2010 IEEE International Symposium on Multimedia

Bridging the Gap between Virtual and Real World by Bringing an Interpersonal Haptic Communication System in Second Life

Abu Saleh Md Mahfujur Rahman, SK Alamgir Hossain, Abdulmotaleb El Saddik
Multimedia Communications Research Laboratory
University of Ottawa
Ottawa,Canada
{*kafi, skahossain, abed*}*@mcrlab.uottawa.ca*

***Abstract*—The sense of touch has much importance in technology-mediated human emotion communication and interaction. Many researchers around the world are aiming to leverage the sense of touch in the communication medium between multiuser 3D virtual world and real environment. Driven by the motivation, we explored the possibilities of integrating haptic interactions with Linden Lab's multiuser online virtual world, Second Life. We enhanced the open source Second Life viewer client in order to facilitate the communications of emotional feedbacks such as human touch, encouraging pat and comforting hug to the participating users through real-world haptic stimulation. These emotional feedbacks that are fundamental to physical and emotional development in turn can enhance the users interactive and immersive experiences with the virtual social communities in the Second Life. In this paper, we describe the development of a prototype that realizes the aforementioned virtual-real communication through a haptic-jacket system. Some of the potential applications of the proposed approach includes distant lover's communication, remote child caring, and stress recovery.**

***Keywords*-Tactile feedback; interaction design; sense of touch; haptics; interpersonal communication; virtual world; Second Life;**

I. Introduction

The use of sense of touch has much significance in inter-human communication. Social emotional touches in the form of handshake, encouraging pat, hug, tickle etc. physical contacts are fundamental to mental and psychological development and hence their applications in interpersonal communication systems have attracted attention of many researchers around the world [1]. In order to convey the emotional feedbacks haptic is given high regards in live communication [2][3] and in immersive virtual environments [4]. The haptic-based nonverbal modality can enhance social interactivity and emotional immersive experiences in a 3D multiuser virtual world that presents a 3D realistic environment, where people can enroll in an online virtual community [5]. One of the most popular and rapidly spreading examples of such systems is Linden Lab's Second Life (SL) [6]. In Second Life, similar to ActiveWorlds [7] and Sims [8], once connected the users can view their avatars in a computer simulated 3D environment and they can participate in realtime in task-based games, play animation, communicate with other avatars through instant messaging and voice. The social communication aspect of Second Life is hugely popular and counts millions of users [6]. Moreover, its open source viewer [9] provides a unique opportunity to extend it further and equip it with other interaction modality such as haptic.

In this pursuit, we explored the possibilities of integrating haptic interactions in Second Life. We enhanced the open source Second Life viewer client and introduced a communication channel that provides physical and emotional intimacy to the remote users. In the prototype system a user can take advantage of touch, tickle, and hug type haptic commands in order to interact with the participating users by using visual, audio or text based interface modalities. A haptic stimulation of touch and other touch based interactions are rendered to the remote user on the contacted skin through our previously developed haptic jacket system [10] that is composed of an array of vibrotactile actuators. This paper illustrates a preliminary prototype exploring the aforesaid haptic interactions between virtual and real environment actors. An overview of the system components is shown in Figure 1.

Our contribution in this paper is three-fold. First, in order to bridge the gap between virtual and real, we present a Second Life viewer add-on, where we provide haptic interaction opportunity between the real users and their respective virtual avatars through a 3D graphical user interface (GUI) using speech, mouse, text and gesture based interaction modalities. Second, we introduce touch, hug and tickle haptic features for the Second Life users through chat and GUI interactions. Third, we incorporate virtual annotation mechanism for the Second Life avatar so that user dependent interpersonal haptic and animation interactions become possible.

The remainder of this paper is organized as the following. In this paper at first we present a related study in Section II. In Section III we illustrate the various components of our proposed system that facilitates the Second Life based interpersonal communication and provide a general overview of the system and its access mechanisms. Further in Section IV we describe the implementation issues and development

978-0-7695-4217-1/10 $26.00 © 2010 IEEE
DOI 10.1109/ISM.2010.40

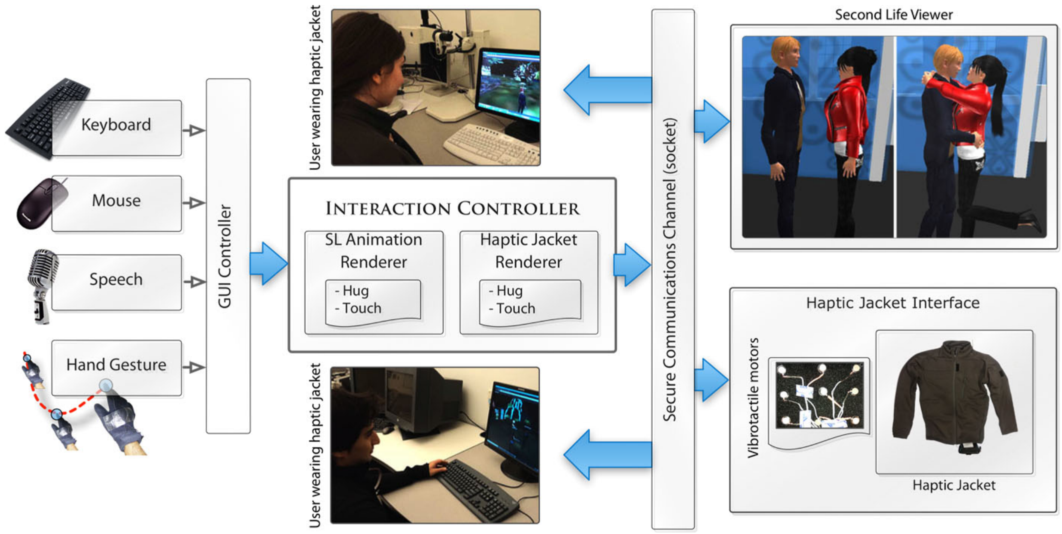

Figure 1. An overview of the Second Life inter personal Haptic communication system.

challenges of different modules. Also, we present response time comparisons for different haptic and animation data, accuracy of different interaction modalities and user study of the system. At the end we provide conclusion of the paper in Section V and state some possible future work directions.

II. Related Study

Thayer [11] states that touch as a opposed to other forms of human-to-human communication will be more trusted by the person touched as a genuine reflection of emotion. Especially in remote communication, touch is a unique channel in affect conveyance as the relation of touch to affect is immediate [3]. Haptic jacket [10] based rendering of touch has been incorporated previously into a conventional teleconferencing system to provide haptic interactions to the remote users. This approach uses marker-tracking technique to specify touchable parts of the user's body. The markers are further tracked using a dedicated camera. The system employs an expensive 3D camera in order to automatically create 3D touchable surface of the user.

In instant messaging, Rovers and van Essen [12] have provided a detailed study on the usage of hapticons that essentially are vibrotactile icons representing smileys. They incorporated six vibrotactile patterns that represent six associated smileys. These smileys could be triggered using mouse or keyboard based interactions. In a 3D virtual environment we attempted to employ similar methodology. In our attempt the smileys are replaced by different type of avatar animations such as hug, tickle and touch that resembles the emotions that the user is trying to communicate to the other.

O'Brien et. al. [13] investigated on an approach relating to intimate communication for couples. In this approach, a person could virtually hold hands by using their proposed probe to share tactile experiences with his or her partners hand. They placed a small microchip inside the ball and when the ball is squeezed by a user the system sends vibrotactile data to the other ball that his or her partner is holding. For couples in long-distance relationships, these communication technologies may be a primary means of exchanging emotions [14]. In distance communication Second Life presents a multi-user communication framework that presents opportunities for interactions that connect people through a shared sense of place. Haptic based input modes have been investigated in Second Life in order to assist the blind people to be able to interact with the Second Life world [4]. The authors have implemented two new input modes that exploit the force feedback capabilities of haptic devices and allow the visually impaired users to navigate and explore the virtual environment. Recently, in Second Life, Tsetserukou et. al. [5] have attempted to analyze the text conversations in Second Life. This system provides emotional haptic feedbacks to the users by using a specially designed wearable hardware. While the different hardware designs for HaptiTickler, HaptiHug, HaptiButterfly and HaptiHeart are commendable, this approach does not seem to consider visual or pointer based graphical interactions in the 3D environment other than the text based conversation system. For example, it seems impossible to interact with specific parts of the virtual 3D avatar that can be used to generate haptic touch stimulation in that respective body part of the real user. Moreover, gesture [15] and audio based interaction modalities can enhance the navigation and interaction experiences of the user in a 3D virtual gaming environment [16]. Hence, in our proposed haptic communication framework we incorporated a flexible GUI based multimodal interaction mechanism in order to provide more natural, easy and accessible interactions in

Second Life.

III. Proposed Framework

In this section we present various components of the system and provide their general description. In the beginning we illustrate the Second Life Viewer and its customization in Section III-A. Later in Section III-B we describe the haptic jacket system and its interfaces to the system. Further in Section III-C, we present the avatar annotation, access control mechanism, animation, and rendering techniques. Lastly, in III-D we present the four interaction modalities and provide their descriptions. The components of the system are depicted in Figure 1 as a block diagram.

A. Second Life Viewer

In this section we discuss the Second Life viewer that provides the mechanism to handle different haptic responses and avatar animation sequences. In order to communicate with the core part of the second life viewer we developed a Second Life event controller as an add-on to the viewer. This coupling architecture provides the option to listen to the communication channel of the Second Life system and incorporate haptic interactions without affecting the functionality of the core system. In the event controller a Nearby Interaction Event Handler module capture the events that are generated from the message transmission module. The event controller performs actions by using text based messaging protocol. A message contains event trigger data, animation data or simple communication data. The Message Transmission Module captures all the messages that are generated in the core section of the Second Life. When through the avatar the user issues events in the 3D environment, e.g., a collision event with other avatars or objects, the message transmission module captures those events and transfers the event messages to the nearby interaction event handler for processing. The event handler module determines the particular event handling routine for a specific event and then packs the event-handling message with the handling routine. The handler then sends the packet to the interaction event decoder. Message transmission module also receives animation data from the animation parcel manager and generates animation sequence for the avatars in the 3D virtual world.

B. Haptic Jacket Interface

Vibrotactile actuators communicate sound waves and create funneling illusion when it comes into the physical contacts with skin. The haptic jacket consists of an array of vibrotactile actuators that are placed in particular portions of the jacket and their patterned vibration can stimulate touch in the user's skin [17]. A series of small actuator motors are placed in a 2D plane in the jacket in a certain manner. An AVR Micro-controller controls the vibration of these actuators. We have configured the Micro-controller so that it processes input commands that are sent from the haptic interaction controller. In order to achieve the input command transmission the haptic interaction controller uses the Bluetooth communication channel. Figure 2 depicts the components of the jacket into more detail.

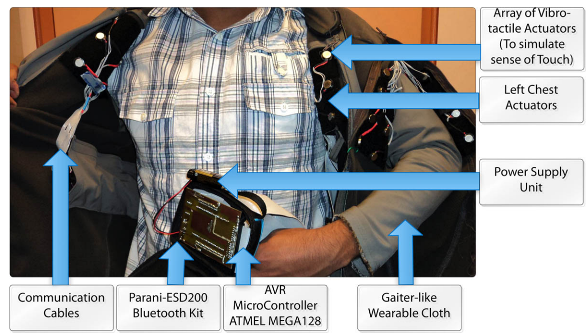

Figure 2. The Haptic jacket.

C. Interaction Controller

In representing the interaction controller, the engine of the system, we will introduce the avatar annotation procedure in Section III-C1. The annotation provides personalized animation and haptic feedback customization options. The way we provide security and authenticity in the avatar-based interactions are described in Section III-C2. We also present the avatar animations and define the associated haptic signal patterns in Section III-C3 and Section III-C4 respectively.

1) Avatar Annotation: In our system we annotated visible body parts of the avatar in Second Life and specified the corresponding physical haptic actuators to render the haptic feedback. For each haptic signal we also annotated the avatar animation. Figure 3 depicts the geometric based avatar annotation scheme. We attached LSL scripts in each of the annotated parts of the avatar that contain the haptic commands as well as the identification number of the animation sequences. For example, we annotated the 3D male avatar's left arm and specified particular vibrotactile actuator stimulation for it. Further, we specified the interacting animations for both the participating male and female virtual avatars. Afterwards, when the user representing the female avatar issues a GUI interaction command to the male avatar arm then the annotated haptic stimulation is rendered at the real male user's arm through the haptic jacket.

For intimate interactions such as a hug, we employed group based annotation scheme. As evident, hugging with parents is different to that with a friend. Hence, separate animation and haptic rendering are required for different hugs. We created groups and incorporated group based annotation of the 3D avatar. For each group we created different avatar animation and haptic rendering options. By using the script based dialog interface any interacting contacts were then assigned to a group (default is formal).

We provided four different groups namely family, friend, lovers, and formal. This group based haptic interaction in Second Life further assisted the user to personalize his/her experience.

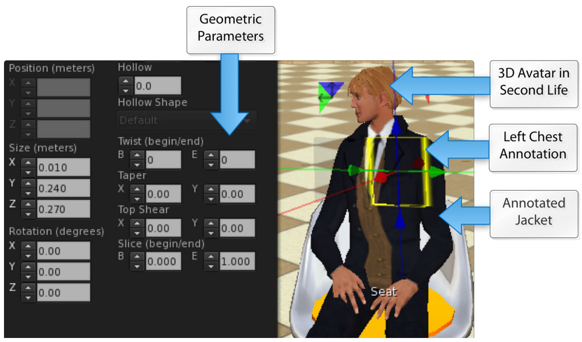

Figure 3. Avatar annotation scheme.

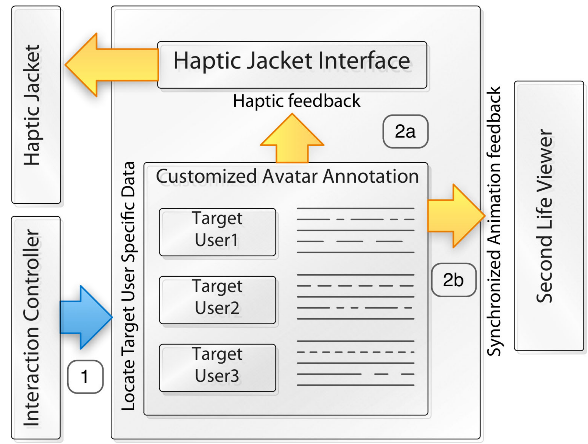

Figure 4. User dependent Haptic interaction access design.

2) Access Control Scheme: In our prototype application we incorporated user profile specific access control mechanism in order to provide the participating users the means of authenticating and personalizing their interactions. For example if user A issues a hug command to user B then the animation and haptic rendering take place only if user B acknowledges the permission. A permission window is shown at user B's Second Life viewer for this purpose, where the interaction could be accepted or rejected. We used Second Life message notification and graphical user interface to display the permission window in which the user is already adapted. In Second Life each user is associated with a string based identification number. Message originated from a user's computer bears that identification number as a preamble to that message. Hence, in order to provide access control we compared the identification number with the list of contacts of the user and decided accordingly.

In order to deliver user specific haptic feedbacks to the user we used the group annotation. In any haptic interaction, the originator user information is mapped to obtain the group of the user. This phenomenon is depicted in Figure 4. In this approach the haptic renderer uses the group specific avatar animation and haptic rendering data in order to deliver customized interactions to the users.

3) Avatar Animation: Animation helps the user to express the emotion in an intuitive manner (if compared to Instant Messaging). The animation rendering depicting a hug for example communicates the user's emotion directly when rendered with the hug haptic feedback. Second Life animation is a BVH (Biovision Hierarchy) file, which contains text data that describes each figure part's rotation and position along a time line. We controlled the avatar position or movement by triggering a message to animation parcel manager, which then executes the BVH animation file for that animation. We created these animation files for hug, touch, and tickle animations by using MilkShape 3D version 1.8.5. Both the participating male and female virtual avatars plays out the defined animation sequences in the Second Life viewer using their respective animation files. Empirically we created four different hug animations for the four groups of users in order to verify our concept that group dependent animations were taking place in the Second Life viewer. In order to control the hug animations we used the SL scripting language of Second Life. In order to start the animation, user A issues a hug, touch or tickle command, the participating second user B consents to it. Afterwards, in hug animation the two virtual avatars of user A and user B (who must be at the same virtual location in Second Life) comes closer by walking and holds each other closely. Similarly, for the touch animation, user A waves a hand emulating a touch activity and user B smiles or frowns (if done repeatedly) indicating that a touch has taken place. If user A touches stomach or neck of the avatar of user B then tickle animation for user B takes place. In tickle animation the avatar of user B moves awkwardly and laughs.

4) Haptic Renderer: The haptic jacket provides the funneling illusion based touch haptic feedback. We leveraged the touch feature to create hug and tickle based haptic feedbacks. We made careful observation about the real life hug and noticed that when two people hug each other both feel a gradual touch feeling in some specific body parts. In a formal hug a user receives touch feedbacks at the chest area and at the back shoulder area. Similarly, during our observations we noticed that in a tickle most users react to the random touch at the stomach area, at the underarm area and sometimes at the neck area. Using these empirical parameters we constructed touch, hug and tickle haptic feedbacks as the following:

- According to the virtual annotation the haptic touch sensation is delivered by incorporating the funneling illusion into the haptich jacket to stimulate real touch at the real user. When one person touches another person then both the participating users receives touch feelings.

- In order to create hug type haptic feedbacks for the participating users we systematically increased the jacket's *leftChest*, *rightChest*, *neck*, *leftBackShoulder*, and *rightBackShoulder* motors intensity levels to produce the funneling illusion. The systematic control of the actuator intensity levels creates the touch effect in those areas and offers a hug type haptic stimulation. The lover type hug is different to that of the formal hug. In addition to the areas defined above we decided to add haptic touch stimulation in the stomach area to emulate the joy emotion [18] [5]. Hence, by following the laws of funneling illusion we activated the arrays of vibration motors attached to the abdomen area of a person.
- As described earlier, following our empirical study the tickle haptic feedback is evoked by incorporating random and unpredictable touch at the stomach area, at the underarm area and at the neck area provided that a GUI interaction at those virtual body places were performed.

D. GUI Controller

The Interaction Controller works as a core service and takes action according to the user inputs from the GUI Controller. The GUI controller enables the usage of keyboard, mouse, speech and gesture based inputs from the user. For example, a user representing a female avatar can point her mouse on a male avatar and produce a click event using the mouse. The GUI controller detects if the annotated body parts of the male avatar has received any GUI commands and sends the avatar body ID and type of action performed to the Interaction Controller. In our prototype the hug command is issued by using the speech, keyboard and gesture based interaction inputs. The GUI commands that were used in the various interaction inputs are discussed in the following:

- Keyboard: While processing the keyboard (text) based inputs from the user the controller analyzes the text messages sent to the jacket owner (receiver). The text message based commands have certain preamble before the commands. Therefore, the interaction controller easily distinguishes the haptic commands that are issued based on the text inputs. The text command forms are HUG $username$, TOUCH $<username\ bodyparts>$ and TICKLE $<username\ bodyparts>$, where $bodyparts$ = $\{$*leftChest, rightChest, stomach, leftShoulder, rightShoulder, leftBackShoulder, leftRightShoulder, leftArm, rightArm, neck*$\}$.
- Mouse: It is extremely flexible to provide touch and tickle commands using a mouse. For each mouse click at the annotated body parts a touch command is issued. When the mouse click happens on the stomach, and neck area of the virtual avatar a tickle command is captured. In order to provide hug command the user clicks a GUI button on the screen and the nearest user is issued a hug command automatically.
- Speech: Similar to our previous speech based interaction methodology [16] in virtual environment we processed the speech based haptic commands from the user. The touch and tickle input commands are similar to that of keyboard interaction, where the user speaks out the type of interaction (touch, tickle) followed by body part names. In order to issue hug input command, the user simply speaks out hug and the nearest user is issued a hug command. User name recognition was not attempted in our approach.
- Gesture: In our previous work we have proposed a novel motion path based gesture interaction system[15]. This system allows the user to define a drawing symbol that can be associated with particular command. We tailored the motion path based gesture interaction approach by introducing three main drawing symbols e.g. h, T, k representing hug, touch and tickle commands respectively. For each body parts we associated the following gesture commands, $bodyparts$ = $\{$*leftChest(L,C), rightChest (*Γ*,C), stomach (S), leftShoulder(L,S), rightShoulder(*Γ*,S), leftBackShoulder (L,b) , leftRightShoulder (*Γ*,b), leftArm (L,m) , rightArm (*Γ*,m), neck(n)*$\}$. The gesture drawing symbols were chosen based on their selection accuracy. For example the gesture recognition rate for Γ is higher than R and since the selection.

IV. Implementation and Results

In Section IV-A we discuss the development details of various modules of our system. Results pertaining to the performance and usage of the system are discussed further in Section IV-B.

A. Development of Different Modules

In this section we present the details of the implementation issues of different modules of our proposed system. We incorporated Microsoft Visual Studio 2005 IDE to develop our system and the primary language used was Visual C++. We adopted Microsoft Foundation Class (MFC) library and asynchronous socket programming scheme to create a socket based secure communication channel. In order to implement voice based interaction we used Microsoft Speech SDK (SAPI version 5.1) [19]. We now briefly illustrate the development of different modules, which are Second Life Controller, Interaction Event Decoder, Permission Manager, and Haptic Renderer. These modules and their inter message communications are depicted in Figure 5.

1) Second Life Controller Module: In order to develop the Second Life add-on, we locally build the Second Life open source viewer Snowglobe [9] version 1.3.2 by using the latest version of CMake (version 2.8.1) [20]. Second Life message transmission module is responsible for dispatching

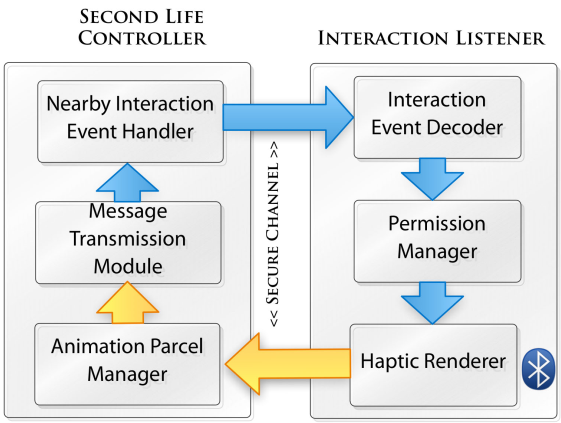

Figure 5. Second Life and Haptic communication system block diagram.

all the messages to handle virtual environment events. All the messages are in XML (Extensible Markup Language) format with detail avatar and virtual environment related data. All the event type messages are filtered by nearby interaction event handler.

2) Interaction Event Decoder Module: Interaction Event Decoder is a component in the interaction listener. It receives all the output messages from the Nearby Interaction Event Handler in an encrypted XML [21] format. The primary responsibilities of the module are to decrypt those received messages and transmit them further to the communication channel towards the Permission Manager.

```
state_entry()
{
    llRequestPermissions(llGetOwner(), PERMISSION_TRIGGER_ANIMATION);
    llSetTimerEvent(1.0);
    llListen(0, "", llGetOwner(), "");
}

touch_start(integer total_number)
{
    // Obtain the calculated body part from the GUI interaction
    llGetAnnotationSelection(0, &vBody_trg, (string)llDetectedKey(0));

    // User specific haptic feedback and animation
    dances=SL_GetUserAnim(SL_GetUser(this), vBody_trg);
    hFeed= SL_GetUserhFeed(SL_GetUser(this), 0, vBody_trg);

    // Render animation and haptic feedback
    llStartAnimation(llList2String(dances, WHICH));
    SLStarthFeed(llList2String(hFeed, WHICH));

    // Animation calibration and user interaction throttling
    if (WHICH++ >= TOTAL)
    {
        WHICH = 0;
    }
}
```

Figure 6. Linden Script to control the user interaction.

3) Permission Manager Module: The Permission Manager (PM) looks up the user dependent access control scheme and produces appropriate permission dialogues in Second Life viewer. The PM issues these dialogues by using SL script and receives appropriate permission parameters. Figure 7 shows the code snippet that is used to control user dependent animation and the vibrotactile motors in the haptic jacket. As shown, before commencing avatar or haptic rendering functions, we call *llRequestPermissions(key AvatarID, integer perm)* function. The function takes two parameters; the first parameter is the user's Avatar ID who requested an event. The second parameter *PERMISSION_TRIGGER_ANIMATION* is a permission type for that event.

4) Haptic Renderer Module: The Haptic Renderer (HR) operates the haptic jacket and notifies the Animation Parcel Manager for synchronized animation feedback. In order to control the jacket motors, it parses an XML file containing haptic patterns and sends a message to the micro controller unit of the jacket accordingly. Portion of the xml file is shown in Figure 7. In our implementation the actuator motors have a total of 16 intensity levels from 0 to 15. Where, 0 means no vibration and 15 indicates the maximum vibration level. To repeat the vibration patterns we set the value for the *numberOfRepetation* attribute.

```xml
<?xml version="1.0" encoding="utf-8"?>
<InteractionRules>
  <hug userType="Friends" hapticFeedback="yes">
    <animationModules>
      <module>
        <animationName>hug1</animationName>
        <animationDuration>MEDIUM</animationDuration>
      </module>
    </animationModules>
    <tactileModules>
      <module name="leftChest">
        <highestIntensity>10</highestIntensity>
        <lowestIntensity>0</lowestIntensity>
        <vibrationType>GRADUAL_INCREASE</vibrationType>
        <intervalTime>500</intervalTime>
        <numberOfRepetation>2</numberOfRepetation>
      </module>
      <module ...
      <module ...
      <module ...
    </tactileModules>
  </hug>
  <touch ...
  <tickle ...
```

Figure 7. Interaction Rules.

B. Results

In this section first we discuss the different parameters that affect the transmission time of the haptic and animation data in our system in Section IV-B1. In Section IV-B2, we illustrate a detail analysis of the impact of different interaction modalities. Further in Section IV-B3 we describe the usability study setup and its analysis.

1) Response Time: We calculate the haptic taransmission time from the sender machine to the receiver jacket by using Equation 1. Where user's average interaction time to interact with the Second Life (SL) viewer is I unit, average data transmission rate via the server is Π, n is the message size and the time for sending data from the receiver machine to the jacket actuators is β_1 unit.

$$R \;=\; I + \frac{n}{\Pi} + \beta_1 \tag{1}$$

After generating a haptic interaction the system approximately requires $R = (3775 + 270 + 344)\, ms$ to complete the transmission. Here, in our experiments the average of the interaction time is $3775ms$, network overhead is $270ms$ and β_1 is $344ms$. The haptic acknowledgement from the receiver machine to the sender jacket is represented by Equation 3. Here, $\frac{n}{\Omega}$ is the average time for transmitting *n* byte feedback message from the receiver machine to the sender machine. We assumed that the transmitted message and its acknowledgment were of the same length.

$$\begin{aligned} S &= I + \frac{n}{\Pi} + \beta_2 + \frac{n}{\Omega} \quad (2) \\ &= R + \frac{n}{\Omega} + (\beta_2 - \beta_1) \\ &= R + \frac{n}{\Omega}, \qquad (\beta_2 - \beta_1) \simeq 0 \end{aligned}$$

On average, the time S is higher than R by $\frac{n}{\Omega}$ unit, which is the network transmission delay. Typically this value is less than 200 ms and hence the added delay does not necessarily affect the interaction experience of the participating users. Furthermore, the haptic rendering and animation rendering are synchronized locally in respective users' machines. Hence the delay is not apparent to the local user during the communications if the system is not paired with external communication channels such as voice.

Table I shows how fast our system delivers haptic and animation rendering. From the result we see that hug interaction needs more time than other interactions as for hug rendering the system is required to process more data than the others.

Table I
HAPTIC INTERACTION AND THEIR RESPECTIVE HAPTIC AND ANIMATION RENDERING TIME

Interaction	Haptic Rendering	Animation Rendering
Hug	7277 ms	7407 ms
Tickle	4123 ms	4899 ms
Touch	2261 ms	3055 ms

2) Analysis of Different Interaction Modalities: A comparison of different interaction modalities used and their suitability for each haptic interactions are given in Table II. The two other parameters are average time needed to produce the command and average accuracy, which are also listed. However, not all haptic input commands were convenient to use for each interaction modalities. For the keyboard (text) interaction modality we found that writing body parts names take time and often spelling mistakes impaired the accuracy of the command. Touch and tickle input commands were very easy to issue using the mouse-based modality. However, while issuing hug input command using the mouse, it became difficult to assign the command to a particular user, hence nearest user was selected automatically. Similar problem occurred while using speech and gesture based interaction modalities as it became cumbersome to recognize the user names using either of those two approaches. From the table we see that the percentage of accuracy is highest for mouse-based interaction modality, which is 99.6%. This and its flexibility for usage in pointing and interacting with annotated body parts made it the ideal medium for haptic input command delivery in our system.

Table II
A COMPARISON OF DIFFERENT INTERACTION MODALITY

Modality	Average Time	Accuracy	Suitability
Keyboard	5110 ms	85%	hug, touch, tickle
Mouse	2075 ms	99.6%	touch, tickle, hug
Speech	3790 ms	55%	hug, tickle, touch
Gesture	4125 ms	78.1%	hug, touch, tickle

3) Usability Study: We conducted usability tests to evaluate the user's quality of experience with our proposed system and to measure the suitability of the approach. The usability tests took place at our university laboratory with 16 (sixteen) volunteers. At a time two volunteers were chosen for the experiments. Two personal computers were used and in each computer the enhanced Second Life viewer was installed to provide animation and GUI based interactions. For the haptic signal transmission, Bluetooth was configured at the PC COM port of the respective computers that interfaces with the hardware controller of the jacket.

At a time, the selected two volunteers were told to put on the haptic jackets and requested to use the prototype by participating in certain haptic interaction based tasks. Their activity was monitored throughout the experiment and recorded for analysis. Afterwards, based on their interaction experiences the users filled out a questionnaire where they were requested to provide ratings of their likeliness, familiarity, ease of usage etc of the system. The user responses are shown in Likert Scale [22] in Figure 8. The ratings of the questionnaire were in the range of 1-5 (the higher the rating, the greater is the satisfaction). The average of the responses of the users were calculated in percentage form and measured after the usability tests. Figure 8 shows the user's responses for each given assertions. It is worth mentioning that more than 80% of the users would like to communicate using the enhanced system through haptic and animation interaction if they were available in Second Life. Overall the users were also satisfied with the synchronized animation and haptic rendering responses.

V. CONCLUSION

In this paper we presented a Second Life HugMe prototype system that bridges the gap between virtual and real world events by incorporating interpersonal haptic communication system in Second Life. The developed system works as an add-on and loosely coupled to the Second Life viewer. The haptic and animation data are annotated in the virtual 3D avatar body parts. The 3D avatar and the annotated body parts representing a real user receive inputs when they are interacted through gesture, mouse, speech or text based input modalities and produces emotional feedbacks such as touch,

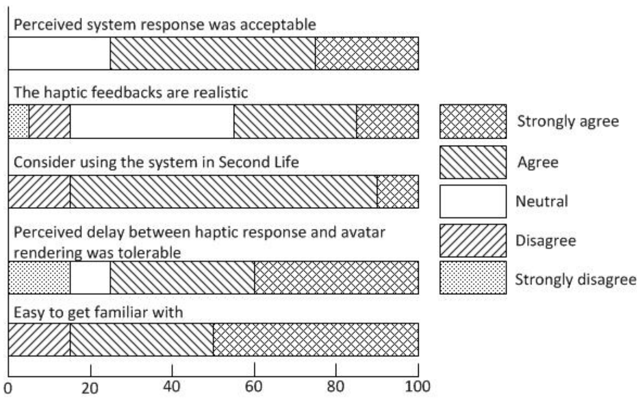

Figure 8. Usability study of the Second Life HugMe system.

tickle and hug to the real user through the haptic jacket. We presented the implementation details of a preliminary prototype exploring the aforesaid haptic interactions in a real-virtual collaborative environment.

References

[1] A. Haans and W. IJsselsteijn, "Mediated social touch: a review of current research and future directions," *Virtual Real.*, vol. 9, no. 2, pp. 149–159, 2006.

[2] S. Brave and A. Dahley, "intouch: a medium for haptic interpersonal communication," in *CHI '97: CHI '97 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 1997, pp. 363–364.

[3] R. Wang and F. Quek, "Touch and talk: contextualizing remote touch for affective interaction," in *TEI '10: Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction*. New York, NY, USA: ACM, 2010, pp. 13–20.

[4] M. de Pascale, S. Mulatto, and D. Prattichizzo, "Bringing haptics to second life for visually impaired people," *Haptics: Perception, Devices and Scenarios*, pp. 896–905, 2008.

[5] D. Tsetserukou, A. Neviarouskaya, H. Prendinger, M. Ishizuka, and S. Tachi, "ifeelim: innovative realtime communication system with rich emotional and haptic channels," in *CHI EA '10: Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2010, pp. 3031–3036.

[6] L. Lab, "Second life," http://en.wikipedia.org/wiki/Second_Life, Tech. Rep., 2010.

[7] Activeworlds, "Activeworlds inc," http://www.activeworlds.com, Tech. Rep., 2010.

[8] Sims, "Electronic arts inc," Sims Online, http://www.ea.com/official/thesims/thesimsonline, Tech. Rep., 2010.

[9] SecondLife, "Second life open source portal," http://wiki.secondlife.com/wiki/Open_Source_Portal, Tech. Rep., Accessed 10 June 2010.

[10] J. Cha, M. Eid, A. Barghout, A. S. M. M. Rahman, and A. El Saddik, "Hugme: synchronous haptic teleconferencing," in *MM '09: Proceedings of the seventeen ACM international conference on Multimedia*. New York, NY, USA: ACM, 2009, pp. 1135–1136.

[11] S. Thayer, *Social Touching. In: Schiff, W., Foulke, E. (eds.) Tactual Perception: A Sourcebook*. Cambridge: Cambridge University Press, 1982.

[12] A. Rovers and H. van Essen, "Him: a framework for haptic instant messaging," in *CHI '04: CHI '04 extended abstracts on Human factors in computing systems*. New York, NY, USA: ACM, 2004, pp. 1313–1316.

[13] S. O'Brien and F. F. Mueller, "Holding hands over a distance: technology probes in an intimate, mobile context," in *OZCHI '06: Proceedings of the 18th Australia conference on Computer-Human Interaction*. New York, NY, USA: ACM, 2006, pp. 293–296.

[14] S. King and J. Forlizzi, "Slow messaging: intimate communication for couples living at a distance," in *DPPI '07: Proceedings of the 2007 conference on Designing pleasurable products and interfaces*. New York, NY, USA: ACM, 2007, pp. 451–454.

[15] A. S. M. M. Rahman, M. A. Hossain, J. Parra, and A. El Saddik, "Motion-path based gesture interaction with smart home services," in *Proceedings of the seventeen ACM international conference on Multimedia*. Beijing, China: ACM, 2009, http://doi.acm.org/10.1145/1631272.1631407.

[16] A. El Saddik, A. S. M. M. Rahman, and M. A. Hossain, "Suitabiulity of searching and representing multimedia learning resources in a 3-d virtual gaming environment," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 9, pp. 1830–1839, 2008.

[17] A. Barghout, J. Cha, A. El Saddik, J. Kammerl, and E. Steinbach, "Spatial resolution of vibrotactile perception on the human forearm when exploiting funneling illusion," in *Haptic Audio visual Environments and Games, 2009. HAVE 2009. IEEE International Workshop on*, 7-8 2009, pp. 19 –23.

[18] C. DiSalvo, F. Gemperle, J. Forlizzi, and E. Montgomery, "The hug: an exploration of robotic form for intimate communication." in *RO-MAN 2003*, I. Press, Ed., 2003, pp. 403–408.

[19] Wiki-Link, "Microsoft speech api, http://en.wikipedia.org/wiki/microsoft_speech_api," Accessed, 11 July 2010.

[20] CMake, "Cross platform build system," http://www.cmake.org/cmake/resources/software.html, Tech. Rep., 2010.

[21] Wikipedia, "Xml-encryption," http://en.wikipedia.org/wiki/XML_Encryption, Tech. Rep., Accessed, 10 July 2010.

[22] Wikipedia-Likert-Scale, "Likert scale," http://en.wikipedia.org/wiki/Likert_scale, Tech. Rep., 2010.