Reality over Web: Pervasive Computing Meets the Web

Muthucumaru Maheswaran
School of Computer Science, McGill University, Montreal, QC H3A 0E9, Canada

Devarun Bhattacharya
School of Computer Science, McGill University, Montreal, QC H3A 0E9, Canada

Follow this and additional works at: https://tsinghuauniversitypress.researchcommons.org/tsinghua-science-and-technology

Part of the Computer Sciences Commons, and the Electrical and Computer Engineering Commons

Recommended Citation
Reality over Web: Pervasive Computing Meets the Web

Muthucumaru Maheswaran* and Devarun Bhattacharya

Abstract: Reality over Web (RoW) is a novel concept, where a window on the web corresponds to a window onto a real space. Once the correspondence is established, users should be able to interact or manipulate the objects or people in the real space through the web window. In this paper, we introduce the RoW concept and highlight the principles that govern its design and implementation. A system architecture for realizing the RoW concept is described along with a proof-of-concept prototype that implements portions of the RoW concept. One essential part of an RoW implementation is accurate locationing of objects and people in a video frame. The locationing problem becomes particularly challenging because we want to reuse existing infrastructure as much as possible. We developed a high-frequency sound-based locationing scheme and implemented it on the prototype. The results from initial experiments performed on the locationing scheme are reported here.

Key words: Internet-of-things; locationing; pervasive computing; telepresence; web; web-of-things

1 Introduction

After remaining an academic curiosity for more than a decade, pervasive computing is gradually getting adopted in the real world[1]. In accordance with the pervasive computing vision[2-3], computing is becoming invisible; yet, it is present in many objects we interact with on a day-to-day basis. For example, thermostats, door locks, and light bulbs are getting smart by the infusion of computing capabilities. Availability of cheap low-power microprocessors and the market pressure to make “smart” gadgets is further accelerating the pace of infusion of computing into common objects.

Another important factor that is pushing us towards a smart world is the emergence of the smartphone as a preeminent personal gadget[4]. The smartphone is beginning to play a role as a hub that captures several behavioral traits of a person. Many smartphones in conjunction with clouds[5] run applications for analyzing the walking, messaging, and social-interaction patterns of people in addition to storing their user preferences. By connecting to a smartphone holding such person-centric information, other devices can offer personalized services to the user. For instance, billboards can show personalized advertisements using the shopping information released by a user’s smartphone. Similarly, a climate control system could adjust the temperature settings depending on the preferences of a user occupying a room. One of the advantages of using a smartphone over other approaches is the sophisticated privacy controls that can be provided for the users[6].

The first phase of the efforts to turn everything smart is well underway[1]. In this phase, objects are getting smart and they are networked to connect with another device such as a smartphone. The second phase is smart services offered by smart devices that discover each other and form opportunistic networks to undertake complex operations that cannot be performed by a single device.

The essence of the smart world idea is to optimize the environment to suit the needs of a user[1]. One interesting question we may ask is how could a smart world help in the quest for developing newer
forms of “presence” (for example, crowd presence) mechanisms. In this paper, we propose Reality over Web (RoW) as part of the answer to that question.

RoW is a novel concept, where a window on the web corresponds to a window onto a real space. Through the web window users should be able to interact with people or manipulate the smart objects in the real space very much like a user who is in the local environment as the smart objects. The RoW enables many different forms of interactions between the users in the web and remote users. It can enable interactions similar to telepresence by allowing the web users to interact with remote users and manipulate things in the remote environment. Further, remote things and people can convey messages to the web users that can be rendered on top of a video feed of the remote environment, which is much similar to Augmented Reality. One of the unique aspects of RoW is the crowd it brings to the real space by mapping a window on it to a web window. People and things in the real space have an “audience” from the collection of active web windows. Coordinating the crowd activity according to the interaction requirements of the things or people in the real space is a unique challenge introduced by RoW.

A system architecture for realizing the RoW concept is described along with a proof-of-concept prototype that implements portions of the RoW concept. One essential part of a RoW implementation is accurate locationing of objects and people in a video frame. The locationing problem becomes particularly challenging because we want to reuse existing infrastructure as much as possible and yet achieve very high localization accuracies. We developed a high-frequency sound-based locationing scheme and implemented it in the prototype.

2 Related Work

To the best of our knowledge, RoW takes a unique approach in its attempt to map a web window to a window onto a real space. However, it borrows ideas from many different concepts proposed so far by other researchers. In this section, we briefly describe them and point out their differences from RoW.

Metaverse is a fusion of virtually enhanced physical reality and physically persistent virtual space[7]; it allows the users to experience it as either. In essence, Metaverse leads to a shared virtual space, where the users can interact with various virtual objects and with each other through their virtual representations. There are a number of different implementations of Metaverse which are gaining in popularity. The open-source Metaverse project[7] provides a Metaverse engine that can be used by a user to deploy custom virtual worlds. The Metaverse engine uses a client-server architecture in which the server generates the 3-D virtual world using the configurations set by the user; the client renders the Metaverse as seen by the user and communicates back the user actions to the server. In addition to the open-source engine, a number of commercial Metaverse engines like Second Life[8] and Active Worlds[9] exist.

Augmented Reality (AR)[10,11] is an area that has been gaining widespread popularity in recent years; many AR apps have been developed for smartphones and tablets. AR is a way to enhance our perception of the real world by artificially adding graphics, information or sound to it[11].

Telepresence[12,13] are a set of technologies that allow a user to experience the sensations of being remotely present and give the impression to others about the user’s presence in their environment. Telepresence systems mainly involve the human sensory elements of vision and audio. A bi-directional communication channel is used to exchange the video and audio information between two physical places.

Internet-of-Things (IoTs)[14,15] is an emerging trend that wants to make things “smart” by putting computing elements into objects that people use on a day-to-day basis. A closely related concept to IoT is the Web-of-Things (WoT)[16]. In WoT, the smart objects (things) are made to speak the “language of the web” for their inter-communication instead of using specialized protocols. By connecting the things to the web, WoT facilitates the creation of mashups including things and other resources in the web.
To realize the objective of RoW, we need to locate the smart objects and people very accurately in a video frame. Locationing computing devices is a well-studied problem; numerous techniques have been proposed using different technologies such as Bluetooth\cite{17,18}, Wi-Fi\cite{17,18}, and Infrared\cite{19}. In locationing, the objective is to determine the spatial location of a device in the real 3-D space. Another much simpler problem is detecting presence, that is detecting whether a computing device is within a given distance from another device. Using existing techniques, presence can be detected to a very high level of accuracy.

In Active Badge\cite{20}, each device emits Infrared and RF beacons. These emissions are picked up by sensors that are distributed over the environment and reported to a central server, which performs the locationing. The Cricket\cite{21} system distributes the beacons into the environment and the beacons emit RF and ultrasound pulses. Each device picks up the pulses and computes its distances from the beacons. SpotOn\cite{18} and RADAR\cite{17} utilize Wi-Fi and Bluetooth received power signal strength values to do the locationing.

Table 1 shows a comparison between RoW and three other technologies: telepresence, augmented reality, and web-of-things. As can be observed from the table, the primary difference is the purpose of the different technologies. RoW is meant to let a web audience interact with a portion of a smart world that is shown in a web window. Through these interactions users can enjoy the functionalities offered by the other three technologies in RoW. However, the interactions in RoW are limited to the scope of the web window and the physical windows that is currently mapped to the web window.

### 3 Design Principles

The design of RoW is governed by several basic principles. We interpret the relevant principles as follows:

1. **Infrastructure reuse** Users can interact with RoW services through different types of end-system devices ranging from handheld devices such tablets or smartphones to desktop computers. To handle the end-system diversity, RoW needs the end systems to run HTML5 compliant web browsers. People or objects wanting to be featured in RoW need computing elements embedded or co-located with them. For example, people need to carry a computing element such as a smartphone or smartwatch that can run an app to talk with RoW services. The remaining part of the RoW framework is the infrastructure that supports the RoW services. This part of the framework has components that are specifically created for RoW. By reusing existing end-systems, RoW allows users to benefit from the services offered by RoW without requiring the adoption of a new genre of gadgets. Further, by reusing smartphones as computing elements, RoW sets a low barrier for people to participate in the system.

2. **End-user control** The ubiquitous nature of

---

**Table 1** Comparing RoW with existing alternatives.

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Mode of deployment</th>
<th>Access for web users</th>
<th>Smart object support</th>
<th>Remote actuation</th>
<th>Visualizing remote events</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reality over web</td>
<td>Video over web</td>
<td>Accessible for single and crowd of users</td>
<td>Yes</td>
<td>Using network connections</td>
<td>Video over web shows remote events</td>
</tr>
<tr>
<td>Telepresence</td>
<td>Video</td>
<td>Accessible for one user at a time</td>
<td>No</td>
<td>Using robotic manipulators</td>
<td>Video shows remote events</td>
</tr>
<tr>
<td>Augmented reality</td>
<td>Video</td>
<td>Display data accessible to all users</td>
<td>No</td>
<td>Not supported</td>
<td>Video overlays</td>
</tr>
<tr>
<td>Web-of-things</td>
<td>Web</td>
<td>Accessible for one user at a time</td>
<td>Yes</td>
<td>Using network connections</td>
<td>Web</td>
</tr>
</tbody>
</table>
the web means any privacy risk posed by a web-based system can reach significant proportions very quickly. Therefore, RoW needs to minimize the privacy risks faced by its users. The best way a system can protect the privacy of its users is by not collecting users’ identifying information unless it is truly essential for its operation[22]. To satisfy this condition, RoW does not gratuitously discover people by using schemes such as face recognition. Instead, RoW relies on schemes where participants maintain control over their detectability at all times so that they can go offline while remaining visible in the frame of a video. In addition to controlling their detectability, users should be in control of how they are identified once they are detected by the system.  

3) Connection transparency The computing elements carried by the people or embedded in the objects can connect with a RoW service point through a variety of networks. Suppose people participating in a RoW session are connecting through their smartphones. The smartphones can have their own wide-area data connections or connect through the RoW service point using access networks such as Wi-Fi or Bluetooth. The RoW framework is responsible for making the people or objects accessible from the web the same way irrespective of the actual network connectivity. 

4) Privacy aware matching The RoW framework performs a matching between the web users and the participants in a RoW session. The web users and participants can have permanent network identities such as IP addresses, social network identities, and email identities. During the matching it is essential for RoW to refrain from releasing such permanent identities belonging to one party to the other party. The association between a web user and a person or object within a RoW session can change with time; existing associations can expire and new ones can become active. The RoW framework is responsible for maintaining the associations and ensuring that the time limits are enforced properly.

4 Reality over Web

4.1 Overall concept

The primary goal of RoW is to implement the idea that a window in a web browser can be mapped to a window onto a real space. The mapping between the two windows should be achieved such that the following characteristics are true: (a) objects and people in the window are separately addressable, (b) events directed towards an object on the web window reach the corresponding object in the real space, (c) events generated by the real object are notified to the web window, (d) level of association between the web window and the real space is controlled by the authorization level secured by the user controlling the web window, and (e) objects in the real space can independently set conditions that should be honored in associating them with the web window.

An example mapping of a web window onto a window in the real space is shown in Fig. 1. The real space has two people and a network-enabled thermostat. The people are associated with the web window via the smartphones they carry. As illustrated in Fig. 1, multiple objects/people can be in a window. The portion of the web window occupied by the object/person is associated with the real object/person. 

The mapping of real objects/people into a web window needs recomputation if already mapped objects/people move out of the window or new objects/people move into the window. Also, the mapping needs recomputation if the real window changes. Such a change can occur when the camera that is capturing the real space pans.

4.2 System architecture

As shown in Fig. 2, a RoW system has three major types of components: browsers, servers, and smart
objects. Users wanting to interact with smart objects or people in a remote space can do so using HTML5 capable browsers. The remote space is captured by a video camera and sent to the browsers as HTML5 video. The HTML5 video is augmented with locational information regarding the smart objects by the RoW servers. The application running in the browsers are responsible for properly rendering the video and locational information to the user. The user interactions are captured and sent back to the servers by the same application.

The organization of the most important functions in the RoW architecture is shown in Fig. 3. The functions that are part of the RoW architecture can be broken down into four types based on what they do: locationing, data mining, data presentation, and input aggregation and coordination. Although the prototype relies on high-frequency sound-based techniques for locationing, the architecture is generic. We could use another locationing technique or a combination of techniques to achieve higher locationing accuracies. One important requirement of locationing is automatic calibration. We could conceive an automatic calibration process, where locationing information from one technique could be used to correct the error out of the other techniques.

The smart objects in the real space can release identifying information that can be used by RoW to accumulate more information about the smart objects. For instance, a smartphone could release the tweet handle of the user to RoW. Using the tweet handle, RoW could query the Twitter service to gather the latest tweets from the user.

Once RoW has all the information about the smart things and people in the real space, it needs an intelligent layouting scheme to present the data in the web window. It should be noted that the web window is primarily showing a video of the real scene. Therefore, the information overlaid on top of the smart things or people should be carefully placed to minimize the obstructions. As people or things move in the scene, the layouting scheme needs to revise its placements.

The RoW architecture that allows large number of web users to simultaneously interact with the smart objects or people introduces many unique challenges. In particular, depending on the functions exposed by the smart objects the activities of the web users have to be aggregated or serialized. Providing a generic framework for RoW that can be customized in this regard in different ways is an interesting implementation problem.

### 4.3 Proof-of-concept prototype

In this section, we describe a proof-of-concept prototype that implements portions of the RoW architecture. In particular, the initial prototype is focusing on the localization part of RoW.

The key idea leveraged by the RoW prototype to locate devices is the directional sound radiation pattern of speakers. Most speakers emit sound such that the sound energy is dissipated more in certain directions and less in others. The higher the frequency of the tone emitted the radiation pattern is more directed; however, the actual radiation pattern can vary depending on the speaker design. We place a speaker on a turning platform and sweep the space that contains the devices as shown in Fig. 4. The RoW server precisely controls turning platform such that the angle swept by the
speakers radiation pattern is known. The devices that want to be located will continuously measure the sound intensity levels at the tone being radiated by the speaker and periodically report them to the RoW server using a Wi-Fi connection. Simultaneously, the space is captured by a webcam that is collocated with the speaker. The rotational axis by which the turntable rotates the speaker is where the webcam is situated. This way the speakers sound beam can properly sweep the space being captured by the video. The video is converted into an HTML5 video feed by the server and composted with a touch sensing transparent HTML5 canvas element. The users can access this composite HTML5 video over the web.

Figure 5 shows an example locationing scenario. Here two smartphones are in the physical space that is scanned by the RoW prototype. After the scanning phase is completed, the RoW prototype draws two boxes to denote the region of the video that contains the smartphones. Any touch events made by the user inside a box will be directed to the corresponding smartphone.

5 Illustrative Techniques for Locationing

As explained in Section 4.3, the RoW prototype uses a high-frequency sound-based locationing scheme. The locationing server receives the sound intensity reports from the devices and uses a detection algorithm to determine the positions of the different devices. In our experiments, so far we have implemented the following algorithms.

- **Peak amplitude method** In this method, the server searches for the maximum sound intensity level reported by a device and assumes that the device is located at that location. A window surrounding that location is reported by the detector.

- **Minimum threshold method** The server finds out the sound intensity levels that exceed a certain threshold. It computes the smallest possible window to include all such reports and returns it as the location of the device. A window computed using this scheme can be quite large and may not reveal the actual location of the device.

- **Maximum energy method** This method is based on the observation that by adding all the sound intensity levels reported by a device, we can estimate the sound energy received by the device. We split the range of sound intensity levels into two sets and compare the energy received by the two sets. The set with the higher energy level is deemed to contain the device. This process is iterated few times to narrow the region that contains the device.

From the experiments, we observed that the maximum energy method provides the best detection accuracy. Out of the methods tested, it is the most robust against reflections caused by other objects in the environment.

Figures 6 and 7 show the results of the experiments we performed to evaluate the accuracy of the
locationing scheme. Accuracy of locationing is computed by observing the difference between actual and detected distances ($y$ and $x$ axis respectively in the graph). The actual distance is the distance from the left margin of the video to the image of the object in the video. The detected distance is the distance from the left margin of the video to the center of the rectangle drawn by the detector. In these figures, we show the results for the maximum energy method because it provided the most robust detection. Figure 6 shows the results we obtained in a sound insulation room – the walls of this room are made of sound absorbing materials that minimize reflections and external disturbances. Figure 7 shows the results we obtained in a lab environment.

An important observation from these experiments is that localization accuracy is still a problem. We are continuing to investigate many different ideas to achieve better localization accuracies. One idea is to use parabolic speaker designs that will focus the sound into narrower beams. Another idea is to combine the sound-based localization with other approaches such as depth camera information or image feature-based detection schemes.

6 Example Applications

Video is already a highly popular media format on the web. Using RoW, we can add a new dimension of interactivity to videos over the web without requiring any extensions to the existing web infrastructure. In this section, we describe example applications that can benefit from the new type of interactivity provided by RoW.

(1) First Person Games (FPGs) They are an important sub-category of video games. In FPGs, the virtual world is rendered from the perspective of the player. The virtual world is maintained by the game engine and contains the characters and environment that make up the game state. With RoW, we could develop new types of FPGs, where the characters and environment are real. That is, real world including people can form the scenery for the game world. Part of the game state could be maintained by the computing elements (smartphones and other embedded computers) in the scene. For example, a paint ball game can be created where real people can be the targets of digital paint balls. The smartphone held by the players can record paint ball hits and misses. Because part of the game state is held in the smartphones that are held by the players, the game state can also evolve due to players bumping into each other.

(2) Tele-education/Tele-instruction Video conferencing is widely used for tele-education purposes. Over the years, many video conferencing systems have been developed with a number of interesting features. Several video conferencing systems provide speaker highlighting features that automatically selects the “talking head” in a given session. With RoW, it is possible to develop video conferencing systems that are even more interactive. We could allow a user to tap on an arbitrary participant from a crowd and initiate a conversation with that person. For example, a professor can pick a wayward student from a classroom (assuming that the professor has a video feed of the classroom – a reverse channel) for a one-on-one conversation. In addition, a user can share data files or send messages with a selected person from a crowd.

Another possibility created by RoW is virtual laboratories. Although virtual laboratories already exist, RoW introduces new modes of interactions that can be leveraged in designing and implementing laboratories with better user experiences. We could use RoW to interact both inside and outside an instrument. For example, a view could show the exterior of a thermostat and tapping on the object will present a menu to control the thermostat. In general, this facility provided by RoW could be extended to look at computing nodes collectively or control them individually. The exterior view allows the users to obtain an independent assessment of operation of the computing node. One example laboratory could be a virtual robotic laboratory consisting of a swarm of micro robots. Using the facilities provided by RoW, students could observe the collective behavior after they program selected robots. Individual robots can be controlled by accessing their control interfaces or dragging and dropping a new control program onto them.

(3) Home automation and control Many household appliances are getting smart – they are embedded with powerful microcontrollers that expose sophisticated and programmable interfaces. The smart appliances use their processing capabilities to optimize power usage, provide easy to use interfaces, and interact with other appliances. The next step is for smart appliances to operate in concert to provide a smart environment in the home. By cooperating with each other, the smart
appliances will be able to provide the optimum service while consuming the minimum amount of energy.

Another interesting development is the cooperation between smart appliances and home automation/security systems that allows remote control and inspection of the appliances in a home. For example, a home owner could check whether the garage door is open or close using a mobile app. With RoW, telepresence could be added to the services afforded to the home owners. A home owner could interact with the smart appliances and see their reactions for her actions via a video link. The experience home owners get from a RoW-based system will be much more realistic than the experience gained via a web-based system.

(4) Interactive virtual showrooms Advertising is a major activity on the web. Adding interactivity to advertising campaigns is an assured way of engaging the audience and make the campaigns more effective. RoW is ideal for setting up interactive web-based virtual showrooms for advertising campaigns. Virtual showrooms could feature electronic gadgets such as televisions and computers. With RoW infused interactivity, users can observe the working of a gadget externally and interact with it in a selective manner. For example they could turn on/off selected features of a gadget and observe its behavior. Even interconnecting smart gadgets to check their inter-networking capabilities could be part of the services offered by the virtual showrooms.

7 Concluding Remarks

The major thrust of the recent developments in Internet-of-things is to create an smart world, where objects in the environment can optimize their collective behavior to minimize the power consumption and maximize the services offered to the users. Simultaneously, web is further strengthening its position with the introduction of HTML5 and the impending revision of HTTP. Therefore, it is timely to investigate the expanded role web that could play in a smart world; the RoW is precisely such an exercise.

In this paper, we introduced the RoW concept and an architecture for realizing the concept. The main idea of RoW is that a web window can be mapped to a window onto real space and using the web window the objects/people in the real world can be manipulated. We designed and implemented a proof-of-concept prototype that implements portions of the RoW concept. One of the important functions required by RoW is determining the positions of the objects within the web window. Although object locationing is a well studied problem, the RoW context introduces unique challenges; it calls for an investigation of this problem with the aim of developing accurate locationing schemes. As part of the proof-of-concept prototype, we developed an new object locationing scheme that uses high-frequency sound. From the extensive experiments we carried out on the prototype, we observe that a high-frequency sound-based scheme could be part of the locationing framework for RoW.

There are many issues that require further research to fully develop the RoW concept. One issue is the accuracy and automatic calibration of the sound-based locationing scheme. From the experiments we carried out on the prototype, it is evident that proper calibration is necessary to get the desired level of accuracy from a sound-based locationing scheme. We need to develop an automatic calibration scheme that will use auxiliary information obtained from image-based detectors to correct the errors in the sound-based schemes. Another issue is the synchronization of the events in the real and virtual worlds. The video captures the events in the real world and the events in the virtual world are managed by the RoW servers. It is important to synchronize the events across the two worlds so that viewers can interact with objects in a web window and not with objects that have moved out of the window.

References

[1] B. Wasik, In the programmable world, all our objects will act as one, Wired, vol. 21, no. 6, pp. 140-147, 2013.
Muthucumaru Maheswaran is an associate professor in the School of Computer Science at McGill University. He got a PhD in electrical and computer engineering in 1998 from Purdue University, West Lafayette and a BScEng degree in electrical and electronic engineering in 1990 from the University of Peradeniya, Sri Lanka. He is in the editorial board of the Journal of Big Data (Springer). His research interests span the areas of cloud computing, online social networks; he has investigated many issues related to resource management, identity, data privacy, and access control in these areas. He has supervised the completion of 8 PhD theses in the above areas. He has published more than 100 technical papers in major journals, conferences, and workshops. He holds a US patent in wide-area content routing.

Devarun Bhattacharya is a software engineer currently working in Cisco Systems, Canada. He received his Master’s degree from McGill University, Montreal where he was part of the Advance Networking Research Lab. He got his Bachelor’s degree from SASTRA University, India. His research interests include computer networks, mobile computing, and distributed systems.