Human Robot Interaction and Control: Translating Diagrams into an Intuitive Augmented Reality Approach

Eranda Lakshantha
School of Information Technology
Monash University Malaysia
Malaysia
eranda.lakshantha@monash.edu

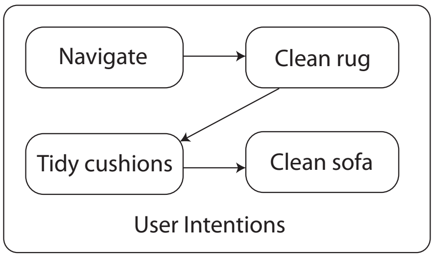
Simon Egerton
School of Information Technology
Monash University Malaysia
Malaysia
simon.egerton@monash.edu

***Abstract*—Robots will play a vital role in our future personal spaces and we need to provide ways for robots and humans to interact with each other in a way that is natural, intuitive, descriptive and unambiguous. In this paper we introduce a framework that enables a human and robot to naturally interact and communicate with each other using the idea of diagrams. The diagrams are formed by a connected set of object markers placed in the environment. These markers can either be physically present in the environment or virtually present using marker-less technology. This paper presents a marker-less method for globally persistent markers. Diagrams are formed by connecting the objects together enabling the user to easily interact and control the robot in complex ways. We report on a proof-of-concept implementation of our framework and show how the framework can be used to program a robot to carry out navigation and action tasks within an environment.**

***Keywords*-human robot interaction; augmented reality; robotics; vision-based navigation;**

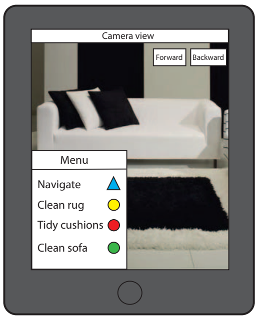
I. Introduction

Robots are the next technological and social revolution in waiting. They will form part of our futures and everyday lives, playing valuable roles in our homes, offices, factories and hospitals. In all of these scenarios robots and humans will be interacting with each other, while trying to make everyday actions more efficient. In many cases people who have little practical experience with these complex machines will need to collaborate with them and even program them or instruct them to perform a list of tasks. It will be the role of the Human Robot Interaction (HRI) sub-system, embedded into the robot's architecture, to bridge these experience gaps and make the collaboration between man and machine intuitive and seamless. We envisage a successful HRI platform being multi-modal, encompassing language, visual, tactile, role-play and imitation interactive mechanisms. In this paper we explore one of the most powerful and earliest forms of human communication, '*diagrams*'. Diagrams enable humans to communicate with each other in an intuitive manner and allow the exchange of information across language and cultural boundaries. In the context of this paper we view diagrams in their broadest sense as a mechanism for referencing physical space, instrumenting that space with markers, instructions and messages. Conceptually, these diagrams translate into 3-Dimensional information trees (or 'maps') rendered in physical space with Augmented Reality (AR) technology.

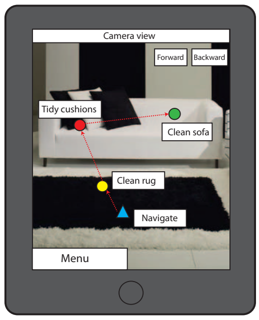


(a) A sequence of actions the user requires a robot to carry out.

(b) The environment in which the user intend to perform the above actions.



(c) The user's tablet interacting with the environment.



(d) The diagram objects are placed within the environment.

Figure 1: Application concept of our HRI framework.

The framework's concept is illustrated in Fig. 1. Here the user wishes to program the robot to carry out a specific set of tasks. The user's intentions are expressed by the illustration, in Fig.1a. Using the framework the user can

978-1-4799-2947-4/14 $31.00 © 2014 IEEE
DOI 10.1109/IE.2014.24

embed their intentions into the environment giving each task a context through spatial references.

The rest of the paper is structured as follows. Section 2 highlights the existing diagrammatic approaches used for human robot interaction. The details of our framework are described in section 3 whereas section 4 reports on our experimental methodology and planned experiments. Section 5 discusses future work and is followed by concluding remarks.

II. Related Work

In general our framework uses environmental markers to convey information to the robot and the robot can also use these markers to convey information back to the user. In this paper we describe our virtual markers rendered through augmented reality. However our framework can also use other types of environmental markers, such as barcodes, quick response (QR) codes and fiducial markers. Some of the more recent uses of these markers by other researchers in HRI are briefly described below.

The work of [1] presents a method of instrumenting the environment with bar-code markers, these markers are then processed to obtain object poses. While their work does not explicitly tag the objects with bar-codes, they point out that this could be done, which would also help facilitate object recognition. This would be particularly useful for any service robot that required to interact with objects within the environment, a super-market assistant for example. Another example of bar-code communication are the simulated worlds of Martinez [2]. Here the robots use special bar-code identifiers to derive their own pose information, position and heading, within the environment.

The faster Quick Response codes have recently been used by [3] as a means to reliably recognize objects within a household environment. They use the information to enable the robot to grasp and manipulate the objects in real-time. QR codes have been used to mediate collaboration between a human and a shopping assistant robot [4]. Here the assistant maintains a shopping list where it is able to select items and alert the user of any missing items after the shopping run.

Fiducial markers extend the ideas of the previous two markers and allow virtual objects to be overlaid upon then via an augmented reality interface. This allows for a richer human robot interaction possibilities. The work of [5] employs fiducial markers to plan optimal trajectories of a stationary robotic arm. A hand-held device which is attached with a fiducial marker cube is used for interacting with the robot. The marker dimensions are typically known to the tracking system allowing depth and perspective to be derived. Hu [6] develops this idea and uses the markers to render a virtual robot which acts on behalf of the real robot. Here the users can control the virtual robot to manipulate the remote real robot indirectly. The work of [7] uses related ideas, they use fiducial marker patterns for planning obstacle and collision free paths for a group of heterogeneous robots.

The above examples of markers all rely on artificial markers, however natural objects within the environment can themselves act as markers. This is more practical than instrumenting the environment artificially with barcodes, QR codes and fiducial markers, but comes at the expense of natural object detection and recognition. This is a fundamentally difficult task. However, the recent works of [8], [9], [10] show promising results in this area, with applications such as people tracking and following.

While artificial markers work well, they can be impractical in general as they require physical placement. Using natural objects as markers offers a nice solution, but this is difficult to implement in practice, especially in dynamic environments.

The marker framework we introduce in the next section seeks to combine the best of these markers while overcoming their shortcomings. The framework makes use of the PTAMM (Parallel Tracking And Multiple Mapping) algorithms [11] which are able to detect and track natural low level environmental features in local geometric spaces and render augmented reality objects into this space.

III. Spatial HRI Marker Platform

Our Spatial Human Robot Interaction Marker Platform (SHRIMP) integrates all of the spatial marker ideas above and introduces a new marker, '*virtual marker*' concept. Using the platform the user can place virtual information markers in the environment, or use existing physical markers and attribute information or tasks to them. The user can then connect these markers to program the robot with a sequence of tasks, and lay down navigation or patrol paths and so on. The user interacts with the platform and robot through a multimedia device, such as desktop or laptop PC, tablets, smart-phones or whatever the next generation of device might be, Google Glass for example. The overall SHRIMP architecture is illustrated in Fig. 2. The platform enables the user to interact with any robot connected into the system. This common interface enables

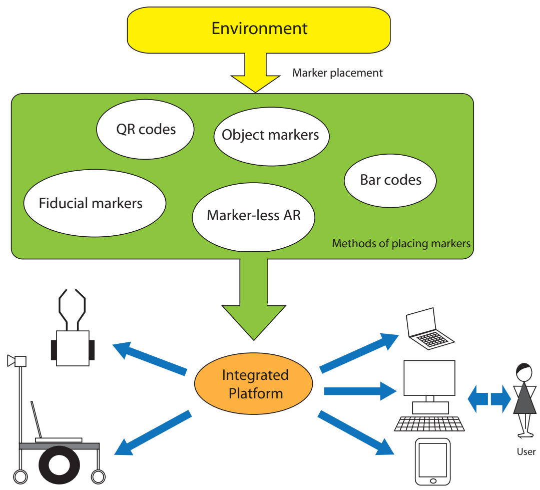


Figure 2: Overall architecture of the Spatial HRI Marker Platform (SHRIMP)

users to smoothly work with a wide range of robots with a minimal learning curve.

The user observes the environment through a video-feed, a mobile tablet or smart phone for example. The SHRIMP interface allows users to position AR markers or objects directly inside the environmental scene, and allows users to add, change, remove and or connect these objects together. These objects are typically tagged with instructions for the robot.

An example application scenario would be a hospital environment, where the SHRIMP platform could be used by a doctor to place routing information for care robots to follow for their rounds. As illustrated in Fig. 3, doctors would be able to lay down a series of augmented reality objects in order to formulate a navigation path for care robots to follow.

Our SHRIMP platform is built on top of the current state-of-the-art in marker-less AR technology, PTAMM (Parallel Tracking And Multiple Mapping) [11]. PTAMM identifies and tracks unique scale invariant features to accurately register AR objects with regard to the environmental scene. It further transforms the environmental scene into a 3-D coordinate system while allowing us to place AR objects at any arbitrary location in space. PTAMM by default, creates multiple local coordinate systems at multiple locations in space. However our SHRIMP platform requires a single globally referenced framework to enable AR objects to be tracked and persist when viewed from new directions.

To overcome this, we have implemented a linear transformation algorithm to integrate the local PTAMM coordinate frames into a single global coordinate frame model. The algorithm takes the initial camera position as the origin of the global coordinate frame, and expresses the camera pose with regard to this global frame origin. The system could create and switch into new *local* coordinate frames as the camera enters into un-explored regions in the environment. At the point of switching into a new coordinate frame, our algorithm expresses the current camera pose with regard to the global frame, by taking linear transformations into account. Transformations between coordinate frames are calculated by the following mathematical equation,

$$x' = \begin{bmatrix} R & | & t \end{bmatrix} x \qquad (1)$$

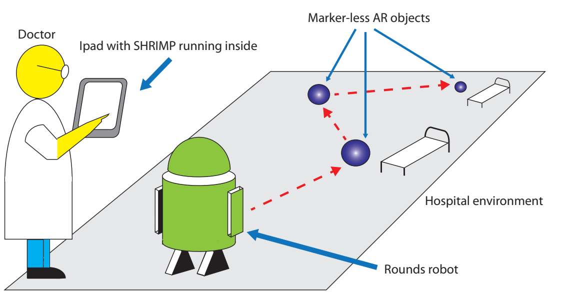


Figure 3: An example application: planning virtual routes for a rounds robot in a hospital


Figure 4: Global tracking persistence of our framework

x' and x are vector representations of a homogeneous point in global frame and local coordinate frame respectively. Matrix R denotes the amount of rotation while the vector t signifies the amount of translation between the two Cartesian spaces. Collectively $\begin{bmatrix} R & | & t \end{bmatrix}$ is known as the transformation matrix which enables a homogeneous point to be expressed among two coordinate systems. Grounded on linear equations, our algorithm allows AR objects to persist under different view angles, while capturing multiple local frames in relation to a fixed global frame. Further explanations of our algorithmic steps can be found in [12].

The series of images in Fig. 4 illustrate the tracking persistence of the resultant marker-less AR framework under different perspectives of the camera scene. The attached video file[1] further demonstrates the scenario highlighted in Fig. 4. As the video shows, we place an AR object in the environment during the initialization of our system. 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 a solid physical object persists at a location.

We have developed our framework under Ubuntu 12.04 and ROS (Robot Operating System). The SHRIMP platform consists of three main components namely, PTAMM, a linear transformation algorithm, and ROS as illustrated in Fig. 5. We merged our extended version of PTAMM with ROS via a custom-built software module in order to translate high-level HRI tasks into control signals. 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. This enables us to experiment with

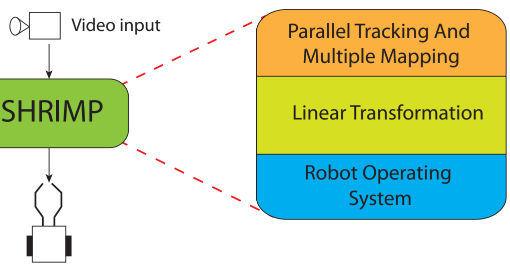


Figure 5: Internal components of SHRIMP

[1] http://youtu.be/cu7BIbyKMNc

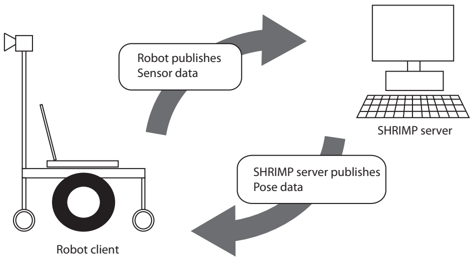


Figure 6: Communication model between SHRIMP and a robot

different robot platforms with minimal changes to the code base.

Fig. 6 illustrates the communication model between the robot client and the SHRIMP server. The robot is publishing sensor data and event information back to the server. The server publishes pose data and event information back to the robot, instructing the robot to carry out tasks, guided by the AR objects input into the system by the user.

IV. Applications of Our Model in HRI

In order to evaluate the usefulness of our framework we have implemented a proof-of-concept system that allows users to set navigation tasks for robots connected to the platform. Initial experiments were carried out with a Parallax robot and later with a LEGO Mindstorms robot. The SHRIMP platform is hosted on a ROS enabled server PC, where the PC receives a live feed from a web-cam connected to the robots. The web-cam captures the robot's environmental scene and is displayed to the user via SHRIMP. Fig. 7 illustrates our first experimental run with the Parallax robot. Here the user is placing a single navigation object within the robot's immediate scene. The SHRIMP platform then navigates the robot to the object marker by issuing the appropriate set of commands for the robot. Fig. 7 shows the robot's start location, and the three

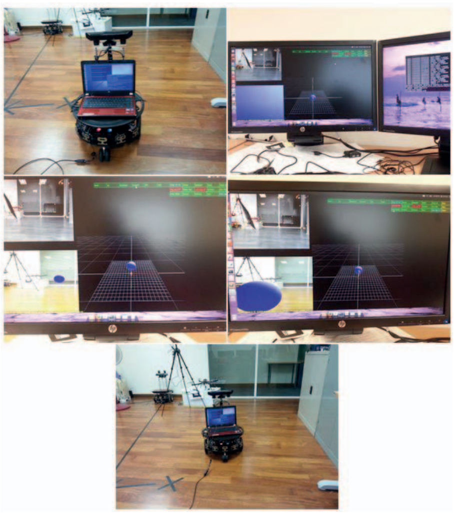


Figure 7: Navigating a Parallax wheeled robot with SHRIMP

lower images in the figure show the placement of the AR navigation objects into the robot's environment scene. The run is demonstrated in the online video footage here [2].

Fig. 8 shows a similar navigation scenario using an NXT Mindstorm robot. Here, SHRIMP has been extended to allow the user to lay down a set of AR navigation objects within the robot's environmental scene. These are then connected together to create a virtual navigation path. Fig. 8a depicts a virtual navigation path with the three AR navigation objects. Once the navigation path is complete the user instructs the robot to follow the path. SHRIMP issues the appropriate set of commands for the robot and the NXT proceeds along the navigation path visiting the AR navigation objects in connected order. This is illustrated in Fig. 8b, Fig. 8c and Fig. 8d respectively. The run is demonstrated in the online video footage here[3].

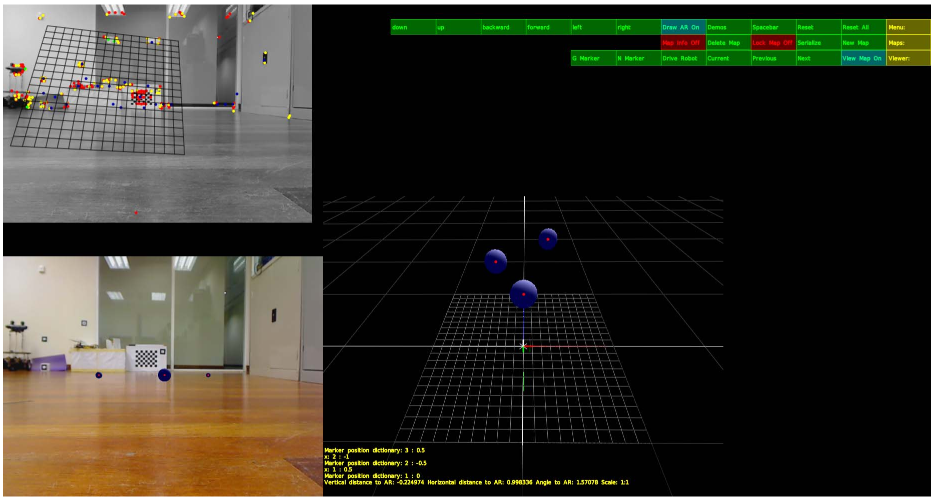
A. An HRI Control Task with SHRIMP

In this experiment navigational AR markers are extended with task AR markers. Here the task AR markers perform an object grip & hold task. In this scenario the user needs to navigate the robot to a specific location and grasp an object. They do this by placing navigational AR markers and task AR markers through SHRIMP. To help the user, the two kinds of AR markers need to be distinguished. There are many options here, using markers with different shape, colour, or textual descriptions. In our experiments we chose to use different colours with a colour coded key, which is the easier option. The control task is illustrated in Fig. 9. Here navigational AR markers are coded blue and the grip & hold task AR markers are coded pink. The run is demonstrated in the online video footage here [4].

V. Limitations & Future Work

This paper has reported an initial proof-of-concept implementation of our spatial human robot interaction marker platform. The system currently requires manual calibration of depth of field. Future work will make this process automatic, without having to require the user's intervention. There are several options here, either calibrating to a known fiducial marker, incorporating structured lights sensors such as the Microsoft Kinect, or using QR codes at known locations. The experimental trial reported above assumes the user has access to the same environment as the robot and is placing markers by directly referencing the space through the camera of a mobile device. It is also possible to place markers when remote from the environment through the robots camera feed. However, it is difficult for users to correctly perceive proper depth of field. To overcome this the robot could construct a 3-dimensional model of its local environment to guide the user. This could be achieved with a structured light sensor. The model could also be used to track the robot through the remote space.

[2]http://youtu.be/N0uQpxihSUo
[3]http://youtu.be/lIlpv3QPH5g
[4]http://youtu.be/ye8wKdJX7IY



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

(b) First point

(c) Second point

(d) Third point

Figure 8: Creating a virtual navigation path by marking multiple locations in space with multiple AR objects.

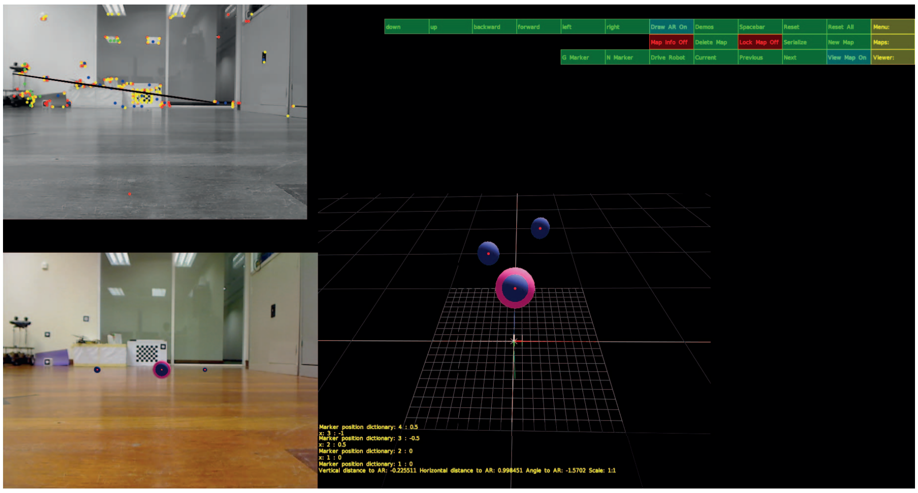


Figure 9: Placing multiple task AR objects in the environment using colour to represent different tasks types

Despite the issue of manual calibration, several other limitations are also noticeable in our current implementation of SHRIMP. For instance, the action tasks represented by AR objects (such as clean rug, tidy cushions, clean sofa, etc.) are highly contextual and abstract. Once given a certain abstract task, the robot has to figure out the sequence of individual actions that cover the overall objective of the task. Consider the action of '*tidy cushions*'. First the robot would have to find its way through a complex path until it reaches the correct location. At this point, the robot also might have to decide on which path to follow, i.e. the shortest path or the obstacle-free path. Next it would have to select a cushion, execute a hold & grip action and then place the cushion at right location. Finally the robot has to make sure that the task is successful and now the place actually looks tidy. Therefore the current system needs to be equipped with more sophisticated control algorithms to meet proper executions of high-level (abstract) tasks. Moreover, the problem gets even complex when we have several such abstract tasks laid onto the AR scene. At the moment, our framework does not drill down those abstract tasks into sub-tasks, neither does it evaluate the semantics of those tasks in isolation. One potential solution would be to enable the user to select and combine a set of predefined low-level actions (such as navigation, gripping, etc.) to construct high-level tasks on the fly. This allows users to customize high-level tasks while adapting those tasks according to their respective contexts. We further seek to address these issues with future enhancements.

VI. Conclusion

The aim of our work is to simplify the human robot interaction experience and allow humans with little or no experience with robot technology to intuitively operate and communicate with a diverse range of complex robot technology. We believe this can be achieved through the use of diagrams to convey information in both directions between the robot and human. To this end we have developed a framework that enables a human to effectively draw diagrams contextualized and embedded in the environment. Nodes in the diagrams are locations or objects in the environment, described by physically present markers or by using the virtual marker technology developed in our SHRIMP framework. Our early experimental tests with our proof-of-concept implementation indicate that non-experts can readily control a range of robots using our system with little to no learning curve. With future and further development we expect our model to play a key role in human robot interaction middleware of the future.

References

[1] Y. Han, Y. Sumi, Y. Matsumoto, and N. Ando, “Acquisition of object pose from barcode for robot manipulation,” *Simulation, Modeling, and Programming for Autonomous Robots*, vol. 7628, pp. 299–310, 2012.

[2] D. Martinez, M. Gonzalez, and X. Huang, “An Economical Testbed for Cooperative Control and Sensing Strategies of Robotic Micro-vehicles,” *Informatics in Control, Automation and Robotics*, vol. 174, pp. 65–75, 2013.

[3] W. Li, F. Duan, B. Chen, J. Yuan, J. T. C. Tan, and B. Xu, “Mobile robot action based on QR code identification,” in *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 860–865, Ieee, Dec. 2012.

[4] M. Garcia-Arroyo, L. F. Marin-Urias, A. Marin-Hernandez, and G. D. J. Hoyos-Rivera, “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*, p. 135, 2012.

[5] H. Fang, S. Ong, and a.Y.C. Nee, “Interactive robot trajectory planning and simulation using Augmented Reality,” *Robotics and Computer-Integrated Manufacturing*, vol. 28, pp. 227–237, Apr. 2012.

[6] H. Hu, X. Gao, H. Sun, Q. Jia, and Y. Zhang, “Design and implementation of the teleoperation platform based on augmented reality,” *2013 IEEE 12th International Conference on Cognitive Informatics and Cognitive Computing*, pp. 126–132, July 2013.

[7] S. Lee and N. Lucas, “Development and human factors analysis of an augmented reality interface for multi-robot tele-operation and control,” *SPIE Defense, ...*, vol. 8387, pp. 83870N–83870N–8, May 2012.

[8] P.-T. Huang, L. Chung-Ying, C.-C. Hsu, and C.-M. Hong, “Object following based on SURF for mobile robots,” in *IEEE 1st Global Conference on Consumer Electronics (GCCE)*, (Tokyo), pp. 382–386, Oct. 2012.

[9] J. Jean and F. Lian, “Robust visual servo control of a mobile robot for object tracking using shape parameters,” *Control Systems Technology, IEEE ...*, vol. 20, no. 6, pp. 1461–1472, 2012.

[10] M. Karkoub, M.-G. Her, C.-C. Huang, C.-C. Lin, and C.-H. Lin, “Design of a wireless remote monitoring and object tracking robot,” *Robotics and Autonomous Systems*, vol. 60, pp. 133–142, Feb. 2012.

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

[12] E. Lakshantha and S. Egerton, “Towards A Human Robot Interaction Framework with Marker-less Augmented Reality and Visual SLAM,” *Jounal of Automation and Control Engineering*, vol. 2, no. 3, pp. 250–255, 2014.