot and Human Interactive Communication*, 2005.

[7] P. Scerri, D. V. Pynadath, and M. Tambe, "Towards Adjustable Autonomy for the Real World," *Journal of Artificial Intelligence Research*, vol. 17, pp. 171–228, 2003.

[8] J. L. Burke and R. R. Murphy, "Situation Awareness and Task Performance in Robot-Assisted Technical Search: Bujold Goes to Bridgeport." [Online]. Available: citeseer.ist.psu.edu/burke04situation.html

[9] M. Baker, R. Casey, B. Keyes, and H. A. Yanco, "Improved interfaces for human-robot interaction in urban search and rescue." in *IEEE International Conference on Systems, Man and Cybernetics, Volume 3*, 2004, pp. 2960–2965.

[10] B. Keyes, R. Casey, H. A. Yanco, B. A. Maxwell, and Y. Georgiev, "Camera placement and multi-camera fusion for remote robot operation," in *IEEE Int'l Workshop on Safety, Security and Rescue Robotics*, August 2006.

[11] D. J. Bruemmer, D. A. Few, M. C. Walton, R. L. Boring, J. L. Marble, C. W. Nielsen, and J. Garner, " 'Turn Off the Television!': Real-World Robotic Exploration Experiments with a Virtual 3-D Display," in *HICSS '05*. Washington, DC, USA: IEEE Computer Society, 2005, p. 296.1.

[12] B. Ricks, C. W. Nielsen, and M. A. Goodrich, "Ecological Displays for Robot Interaction: A New Perspective," in *Proceedings of IROS 2004*, 2004.

[13] M. Quigley, M. A. Goodrich, and R. W. Beard, "Semi-autonomous human-UAV interfaces for fixed-wing mini-UAVs," in *Proceedings of IROS 2004*, September–October 2004.

[14] M. Sugimoto, G. Kagotani, H. Nii, N. Shiroma, M. Inami, and F. Matsuno, "Time Follower's Vision: A Teleoperation Interface with Past Images," *IEEE Comput. Graph. Appl.*, vol. 25, no. 1, pp. 54–63, 2005.

[15] P. Gaussier, C. Joulain, S. Zrehen, J. Banquet, and A. Revel, "Visual navigation in an open environment without map," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'97)*, 1997, pp. 545–550. [Online]. Available: citeseer.ist.psu.edu/gaussier97visual.html

[16] C. W. Nielsen, B. Ricks, M. A. Goodrich, D. J. Bruemmer, D. A. Few, and M. C. Walton, "Snapshots for semantic maps," in *SMC (3)*, 2004, pp. 2853–2858.

[17] B. A. Maxwell, L. Meeden, N. S. Addo, P. Dickson, N. Fairfield, N. Johnson, E. G. Jones, S. Kim, P. Malla, M. Murphy, B. Rutter, and E. Silk, "Reaper: A reflexive architecture for perceptive agents," *AI Magazine*, vol. 22, no. 1, pp. 53–66, 2001.

[18] B. A. Maxwell, N. Fairfield, N. Johnson, P. Malla, P. Dickson, S. Kim, S. Wojtkowski, and T. Stepleton, "A real-time vision module for interactive perceptual agents," *Machine Vision and Applications*, vol. 14, no. 1, pp. 72–82, 2003.

[19] M. W. Kadous, R. K.-M. Sheh, and C. Sammut, "Effective user interface design for rescue robotics," in *HRI '06: Proceeding of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*. New York, NY, USA: ACM Press, 2006, pp. 250–257.

[20] S. Peleg, B. Rousso, A. Rav-Acha, and A. Zomet, "Mosaicing with strips on adaptive manifolds," pp. 309–325, 2001.

[21] "Rotundus," http://www.rotundus.se/, May 2007, durable mobile robots for outdoor surveillance.

[22] P. Johansson and M. Seeman, "Graphical user interface for control of a spherical robot," Master's thesis, Uppsala University, 2006.

[23] T. Fong and C. Thorpe, "Vehicle Teleoperation Interfaces," *Auton. Robots*, vol. 11, no. 1, pp. 9–18, 2001.