# JackIn: Integrating First-Person View with Out-of-Body Vision Generation for Human-Human Augmentation

**Shunichi Kasahara**
Sony Corporation
2-10-1 Osaki, Shinagawa-ku, Tokyo, Japan
Shunichi.Kasahara@jp.sony.com

**Jun Rekimoto**
Sony CSL, The University of Tokyo
3-14-13 Higashigotanda, Shinagawa-ku, Tokyo, Japan
rekimoto@acm.org

## ABSTRACT

*JackIn* is a new human-human communication framework for connecting two or more people. With first-person view video streaming from a person (called *Body*) wearing a transparent head-mounted display and a head-mounted camera, the other person (called *Ghost*) participates in the shared first-person view. With JackIn, people's activities can be shared and assistance or guidance can be given through other peoples expertise. This can be applied to daily activities such as cooking lessons, shopping navigation, education in craftwork or electrical work, and sharing experiences of sporting and live events. For a better viewing experience with frist-person view, we developed the *out-of-body* view in which first-person images are integrated to construct a scene around a Body, and a Ghost can virtually control the viewpoint to look around the space surrounding the Body. We also developed a tele-pointing gesture interface. We conducted an experiment to evaluate how effective this framework is and found that Ghosts can understand the spatial situation of the Body.

### Author Keywords

Augmented Human; First person view streaming; Augmented Reality; Out-of-body

### ACM Classification Keywords

H .: 5.2 [Information interfaces and presentation]: User Interfaces: Input devices and strategies; H.5.1 [Multimedia Information Systems]: Artificial, augmented, and virtual realities; I.3.6 [Methodology and techniques]: Interaction techniques.

## INTRODUCTION

Tele-communication technology has enabled users to talk remotely to each other using face-to-face video allowing a sense of co-existence. Research on tele-existence has focused on enabling humans to have a real-time sensation of being at a place other than where they actually are and interact with that environment [1].


AH '14, March 07 - 09 2014, Kobe, Japan

http://dx.doi.org/10.1145/2582051.2582097

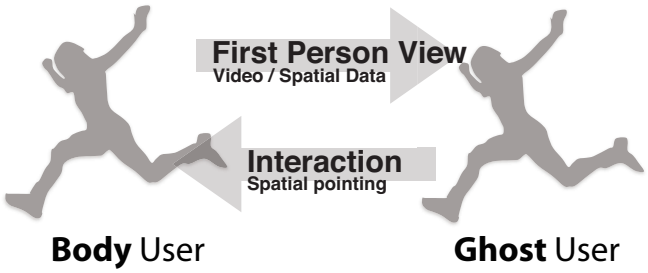



**Figure 1. JackIn overview. User, referred to as "Body", wears transparent head-mounted display (HMD) and head-mounted camera. First-person video from head-mounted camera is streamed to "Ghost" in remote location. Ghost can see and control generated wider scene of Body and point remotely into it. Body can detect pointing with transparent HMD.**

We frequently want to share our experiences with others. For instance, when we are traveling, we want to share all of what we feel and see

Photos and videos have partially enabled this kind of sharing experience, and recent developments in wide-angle wearable cameras, such as GoPro [23], has made it possible to record and share one's experiences. First-person view video streaming of what we experience would be a significant direction in video communication.

As an extension of first-person view streaming, bi-directional interaction through the first-person view should also be considered. For example, when rescue workers in a disaster area need expert information to make decisions, they might want to send specific information about the situation via first-person video to experts in a remote location. In other words, an expert is allowed to enter a body to perceive the shared experience.

**JackIn** is an interaction framework that allows us to stream and share our experiences or receive assistance or guidance by communicating with others in a remote location. The term "jack in" was originally described as an action in which a person enters cyberspace in the novel "Neuromancer" by William Gibson [22]. We expanded this concept an interaction in which people enter another person.

In the **JackIn** framework, we define a **Body** as a person doing an activity in an actual environment and a **Ghost** as a person who receives the transmitted experience and gives guidance via a network. The number of Ghosts can be a one or

more, and also be a large number when they are watching streaming first-person video. In an implementation of jackIn interaction, a Body wears a transparent head-mounted display (HMD) and a head-mounted camera to stream the first-person view video, and a Ghost can look around the Body's environment and visually participate in that first-person view.

JackIn can be considered a kind of augmented reality framework. A Body is allowed to receive navigation instruction and guidance via an optical transparent HMD, which is a type of information overlay common in augmented reality. In conventional augmented reality frameworks, the overlaid information for visual augmentation is acquired from computer intelligence and databases; however, in our JackIn framework, humans provide information for augmenting other humans. This form of communication will enable us to not only share personal experiences but also help each other remotely.

We also consider JackIn as a kind of tele-presence. Conventional tele-presence frameworks allow users to jack into machines at a remote location and see the remote environment from the viewpoint of the machine. However, our focus is communication where humans jack into humans. In human-machine tele-presence, humans are *masters* and machines are *slaves*. On the other hand, when humans jack into humans, the behavior of the Body is not controlled by the Ghost. How to design interfaces to allow individual independence is an issue.

Applications with the JackIn framework include, cooking lessons, shopping assistance, navigation, education in craftwork, and sharing experiences at sporting and other live events.

The remainder of this paper is as follows. We begin by explaining the JackIn interface and related work. We then introduce the JackIn architecture and its implementation. We also report on an evaluation of our framework in terms of it effectiveness. Our discussion focuses on these results and future work.

## JACKIN INTERFACE AND RELATED WORK

We now describe the interface for JackIn and related work from two aspects; first-person view video streaming and remote tele-pointing.

### First-person view video streaming

As a modality for streaming one's experiences, we focus on the first-person view as the first phase. There have been several studies in remote collaboration on finding ways to enable remote collaborative physical tasks with video mediated systems [4, 11], focusing on how an expert user in a remote location understands the situation of the local user and gives instructions.

To understand the detailed situation of a local user, local users with a head-mounted camera and remote users who are looking at the streaming video can communicate with a common understanding, i.e., a sense that they are watching the same situation [9].

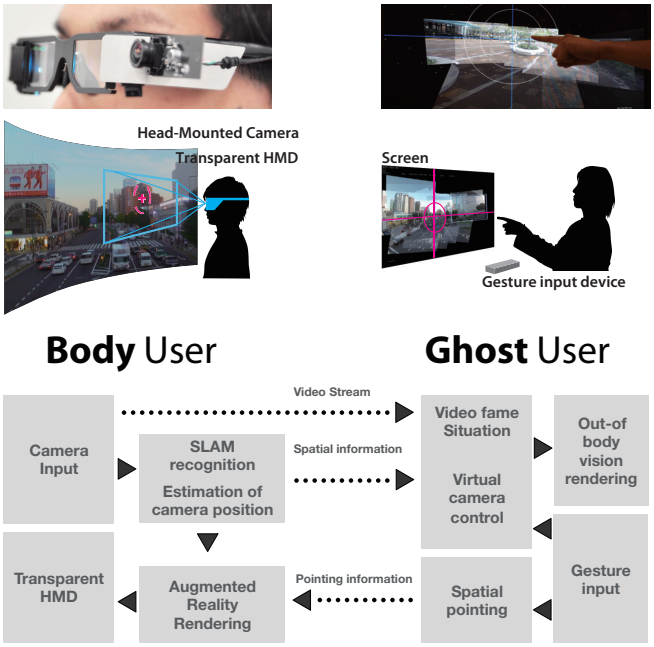



**Figure 2. JackIn Architecture, "Body" wears transparent HMD and head-mounted camera. First-person view video from camera is streamed to "Ghost" in remote location. Ghost can see and control view point for integrated wider scene around Body and point remotely into scene. Body can detect pointing with transparent HMD.**

However, video images from a head-mounted camera are often too shaky, which may cause motion sickness in the other person [18] and detract from video content. Fussell et al. also pointed out these problems with first-person video mediated collaboration [5]. In particular, the remote user is focused on the same viewpoint as a local user. To give guidance from a different viewpoint, the remote user has to instruct the local user to move toward the desired position.

To solve these problems, we propose an out-of-body view, which is generated from the integration of the first-person video that provides a spatially registered bird's-eye view around the local person. This enables the other person to virtually look around the first person's environment independently from the head motion of the local person.

Out-of-body view can be partially achieved with an external camera in addition to a head-mounted one; however, installing such equipment is not practical. To generate an out-of-body view, 3D re-construction around the Body such as by using kinect Fusion [15], can be an option, but this depth image sensing is not suitable for outdoor use. Research on the visualization of an environment and integration of photos enables a wider vision of the pre-captured situation ([12]). Previous studies also achieved 2D camera image stitching for providing a wider view of the local user ([9, 16]). However, active viewpoint control by remote users would be required for remote assistance use.

Therefore, we propose **out-of-body view generation from first-person video from a Body's head-mounted camera** and **viewpoint control of out-of-body view for a Ghost independently from the Body's movement**. Actual imple-

mentation is a system virtually constructed around the view scene from first-person video on the basis of simultaneous localization and mapping (SLAM) [24]). The virtual camera for rendering the constructed scene can be controlled by the Ghost to enable out-of-body view.

#### Remote tele-pointing interface

We also developed a method for interacting with the local people through the first-person view. In a video-mediated collaboration system, gesture expression is an important factor as well as verbal communication for establishing common understanding of a space. ( [6, 7]).

Examples of such scenarios are "Press this button" for assistance in operating a machine, or "Grab that fruit instead of this" for remotely ordering in a supermarket. With this common spatial understanding, local and remote users can have a conversation using deictic words, e.g., "this one", "that way", and "it" while feeling a sense of spatial coexistence ( [8]).

To visualize tele-pointing, the pointing position on the video pixel should be converted into a spatial 3D point in the Body's space. In previous work with image recognition or 3D sensing technology, systems that allow the user to transfer 3D hand gestures to remote users with mobile screens and side setup displays have been proposed [2, 3, 10]. However, these implementations do not allow the Body to see real-world augmentation, and indicating gestures outside the current view of the Body is not supported.

To solve these problems, we also propose a **method for tele-pointing registered in the Body's space (not in pixels)** and n **graphical user interface for indicating pointing outside the Body's view.**

## JACKIN ARCHITECTURE

In this section, we describe the architecture for implementing the JackIn framework, as shown in Figure 2). The Body wears a transparent HMD and head-mounted camera. First-person video from the head-mounted camera is streamed to the Ghost in a remote location. The Ghost can see and control a wider scene around the Body and remotely point into it with a gesture detection device. The Body can detect the pointing with the transparent HMD.

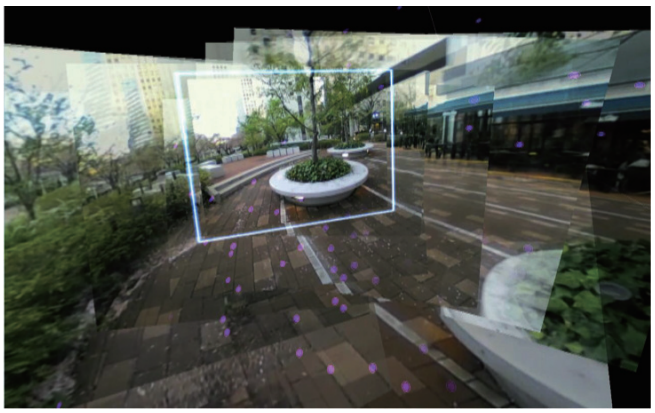


**Figure 4. Out-of-body view for Ghost is constructed from latest and previous frames. Video frames are spatially stitched with spatial information in real time.**

There are three important feature of the out-of-body view in the JackIn architecture. First, it consists of spatially situated video frames of the first-person video to provide a wider scene around the Body (Figure 4). Second, it provides independent control of the viewpoint from the head motion of the Body. Third, the Ghost can point though the out-of-body view, which also appears as overlaid graphics on top of the Body's actual environment with spatial navigation. Each feature is based on the spatial relationship between the estimated camera position and the environment acquired from SLAM. The entire generation and control process of the out-of-body view is shown in Figure 2. We describe this process and features in detail in the following section.

#### Generation of out-of-body view

The framework performs real-time SLAM from the captured video frame and generates extended image view by spatially stitching the current and previous video frames (Figure 4), and the current video frame is rendered as real-time video.

When SLAM tracking begins, the Body's estimated camera position ($M_{bc}$) in the reference frame of the space and 3D image feature point group $P_0, P_1..., P_N$ are acquired (Figure 3 (a)). Here, $M$ represents a 16-dimention matrix, which includes position and rotation information.

When a new video frame is going to be situated, the position of the video frame ($M_f(t)$) is defined as $M_f(t) = M_{bc}(t)T(d)$ (Figure 3 (b)).

Here, $T(d)$ means the translational offset toward the Body's camera direction. This is defined on the basis of the camera angle and distance between position of the Body camera and point $P'$ which is nearest to the Body's camera direction from image feature point group $P_0, P_1..., P_N$. Then view rendering is performed to visually stitch current and past video frames.

A new video frame to be situated is compared with existing ones. If the new one is relatively new compared with those that exist around it, or if there is no existing frame around, the new frame is situated (Figure 3 (c)).

Through this process, video frames are situated in a recognized reference coordinate and rendered as a scene from the virtual camera position controlled by the Ghost.

#### Interaction with out-of-body view

These constructed spatial frames are rendered from the virtual camera position ($M_v$) where the Ghost will see the out-of-body view including the current frame (Figure 4), and the Ghost can control it. Each situated frame is rendered with alpha control by comparing the current virtual camera orientation ($M_v$) to stitching rendering failure.

Interaction with the out-of-body view has three different modes: **Follow Mode** where the Ghost can passively view the out-of-body view, **Viewpoint Control Mode** where the Ghost can control the viewpoint, and **Pointing Mode** where the Ghost can tele-point in the out-of-body view (Figure 5).

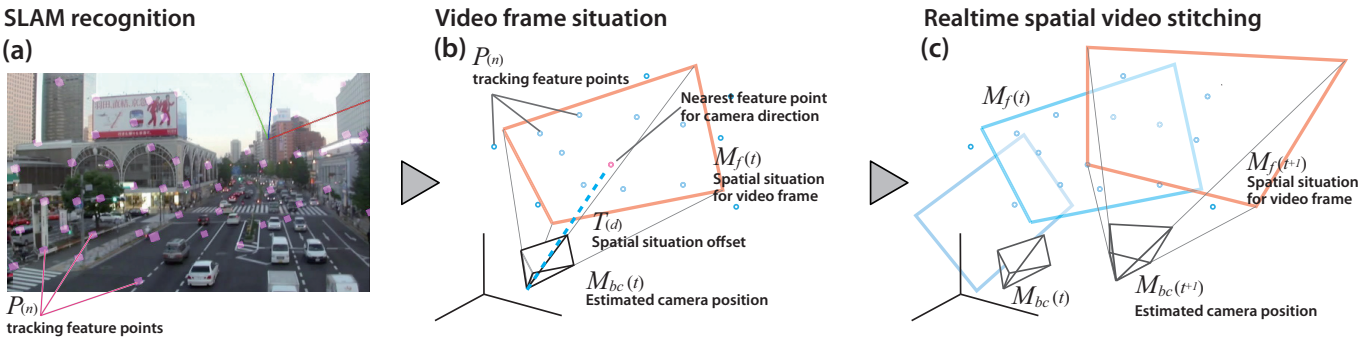



**Figure 3. Generation of out-of-body view: it virtually constructs wider view from first-person video on basis of SLAM.**

*Follow Mode*

In the follow mode, motion of the virtual camera is controlled, as shown in Figure 5-(a). In each frame update at time $t$, on the basis of SLAM recognition, the Body's camera position $M_{bc}(t)$ is acquired. It is smoothed by temporal filtering where the translational value $T_{bc}(t)$ and rotational value $R_{bc}(t)$ are smoothed with parameters $k_T$ and $k_R$, respectively. Then a smoothed camera position $M'_{bc}(t)$ is acquired from $T'_{bc}(t)$ and $R'_{bc}(t)$.

$$T'_{bc}(t+1) = k_T T_{bc}(t) + (1-k_T)T'_{bc}(t)$$

$$R'_{bc}(t+1) = k_R T_{bc}(t) + (1-k_R)R'_{bc}(t)$$

Then the virtual camera position ($M_v$) is controlled to obtain a closer position where is translated toward the backward with a certain offset $T_{offset}$ from $M'_{bc}(t)$. This allows the Ghost to simultaneously understand the spatial situation with a wider view and the current video frame.

The response speed of the virtual camera position against the Body camera position is adjusted with parameter $k_v$. This parameter is adjusted to be small when the current frame is within the viewpoint of the virtual camera $M_v(t)$ (Hence, the virtual camera moves slowly and renders a spatially static view), and becomes larger when the current frame goes outside the viewpoint of the virtual camera to follow the real-time frame.

$$M_v(t+1) = k_v M'_{bc}(t) T_{offset} + (1-k_v)M_v(t)$$

With this process, the virtual camera allows the Ghost to observe a spatially extended video frame with smoothed motion. These motion controls are performed automatically without manipulation from the Body or Ghost.

*Viewpoint Control Mode*

When a user starts pointing from a certain distance or more against the screen display, the viewing mode changes to the viewpoint control mode and the Ghost can control the viewpoint of the out-of-body view (Figure 5 (b)).

With a gesture detection device, the system acquires a 2D position on the screen. A hit-test of pointing for each frame is carried out to determine the focus frame ($f'$) from the spatially situated frames.

The position of the virtual camera is controlled to get closer to the position where the virtual camera would look at the focus frame ($M'_f$) with the offset $T(d)_{offset}$ Here, $T(d)_{offset}$ is defined so that the camera position is located at a farther position, then the Ghost can observe a wider view when moving.

In this view mode, the Ghost can control the viewpoint independently from the Body's movement.

*Pointing mode*

Finally, when the finger of the user gets closer to the screen within a certain distance, pointing mode begins. In this mode, the position of the virtual camera is controlled with the same algorithm as the viewpoint control mode with a smaller offset, which makes the virtual camera get closer to the focus frame. This enables the Ghost to observe the focus frame with a zoom-in view; thus, confirming the details inside the video frame.

As a main function in this mode, the Ghost's pointing is registered in the space around the Body. When the Ghost points, the 2D position on the screen is compared with the projected 2D positions of each spatial feature point. Then the nearest three points $P'_0, P'_1, P'_2$ are determined from the image spatial feature point set $P_0, P_1 ..., P_N$.

The 3D pointing position $P'(x, y, z)$ is defined under the assumption that spatial pointing is on the same plane with these three points.

$$P'(x, y, z) = a(P'_2 - P'_0) + b(P'_1 - P'_0)$$

Then, the 3D point on the plane can be calculated by solving $a$ and $b$. Through this process, pointing to a pixel of the video the Ghost pointed is converted into a 3D point on the basis of spatial recognition.

**Interface between Body and Ghost**

The Body can see the spatial pointing that the Ghost generated as augmented reality graphics. These graphics are rendered to match the view of the actual environment with an optical transparent display (Figure 6). To display this augmented reality graphics on the transparent display, the spatial

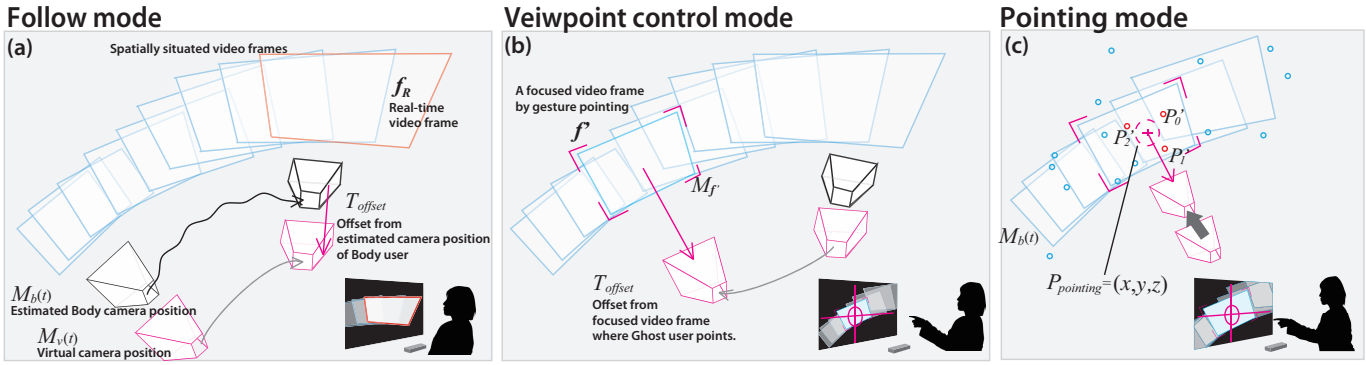



**Figure 5. Ghost can control viewpoint of out-of-body view by gesture input**

relationship between the camera image of the wearable camera and the view of the naked eye are calibrated in advance.

The field of view of the transparent display is limited and can be smaller than that of the camera. With the out-of-body view, the Ghost can see a wider view than that of the Body. Therefore, we need to provide graphical navigation for annotations that are outside the field of view of the HMD.

In the JackIn architecture, we applied the halo approach [17] to navigate toward a spatial position where an annotation exists. This halo approach is used for visualizing off-screen objects for small displays, which draws a circle with the center at the point of the off-screen objects. A user can intuitively estimate the approximated position through the direction of the arc and radius even if objects exist in the off-screen region. The process of halo-approach augmented graphic navigation is shown in Figure 7. The graphical behavior is defined as follows.

When the spatial annotation is located outside the field of view of the HMD and far enough away from the center view of the HMD, an arrow graphic is rendered to show the direction (Figure 7- a).

When the Body's head moves and the spatial annotation gets closer within a certain distance, a circle is rendered. The center position of this circle is located at the annotation, and the radius of the circle is defined as the distance between the center of the HMD viewing area and the position of the target annotation (Figure 7-b).

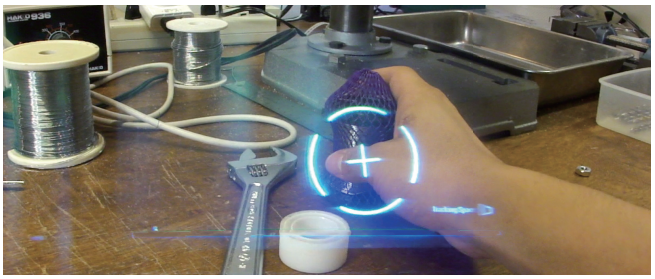


**Figure 6. Visualization example of spatial remote pointing from Ghost. These graphics are displayed in transparent HMD of Body.**

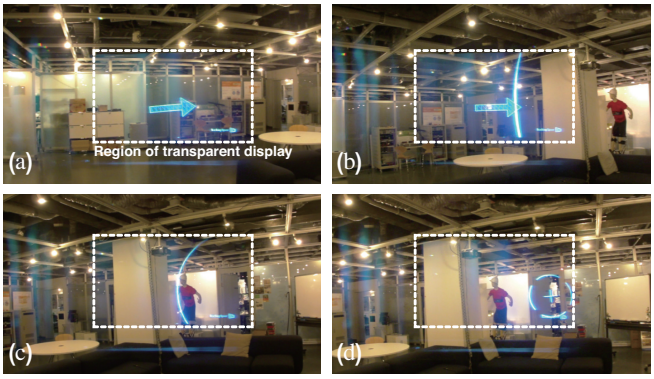



**Figure 7. Halo-approach augmented graphic navigation for spatial pointing**

When the Body moves his/her head to get the center of the circle within the display region, the radius becomes smaller and vice versa (Figure 7-c). This enables users to intuitively estimate the spatial position outside the display. When the target annotation enters the HMD view area, the circles change into target indicators that appear on top of the physical position (Figure 7-d). Through this graphical user interface, the Ghost can understand and estimate the spatial situation of the target annotation even though the annotation is outside the field of view of the HMD.

## IMPLEMENTATION

The Body wears a transparent HMD (LUMUS DK-32 [14]) with 40 diagonal angle of view and 1280 x 720 resolution, and a USB camera with 80 horizontal degrees is also attached on the left or right of the head based on the Body's dominant eye. They are connected to a laptop computer running the JackIn Body-Side application that performs SLAM, as well

as spatial recognition data streaming, video streaming of first-person view with motion jpeg compression, 20 Fps, 640 x 480 VGA images, and rendering remote pointing graphics created by the Ghost. We use the software library SmartAR [13] for SLAM tracking, which tracks the spatial feature points and estimates the 3D situation between the environment and camera.

The Ghost uses Leap motion [21] as a gesture input device and a 40-inch screen for displaying the out-of-body view. The Ghost-Side application generates the out-of-body view from received streaming video and spatial recognition data, controls the out-of-body view through gestures, and sends the spatial pointing information to the Body side. The Ghost and Body laptop computers are connected over a WLAN. The video streaming and data of the spatial recognition and pointing are exchanged using the UDP protocol. Approximate video latency is 400 msec over a network environment.

## USER STUDY

We conducted a user study to evaluate the advantages of out-of-body view for video mediated communication with the JackIn system . We investigated how the JackIn system including out-of-body view changes communication between the Ghost and Body through recorded video and motion data. We also collected informal and anecdotal data from users on how the Ghost and Body felt throughout the experiment.

### Task design

We chose remote task assistance, which is based on a use case in which the Ghost remotely assists the Body in choosing laboratory tools from scattered items in a lab workspace (Figure 8).

We used a half-assembled Lego block as the target item, which is difficult to identify with only verbal expressions. The Ghost, as the expert, knew what item to use, but had no idea where it was in the spatial situation around the Body.

Each participant was asked to perform the task as a Ghost and to communicate and instruct the Body in a different room to pick up the target item. On the Ghost side, the following two conditions with difference modes were compared, 1) the JackIn system with out-of-body view and 2) the JackIn system with first-person view video, and both test modes included other functionalities such as remote pointing into 3D space and voice communication.

To minimize individual variation and background knowledge, one person who was the Body. The Body was not able to know which test condition was to be used.

### Procedure and Participants

In the task, 12 half-assembled Lego blocks with unique IDs were placed randomly around the lab workspace. The same set of items was used for task instruction for the Ghost. The Ghost was allowed to control the out-of-body view (mode 1 only) and participate in any conversation except telling the Body the ID of the target object. The task completed when the Ghost and Body confirmed that the Body retrieved the target object. Under each test condition, the spatial placement

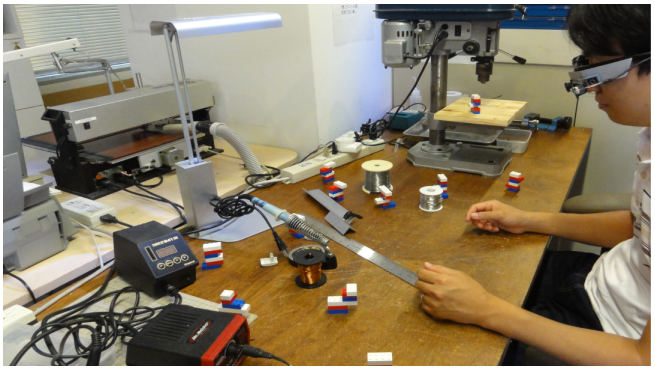


**Figure 8. Environment of user study experiment**

**Table 1. Questionnaire results**

| | Without Out-of-body | | | With Out-of-body | | |
|---|---|---|---|---|---|---|
| Question | Mean | SD | % | Mean | SD | % |
| Q1 | 5.0 | 1.48 | 6/10 | 6.0 | 1.07 | 8/10 |
| Q2 | 4.0 | 1.35 | 4/10 | 6.0 | 0.63 | 10/10 |
| Q3 | 6.0 | 1.08 | 9/10 | 5.0 | 1.15 | 7/10 |
| Q4 | 5.5 | 1.16 | 8/10 | 5.5 | 0.70 | 10/10 |

of the target item was replaced randomly. The task for each condition was performed three times and the order of the two different conditions was counterbalanced randomly for each participant.

We recruited ten participants, ranging in age from 20 to 40 with regular computer skills. Five participants had used some type of spatial gesture device such as Microsoft Kinect and Leap Motion. Participants were observed and video recorded for later analysis, and they filled out a post-test questionnaire.

### Results

Table 1 lists the results from the questionnaire regarding Q1:"ease in finding the target", Q2: "ease in understanding the remote situation", Q3:"ease in learning how to use" and Q4: "ease in operating the system". The questions were based on a 7-point Likert scale; 7=agree and 1=disagree.

The results show the mean, standard deviation (SD), and percentage of positive responses (represented as %, 4 to 7 on a 7-point Likert scale) for each question. For Q2, four users answered positively without out-of-body view, on the other hand, all users answered positively with out-of-body view.

The results showed that with out-of-body view, participants could understand the remote spatial situation easier than with fist-person view. Regarding this question, eight participants mentioned that they felt less anxiety with out-of-body view. Six participants also commented that they could not remember the situation after only glancing at the first-person view. On the other hand, four participants commented that they felt more confident when they chose the target object with out-of-body view because they could compare probable targets by changing the viewpoint and choose the most likely one.

Some participants mentioned that they needed more practice with the gesture input interface. They sometimes felt it difficult to point at a certain spatial point in both the out-of-

body and first-person views. This is caused by error in spatial conversion (described in the JackIn architecture section) and gesture error when they changed the mode from pointing to control.

Note that task completion time varied from 5 sec to approximately 120 sec, which seems to be highly dependent on the locations of the target items and individual skills such as verbal description and retrieval with 3D dimensional rotation of the object.

### Sequence observation of interaction

We also observed the communication and interaction sequences of each task by analyzing the recorded motion data. During each task, we recorded the estimated Body's camera rotation and virtual camera rotation, which was controlled by the Ghost,.

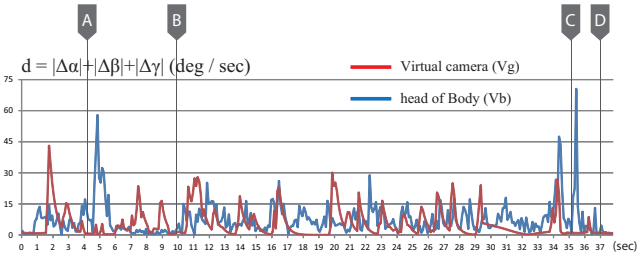
We plotted the temporal changes in the camera angle per second (top of Figure 9-(a) to analyze the degree of activeness of each camera motion. Larger values indicate the active movement of each camera angle. The differential angles between the Body's camera and the virtual camera of the Ghost were also plotted (bottom of Figure 9-(b), the larger values mean that the Body's camera angle and the virtual camera headed in different directions.

Figure 9 shows an example of a frequent interaction pattern. After 2 sec from the beginning, the Ghost moved quickly to see a different viewpoint and found the probable target then instructed the Body to verify by pointing (time A). Just after the instruction, the Body changed his orientation to see the indication, then the Ghost and the Body looked at the object from almost the same viewpoint (time B), then the Ghost checked it again with real-time video. However, the Ghost found that the probable target was not correct. The Ghost then described the target object for the Body and started to look around again.

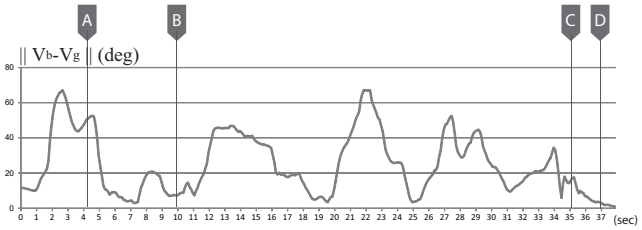
From time B to C, the Ghost and Body looked around independently, as we know from several small peaks and constant change in graph A. At 17-20 sec and 30-34 sec, the Ghost moved little and stayed almost at the same viewpoint even though the camera angle of the Body changed constantly. During these times, the Ghost looked at a certain area in the past video frame of the out-of-body view and kept comparing an item with one Ghost is having in the hand. Finally, the Ghost found a more probable target and pointed (time C), then the Body changed his orientation toward the target and confirmed that it was the answer (time D).

This kind of interaction sequence was found throughout the experiment. We observed that the Ghost could explore a remote situation independently from Body's motion.

As discussed in several previous studies [2, 8], with telepointing with both out-of-body and first-person views, we observed that the Ghost frequently used deictic words such as "this one", "that way", and "this, this", which signified that the Ghost and Body established a common understanding of the spatial situation. For instance, when they found the object




(a) Time-series variations in temporal orientation change in virtual camera (Vg) and head orientation of Body (Vb)




(b)Time-series variations in difference between orientation of virtual camera and estimated head orientation of Body

**Figure 9. Sequence graph for (a) temporal changes in camera (both Ghost and Body) angle per second, and (b) differential angle between Body's camera and virtual camera of Ghost**

was the wrong one, they checked with one another with just verbal description such as "check the left one".

Almost all participants used a combination of pointing and verbal expressions to mitigate pointing errors. Through these observations, we found that tele-pointing is effective for not only pointing but also for establishing a common understanding in a spatial situation.

We observed changes under the two different conditions regarding deictic words. With the out-of-body view and pointing, the Ghost used "this way" for direction as well as "this" for pointing. An interesting aspect of this observation is that these usage variations seemed to be based on whether the pointing was in the current video frame of the Body. It can be assumed that the Ghost might think that the current video frame is the region where the Body actually sees then change his/her verbal expression to describe the position.

## CONCLUSION AND FUTURE WORK

We proposed JackIn, a human-human communication framework in which a person can virtually enter another person. We described the JackIn architecture, which enables a user to observe a remote situation through a spatially extended first-person video called out-of-body view and provides an interface for remote navigation for a local user with augmented reality. We confirmed the effectiveness of our architecture and gained significant insights that will push our exploration forward.

However, we also found limitations with the current implementation regarding spatial recognition, which limits system feasibility in a wider workspace. For future work, we will investigate other sensing technologies such as integration with motion sensors and 3D reconstruction with depth sensors.

We believe that JackIn has the potential in which the abilities of one person (Body) can be augmented by another (Ghost). However, we focused on navigation and tele-pointing as interaction for this study. Other activities can be also augmented. For instance, interaction for giving more complex instructions to manipulate tools from remote location should be considered [26].

For this study, we focused on visual information and conversation as modalities for guiding a person; however, our JackIn framework can be also applied for other modalities such as tactile feedback, virtual force sensation [25], and electrical muscle stimuli. Assistance from multiple Ghosts should be also investigated [19].

## REFERENCES


1. Tachi, S., Arai, H., Maeda, T., Oyama, E., Tsunemoto, N. and Inoue, Y., "Tele-existence in real world and virtual world," Advanced Robotics, 1991.193-198
2. Rajinder S. Sodhi, Brett R. Jones, David Forsyth, Brian P. Bailey, and Giuliano Maciocci. 2013. BeThere: 3D mobile collaboration with spatial input. In Proc, CHI 2013. ACM Press (2013), pp.179-188.
3. Tecchia, F., Alem, L. and Huang, W. 2012. 3D helping hands: a gesture based MR system for remote collaboration. In Proceedings of the 11th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry (VRCAI '12). pp.323-328.
4. Kuzuoka, H., Spatial workspace collaboration: a SharedView video support system for remote collaboration capability, CHI 1992, pp.533-540, 1992.
5. Fussell, S.R., Setlock, L.D. and Kraut, R.E. 2003. Effects of head-mounted and scene-oriented video systems on remote collaboration on physical tasks, CHI 2003, pp. 513-520. 2003.
6. Kirk, D., Rodden, T. and Fraser, D. S. 2007. Turn it this way: grounding collaborative action with remote gestures. CHI 2007, pp.1039-1048, 2007.
7. Steve M., Telepointer: Hands-Free Completely Self Contained Wearable Visual Augmented Reality without Headwear and without any Infrastructural Reliance. In Proceedings of ISWC '00, 177-178. 2000.
8. Bauer, M., Kortuem, G. Segall, Z. "Where Are You Pointing At?" A Study of Remote Collaboration in a Wearable Videoconference System. ISWC 1999, pp.151-.158, 1999.
9. Kurata, T., Sakata, N., Kourogi, M., Kuzuoka, H. and Billinghurst, M. 2004. Remote Collaboration using a Shoulder-Worn Active Camera/Laser. In Proc. ISWC 2004. 62-69.
10. Gauglitz, S., Lee, C., Turk M. and Hllerer, T. Integrating the physical environment into mobile remote collaboration. In Proc. MobileHCI '12 . ACM Press (2012) 241-250.
11. Ou, J., Fussell, S.R., Chen, X., Setlock, L.D. and Yang, J. Gestural communication over video stream: supporting multimodal interaction for remote collaborative physical tasks. In Proc. ICMI '03. ACM Press (2003), 242-249.
12. Snavely, N., Seitz, S. and Szeliski, R. Photo tourism: exploring photo collections in 3D. ACM Transactions on Graphics (TOG) 2006. 83-846
13. SmartAR (http://www.sony.net/SonyInfo/News/Press/201105/11-058E/index.html)
14. LUMUS (http://www.lumus-optical.com/)
15. Izadi, S., Kim, D., Hilliges, O., Molyneaux, D., Newcombe, R., Kohli, P., Shotton, J., Hodges, S., Freeman, D., Davison, A. and Fitzgibbon, A. KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. In Proc. UIST '11. ACM Press (2011) 559-568.
16. L. Cheng and J. Robinson, Dealing with speed and robustness issues for video-based registration on a wearable computing platform. In Proc. ISWC98, pages 84.91, 1998.
17. Baudisch, P. and Rosenholtz , R. Halo: a technique for visualizing off-screen objects. In Proc. CHI '03. ACM Press (2013) 481-488.
18. Benson, Alan J. (2002). "Motion Sickness". In Kent B. Pandoff and Robert E. Burr. Medical Aspects of Harsh Environments 2. Washington, D.C.: Borden Institute. pp. 1048-1083. Retrieved 4 Dec 2012.
19. Goldberg, K., Song, D., Khor, Y., Pescovitz, D., Levandowski, A., Himmelstein, J., Shih, J., Ho, A., Paulos, E. and Donath, J. Collaborative Online Teleoperation with Spatial Dynamic Voting and a Human "Tele-Actor" In Proc. ICRA '02. 1179-1184.
20. Tamaki, E., Miyaki, T. and Rekimoto, J. PossessedHand: techniques for controlling human hands using electrical muscles stimuli. In Proc. CHI2011, ACM Press (2011) 543-552.
21. Leap Motion (https://www.leapmotion.com/)
22. William Gibson, Neuromancer, Ace Science Fiction, 1984.
23. GoPro Official Website, http://gopro.com/
24. Leonard, J.J. and Durrant-whyte, H.F., Simultaneous map building and localization for an autonomous mobile robot. Proceedings IROS'91. IEEE/RSJ International Workshop, pp.1442-1447, 1991.
25. Rekimoto, J. "Traxion: A Tactile Interaction Device with Virtual Force Sensation, ACM UIST 2013, pp.427-432, 2013.
26. Gerhard, R., Ethan, E., and Tom, W., D. Semi-automatic Annotations in Unknown Environments. In Proceeding of ISMAR '2007, pp. 67-70, 2007.