The Sentient Visor: Towards a Browser for the Internet of Things

Paúl E. Estrada-Martínez, Jorge Álvarez-Lozano, J. Antonio García-Macías and Jesús Favela
Computer Science Department, CICESE Research Center
Ensenada, Baja California, Mexico
{pestrada, jalvarez, jagm, favela}@cicese.mx

***Abstract*—As we advance towards an Internet of Things (IoT), it becomes necessary to conceive new forms of interaction between people and the everyday objects that surround them. A world full of internetworked objects with sensing capabilities requires effective tools in order to interpret contextual data gathered from the environment and services provided by smart objects. We aim to provide a browser for the IoT through what have referred to as a sentient visor. In this paper, we present a working prototype of a sentient visor by exposing its core technology and applications. The sentient visor incorporates mechanisms to identify smart objects, interact with them and infer information from contextual data by using semantic web technologies. We also pose some limitations and future directions of our approach.**

***Keywords*—Sentient Visor, Internet of Things, Semantic Information**

I. INTRODUCTION

The Internet of Things (IoT) is a paradigm where every single object or thing is interconnected and addressable through standard communication protocols [1]. In order to make this vision a reality, everyday objects need to be augmented with networking capabilities and sensory ability, creating “smart objects”.

Smart objects can be built by embedding current technologies in a certain object or thing. For instance, we can incorporate RFID, sensors that measure physical parameters (e.g. temperature, light, weight, humidity) and actuators in home appliances, cookware, automobiles, furniture, and other common objects that we use on a daily basis. Wi-Fi and Bluetooth can be used for communication.

Besides, we can add mobile devices capabilities such as GPS, accelerometers, compasses and gyroscopes. As a result, embedding smart objects in the environment will create an augmented physical space where different smart objects provide information and services.

Previous research in smart objects has resulted in new terms and applications, such as ambient displays [2], sentient artifacts [3] and sentient displays [4], which gather contextual data to provide additional and more useful information than the users can perceive by themselves.

However, as research efforts continue to make the IoT a reality, embedding smart objects in the environment faces important challenges. For instance, how can a user interact with smart objects in augmented physical spaces? How to determine whether or not smart objects exist in a certain location? How to navigate through the augmented environment in order to identify individual objects? Finally, how to access and display data related to such objects?

Our approach to these problems is focused on having a universal browser for the IoT running on mobile devices (e.g. a smartphone or a tablet). We call this browser a sentient visor [5], [6]. In Fig. 1 we exemplify this concept. However, a browser for such purpose should meet important characteristics dictated by the IoT.

In a previous work [7], we explained the rationale of designing a sentient visor by exposing the differences between browsing in the internet and browsing in the IoT. We also proposed a design space for sentient visors comprising the following six design dimensions:

1) Object identification and service discovery: A smart object can be identified through different techniques such as visual ones (e.g. 2D codes or object recognition algorithms) and electronic ones (e.g. RFID or NFC). Once the smart object is identified, a sentient visor must be able to discover the services provided by such object. These are necessary first steps to interact with a smart object.

2) Context-awareness: A sentient visor must harness data gathered from different sources such as accelerometers,

Fig. 1. Using a sentient visor system to interact with a smart food product.

microphones, environmental parameters and other contextual data to determine what this information means and how to react accordingly to it. With such an adaptive behavior, a sentient visor could present personalized information to users or even trigger the automatic execution of a service.

3) Multimodal information: The interaction with smart objects may occur in different situations and locations. Therefore, sentient visors need to handle multimodal information, adapting input and output to user preferences. For example, a user could prefer voice commands over touch-screen gestures. Also, it might be more effective to display an animated graphic on screen instead of a plain text message or even a smart object could serve as an output by activating an actuator.

4) Mixed-Reality User Interface: This dimension ranges from augmented reality (AR) to augmented virtuality. A sentient visor might use an AR display while another could use real world entities in virtual worlds.

5) Contextual reasoning: A sentient visor must also react to more subtle inputs that are not important for a traditional browser such as pointing to places or identifiying an object's owner. This information along with other contextual data might be used to infer spatial and temporal events, thematic activities or perhaps the user goals.

6) Semantic communication: Using semantics to exchange information with smart objects allows the sentient visor to determine what the object is trying to convey. For instance, the type and format of readings or the characteristics of their services. This enables better inferences and richer interactions.

With this background, in this paper we focus on the implementation and technical aspects of the system.

As we show in the next section, a sentient visor can be used in several scenarios that demand specific features to accomplish user's goals while interacting with different smart objects. Thus, we have developed our prototype system with the core mechanisms to identify individual smart objects and the services they provide, context-awareness for gathering data from the surroundings and displaying relevant information to users. With this work, we want to show a sentient visor prototype system considering the six design dimensions.

In this paper, we present a detailed implementation of the sentient visor, exposing its core-technology and showing some sample applications. The remainder of this paper is organized as follows: In section II, we describe application scenarios. In section III, we show the architecture of the sentient visor. Then, in section IV, we detail its implementation. Later, in section V, we present a discussion about the sentient visor and related work. Finally, in section VI, we talk about the future work and conclusions of our research.

II. APPLICATION SCENARIOS

In order to exemplify the concept and usage of a sentient visor, we now describe 3 application scenarios within the healthcare domain, where the sentient visor is used for browsing the information space in the IoT.

A. *Dietary Care*

In the first scenario we introduce Mr. Velazquez, a 73 year old person who follows a strict diet since diagnosed with diabetes. Also, his Body Mass Index (BMI) is higher than recommended. When going to the supermarket, Mr. Velazquez uses the sentient visor software installed on his new cell phone. He points with the camera of the cell phone at a product and sees a chart indicating the nutritional information about that specific product. Also, the sentient visor gives him advice about whether or not the product is recommended for his diet by displaying text on the screen. However, he cannot read properly due to the small font and visual impairment. Being aware of this situation, the sentient visor chooses to give him auditory advice while presenting a bigger chart on screen.

B. *Medical Monitoring Aid*

This second scenario focuses on aiding medical personnel. During her morning rounds, the nurse Diaz visits all her patients carrying a tablet-sized device, with its sentient visor software. When monitoring a patient, the nurse points the device to the patient's bed, and views on the screen the overall health status of the patient, including his vital signs and medical treatment. Looking at the sentient visor nurse Diaz finds two annotations that Dr. Gomez left a couple of hours ago, indicating the need to perform a blood and glucose exam. Because the patient was diagnosed with blood pressure problems and has an upcoming surgery, it is required to monitor the patient's vital signs, to verify that they are stable. For this purpose, Dr. Gomez points at the patient with his sentient visor enabled device and he sees the history record of the patient for the last 8 hours.

C. *Services Guide*

In the third scenario we have Mrs. Perez, who was recently diagnosed with cancer. Having the need for several medical examinations, she frequently visits different medical facilities. Thus, she often needs help finding the office of laboratory where she needs to go at each building. Fortunately, the hospital buildings have augmented services to help her and other patients. The doctor has required her to have a blood examination at the General hospital but Mrs. Perez is not familiar with the hospital facilities. She turns on her sentient visor enabled cell phone and discovers available augmented services. Then, she gets a menu on her device's screen and she looks for the required laboratory. Once she selects the desired destination, the sentient visor determines the location of Mrs. Perez and through the sentient visor she gets visual aid and audio instructions for arriving to the desired place.

III. Architecture

Analyzing our vision of a browser for the IoT and application scenarios of previous sections, we defined the architecture of the sentient visor (see Fig. 2), composed by a mobile client device, a server for contextual data processing and another server which encapsulates third-party web services that extend the functionality of the sentient visor.

A. *Context Capture Component*

One of the key elements in the previous scenarios is the need to identify individual smart objects in the environment as well as services provided in different locations. Hence, the sentient visor has to provide both identification and service discovery mechanisms for recognizing smart objects in the surroundings and finding available services for users.

Additionally, for displaying relevant information for each user and determining what is happening at a given location, the sentient visor supports the categories of features for a context-aware application described in [8]. These categories are: *presentation* of information and services to a user, automatic *execution* of a service for a user, and *tagging* of contextual information to support later retrieval. Also, it uses information about the network status, date/time and user data.

With these elements, the context capture component provides mechanisms to enable the continuous sensing of the environment, obtaining user profile and location data, detecting available services in the surroundings and identifying individual objects in the line of sight of the device's camera as well as the current date and time.

B. *Augmented Reality Component*

To provide a fluid interaction between the application and the user, the sentient visor has to provide a nonintrusive and

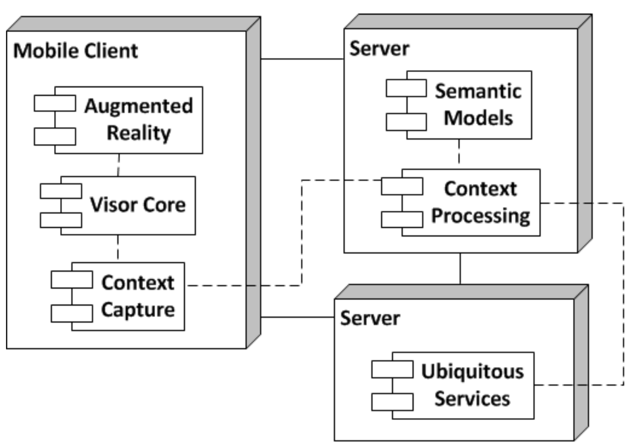

Fig. 2. Architecture of the sentient visor.

user friendly interface, simple enough to be used by people with little technical knowledge and less experienced in using cell phones or other devices.

In addition, since the sentient visor must allow exploring the environment, users should be capable of seeing the landscape and pointing to an object through the device's camera. This feature should help users to visually know where the information comes from. For this purpose we use AR to augment the perception of the user, since AR allows "experiencing the real world and augmenting the experience" [9]. In that way, AR supports the sentient visor in aiming to provide a user interface between users and smart objects.

Currently, it is feasible to deploy mobile AR applications on cell phones. For instance, several mobile AR applications are already available for wide spread use (e.g. Layar [10], Wikitude [11], SekaiCamera [12]), and also, corporations like ARToolworks [13] provide multiple development tools for AR applications, with free and commercial licenses available.

Then, the information received from the servers is used by the AR component to display a virtual entity to the user. Unlike traditional AR applications, the sentient visor allows to handle multimodal information such as text, audio and imagery.

C. *Visor Core Component*

While running on a device, the sentient visor needs to perform several tasks such as identify a smart object, communicate with external servers, display information and react to user input. The visor core component manages all these tasks as separate threads, providing a continuous execution of the sentient visor in the mobile client.

D. *Context Processing Component*

The contextual data gathered by the context capture component is not useful if we consider them as isolated data. Therefore, the context processing component combines the gathered contextual data to make inferences using a forward chaining procedure. These inferences first determine which information will be displayed to the user, accordingly to the user role, date and time data, among others contextual aspects. Then, another inference is made to determine how the sentient visor will display the information based on the user's preferences.

E. *Semantic Models Component*

Smart objects, user profiles and other resources are identified and addressed with a Uniform Resource Identifier (URI). This component obtains data related to each resource that is then processed by the context processing component to make inferences. The resources are represented as semantic models by augmenting their web representation with semantic metadata (see section IV.C).

F. *Ubiquitous Services Component*

This component adds extra functionality to the sentient visor through the use of UbiSOA [14], a platform that was developed in our research group to create ubiquitous computing applications that consume web services. For instance, UbiSOA provides geo-location based on the RSSI signal, but also provides access to third-party web services, such as Google Maps, Twitter, among others. It also supports adding more services through standard web mechanisms.

IV. IMPLEMENTATION

Having defined the architecture of the sentient visor, we developed a working prototype with the Java programming language. For its implementation we used some of the current semantic web technology standards, such as RDF and CC/PP. Also, we used vocabularies and inferences. These elements are considered by the W3C as a part of the technology stack for supporting the semantic web. We describe below the implementation of the sentient visor and the tools we used.

A. *Smart Object Identification and Service Discovery*

We have used a resource-oriented approach representing each smart object as a web resource with a unique name accessible through well-known web mechanisms in a RESTful style (e.g. GET, POST). Thus, the unique URI of each web resource serves as an ID. Besides, we used the multicast DNS (mDNS) protocol with DNS based service discovery (DNS-SD) for both the registering and discovery of services available in the local area network.

We acknowledge that a smart object can have communication and processing capabilities that can serve to notify its identity and services by itself. However, for now we only use two-dimensional codes taking advantage of their storage capacity, specifically the QR codes, by attaching one of these codes to a smart object.

This labeling provides a reference for the sentient visor, serving as an identification mechanism to obtain the URI of the smart object. We used the ZXing library [15] for decoding QR codes. For enabling the service discovery we have implemented the mDNS/DNS-SD by using Apple's Bonjour [16] that enables automatic discovery of devices and web services over IP networks without the need of having a directory server.

B. *Semantic Information*

As the sentient visor communicates and shares information between different smart objects, it needs a common vocabulary that is adopted by both the sentient visor and the smart objects, enabling the understanding of the information transmitted between entities. Semantics play a key role in this aspect as they provide meaning to the contextual data by adding metadata annotations. This allows software entities to share, reuse, understand and make inferences about the information.

However, to enable this behavior it is necessary to define such a vocabulary. This requires the definition of objects, concepts, relations and other entities in the domain of interest.

For this purpose we used ontologies [17] that allow the definition of a knowledge representation model based on conceptualization, which derives into a common vocabulary that software entities can use to share information.

For sharing information between smart objects and the sentient visor, we have designed the ontology using Protégé [18] that provides tools to build domain models and ontology-based applications, allowing the creation, visualization and manipulation of ontologies in several formats.

The designed ontology comprises all the aspects for describing contextual data, concerned to location, date and time, user data and also, the information of smart objects and their services. The main classes of the ontology are: *Person*, *Location*, *Presentation*, *SmartObject*, *Service*, among others. In Fig. 3 we present a fragment of the ontology.

An important feature of the sentient visor is to handle multimodal information. The ontology also defines the necessary concepts to help the AR component perform this task. In Fig. 4 we show a fragment of the ontology comprising the multimodal information concepts.

C. *Knowledge Representation*

Once we implemented the ontology, it was necessary to model the contextual information using that ontology. For this purpose we used Jena [19], a framework that allows building semantic web applications and provides a programming environment for RDF, RDFS and OWL. When the information component obtains the data related to a smart object's URI, it creates a RDF document with all the contextual data following the designed ontology model. An example of a RDF document is shown in Fig. 5.

D. *Inference Engine*

The application scenarios presented before involve a significant amount of contextual data, which is used to determine what information will be displayed. Whereas we highlight the importance of semantics in the interaction and communication between entities, semantics can also be used to apply rules in the contextual data to make inferences.

Because the preferences and needs of each user are significantly different, we use inferences to determine what information best suits the user profile. For instance, we can retrieve information according to its gender, weight, date, and time, among others. Also, inferences are used to decide how to display the information according to the user needs. For example, if the user has visual impairments, it can have an audio advice instead of only seeing graphics on screen.

The inference engine determines the appropriate visualization accordingly to the Composite Capability/Preferences Profile (CC/PP) [20] of each user.

The context processing component integrates Pellet [21] to make inferences on the gathered contextual data. Pellet is an OWL reasoning engine that also supports rule-based inferences. Two types of rules are defined for this prototype, the first is the content of information and the second refers to the presentation of that content.

We used a rule-based inference engine because the application scenarios involve significant conditional branching and decision-making to give recommendations and advice based in the gathered contextual data. For instance, the sentient visor can retrieve information related to the user's role, gender or weight, and it can decide the presentation of information (e.g. audio, imagery, text) accordingly to the user preferences. An example of the implemented rules is as follows: *"If the food is low in sugar and is natural, then the food is a good alternative as an appetizer"*.

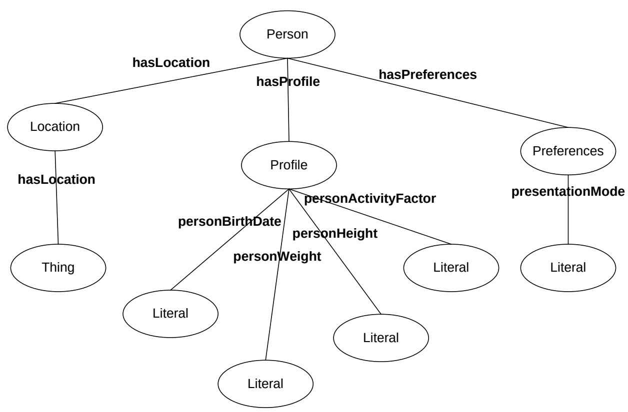

Fig. 3. Fragment of the ontology designed for the sentient visor.

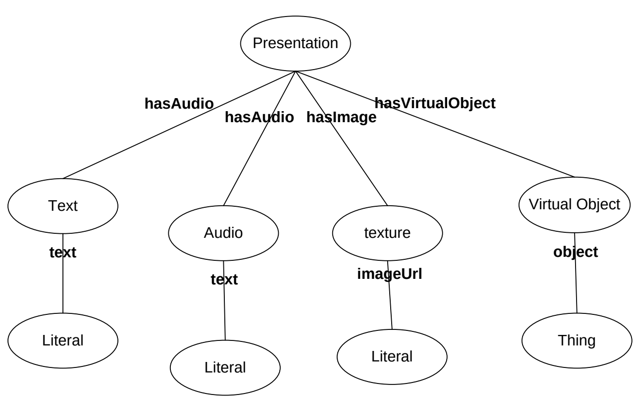

Fig. 4. Fragment of the ontology showing the concepts for handling multimodal information.

Once the inference task is finished, the context processing component creates a RDF document with the content and desired presentation of the information (see Fig. 6). The sentient visor receives this RDF document and then, it is used by the AR component to activate the appropriate virtual entity (see Fig. 7 and Fig. 8).

E. Augmented Reality

For the current prototype, we are not using any AR specific software. Instead, we achieve AR using the QR code as an AR marker by tracking the control points of the QR code. We perform the tracking with the same library we use to decode the QR code (see section IV.A). For rendering 2D and 3D graphics we used the Processing [22] programming language.

Moreover, due to the behavior and changing environment of the application scenarios, the sentient visor must be able to display multimodal information dynamically (e.g. imagery, text, audio, virtual objects). As exemplified in the scenarios presented in section II, this is very important in order to appropriately display information based on the user's preferences. Therefore, we have provided the AR component with mechanisms to handle and display multimodal information.

```
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
xmlns:SV="http://www.semanticweb.org/ontologies/
sentientVisor#">
<rdf:Description
rdf:about="http://localhost:8182/ubicomp/users/2
">
<SV:birthdate>12-12-1980</SV:birthdate>
<SV:activityFactor>1.375</SV:activityFactor>
<SV:weight>80</SV:weight>
<SV:height>170</SV:height>
<SV:gender>Male</SV:gender>
<SV:lastName>Smith</SV:lastName>
<SV:firstName>John</SV:firstName>
</rdf:Description>
</rdf:RDF>
```

Fig. 5. RDF document describing information about a user's profile.

```
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
xmlns:SV="http://www.semanticweb.org/ontologies
/sentientVisor#">
<rdf:Description
rdf:about="http://localhost:8182/ubicomp/presen
tation/audio">
<SV:text>NOT RECOMMENDED, IT CONTAINS 50 % OF
ALLOWED DAILY INTAKE</SV:text>
</rdf:Description>
</rdf:RDF>
```

Fig. 6. RDF document describing the content and presentation of information to be display with AR.

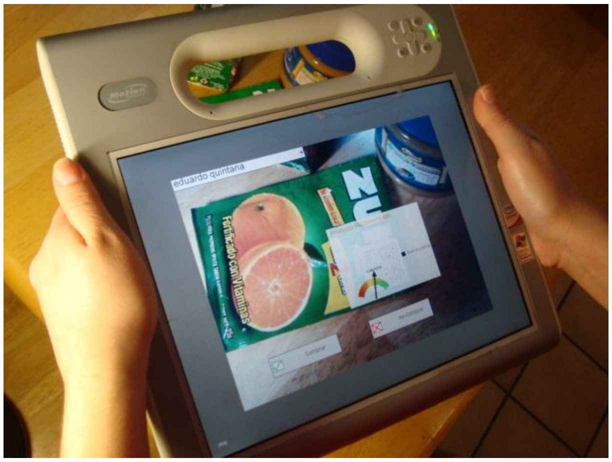

Fig. 7. The sentient visor displaying a sugar meter and giving audio advice.

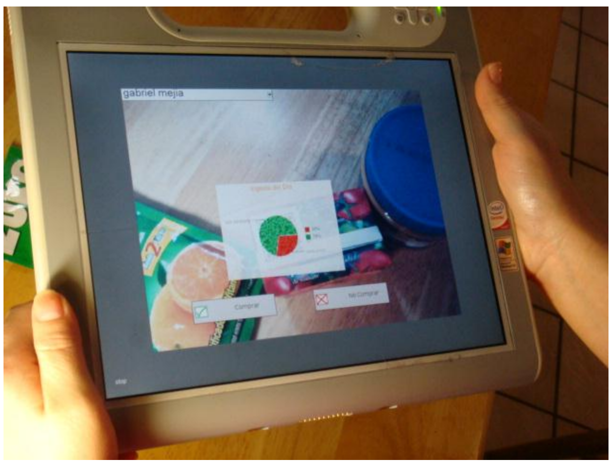

Fig. 8. The sentient visor displaying a pie chart with the calories of a tomato sauce.

V. Discussion And Related Work

Taking into account the design dimensions we mentioned before (see section 1) for developing a sentient visor, we consider it is possible to enable the navigation of the information space in the IoT.

Nevertheless, the navigation metaphor in information spaces has been addressed before in different research projects and consumer-oriented applications such as location-based mobile AR applications and context-aware systems. However, our concept of a sentient visor is to have a universal browser to interact with all the smart objects and services of the IoT. This approach comprises a wider scope than other IoT projects. Hence, we have identified some state of the art systems that we consider are closer to our sentient visor and now we analyze their differences according to the design space of the sentient visor. A comparison between the sentient visor and such systems is summarized in Table 1.

As we can see, other systems do meet some of the design dimensions of the sentient visor. For instance, they all can identify some kind of real world objects and provide some degree of context awareness. Also, they make use of multimodal information and a mixed-reality UI.

However, mobile AR systems such as [23] and [24] are not sufficient for navigating the IoT. Even though these systems follow the idea of having a browser to display contextual information and using the real world as a user interface, if we tried to use them for the IoT we would be limited because they do not consider the possible interaction with individual smart objects by means of a web presence for such objects and their services. Therefore, a user would not be able to identify a smart object and interact with it.

It is also worth noting that their context-awareness is low, compared to the sentient visor. The applications they are focused on are location-based systems whereas the sentient visor makes use of multiple contextual data as we explained in previous sections. These navigation systems were not intended specifically for the IoT as the sentient visor. As a result, the sentient visor is more capable of interacting with smart objects and using the services available in augmented spaces.

In [25] the authors proposed a framework for accessing services of physical products. The framework is envisioned as a browser for the IoT. This work shares the idea of a universal browser with our sentient visor, and also takes into account location data and services of objects. To our knowledge, this is the only work exploring the paradigm of a browser intended for the IoT as we do.

However, we have some important differences about what characteristics should such a browser possess. For example, they do not use contextual reasoning, semantics or a mixed-reality UI as defined in our design space. This main difference makes the sentient visor an adaptive system that autonomously infers the content and presentation of information (e.g. text, audio, imagery), supporting the user preferences and needs by reasoning over contextual data including users and objects information.

TABLE I. COMPARISON BETWEEN THE SENTIENT VISOR AND OTHER SYSTEMS

System	Design Dimensions of the Sentient Visor					
	Object ID and Service Discovery	***Context-Awareness***	***Multimodal information***	***Mixed-Reality UI***	***Contextual reasoning***	***Semantic communication***
Sentient Visor	X	X	X	X	X	X
[23]	X*	X*	X	X		
[24]	X*	X*	X	X		
[25]	X	X	X			
[26]	X	X	X*			

*Limited

A well-known project is CoolTown [26], which explores an infrastructure to support web presence for people, places and things, putting information about people and places in web servers, and also embedding web servers into objects (e.g. printers). Then, using a PDA in places with such web presence, people can use the available services and retrieve local information. This project was one of the first to explore the concept of a universal platform to interact with real world objects. However, it lacks some important characteristics we discussed above, such as semantic communication, contextual reasoning and also, a mixed-reality UI, because the authors only considered displaying information using web pages.

We have found that semantic communication and contextual reasoning are important aspects of the sentient visor that are not considered by the above projects. These aspects are needed to build a browser for the IoT. For instance, a person using a web browser commonly types queries in a search engine or navigates from link to link. However, a browser for the IoT should consider other implicit user input such as the proximity between the user and smart objects, the information provided by their services, user goals, among others. All this information can be used to infer how and when to trigger a specific action (e.g. display a specific message or animated graphic).

We argue that a browser for the IoT should meet the design dimensions we proposed because of the differences we encountered between browsing the traditional internet and the IoT. Also, there are scenarios that demand the fulfillment of these characteristics. For instance, we could have an automobile with a transparent display as a front window. Then, a person driving the car could see information about the weather, the traffic or the proximity of other cars, while still being able to drive without disturbing the view of the road. In this scenario, it is clear that a mixed-reality UI has its advantages over dedicated display devices, and since the person is driving, he needs a proactive system that lowers the need for explicit user input. Possibly limited to voice commands or even to none, making the browser use contextual reasoning and semantics to infer what it needs to do for the user. Hence, we consider that the sentient visor is a more capable browser for the IoT.

VI. CONCLUSIONS AND FUTURE WORK

We have presented a working prototype of our sentient visor, a browser for the IoT, and whereas we have succeeded enabling the navigation of the IoT with this preliminary approach, there are still some issues that we need to address to further advance our vision.

As shown in this work, we used contextual data to infer the presentation and content of information that is displayed by the sentient visor. However we plan to extend its functionality to infer more information from sensors and actuators embedded in smart objects and the environment, such as RFID, proximity and temperature sensors, among others, enabling more sources of knowledge. For instance, it would be desirable in certain scenarios for the sentient visor to change in real-time the displayed graphics or show more detailed information based on the proximity between the smart object and the sentient visor.

Semantics will play a key role in the interaction between the sentient visor and sensors due to a variety of sensors with different capabilities that can provide different spatial, temporal and thematic information. Semantic annotation of raw sensor data can provide meaning for such sensory data, serving as new source of knowledge. Therefore, we plan to do further research in the integration of semantic technologies and sensors with the sentient visor.

Sharing the vision of a web of data, commonly referred as the semantic web, we have provided the sentient visor with standard web mechanisms and compliance with some of current W3C standards and recommendations, but as we advance towards an IoT, standardization is also a concern for both the physical objects and software entities of the sentient visor. A privacy control over the smart objects is also needed to avoid people seeing sensitive information, for instance, in medical scenarios.

Furthermore, since the sentient visor is a browser intended to run on several devices, we need to test and evaluate its performance in different hardware configurations for analyzing the processing power and memory requirements for its usage, as well as the impact on the device's battery consumption.

We also acknowledge that our sentient visor approach has some limitations. For example, always carrying a device with the sentient visor could not be the best solution for all scenarios. People could be required to avoid the usage of cell phones or other camera enabled devices while in certain places. For these situations, it is more likely that public displays could be a better choice for displaying and sharing information.

However, we deem that a sentient visor can be better to browse the information space in the IoT considering the scenarios described in this work as well as in other similar situations of our daily lives.

ACKNOWLEDGMENT

Financial support for this project was provided by the Mexican Council for Science and Technology (CONACYT).

REFERENCES

[1] Atzori, L., Iera, A. y Morabito, G. "The Internet of Things: A Survey". Computer Networks, 54(15): 2787-2805 p. ISSN 1389-1286. 2010.

[2] Mankoff, J., Dey, A.K., Hsieh, G., Kientz, J., Lederer, S. and Ames, M., "Heuristic evaluation of ambient displays". In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Ft. Lauderdale, Florida, USA, April 05 - 10, 2003). CHI '03. ACM, New York, NY, 169-176. 2003.

[3] Kawsar, F., Fujinami, K., and Nakajima, T. "Augmenting everyday life with sentient artefacts". In Proceedings of the 2005 Joint Conference on Smart Objects and Ambient intelligence: innovative Context-Aware Services: Usages and Technologies (Grenoble, France, October 12 - 14, 2005). sOc-EUSAI '05, vol. 121. ACM, New York, NY, 141-146. 2005.

[4] Segura D., Favela Vara, J., and Tentori Espinoza, M. E., "Sentient displays in support of hospital work", 3rd Symposium of Ubiquitous Computing and Ambient Intelligence, SpringerLink Press, 2008.

[5] Avilés-López, E., Villanueva-Miranda, I., García-Macías, J. A., and Palafox-Maestre, L. E. "Taking care of our elders through augmented spaces". In Proceedings of the 2009 Latin American Web Congress (La-Web 2009) (November 09 - 11, 2009). LA-WEB. IEEE Computer Society, Washington, DC, 16-21. 2009.

[6] Avilés-López, E. and García-Macías, J. A." Building and employing cross-reality. Sentient visors for browsing the world". IEEE Pervasive Computing. 8(3): 57 p. 2009.

[7] García-Macías, J. A., Álvarez-Lozano, J., Estrada-Martínez, P., Avilés-López, E. "Browsing the internet of things with sentient visors". IEEE Computer. 44 (5), 46-52 p. May 2011.

[8] Dey, A. K., "Understanding and using context", Personal and Ubiquitous Computing, vol. 5, no. 1, pp. 4-7, Jan, 2001.

[9] Pulli, K., Chen, W., Gelfand, N., Grzeszczuk, R., Tico, M., Vedantham, R., Wang, X., y Xiong, Y. "Mobile visual computing". In Proceedings of the 2009 international Symposium on Ubiquitous Virtual Reality (July 08 - 11, 2009). ISUVR. IEEE Computer Society, Washington, DC, 3-6. 2009.

[10] Layar. [Online]. Available: http://www.layar.com (Last Access: May 18, 2011).

[11] Wikitude. [Online]. Available: http://www.wikitude.org (Last Access: May 18, 2011).

[12] Sekai Camera. [Online]. Available: http://sekaicamera.com (Last Access: May 18, 2011).

[13] ARToolworks. [Online]. Available: http://www.artoolworks.com/ (Last Access: May 18, 2011).

[14] Avilés-López, E. and García-Macías, J. A. "UbiSOA Dashboard: Integrating the Physical and Digital Domains through Mashups". HCI International 2009. Springer Berlin / Heidelberg 466-474 p. ISSN: 0302-9743. 2009.

[15] ZXing. [Online]. Available: http://code.google.com/p/zxing/ (Last Access: May 18, 2011).

[16] Bonjour. [Online]. Available: http://www.apple.com/es/support/bonjour/ (Last Access: May 18, 2011).

[17] Ontology-W3C. [Online]. Available: http://www.w3.org/standards/semanticweb/ontology (Last Access: May 18, 2011).

[18] Protégé. [Online]. Available: http://protege.stanford.edu (Last Access: May 18, 2011)

[19] Jena. [Online]. Available: http://jena.sourceforge.net (Last Access: May 18, 2011).

[20] CC/PP. [Online]. Available: http://www.w3.org/TR/CCPP-struct-vocab (Last Access: May 18, 2011).

[21] Pellet. [Online]. Available: http://www.clarkparsia.com/pellet (Last Access: May 18, 2011).

[22] Processing. [Online]. Available: http://processing.org/ (Last Access: May 18, 2011).

[23] Ajanki, A., Billinghurst, M., Jarvenpaa, T., Kandemir, M., Kaski, S., Koskela, M., Kurimo, M., Laaksonen, J., Puolamaki, K., Ruokolainen, T. and Tossavainen, T. "Contextual information access with augmented reality". IEEE International Workshop on Machine Learning for Signal Processing (MLSP), pp.95-100, Aug. 29 2010-Sept. 1 2010.

[24] Schmalstieg, D. and Reitmayr, G. "The world as a user interface: augmented reality for ubiquitous computing". In: Location Based Services and TeleCartography (eds. Georg Gartner, William Cartwright, Michael P. Peterson), pp. 369-382, Springer Berlin Heidelberg New York, ISBN 3-540-36727-6, 2007.

[25] Roduner, C. and Langheinrich, M. "BIT – A framework and architecture for providing digital services for physical products". In Proceedings of the Internet of Things (IOT), 2010, pp.1-8, Nov. 29 2010-Dec. 1 2010.

[26] Kindberg, T., Barton, J., Morgan, J., Becker, G., Caswell, D., Debaty, P., Gopal, G., Frid, M., Krishnan, V., Morris, H., Schettino, J., Serra, B., and Spasojevic, M. "People, places, things: web presence for the real world". Mob. Netw. Appl.7, 5 (Oct. 2002), pp. 365-376. 2002.