# Remote Context Monitoring of Actions and Behaviors in a Location through 3D Visualization in Real-Time

John Conomikes[1], Zachary Pacheco[1], Salvador Barrera[2], Juan Antonio Cantu[2], Lucy Beatriz Gomez[2], Christian de los Reyes[2], Juan Manuel Mendez-Villarreal[2] Takeo Shime[3], Yuki Kamiya[3], Hedeki Kawai[3], Kazuo Kunieda[3], and Keiji Yamada[3]

[1] Carnegie Mellon University, Entertainment Technology Center (ETC), 800 Technology Drive, Pittsburgh, PA, 15219, USA
[2] Universidad de Monterrey (UDEM), Engineering and Technology Division, Av. Morones Prieto 4500 Pte. San Pedro Garza Garcia, C.P. 66238, N.L. Mexico
[3] NEC C&C Innovation Research Laboratories, 8916-47, Takayama-Cho, Ikoma, Nara 630-0101, Japan
{JohnConomikes,zakpacheco}@gmail.com, {sbarrea1,jcantaya,lgomez20,xpiotiav,jmndezvi}@udem.net, t-shime@ce.jp.nec.com, y-kamiya@fn.jp.nec.com, h-kawai@ab.jp.nec.com, k-kunieda@ak.jp.nec.com, kg-yamada@cp.jp.nec.com

**Abstract.** The foal of this [project is to present huge amounts of data, not parse-able by a single person and present it in an interactive 3D recreation of the events that the sensors detected using a 3D rendering engine known as Panda3D. "Remote Context Monitoring of Actions and Behavior in a Location Through the Usage of 3D Visualization in Real-time" is a software applications designed to read large amounts of data from a database and use that data to recreate the context that the events occurred to improve understanding of the data.

**Keywords:** 3D, Visualization, Remote, Monitoring, Panda3D, Real-Time.

## 1 Introduction

This prototype is the result of a long project development made at the Entertainment Technology Center where work was done in conjunction with NEC and the Universidad de Monterrey.

While there is a lot of work in this field one of the unique angles of this project is the type of data is designed to build the recreation from. This data is from NEC's LifeLog system which tracks a wide variety of detailed information on what each employee in the monitored space does daily on a second to second basis.

Additionally, the data can be viewed from anywhere in the world, not just the monitored laboratory.

R. Shumaker (Ed.): Virtual and Mixed Reality, Part I, HCII 2011, LNCS 6773, pp. 40–44, 2011.

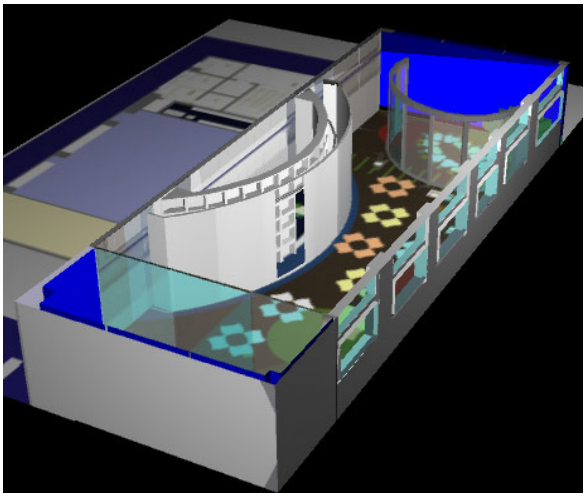

**Fig. 1.** Initial 3D shaded model for the Southern laboratory

## 2 Methodology

One of the requirements for this project is the ability to view the current state of the office, i.e. keeping up with the sensor data in real-time.

Due to the large amounts of data that must be parsed every frame a rolling parsing system had to be implemented where only a portion of the data is parsed and updated each frame rather than all of it in a single frame per second. This is done because the number of frames per second must be kept above 20 in order to maintain a smooth appearance. This gives us only 50 ms of parsing time, minus the overhead of rendering the 3D environment.

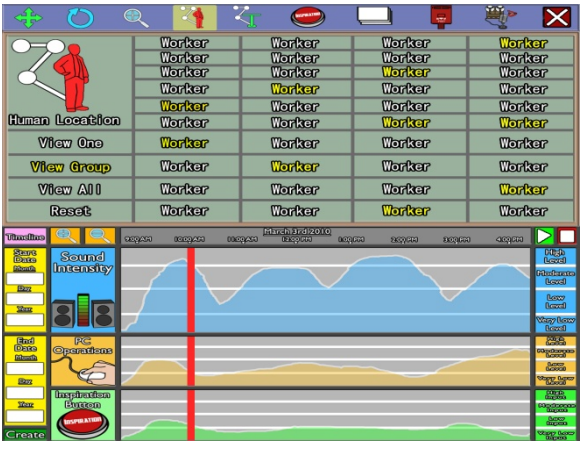


**Fig. 2.** Initial UI design

As the sensors only poll data at most once per second, this system allows us to keep the data real-time without sacrificing frame rate. Originally it was thought to use threading to help alleviate this problem, however the 3D rendering engine used (Panda3D) has very limited inherent support for threading so this was not possible.

Another problem that was tackled was that of a user interface, as the people using this tool may not be high end computer users and there is a large amount of data available to analyze.

We went over a large number of different designs (see Figure 2 above for an example of one of the previous user interface designs) before settling on this latest one which combines ease of use (similar to Office 2007[1] style tabbed buttons) while still allowing the user a large amount of freedom to show and hide data as needed. See Figure 3 below for the final user interface design of the software.

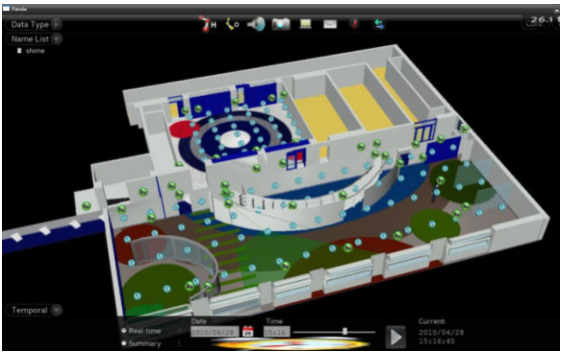


**Fig. 3.** Final UI design

## 3 System Architecture

Our entire system is built on NEC's LifeLog system which is responsible for gathering the large amount of data that is needed for the software to operate. See Figure 4 below for a view of the ceiling with installed sensors.

**Fig. 4.** Ceiling of the South Laboratory with installed sensors

Employee location is detected through the use of IR emitters on employees and receivers mounted on the ceiling, though approximately 25% of all location data is "8022" which is the code for a person who is not detected by any IR receiver on the premises.

Ambient sound level data is collected by over 90 microphones installed in the ceiling. There are also over 30 cameras (like the one shown in Figure 5 below) in place on the ceiling to provide up to 10 images per second.

**Fig. 5.** Close up of one of the many cameras installed in the ceiling

All E-mails send to or from monitored employees are also stored, though addressees that are not monitored are stored only as “Company Employee" or "Recipient Outside Company".

Additionally, extensive information is pulled from the computer operations of each monitored employee. Statistics such as key presses, mouse clicks and mouse movements in the past second. Further, they track the currently active process running on the computer and the most recently accessed file. Even all of the currently running processes in the background. Finally they log all of the employee's internet access, though this last piece of information can be disabled by the employee. Finally, each employee has a wireless button that they carry with them that records when it was pressed and if pressed for more than one second, it also reports the duration of the press.

Also, while not related to people, 16 RFID readers are used to track the location of resources (e.g. books, laptops) which have RFID tags on them, as they move around the office. It also tracks which employee is using each particular resource.

The flow of information is quite simple, the LifeLog system polls the sensors for their latest information. It then takes this information, timestamps it and outputs it to a simplified YAML[2] format and stores this information on a server. Out program then connects to the server and requests the files required to view the time the user wishes to view, loads the needed information into memory in python data structures and displays the recreated events to the user.

Due to security restrictions at NEC, the data is only accessible locally or through a Virtual Private Network (VPN) connection. However, since the only remote action that is being performed with the software is reading data from the server, with less strict security measures, the software can function anywhere without the need for any special access permissions.

## 4 Experimental Results

In testing the software it was found that starting the software up takes approximately one minute per hour of data the user wishes to view. This is because the user needs to be able to jump around to any point in the data and the only way this could be done seamlessly while playing the data is to load all needed data up front. However, after this load time, the user can easy jump to any point in time for the loaded data, in addition to being able to view the most recent data. This load time could also be reduced by having direct, local access to the server or lengthened by a slow internet connection.

## 5 Comments and Conclusion

While the system does use a large concentration of sensors in a small area and is generally very invasive, it does mean there are many promising opportunities for future research to improve on both the technology and software. While not ready for industry yet, with the inclusion of other research as well further improvement of the current software this seems to be a promising technology and may prove to be the next big step in combining multiple different information gathering technologies.

## References

[1] Ebara, Y., Watashiba, Y., Koyamada, K., Sakai, K., Doi, A.: Remote Visualization Using Resource Monitoring Technique for Volume Rendering of Large Datasets. In: 2004 Symposium on Applications and the Internet (SAINT 2004), p. 309 (2004)

[2] Hibbard, B.: Visad: connecting people to computations and people to people. SIGGRAPH Computer Graphics 32(3), 10–12 (1998)