Is Augmented Reality the Future Middleware for Improving Human Robot Interactions: A Case Study

Eranda Lakshantha
Monash University
eranda.lakshantha@monash.edu

Simon Egerton
Monash University
simon.egerton@monash.edu

ABSTRACT

With robots appearing frequently within our society, the cases will be higher where people with little or no practical experience in robotics would have to supervise robots. Future interfaces should make Human Robot Interactions (HRI) intuitive for such less-experienced users. A key requirement to have an intuitive interface is to improve the level of HRI performance. In this study we try to improve the HRI performance by developing a system namely, SHRIMP (Spatial Human Robot Interaction Marker Platform) based on Augmented Reality (AR) technology. We present SHRIMP as a new type of middleware for HRI, that can mediate high-level user intentions with robot-related action tasks. SHRIMP enables users to embed user intentions in the form of AR diagrams inside the robot's environment (as seen by the robot's camera view). These AR diagrams translate into action tasks for robots to follow in that environment. Furthermore, we report on the HRI performance induced by our SHRIMP framework when compared to an alternative more common robot control interface, a joystick controller.

Keywords

Augmented reality, Human robot interaction, Vision based navigation, Robotics

1 INTRODUCTION

In future, robots will play a major role in our personal spaces, while making our everyday activities more efficient. However it is not guaranteed that people who control robots in personal spaces such as homes, offices, schools, and hospitals will always have a specialized experience in controlling robots. In such cases, a successful HRI depends on a powerful interface that can bridge those experience gaps and make the collaboration between man and machine intuitive and seamless. A key ingredient in making an intuitive interface is to have a higher level HRI performance. There are several options available for this purpose including visual, tactile, verbal, or multimodal communication mechanisms. In this paper, we explore the levels of HRI improvement realizable through one of the most powerful and most advanced visual communication mechanisms, *Augmented Reality* (AR).

AR refers to the representation of virtual graphics objects on top of a real-world scene [Payton et al., 2001]. In the context of our study we view AR as a new form of middleware to carry out HRI activities, a middleware that could work as a generalized framework across multiple robot platforms and hardware devices rather than a stand-alone application specific interface. In its core, AR uses diagrams for referencing physical space, instrumenting that space with markers, instructions and messages. A middleware assisted by AR could express these diagrams as robot related action tasks, which user intends to carry out in a physical space. SHRIMP is a proof of concept framework that we developed to realize the existence of such a middleware.

SHRIMP enables users to interact with any robot through the placement of AR diagrams within the environment. These AR diagrams (or AR objects) can be tagged with instructions, such as 'follow', 'wait', 'hold' and higher level functions or behaviors such as 'vacuum the floor', 'clean the table' are also possible. AR object placement is achieved by directly placing them in the robot's environment through a real-time video centric interface. We believe that this gives room for the less-experienced human operator to improve his spatial awareness and manipulate robots intuitively. For example consider a home environment. With SHRIMP in place, a house wife could visually program or leave messages for her cleaner robot by placing and tagging these AR objects in the environment. Another scenario would be a hospital environment, where SHRIMP platform could be used by a nurse to place routing information for care robots to follow for their rounds. In later sections we further discuss the way our middleware operates and its utilization of HRI performance.

The remainder of this paper is structured in the following manner. Section 2 highlights the current state-of-the-art diagrammatic mechanisms used in robotics. The details of our solution are described in Section 3 whereas Section 4, 5, and 6 report on our applications, experiments and their results respectively.

2 RECENT SOLUTIONS

Although we describe our proposed middleware as a diagrammatic mechansim rendered through AR, there are other methods that use diagrams as the primary mode of communication for HRI. In general, we can classify them into four distinct categories namely, digital codes, fiducial markers, object markers & marker-less methods. Mentioned below are some of the recent applications from each category.

Digital codes have two different representations, bar-codes & QR (Quick Response) codes. The work described in [Han et al., 2012] presents a method for interacting with robots through bar-codes. According to them, robots can track object poses by having a bar-code on the observed object. They further suggest the suitability of such a mechanism for assistive robots in super-markets to grab, hold and lift commodities tagged with bar-codes. On the other hand, a recent study by [Martinez et al., 2013] describes a simulation and a test bed application for mobile robots. Their test bed application implements a strategy for tracking the position and the heading angle of mobile vehicular robots via a special bar-code ID.

The second type of digital codes are the QR codes. They are considered as a faster identification method for object recognition, especially when robots are used in household environments [Li et al., 2012]. Further demonstrations by [Li et al., 2012] suggest that robots can move or grasp objects, with those objects marked with QR codes. The work of [Garcia-Arroyo et al., 2012] further supports this claim with their shopping assistance robot system. The shop assistant robot cooperates with the human user to maintain his shopping list, select items and alert for any missing items from the list through QR codes.

In addition to digital codes, fiducial markers act as an alternative diagrammatic mechanism, especially in robot path-planning activities. An example application can be seen in [Fang et al., 2012], where they used fiducial markers to plan optimal trajectories of a stationary robotic arm. Here the communication between the user and the robotic arm is performed with a special handheld device which is attached with a fiducial marker cube. A study presented by [Hu et al., 2013] delivers a similar idea of motion planning where they place a fiducial marker to create a 'virtual robot' which acts on behalf of a real robot. They further suggest that human operators can control the virtual robot to indirectly manipulate the remote real robot. A similar notion can be seen in [Lee and Lucas, 2012], where they use fiducial marker patterns for planning obstacle and collision free paths for a group of heterogeneous robots.

Object markers work by tracking the motion of natural objects within the environment. Thereby it creates a less artificial form of interaction when compared to digital codes and fiducial markers, especially where object-tracking robots and human-following robots are used. Recent works described in [Jean and Lian, 2012] and [Karkoub et al., 2012] indicate the usefulness of object markers for such applications.

The above body of work justifies the claim that bar-codes, QR codes, fiducial markers, and objects markers are well established diagrammatic mechanisms in HRI. Their application is substantially verified in a variety of HRI tasks and thus research with these mechanisms has grown into a well matured state. Even though this is the case, they pose the problem of specifically instrumenting the environment with those objects, before having any interaction with a robot.

On the other hand, marker-less methods does not require the environment to be instrumented with special markers and can be used in a readily available environment. Studies presented in [Chen et al., 2008, Leutert et al., 2013, Abbas et al., 2012] describe the use of marker-less methods such as *marker-less AR* for interacting with robots. However, their outcomes appear questionable when robots are operated in more practical and everyday environments, as in homes and office spaces. Furthermore, it is an open question to see how well marker-less AR can fill in the experience gaps for non-robotic users. Following section highlights our solution as we explore marker-less AR's potential as a diagrammatic mechanism in improving HRI performance.

3 PROPOSED SOLUTION

In making a solution, our first step was to focus on structuring our framework with a stable marker-less AR approach. The current state-of-the-art in marker-less AR technology is arguably the Parallel Tracking and Multiple Mapping (PTAMM) [Castle et al., 2008] platform. PTAMM identifies unique scale invariant features to attach AR objects (i.e. virtual graphics elements) and locally tracks them through a scene.

However, PTAMM by default generates multiple local maps and so it could not track a single AR object in a persistent manner. For example, imagine you create an AR object and move the camera towards it. Once you move past the AR object, turn the camera back in an angle of 180 degrees. At this point the camera will be looking at the path it has travelled, and we would expect the AR object to be seen persistently anchored at its original position. We ran several trials with the default PTAMM implementation and found it difficult to achieve this behavior. Since we intend to apply PTAMM to guide robots, tracking robustness under such wide camera angles is considered one of our major design decisions.

We have addressed this shortcoming by introducing a *linear transformation* algorithm into PTAMM. The algorithm combines all local maps generated by PTAMM into a single global map with linear equations, carried out at frame-rate. At the beginning of the algorithm, it takes the initial camera position as the global map origin. All the subsequent local maps are expressed with regard to the global map origin (via linear equations), hence giving a global camera pose throughout the course of the camera's motion. Finally the rotation & the translation matrices embedded in the global camera pose are fed into the graphics rendering pipeline. More details of our algorithm can be found in [Lakshantha and Egerton, 2014].

Our linear transformation algorithm enables the marker-less AR object to be globally tracked in a persistent manner, in the same way a physical object would be. This helps our proposed middleware framework (i.e. SHRIMP) to maintain and track AR objects when they fall out of camera view, or more importantly, approached from a different location.

The series of pictures in Figure 1 demonstrates the persistence of our marker-less AR framework. The first three images show local vantage points and placement of the AR object in the environment. Then the camera is turned away from the AR scene, returns from a different vantage point and the AR object is observed, demonstrating persistence in the same way the solid physical objects are within the scene.

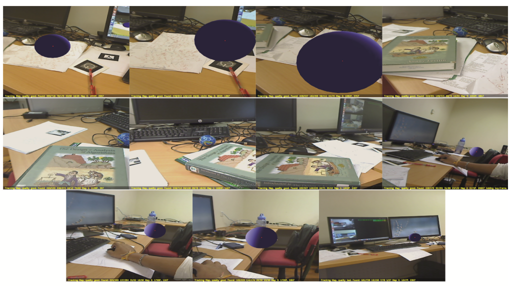


Figure 1: Tracking persistence of SHRIMP under wide camera angles

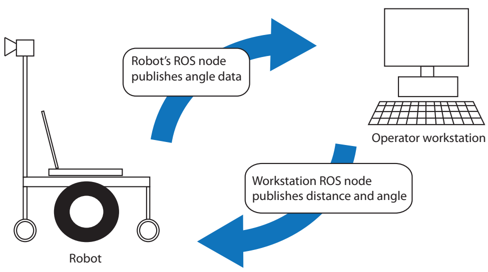


Figure 2: SHRIMP communication architecture

We have implemented our framework under Ubuntu 12.04 and the Robot Operating System (ROS) framework. The framework currently runs as a client server model, where the robots act as clients to the SHRIMP server hosted on an i7 desktop PC, as illustrated in Figure 2. This enables us to 'plugin' and experiment with different robot platforms with minimal changes to the code base.

As shown in Figure 2 the human operator performs the placement of AR objects through the desktop PC which in turn hosts the SHRIMP service. A standard USB web-cam which outlooks the target environment, provides the user with the view of the robot's view frustrum. The placement of the AR objects is carried out on top of this live video feed, in a real-time manner. After positioning an AR object within the robot's vicinity, SHRIMP broadcasts the amount of distance and the rotation required, in order to reach the target location - in this case the AR object's location. The robot keeps on listening to this broadcast and captures this data, followed by executing the required motion. The conversation between the robot and the SHRIMP server continues until it reaches the target location.

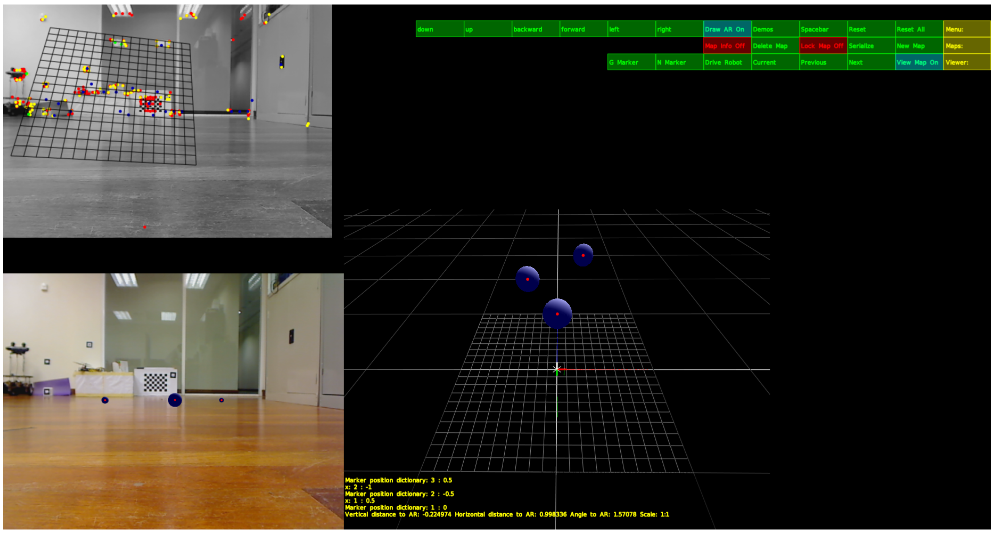
4 APPLYING OUR SOLUTION TO HRI

We tested SHRIMP with two different robot development platforms, firstly on a Parallax® Eddie robot platform and then on a LEGO® Mindstorm NXT. In the next two sections we demonstrate two HRI scenarios where we discuss our framework's functionality with these robots.

4.1 Navigation Scenario

For the first case, we chose HRI *navigation* task as it is the most fundamental and widely-used functionality in mobile robotics. SHRIMP permits the human operator to lay down a series of virtual AR objects on the scene, whereby each object acts as a navigation point for the robot. These AR objects can be organized into a set of way-points which ultimately constitute a virtual navigation path on top of the robot's field-of-view. The scenario highlighted here is illustrated in Figure 3.

In the demonstration shown in Figure 3a, the navigation path is trailed by three AR objects. The right-hand side (area under black background) provides a 3D map view of the environment to enhance depth perception. Once the AR objects are laid, the robot is ready to move along the path. The navigation is performed in the order AR objects are placed in the scene. Accordingly, robot starts moving on to the first AR object in Figure



(a) Laying a series of way-points with marker-less AR

(b) First point (c) Second point (d) Third point

Figure 3: Creating a virtual navigation path by marking multiple locations in space with multiple AR objects.

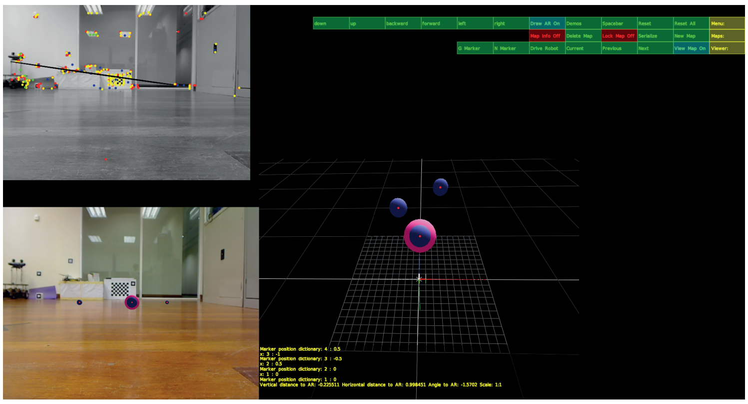


Figure 4: Multiple HRI tasks are represented with multi-colored AR objects

3b, then moves on to the second AR object in Figure 3c. The robot terminates its navigation after moving on to the last AR object in Figure 3d. We tested this scenario with a LEGO® Mindstorm NXT robot. The accompanying video file[1] further explains this demonstration.

4.2 Hold & Grip Scenario

In our second scenario we investigate the SHRIMP's operation against a *hold & grip* task. Here, we navigate the robot towards a location in space and then perform a hold & grip action. Multiple AR objects with different colors are employed for this purpose, in a manner where each color maps into a unique HRI task. Let's consider the illustration in Figure 4. According to the illustration a blue-colored AR object represents a navigation task whereas the pink AR object signifies a hold & grip action. In this case the blue AR object (the foremost one) and the pink AR object shares the same point in space, which carries the idea that the robot should perform a hold & grip task at the location of the first

[1] `http://youtu.be/lIlpv3QPH5g`

Subject ID	School / Department	Gender	Age group (years)	Level of experience using computers (years)	Level of experience using computer games (years)
1	IT	Male	26-30	10+	10+
2	IT	Female	26-30	8-10	8-10
3	IT	Female	26-30	8-10	8-10
4	IT	Male	21-25	8-10	8-10
5	IT	Male	35+	10+	less than 2
6	Engineering	Male	21-25	5-7	5-7
7	IT	Female	15-20	less than 2	less than 2
8	Engineering	Female	21-25	10+	2-4
9	Arts	Female	21-25	8-10	8-10
10	Engineering	Female	21-25	10+	2-4
11	IT	Male	21-25	10+	8-10
12	Other	Female	26-30	8-10	2-4
13	IT	Male	26-30	10+	2-4
14	IT	Male	31-35	5-7	10+
15	Other	Male	35+	10+	8-10
16	IT	Male	26-30	10+	10+
17	Other	Female	26-30	8-10	5-7
18	IT	Male	35+	10+	less than 2
19	IT	Male	35+	10+	less than 2
20	IT	Female	26-30	8-10	less than 2

Table 1: Participant profiles

AR object. The online video footage[2] provides a more comprehensive demonstration of this scenario.

5 A CASE STUDY: NAVIGATION

To investigate the SHRIMP's HRI performance for less-experienced users (i.e. average users) we carried out a case study. In this case study our robot client is a Parallex® wheeled robot. The Parallex® is a Microsoft® robot reference design with its driver layer adapted to the ROS environment. The case study addressed the following hypothesis,

> *"Does the SHRIMP framework improve the HRI performance for the average person ?"*

To answer this hypothesis we set up a comparative navigation task experiment. A total of twenty participants were employed for this task and we asked each participant to remotely operate the robot and navigate it over a predefined path. Details of participant profiles such as age, gender, and experience levels are summarized in Table 1.

The participants only had access to the robot camera view and were also asked to observe the environmental scene through the robot's camera view while performing the task. The participants completed the task twice, once, remotely operating the robot using a PS3 joystick controller and again using our SHRIMP AR framework. In order to minimize bias, we shuffled the order of execution between the two conditions PS3 and SHRIMP for each participant. For instance, if one participant had PS3 as the first trial & SHRIMP as the second, then the next participant had SHRIMP as the first trial & and PS3 as the second. Here we used a PS3 joystick controller as a benchmark since joystick controllers are one of the most widely used HRI methods.

To test our hypothesis we measured task completion times and measured operator performance-levels (i.e. task load level) via a post observational questionnaire where we asked each participant to recall a set of special elements within the environment. These special el-

Figure 5: Experimental set up for our case study.

[2] `http://youtu.be/ye8wKdJX7IY`

ements were symbolized by a set of fiducial markers which in turn were positioned randomly across the environment. Users managed the robot only through the camera view whereby a solid screen in the middle further prevented users from directly viewing the environment. Our experimental set up is illustrated in Figure 5 above.

If our hypothesis is true then we would expect average performance levels for our SHRIMP model to be higher than PS3. We quantified the operator performance level (W) by taking weighted ratio between the number of correct observations (C) and the task completion times (T), calculating the following formula.

$$W = \frac{C}{T} \tag{1}$$

The idea captured here is the notion that subjects completing the task with a higher number of correct post observational questions with a lower navigation task time are considered to have higher performance levels. In this case the performance factor is measured by two well-known HRI metrics, *situation awareness* & *task completion time* [Steinfeld et al., 2006].

Situation awareness is evaluated by the human operator's capacity to pay attention to the environmental scene while controlling the robot. According to [Parasuraman et al., 2000], the amount of situation awareness positively correlates with the users' interaction performance. We assessed the amount of situation awareness by using the number of successful recalls, which are captured through the post observational questionnaire. The questionnaire included five multiple choice questions, each providing a series of markers. Out of those markers, the user had to recall and select the correct marker that was present in the environment. Consequently, a higher amount of successful recalls lead to higher values for W, which in turn indicate higher levels of interaction performance. The raw data including correct observations, task completion times and W values are summarized in Table 2[3].

6 RESULTS

Based on the data of Table 2 we plot the histogram between the number of participants and the W values (i.e. performance indicator) in order to investigate performance patterns posed by the both types of interfaces. Figure 6 illustrates the resulting histogram for our SHRIMP framework. The peak of its normal distribution indicates an average value (W^S) of 0.80721632. Similarly Figure 7 depicts the histogram with its normal distribution for PS3 gamepad whereas its peak point marks an average value (W^P) of 0.500894682.

[3] In Table 2 data points in W column with 0 values highlight the cases where users could not sucessfully recall elements in the environment. These are extreme cases and further experimentation will be done with those cases removed.

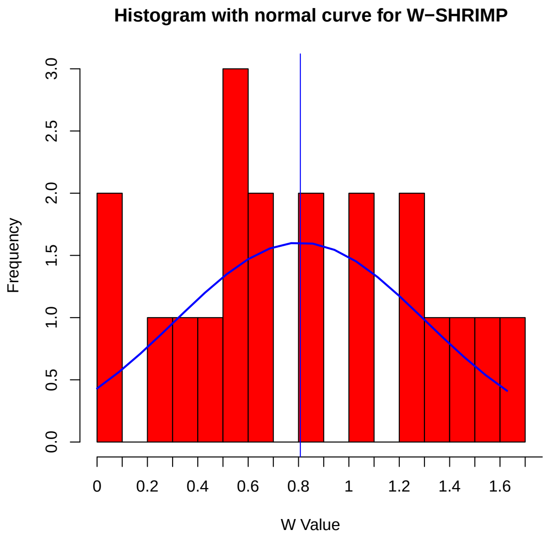


Figure 6: Histogram of SHRIMP

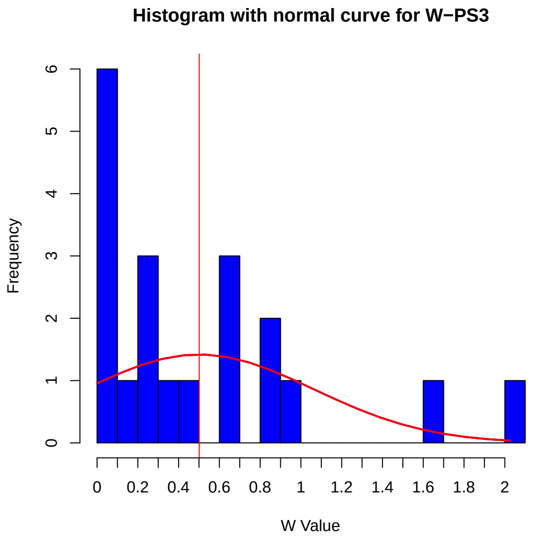


Figure 7: Histogram of PS3 gamepad

According to the graphs in Figure 6 and Figure 7 it is evident that,

$$W^S > W^P \tag{2}$$

In order to better highlight this difference we rescale both W^S and W^P in the following manner while bringing them into a common range.

$$W^S_{rescaled} = \frac{W^S}{W^S + W^P} \tag{3}$$

$$W^P_{rescaled} = \frac{W^P}{W^S + W^P} \tag{4}$$

Subject ID	Correct observations with PS3 (%)	Correct observations with SHRIMP (%)	Time-PS3 (s)	Time-SHRIMP (s)	W-PS3	W-SHRIMP
1	40	100	44.8	69.01	0.89	1.45
2	80	100	175.91	122.41	0.45	0.82
3	100	40	165.56	167.18	0.60	0.0014
4	80	60	48.36	43.36	1.65	1.38
5	0	60	83.7	74.92	0	0.80
6	0	100	65.33	98.05	0	1.02
7	40	20	119.67	76.08	0.33	0.26
8	20	0	138.78	62.72	0.14	0
9	0	40	86.58	70.81	0	0.56
10	40	100	159.72	79.09	0.25	1.26
11	0	40	64.96	58.66	0	0.68
12	40	100	57.8	61.42	0.69	1.63
13	0	20	61.45	64.78	0	0.31
14	40	40	42.69	68.08	0.94	0.59
15	40	60	60.78	59.51	0.66	1.01
16	0	40	46.13	64.92	0	0.62
17	60	100	72.15	63.47	0.83	1.58
18	40	40	148.96	92.6	0.27	0.43
19	80	40	39.53	77.45	2.02	0.52
20	20	100	73.54	81.62	0.27	1.22

Table 2: The raw dataset obtained for each participant.

$W^S_{rescaled}$ and $W^P_{rescaled}$ are the rescaled values of W^S and W^P respectively.

The pie chart in Figure 8 summarizes the results of $W^S_{rescaled}$ (performance level of SHRIMP) and $W^P_{rescaled}$ (performance level of PS3). The results indicate that average performance-levels with PS3 (38%) are approximately halved compared to the SHRIMP (62%). The raw results are statistically significant against a paired t-test with a p-value of 0.05. The significance of these results suggests that the average performance level occurred by PS3 gamepad is lower than the average HRI performance level of SHRIMP.

Further analysis of the normal distributions in W-values highlights the significance of performance differences between our SHRIMP framework and the PS3 gamepad. As illustrated in Figure 9 SHRIMP's standard deviation in W-values (0.497810641) is comparatively lower than the standard deviation of W-values in PS3 (0.562250748). This results in having a leaner normal distribution for SHRIMP in contrast to the distribution of PS3. These results enable us to conclude that our hypothesis is valid and the AR based HRI is more intuitive and performance-wise higher than traditional directly controlled HRI methods.

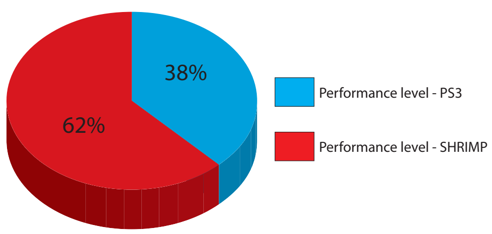


Figure 8: Performance-levels of PS3 & SHRIMP

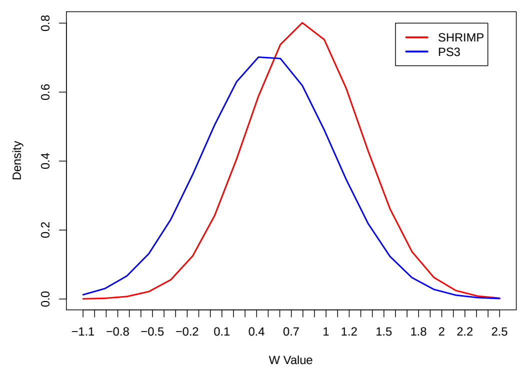


Figure 9: Normal distribution of W-values in SHRIMP & PS3

7 FUTURE WORK

Future work will focus on displaying AR markers within a re-constructed model of the environment. This could be achieved by the integration of our model with some existing 3D re-construction system and will also enable the robot to be tracked (in a 3D map), enhancing the visual feed-back to the user.

8 CONCLUSION

Future applications of robotics will involve people with little or no practical experience in controlling robots. We aim to improve the level of HRI experience for such people by introducing a new form of HRI middleware, namely SHRIMP. The proposed HRI middleware (i.e. SHRIMP) can work as a generic framework across different robot platforms and with different HRI tasks. In this paper we showed two distinct HRI tasks, a navigation task and a gripping task, carried out through SHRIMP. We further discussed our investigations with SHRIMP on measuring the HRI performance in contrast to a PS3 gamepad. Based on our results, we saw that AR-based SHRIMP implementation is less stressful when compared to directly controlled methods. Further experimentation should be performed with an increased sample size to make the statistical analysis more robust. Finally the study leaves us with the following question. Will augmented reality be the HRI middleware of the future?

9 REFERENCES

[Abbas et al., 2012] Abbas, S., Hassan, S., and Yun, J. (2012). Augmented reality based teaching pendant for industrial robot. In *International Conference on Control, Automation and Systems*, pages 4–7.

[Castle et al., 2008] Castle, R., Klein, G., and Murray, D. W. (2008). Video-rate localization in multiple maps for wearable augmented reality. *2008 12th IEEE International Symposium on Wearable Computers*, pages 15–22.

[Chen et al., 2008] Chen, I., MacDonald, B., and Wünsche, B. (2008). Markerless augmented reality for robots in unprepared environments. In *Australasian Conference on Robotics and Automation*.

[Fang et al., 2012] Fang, H., Ong, S., and a.Y.C. Nee (2012). Interactive robot trajectory planning and simulation using Augmented Reality. *Robotics and Computer-Integrated Manufacturing*, 28(2):227–237.

[Garcia-Arroyo et al., 2012] Garcia-Arroyo, M., Marin-Urias, L. F., Marin-Hernandez, A., and Hoyos-Rivera, G. D. J. (2012). Design, integration, and test of a shopping assistance robot system. *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction - HRI '12*, page 135.

[Han et al., 2012] Han, Y., Sumi, Y., Matsumoto, Y., and Ando, N. (2012). Acquisition of object pose from barcode for robot manipulation. *Simulation, Modeling, and Programming for Autonomous Robots*, 7628:299–310.

[Hu et al., 2013] Hu, H., Gao, X., Sun, H., Jia, Q., and Zhang, Y. (2013). Design and implementation of the teleoperation platform based on augmented reality. *2013 IEEE 12th International Conference on Cognitive Informatics and Cognitive Computing*, pages 126–132.

[Jean and Lian, 2012] Jean, J. and Lian, F. (2012). Robust visual servo control of a mobile robot for object tracking using shape parameters. *Control Systems Technology, IEEE . . .* , 20(6):1461–1472.

[Karkoub et al., 2012] Karkoub, M., Her, M.-G., Huang, C.-C., Lin, C.-C., and Lin, C.-H. (2012). Design of a wireless remote monitoring and object tracking robot. *Robotics and Autonomous Systems*, 60(2):133–142.

[Lakshantha and Egerton, 2014] Lakshantha, E. and Egerton, S. (2014). Towards A Human Robot Interaction Framework with Marker-less Augmented Reality and Visual SLAM. *Jounal of Automation and Control Engineering*, 2(3):250–255.

[Lee and Lucas, 2012] Lee, S. and Lucas, N. (2012). Development and human factors analysis of an augmented reality interface for multi-robot tele-operation and control. *SPIE Defense, . . .* , 8387:83870N–83870N–8.

[Leutert et al., 2013] Leutert, F., Herrmann, C., and Schilling, K. (2013). A Spatial Augmented Reality system for intuitive display of robotic data. *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 179–180.

[Li et al., 2012] Li, W., Duan, F., Chen, B., Yuan, J., Tan, J. T. C., and Xu, B. (2012). Mobile robot action based on QR code identification. In *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 860–865. Ieee.

[Martinez et al., 2013] Martinez, D., Gonzalez, M., and Huang, X. (2013). An Economical Testbed for Cooperative Control and Sensing Strategies of Robotic Micro-vehicles. *Informatics in Control, Automation and Robotics*, 174:65–75.

[Parasuraman et al., 2000] Parasuraman, R., Sheridan, T. B., and Wickens, C. D. (2000). A model for types and levels of human interaction with automation. *IEEE transactions on systems, man, and cybernetics. Part A, Systems and humans : a publication of the IEEE Systems, Man, and Cybernetics Society*, 30(3):286–97.

[Payton et al., 2001] Payton, D., Daily, M., Estowski, R., Howard, M., and Lee, C. (2001). Pheromone robotics. *Autonomous Robots*, 11(3):319–324.

[Steinfeld et al., 2006] Steinfeld, A., Fong, T., and Kaber, D. (2006). Common metrics for human-robot interaction. *Proceeding of the 1st ACM SIGCHISIGART conference on Humanrobot interaction HRI 06 (2006)*, 15(2):33.