Agent Augmented Reality: Agents Integrate the Real World with Cyberspace

Katashi Nagao
Sony Computer Science Laboratory Inc.
3–14–13 Higashi-gotanda, Shinagawa–ku, Tokyo 141, Japan
E-mail: nagao@csl.sony.co.jp

Abstract

Agent augmented reality is proposed as a new research area that uses agent technologies for the augmentation of our real world environment by actively integrating information worlds. A special agent called a *real world agent* is introduced. A real world agent is a software agent that can support the user in performing tasks in real world environments, such as place-to-place location guidance, instruction in physical tasks, and the augmentation of human knowledge related to the physical environment. In order for the agent to achieve these tasks, the agent should be aware of the user's real world situation. The detection of real world situations is performed through the integration of various methods, including location-awareness using a global positioning system, object recognition through machine-recognizable IDs (barcodes, infrared rays, etc.), and the processing of visual/spoken inputs. IDs of static objects (e.g., doors, ceilings, walls, etc.) can also give clues to the location-awareness system. When a real world agent correctly deduces the real world situation and the intentions of the user, it can access information worlds (e.g., the Internet) in the same manner as other software agents. In this way, the real world and information worlds can be dynamically integrated. Real world agents can detect real world situations, interact with humans, and perform tasks in information space. The agents also work as facilitators of human-human communicaition, because they can exchange information on their users' current situation and the users can be mutually aware of their contexts.

1 Introduction

Recently, the big trend in human-computer interaction has been to bridge between computer-synthesized worlds and the real world. This research area is called *augmented reality* and has as its main theme the overlay of computer-synthesized images onto the user's real world view. Examples are the work of Bajura et al. [1] and Feiner et al. [5].

We extended the concept of augmented reality so that it covers interactive systems that can enhance the real world informationally. Such systems support real world human tasks by providing information such as descriptions of objects as they are viewed, navigational help in some areas, instructions for performing physical tasks, and so on. Augmented reality systems essentially require the ability to recognize real world objects and situations. There are several approaches for detecting real world objects/situations, approaches such as scene analysis by visual processing, marking with machine-recognizable IDs, detecting locations using the global positioning system (GPS), communicating with physically embedded computers, and so on.

If a situation is correctly recognized, the interpretation of human messages will become much easier, even if they are ill-formed. This type of interaction is called *situated interaction*. Situated interaction is the efficient interaction between a human and a situation-aware computer.

Using situation awareness, humans can naturally interact with computing systems using normal speech without having to be especially conscious of the systems' domains or regulations. In this case, language use will be more flexible and robust. By recognizing real world situations and knowing normal human behavior for these situations, the systems will be implicitly aware of humans' intentions and accurately predict future needs and actions.

On the other hand, people can and do physically move from one situation to another. When moving toward a situation, one will get information related to the new situation being encountered. This is an

intuitive way of carrying out the information seeking process. Walking through real world situations is a more natural way of retrieving information than searching within a complex information space.

We again extended the concept and functions of an augmented reality by introducing agent technology that is implemented as so-called 'software agents' such as Maxims [14] and Softbots [4]. We call this new concept *agent augmented reality* and introduce a special agent called a *real world agent*. A real world agent is a software agent that recognizes the user's real world situation, is customized to her preferences and interests, and performs tasks in the information worlds (e.g., the Internet). Real world agents can communicate with humans using natural language and with other software agents using an agent communication language such as KQML [6]. Situation awareness will help the agents to recognize human intentions by using context-related information from their surroundings.

In the rest of this chapter, we discuss in greater detail augmented reality and agent augmented reality, with some implemented prototypes.

2 Augmented Reality

There are some current, concrete examples of augmented reality systems. One of them is the car navigation system. Using GPS to stay constantly aware of a car's current position, the system gives the driver timely advice about the best route to follow to a destination. For example, as the car approaches an intersection, the system may say "turn left at the next corner." The system also provides some location-dependent information, such as information on good restaurants in the neighborhood, and so on. In a sense, this system augments the real world with digital maps and digital guidebooks.

Another example is the KABUKI (a type of traditional Japanese drama) guidance system. It uses a small radio-like device and a headphone. The system gives the audiencea better understanding using a real time narrative that describes the action, without disturbing overall appreciation of the KABUKI drama. The pace of the action dynamically changes with each performance and if the narration were to get ahead of, or drop behind the action, it would become meaningless, so synchronizing the narration with the action is very important and also very difficult. Currently, narrations are controlled manually, but it is possible for the system to be automated by using computers and real world situation awareness techniques.

In the rest of this section, we discuss the required functions of augmented reality systems.

2.1 Situation Awareness of the Real World

There are two major ways for the computer to achieve situation awareness. One is called *mobile computing* and the other *ubiquitous computing* [28].

Situation detection by mobile computing is classified into ID-based and location-based methods. ID-based methods (called ID-awareness) mark real world objects with machine-recognizable IDs (e.g., barcodes, infrared rays, radio waves). Recognition of objects can be extended to the recognition of situations. Suppose that there is an ID on every door in a building. When the user stands in front of a door, the mobile system detects the location by scanning the ID on the door and by processing the information related to this position may derive some understanding of what the user intends to do. Location-based methods (called location-awareness) include GPS, three-dimensional electromagnetic sensors, gyroscopic sensors, and so on. Spatial information is also a useful input for attaining situation awareness. In contrast to ID-awareness, location-awareness is more scalable, because it doesn't require that objects be tagged. However, when the location of physical objects changes, the system has no way of recognizing this movement and so will fail to identify and call them to the user's attention properly. Therefore, it would be better to apply a hybrid approach that uses both ID-awareness and location-awareness to complement each other.

Using ubiquitous computing, recognizing the human environment will become easier, because it proposes that very small computational devices (i.e., ubiquitous computers) be embedded and integrated into the physical environment in such a way that they operate smoothly and almost transparently. These devices are aware of their physical surroundings and when a human uses a physical device that contains ubiquitous computers or enters some area where physically-embedded computers are invoked to work, these computers are aware of the human's activities. From the viewpoint of reliability and cost-performance, ubiquitous computing does not compare well with mobile computing, since ubiquitous computers require very long battery lives and are significantly difficult to maintain. In addition, when ubiquitous computers are personalized to users, as in an active badge system [27], for example, all user

personal data is processed in the shared environment, while in mobile computing, each user's personal information can be encapsulated within their own machine. So, ubiquitous computing also experiences privacy problems.

2.2 Situated Interaction

We use the term 'situated interaction' to mean the interaction between humans and computers. This can be very efficient, because of the sharing of situations or sometimes incomprehensible if there is no mutual awareness of the situation. An example of situated interaction could happen in the KABUKI guidance system mentioned above. The dramatic narrative will be meaningless if the system misunderstands the user's situation, for example the current scene of the drama.

A real world situation includes the place where the human is, the time when an event occurs, living and non-living things that exist in the vicinity, and any physical action that is being performed (e.g., looking at something).

Using situation awareness, humans can naturally interact with computers without being especially conscious of the computers' domains or regulations. Situations and normal human behavior in these situations can be important clues allowing the computing systems to recognize human intentions and to predict what they will do next. Also the systems may be able to clarify human desires by accepting the input of information both vocally and/or by physical interaction.

As mentioned earlier, humans can move from one situation to another through physical action (e.g., walking). When moving towards a situation, the user can retrieve information related to the situation that is being confronted. This can be an intuitive information seeking process. Walking through real world situations is a more natural way to retrieval information than searching within complex information spaces. Situated interaction can be considered as matching retrieval cues to real world situations. For example, if a person wants to read a book, they naturally have the idea of going to a place where a bookshelf exists. This means that a situation that includes a bookshelf can be a retrieval cue related to searching for books.

2.3 Personalization

Personalization adapts a system to a specific user. Usually, this is done by registration of the user and preparing preference data with a user ID. The system attunes itself to the user with the preference data. A machine learning mechanism can work for dynamically adjusting the system for fine tuning itself to the user.

Some types of augmented reality systems always accompany with the users like *wearable computers* [24] and are customized to them. The system acquires the user's individual habits and preferences by observing the user's repetitive behavior and by asking the user personal information.

Personalization helps the system to implicitly recognize the user's intentions, determine the time to begin interactions, and to select information most salient to the user.

On the other hand, the system could also be personalized using psychophysiological signals such as electroencephalograms (EEG, brain waves), electrocardiograms (EKG, heart beats), or skin conductances (GSR, amount of sweating) [8]. Since these signals are very person-specific, the system could perform in ways that might not make sense to anyone except the user. The MIT Media Lab's group is pursuing this kind of system and calls their approach *affective computing* [19]. Biomedical sensors are used to recognize human mental/emotional status. Thus, the system can react to the user according to the mental situation. For example, if the user is impatient with the answer to a query, the system shortens the answer according to the degree of impatience and changes the order of information presentation based on the relevance to the context.

2.4 Augmentation of Human Memory

One of the interesting functions of augmented reality systems is the augmentation of human memory. The system stores summarized descriptions of the user's behavior in association with situations (including time) where the behavior occurred. A behavior description includes time, location (or ID), and related things (object IDs and/or human IDs). It may also contain visual/spoken information, if available.

Thus, human memory can be indirectly augmented by accumulating memory retrieval cues related to real world situations. Human memories consist of mixtures of real world situations and information that was accessed from those situations. Therefore, recognizing a real world situation can be a trigger

for extracting a memory partially matched with the situation and associating information related to the memory.

The context of an episodic memory provides lots of cues for remembrance. These cues include the physical environment of an event, who was there, what was happening at the same time, and what happened immediately before and afterwards [25]. This information both helps us recall events given a partial context, and to associate our current environment with past experiences that might be related.

Augmented reality gives the opportunity to bring new sensors and context-dependent information processing into everyday life, such that these pieces of information on the physical context can be used by the system to supplement our human memory.

Rhodes's Remembrance Agent [23] is a memory supplement tool that continuously "watches over the shoulder" of the user of a wearable computer and displays one-line summaries of previous electronic mails, online papers, and other text information that might be relevant to the user's current context. These summaries are listed on the bottom few lines of a heads-up display, so the user can read the information with a quick glance. To retrieve the whole text described in a summary line, the user hits a quick chord on a chording keyboard.

3 Examples of Augmented Reality Systems

Following are some experimental systems that can be considered to have some of the functions of the augmented reality systems already mentioned.

Digital Desk Wellner's Digital Desk [29] is an augmented physical desk that has cameras that record a user's activity on the desktop. This computer vision enables the system to respond to such movements as pointing and sketching. A projector displays computer synthesized images on the physical desktop. For example, a projector can create the image of a form, and a visual recognition system can identify the regions of the form being filled and the marks being made from the user's finger movements.

Forget-me-not Lamming's Forget-me-not [12] is a personal information management system. It is based on Weiser's ubiquitous computing [28] noted earlier. The ParcTab portable device uses infrared signals to continously send the user's ID to the ubiquitous computing environment. ParcTab can communicate with other ParcTabs through infrared. If someone wants to pass an electronic document to someone else, all they have to do is pass the document's ID between the ParcTabs. The system also memorizes humans' activities chronologically. The main function of this system is to handle information queries concerning daily activities (places visited, people met, documents submitted, etc.) by using time as a retrieval key. This system thus, extends human memories indirectly.

Chameleon Fitzmaurice's Chameleon [7] is a spatially-aware palmtop computer. It shows situated information according to its spatial location and orientation on a small LCD (Liquid Crystal Display) screen. An example application of this system is called *Active Map*. There is a paper copy of a world atlas on the wall. When a user puts the palmtop device close to a certain region on the map, the system screen shows geographical information about that region. However, this system is not as robust as others, because it can only respond to situation from its own location; if physical objects change location, it cannot adapt to or even recognize this movement.

NaviCam and Ubiquitous Talker Rekimoto's NaviCam (Navigation Camera) [20, 21] is a handheld system consisting of a CCD (Charge Coupled Device) camera for recognizing color-bar ID codes on real world objects, and an LCD screen which reproduces an image of what the user is looking, as if through transparent glass. A basic principle of NaviCam is the *magnifying glass metaphor*. An object, recognized by its ID tag, has some electronic information added to it on the screen, magnifying it not visually, but informationally.

Nagao and Rekimoto's Ubiquitous Talker [17] is an extension of NaviCam that is integrated with a spoken dialogue system.

In order to make natural language processing, especially spoken language processing, more practical, we must restrict or constrain the domains, contexts, or tasks, since it requires, potentially, an unrealistically broad search space on the phonetic and linguistic level. Various sorts of nonverbal information can play a role in fixing the situational context, which is useful in restricting the hypothesis space constructed

during language processing and the interpretation of utterances becomes much easier, even if the utterances are ill-formed. In other words, the correct interpretation of natural language utterances essentially requires the integration of both linguistic and non-linguistic contexts. Especially, comprehending multimodal dialogues is not possible without some acknowledgment of the role of the non-linguistic context. Delving into the above, results in knowledge bases that are very efficient and robust. We have, therefore, introduced robust natural language processing into a system of augmented reality.

NaviCam and the Ubiquitous Talker augment reality with some additional information related to a recognized object/situation. Such information is conveyed using the LCD and voice (in the case of the Ubiquitous Talker). The Ubiquitous Talker accepts and interprets user voice requests and questions. The user may feel as if they are talking with the object itself through the system.

4 Agent Augmented Reality

Integrating the augmented reality and agent technologies has created a new research field called *agent augmented reality*. This field requires a special agent that recognizes real world situations, moves around in information worlds, searches for information related to the intentions of the user, communicates with humans and other agents, and performs, on behalf of the user, some tasks in information space. We call such an agent a *real world agent*.

The real world agent is a kind of software agent that can support the user's tasks in a real world environment, such as location guidance from place-to-place, instruction in physical tasks, and the enhancement of human knowledge related to the physical environment. After the real world agent detects a real world situation and the intentions of the user, it can access information worlds (e.g., the Internet) like other software agents. Thus, it can dynamically integrate the real and information worlds.

One of the most important problems for software agents is clarifying the user's requests. Communication between agents and humans should be more flexible and robust. Recent advances in multimodal interface techniques must be introduced into the human-agent interaction. One direction research has taken is anthropomorphic agents. Nagao and Takeuchi [18] have developed an agent that has a computer-synthesized anthropomorphic appearance and behavior. It can communicate with humans using voice, facial expressions, and head movements. However, this approach will require continued research utilizing the fields of psychology and sociology, because humans' reaction to and acceptance of such interfaces is a highly delicate matter.

On the other hand, our real world agent does not need to have an anthropomorphic appearance, because it accompanies the user and stays aware of the surroundings and activities of the user. Therefore, the agent has a rich source of nonverbal clues with which to clarify the user's requests and to interpret intentions. The efficiency of communication is affected by situated information, which can also be a cause of ambiguity in messages. Situation awareness, a main function of real world agents, can thus play a role in 'disambiguation.'

Next, we will discuss some other important functions of real world agents.

4.1 Situated Conversation

Conversation is an important function of real world agents. Speech is usually based, not only on linguistic contexts, but also on non-linguistic contexts relating to a real world situation. This is called *situated conversation*. In situated conversation, the topic and focus of speech depend on the situation and are easily recognizable when the participants are aware of their surroundings. Our agents can handle situated conversation by virtue of the basic properties of augmented reality systems.

Knowing the user's intention is necessary for natural human-agent interaction. Although a real world situation would provide just a clue about the intention, being able to integrate the non-linguistic context introduced with that situation, with the linguistic context constructed by dialogue processing, is an important step forward.

In general, a user's intentions are abductively inferred using a plan library [16]. A plan library is represented as an event network whose nodes are events with their preconditions and effects, and the links are is-a/is-part-of relationships [11].

For example, in a situation where a person stands in front of a bookshelf, for example a bookshelf on computer science, the situation motivates the person to search for a book on computer science, read it, and study it. Therefore, if the dialogue system is made aware of the situation through recognizing the bookshelf's ID, the plan library shown in Figure 1 is introduced and used for further plan inference.

In this figure, the upward-pointing thick arrows correspond to is-a (a-kind-of) relationships, while the downward-pointing thin arrows indicate has-a (part-of) relationships.

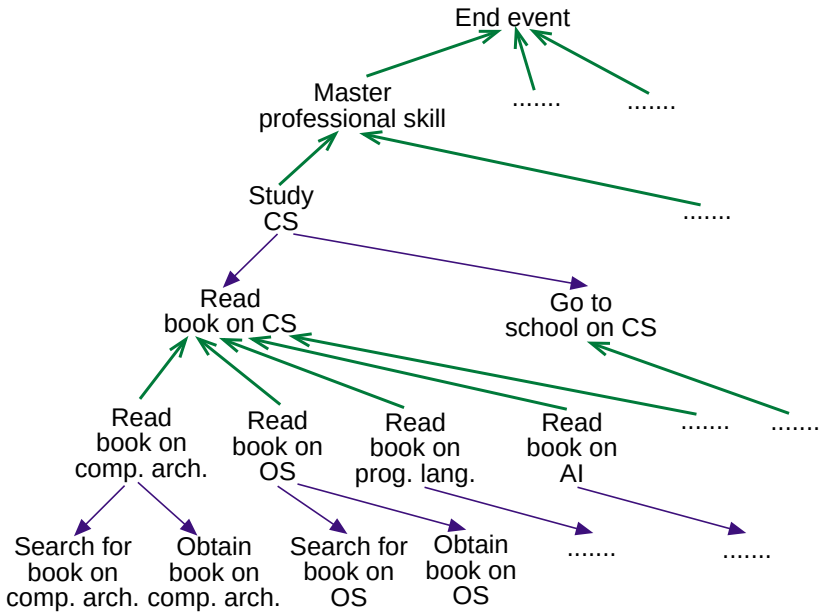

Figure 1: Plan Library `comp-sci-bookshelf-plan`

Introducing and focusing a specific plan library makes plan recognition easier and more feasible.

Another connection between linguistic and non-linguistic contexts is *deictic centers* [31] that are possible referents of deictic expressions. The object and the location in a non-linguistic context can be current deictic centers. Also graphical and textual information on the screen includes deictic centers. Preferences on possible deictic centers as a referent are determined based on the coherence of a dialogue as in the case of anaphora/ellipsis resolution in a linguistic context [26].

4.2 Learning and Adaptation

Similar to the work of Maes and her colleagues [14], our agent should also have mechanisms for learning and adapting. In this case, to learn is to acquire the user's habits, preferences, and interests and to be able to adjust the parameters of probabilities, utilities, and thresholds like Maes's *tell-me* and *do-it* thresholds. Thus, agent learning plays a role of user personalization that can determine the system's behavior considering the user's preferences. This function is useful when determining what tasks are to be performed and at what time to begin them, as well as the contents of what information is to be presented and the timing of the presentation. For example, while walking in a town, one suddenly is consumed by desire for a hamburger and expresses this craving, resulting in the agent searching for the most favored (according to the user's preferences) hamburger shop in that locale (this is done by using the Geographic WWW Server, which will be described later) and placing the user's usual order with that shop.

To adapt is to change behavior according to environmental variables, such as the user's attitude (hurrying or not), the location, time, availability of network communication media (e.g., infrared signals, digital cellular, etc.). Much research on agent adaptation is being done in robotic agents. In this case, most environmental variables are obtained from the physical world through electromagnetic, infrared, ultraviolet, tactile, and visual sensors. Our real world agent must consider these variables, as well as the environmental variables related to the information world.

Agent learning can also have a role in the augmentation of human memory by maintaining a record of the user's daily activities. It accepts user inquiries about places visited, people met, and so on. Like the Remembrance Agent and Forget-me-not mentioned earlier, the real world agent acquires several memory cues from the user's inputs (texts and speech) and signals from the physical environment (GPS radio waves, infrared beacons, object/person IDs).

4.3 Collaboration

A real world agent collaborates with other agents as in a *multiagent system*. The agent communicates with such Internet agents as Softbots [4] when searching for information on the Internet. It also communicates with other people's real world agents to gain knowledge about the humans whom the user is talking with or planning to meet.

Collaboration among real world agents also contributes to group gatherings. In this case, the agents would inform each other of their users' current locations, collaborate in determining an appropriate meeting point, and then navigate the users to the rendezvous. Since the agents are always aware of the users' real world situations, they can work on, not only meeting scheduling, but also coordinating physical meetings.

Multiagent collaboration also works for information filtering. We call this *agent social filtering*. The agents are personalized for their specific users and have knowledge of the users' interests. So, the agent can automatically search for information that will be required by the user in the near future, then filter it based on other users' recommendations that are sharable and accessible to the agent.

Resnick and Varian's Recommender System is a prototype system of collaborative filtering based on recommendation [22]. The system asks the user's preferences using some prepared questions and, then match a list of recommendations to the preferences. The major drawback to this kind of system is that it is only effective when the each user's preferences are broadly distributed over various interest areas.

We are designing and implementing an agent social filtering based on the Recommender System with a credit assignment capability via multiagent communication. In this system, real world agents produce recommendations through interaction with the user. Information-seeking agents select the most appropriate recommendation that is produced by the agent of the highest credit.

5 An Architecture for Agent Augmented Reality

An architecture for the real world agents must consider the following issues:

1. Connection between the real world and information worlds
2. Conversation with the user
3. Communication with other agents
4. Learning for personalization
5. Personalized information retrieval and filtering
6. Protection of user privacy

As mentioned before, recognition of the real world situations can be done using several methods for ID-awareness and location-awareness. We prepared associations between IDs/locations and online resource identifiers in a uniform format similar to the Uniform Resource Locators (URLs) on the World Wide Web (WWW). An example of an association between physical locations and URLs is shown by the Geographic WWW Server described in the next section. Real world agents retrieve such associations through a wireless network.

A human-agent conversation mechanism is designed to integrate linguistic and non-linguistic contexts. The situation awareness module closely interacts with the conversation module. As mentioned before, situated conversation will be more flexible and robust than ordinary conversation. Thus, the potential for cognitive overload of humans during communication will be greatly reduced. Also, computational resources for spoken language processing will be kept to a tractable size by changing to the appropriate phonetic/linguistic dictionaries and knowledge bases according to current situational needs.

Communication between agents is based on an extension of the Telescript technology [30]. Interagent communication is done in special computational fields called *places*. In our architecture, there is a *public place* and a *personal place*. As their names indicate, agents in the personal place can access any personal information resources (e.g., ID number of a credit card). Agents in the public place cannot access such personal information, but can use some public information resources (e.g., product catalogs). Interagent communication would occur when the agent obtains an online resource identifier (e.g., URL) from a recognized object/location and the identifier is related to a place and its inhabitant agent(s). In this case, the communication occurs at the public place, as shown in Figure 2. Interagent communication

would also occur whenever agents visit someone's personal place and contact the real world agent of that person. For example, when a shop's salesman agent recognizes guests who are entering the store (this is done by receiving broadcasts detailing the guests' personal IDs from their real world agents), the shop agent will try to customize his sales points using the guests' personal information and their conversation will be mediated by their real world agents. This is illustrated in Figure 3.

Figure 2 shows an example of interagent communication at the public place and Figure 3 demonstrates the personal place.

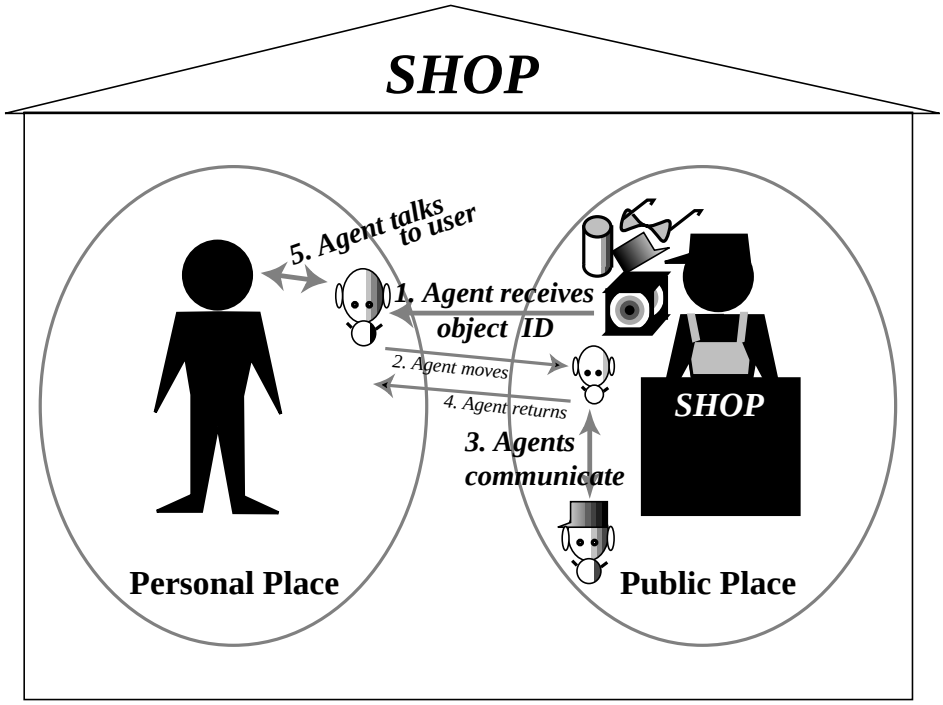

Figure 2: Interagent Communication at the Public Place

Learning for personalization requires data acquisition and model modification processes. The agent memorizes user's behavior related to real world situations, which also works as an input of the data acquisition process. The model modification process uses an initial user model and a statistical learning mechanism. The initial user model will be constructed based on the user profile mentioned below. While the statistical learning is automatic and unsupervised. However, a user can give clues to the model modification process using other mechanisms.

The agent also performs certain actions that will help the user to act in the real world. The user can give some emotional feedback to the agent via voice, hand, or head action. These are regarded as reinforcement signals for the agent, and we employ a reinforcement learning algorithm for modification of a user model. Presently, it is still hard to teach agents the difference between rewards and penalties. Fortunately, there are some hints available to facilitate this decision making process, such as using prosodic features including angry/pleased voice tones, or behavioral features of body movements.

The main task of the agent is the retrieval and filtering of information that is related to the user's real world environment and preferences. There are some techniques for customizing the retrieval and filtering. One is a user profile that includes some keywords and category markers of frequently accessed information (e.g., Web pages, online documents, etc.). Through the utilization of learning mechanisms, the agent acquires the user's model of interests and uses it for information retrieval and filtering.

When the agent uses personal user data for information processing, it may share this information with other agents for some tasks. However, the agent must exercise discretion concerning certain information that is used for identifying individuals, especially when the agent is in public places. However, some agents may have a legitimate need to visit other agent's personal place and retrieve some personal data. To prevent privacy problems, our architecture prohibits any agents who visit other's personal places from leaving. That is, when agents visit personal places, their behavior is restricted; they can only partially access personal information and provide messages; they cannot retrieve responses. If any response is

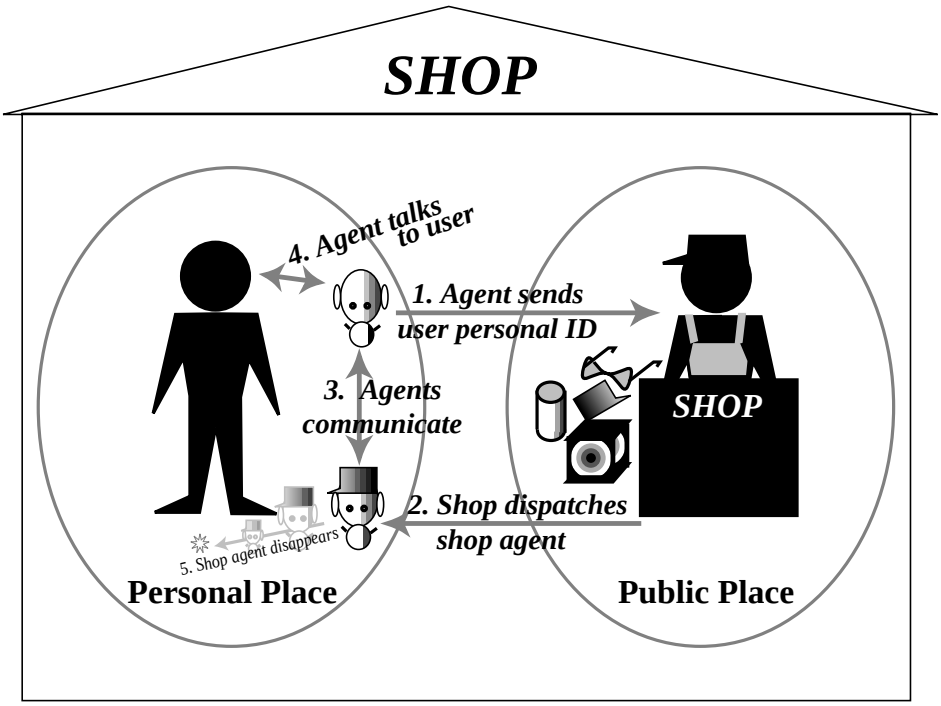

Figure 3: Interagent Communication at the Personal Place

needed, the visiting agent must send a request to the inhabitant agent of the personal place first. There are some cryptographic techniques used in data communication, such as public key cryptosystems. For stronger protection of privacy, interagent communication will require security architecture based on such cryptographic techniques.

6 An Implementation of Agent Augmented Reality

As mentioned in the previous section, the architecture of agent augmented reality is based on the design principles of Telescript technology [30]. In Telescript technology, agents are implemented as mobile (migrate-able) programs that are interpreted in particular "places." Places can perform the following:

- Places can identify visiting agents using their authorities.
- Places can interpret programmed behavior of visiting agents.
- Places can give the right to access resources to appropriate agents.
- Places can mediate communication among visiting agents.
- Places can protect security of resources from agents.

To implement mobile agents that can move around cyberspace such as the Internet, we employed a Java-based mobile agent programming system. While Telescript technology provides its own programming environment for mobile agents, it is not easy to use for extending language specification. A Java-based agent programming language, on the other hand, such as APSL (Agent Programming System and Language) [10], allow us to easily extend the language so as to include personalization, cryptography, and situation awareness techniques.

APSL operation environment assumes that agent programs are running on one of three types of hosts: a Customer Host, an Agency Host, or a Service Host. A Customer Host is a user's mobile host that is connected to the network using some type of wireless data communication. Obviously, there are times that it is disconnected from the network and its computational performance degrades accordingly. Therefore, *remote programming* by means of mobile agents is very important. A user of the Customer Host will access an Agency Host where the agents have been developed and stored, and download an agent customization environment including templates of agent programs and a user interface. Through

the user interface, the user is able to supply some parameters to the agent that can be used during its operation. The user sends the mobile agent to one of Service Hosts where it can obtain various services. When it finished, the Customer Host will be disconnected. The mobile agent visits several Service Hosts and reaches the Agency Host when it has finished all its jobs. When the Customer Host is reconnected to the network, it accesses the Agency Host again and the mobile agent returns to the Customer Host from an agent repository environment running on the Agency Host. Finally, the mobile agent gives any work result to the user. Figure 4 shows the overall process of a mobile agent operation in APSL.

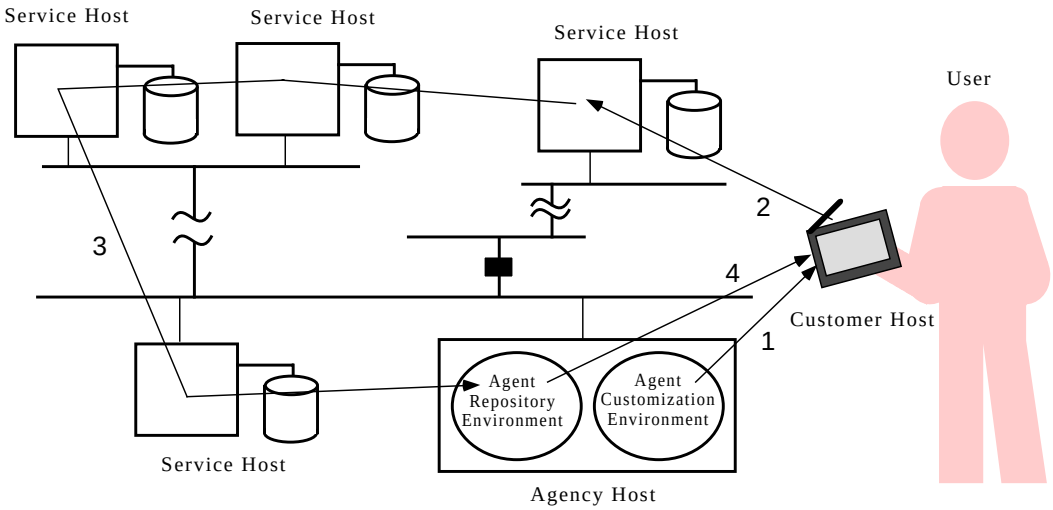

Figure 4: APSL Operation Environment

In APSL, places are implemented as agent programs called *place agents*, however they can not move around the network, in contrast to mobile agents. Place agents provide a variety of services to mobile agents by communicating with them. One of advantages of this implementation of places as agents is the protection of secured data embedded in the mobile agents, because place agents can access mobile agent information not by interpreting programs, but by exchanging messages generated by the mobile agents. Place agents also work as mediators that enable several mobile agents to exchange their messages.

APSL also provides a *communication package* that supports communication between a mobile agent and a place agent. The communication package implements a communication protocol according to available resources. Making implementation of communication protocols independent from agent programs is very important since a protocol should be shared by communicators, and implementation depends on the resources that communicators can provide. In APSL, place-mobile agent communication is initiated by supplying the communication package to the recipient agent. The list of resources and their access permissions are checked before initiating each communication, then the initiating agent selects an appropriate communication package. Agents must use a communication package for every message exchange.

Situation awareness techniques are implemented in place agents. The place agents are categorized as personal place agents and public place agents. The differences between these agents are the same as the differences between personal and public places, which were described in the previous section. Place agents accept user/object/location IDs from the physical environment and relate these IDs with the appropriate information resources such as customer databases, product catalogs, local area maps, Web pages, and so on. If information resources are located at remote hosts, then a mobile agent is instantiated and transferred to the remote hosts. The instantiation is done by a parameter setting to a template of agents. So, place agents have an ability not only to communicate with mobile agents but also to create them. The mobile agents access the information resources that are related with real world situations recognized by the place agents. After communicating with agents at remote hosts, the mobile agents return to their original host and report to the place agents.

Personalization techniques are also implemented in place agents. Personal place agents can access user profile data. They also have a model of user interests and a statistical learning mechanism for user model modification. A user model is described using weighted feature vectors, decision trees, Bayesian networks [3], Gibbs's distributions [2], and so on. We have to select an appropriate representation, one suitable to a particular task and domain, since a more complex model requires a more time-consuming

learning algorithm.

7 Agent Augmented Reality Applications

We have been developing the following experimental systems based on the concept of agent augmented reality.

7.1 ShopNavi: A Shopping Assistant

ShopNavi is a system for commercial information guidance and navigational help in stores. ShopNavi is also functional as a real world agent. Since shopping is a very personal affair, personalization of each user's own ShopNavi is essential. The ShopNavi unit consists of a wireless tag reader (this is similar to a barcode scanner) for object identification, three-dimensional electromagnetic sensors for the recognition of head orientation and hand position, an infrared receiver for location detection, a portable LCD for information presentation, a wireless data communication facility, and a spoken dialogue system.

Figure 5 is a snapshot of the ShopNavi in use, and Figure 6 shows an example of its screen.

Figure 5: ShopNavi in Use

Figure 6: Example of the ShopNavi Screen

Figure 7 shows the system configuration of ShopNavi.

7.1.1 Situation Awareness

Since GPS does not operate inside buildings, ShopNavi uses infrared IDs (location beacons) for location-awareness. These IDs are continuously transmitted from critical points around the store, for example,

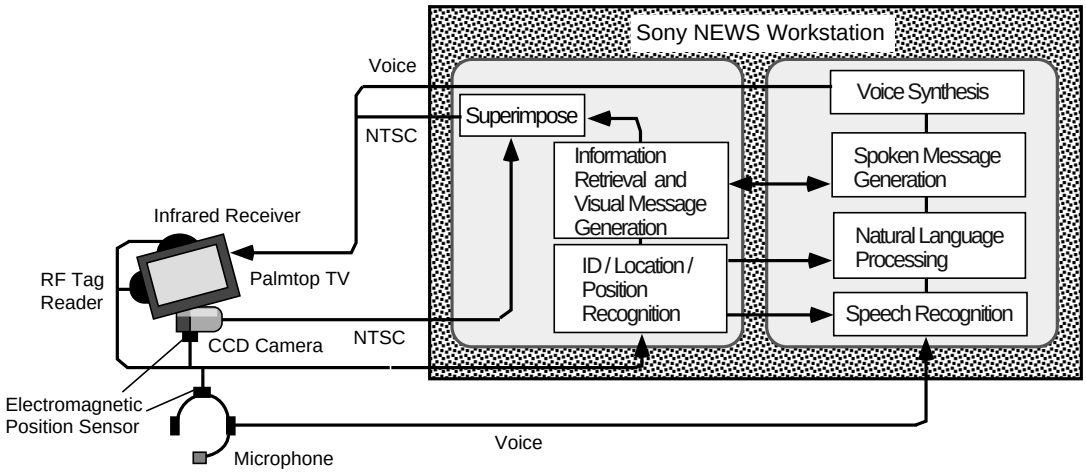

Figure 7: ShopNavi System Configuration

the entrances, counters, sales floors, and so on.

For object recognition, the ShopNavi handheld module is provided with an RF (radio frequency) tag sensor that transmits radio waves to special batteryless wireless tags (RF tags), and receives their ID-coded responses.

The ShopNavi module also uses electromagnetic sensors for recognition of hand position and head orientation. These sensory data are used to recognize the user's viewpoints and focus of attention.

7.1.2 Visual Information Presentation

The system generates a synthesized image by superimposing visual messages related to the real world situation detected from the object and location IDs. The output image is shown on the LCD screen. Image processing is done by software except for the conversion between NTSC video signals and bitmap digital images. The output image is updated at a rate of 10 frames per second.

7.1.3 Spoken Dialogue Processing

The speech dialogue subsystem works as follows. First, a voice input is acoustically analyzed by a built-in sound processing board. Then, a speech recognition module is invoked to output word sequences that have been assigned higher scores by a probabilistic phoneme model and phonetic dictionaries that are changed according to the situation.

These word sequences are syntactically and semantically analyzed and disambiguated by applying a relatively loose grammar and restricted domain knowledge. Using a semantic representation of the input utterance, a plan recognition module extracts the speaker's intention. For example, from the utterance, "I want to find SUKIYAKI beef," at a shop entrance, the module interprets the speaker's intention as "The speaker wants to get information about SUKIYAKI beef (i.e., the place it can be obtained.)."

Once the system determines the speaker's intention, a response generation module is invoked. This generates a response to satisfy the speaker's request. Finally, the system's response is outputted using a voice synthesis module. This subsystem also sends a message to the visual message generator about what graphical and/or textual information should be displayed with the voice response.

Continuous speech inputs are accepted without special hardware. To obtain a high level of accuracy, context-dependent phonetic hidden Markov models are used to construct phoneme-level hypotheses [9]. The speech recognizer outputs N-best word-level hypotheses. As mentioned above, an appropriate phonetic dictionary is dynamically selected by considering the speaker's real world situation. Therefore, perplexities and hypothetical spaces are always maintained in tractable sizes without more advanced (and high-cost) speech technologies.

The semantic analyzer handles ambiguities in syntactic structures and generates a semantic representation of the utterance. We applied a preferential constraint satisfaction technique for disambiguation and semantic analysis [15]. For example, the following semantic representation is constructed from the utterance, "I want to find some SUKIYAKI beef," at the food store entrance.

```
(*want-1
```

```
;; *want-1 indicates that it is an instance
;; of frame *want.
   (:agent *i-1)
   (:theme (*find-1
              (:agent *i-1)
              (:theme *SUKIYAKI-beef-1)))
   (:situation *food-store-entrance-1)
   ;; :situation is added by the situation
   ;; awareness module.
)
```

The plan recognition module determines the speaker's intention by constructing her belief model and dynamically adjusting and expanding the model as the conversation progresses [16]. We use a plan library that is selected according to the situation. For the above example, `food-store-entrance-plan` is selected. Then, the recognized intention will be as follows[1].

```
(*intend-to-know-1
;; *intend-to-know comes from (*want
;; (:theme *find)) in library-front-plan.
   (:agent *speaker-1)
   ;; *i-1 is replaced with *speaker-1.
   (:theme (*location-of-corner-1
              (:source *food-store-entrance-1)))
              (:item *SUKIYAKI-beef-1)))
   ;; *location-of-corner is inserted by
   ;; means of plan inference.
)
```

The spoken message generation module generates a response by using a domain-dependent knowledge base and text templates (typical patterns of utterances). It selects the appropriate templates and combines them to construct a response that satisfies the speaker's request.

The system has the function of *situated conversation* mentioned earlier that combines linguistic and non-linguistic contexts. When the system detects a real world situation, it performs not only a selection of knowledge sources (e.g., phonetic/linguistic dictionaries), but also the introduction of a non-linguistic context. A non-linguistic context includes the object at which a user is currently looking, the location where the user currently is, graphical information displayed on the screen, and chronological relations of situation shifts. On the other hand, a linguistic context involves the semantic contents of utterances, displayed textual information, and the inferred beliefs and intentions (plans and goals) of the user.

In the ShopNavi system, plan inference is initially triggered by introducing a new non-linguistic context, since the motivation of our situated interaction is closely related to the physical actions that accompany a new situation. For example, in a situation where a person is standing in front of a shelf, for example a shelf of drinks, the situation will motivate the person to search for a drink, pick it up, and buy it. When the system detects the current situation from the shelf ID, a situation-dependent plan library is introduced and used for inference.

7.1.4 Personalization of the ShopNavi System

The ShopNavi portable module stores the user's personal information, such as today's menu plan, a budget, past shopping records, favorite items, and shopping habits.

When a user enters a store equipped with a ShopNavi, the system can supply up-to-date information on products in the shop. This is done by giving special consideration to any differences between past and present visits, and the particular shopping requirements of that day. Next, after a preference order has been established, the user is navigated around the different sales areas to the desired items. If necessary, the user can discuss things with the ShopNavi. The system recognizes items by scanning their tags and then prepares the appropriate resources for a spoken dialogue with the user. In this case, the system has to deal with a situated conversation in which most of the contextual information is conveyed implicitly. The system should be aware of the surrounding situations, including the user

[1] Actually, the intention may have several candidates that are assigned numerical preference values.

looking at an object, locations, hand gestures, head orientation, and so on. Furthermore, the system also memorizes the user's activities, for reference and utilization on future shopping trips. This mechanism can also record mnemonic triggers, acting as the user's back up memory. This is an important function of augmented reality systems.

The agent that inhabits the ShopNavi has the task of searching for information in a network that connects the shops, suppliers, and product manufacturers. This allows the user to make queries for information on an item's place of origin, the identity of its manufacturer, and how it was made. The ShopNavi manages the user's personal information, the shop's public information, and information on the network between the stores and their surroundings.

7.2 WalkNavi: An Interactive Tourist Guide

WalkNavi is a location-aware interactive navigation guidance system. It detects the user's current location by using GPS and infrared IDs, recognizes the user's intentions from voice inputs, retrieves location-related information from the WWW, and finally, provides any necessary navigational help or guidance information. In order to relate real world locations with URLs, WalkNavi accesses the Geographic WWW Server that has a network-accessible database of latitude/longitude information and URL links connected to location-related WWW pages. So, WalkNavi is a networked navigation system for pedestrians that uses digital maps/guidebooks.

WalkNavi gives route navigation advice to a desired destination by showing photographs of landmarks that can be used for navigation. It also provides information on notable places and buildings along the way.

WalkNavi is also provided with agent-oriented functionality. It uses an agent to gain access to on-line services, such as reservations at restaurants and electronic ordering of goods at shops through a wireless network. Using situation awareness, the agent will be able to recognize the user's intention implicitly.

Figure 8 is a snapshot of the WalkNavi in use, and Figure 9 shows an example of its screen.

Figure 8: WalkNavi in Use

Conversational interaction with the agent is also an important aspect of the real world agent. The WalkNavi module is integrated with a spoken dialogue system.

Figure 10 shows the system configuration of WalkNavi.

The system consists of the following three basic components, plus the Geographic WWW Server.

7.2.1 Location Awareness System

As stated, location-awareness is achieved using GPS and infrared IDs. GPS gives the system geographical information, including the current latitude and longitude. Locational information from GPS is purposely designed to be accurate to only about 100 meters, but additional information from nearby landmarks, given by the location IDs and/or human voice input helps the system to refine the current position estimate.

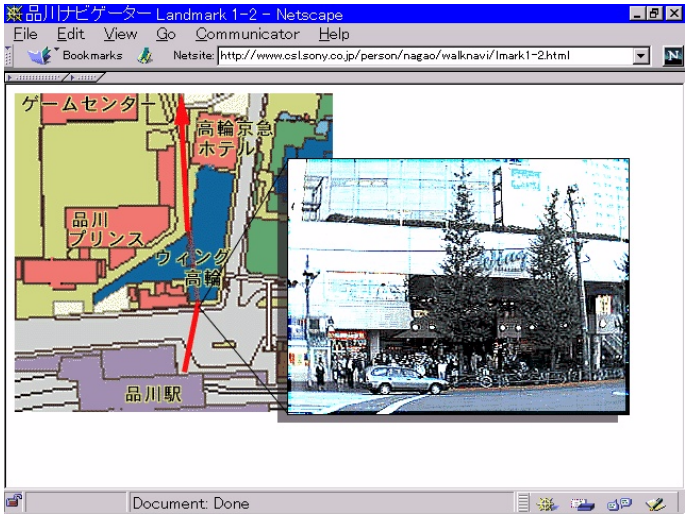

Figure 9: Example of the WalkNavi Screen

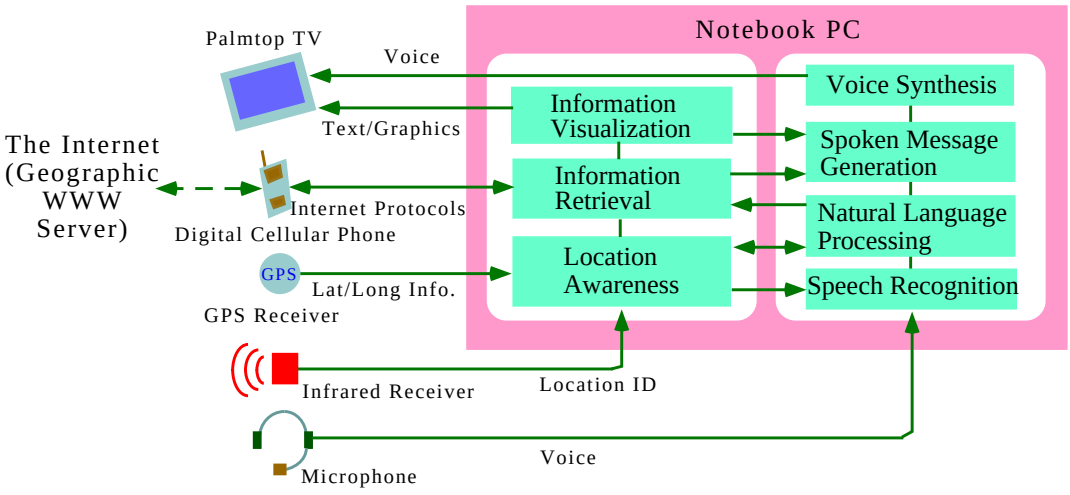

Figure 10: WalkNavi System Configuration

7.2.2 Spoken Dialogue System

Voice is an essential input modality for mobile computers, because it is difficult for someone walking to keep attention on a computer and spoken language reduces the cognitive load of interaction with the computer. The WalkNavi module implements a mechanism of situated conversation. It integrates linguistic and non-linguistic contexts and manages system resources in accordance with real world situations.

7.2.3 Location-Aware Mobile WWW Browser

The outputs of the system are presented through a WWW browser, because we use the WWW as a major information resource. The WWW includes location maps, related information (HTML (Hypertext Markup Language) texts, photographs, and sounds). The mobile Web browser accesses the Geographic WWW Server (mentioned below) via a wireless network using a digital cellular phone. Then, it retrieves location-related Web pages, selects the most appropriate one by using calegorical indices, and shows them on a portable LCD (palmtop monitor screen).

Figure 11 shows an example of a browser screen.

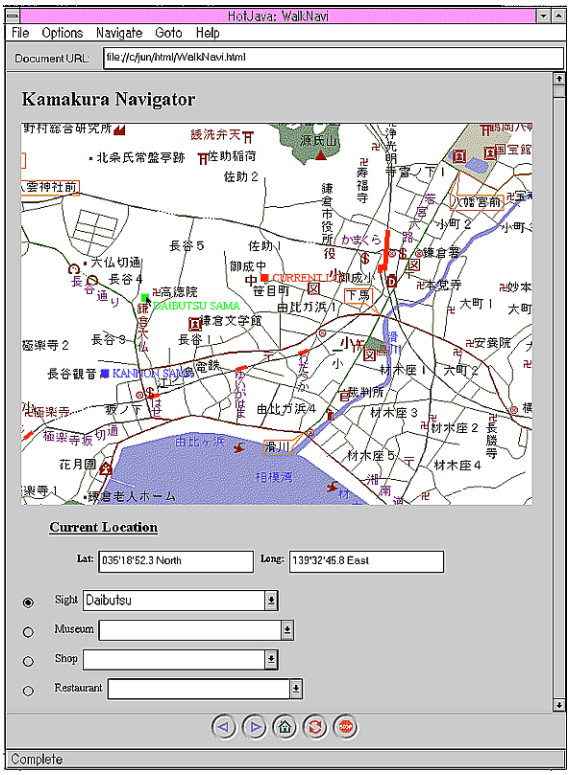

Figure 11: Example of the WWW Browser in Action on the WalkNavi

7.2.4 Geographic WWW Server

Our Geographic WWW Server relates latitudes/longitudes to physical addresses with URLs. The mobile Web browser accesses the server and retrieves URLs related to the current location through a wireless network. When a place's URL is registered with the server, the name of the place, its latitude/longitude, its category (e.g., scenic view, restaurant, etc.), its physical (snail-mail) address, and any additional comments are all input. When one accesses the server, the URLs are obtained by querying the server's URL using the parameters of latitude, longitude, distance, and category.

An example of the Geographic WWW Server's Web page screen is shown in Figure 12.

The main characteristics of the WWW are its openness and the ability for people to freely add their own information. Since WalkNavi uses the WWW as its knowledge resource, the system can handle time-sensitive information such as the announcements of exhibitions. Our geographic server allows users access to information related to their current location, e.g., neighborhood shops and restaurants.

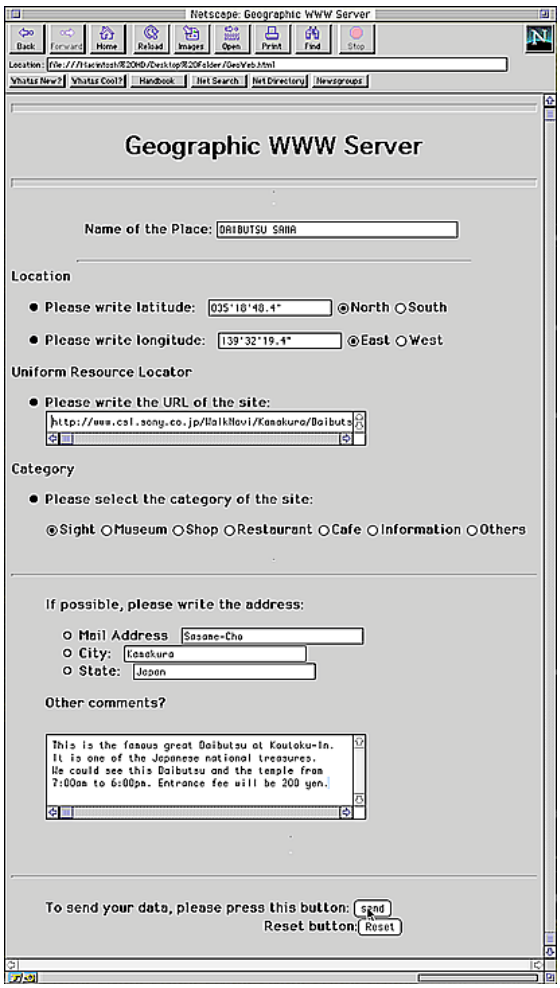

Figure 12: Example of the Geographic WWW Server Screen

Location can also be used for information filtering. Since a query by keywords, without some conditional restrictions, will often result in a great deal of useless information, locational restriction (i.e., a radius of X meters from the current position) can implicitly reduce the number of query candidates and make retrieval more efficient and intuitive.

There were some obstacles to be overcome in scaling up this system. One problem was the use of wireless network communication. Current digital cellular phones only support a maximum communication speed of 9600bps. In addition, the common TCP/IP protocol has too much overhead when used in unreliable (easily disconnected) networks. Consequently, we gave up mobile network computing during the movement phases (walking, etc.) and used our prototype system to download most resources whenever the user was standing still, before continuing to the next point.

Another problem was location-awareness accuracy. Standard GPS is only accurate within a radius of 100 meters, so it was difficult to precisely determine the user's current location in the information world (a digital map). In the near future, some GPS correction methods (e.g., differential GPS) will be applied to make GPS-based location-awareness more feasible.

7.3 HyperCampus: A Location-Aware Personalized Campus Navigation System

Our third example is a kind of combination of above two systems. The task is a guidance system on a university campus.

The HyperCampus system determines a user's location using GPS and infrared beacons, and provides user-specific, context-dependent information on campus and university activities heighten the user's understanding and interest in the university.

Figure 13 shows a snapshot of the HyperCampus system in use.

Figure 13: HyperCampus in Use

7.3.1 Contents of the HyperCampus System

Information contents, including a user registration form, are created on the WWW using HTML, Java, and JavaScript. Personal information is managed by cookies. Typical contents are:

1. Campus area information
   Campus map, building locations, information on outdoor services, etc.

2. Architectural information
   Outlook images of buildings, history of construction, information on events related to buildings, etc.

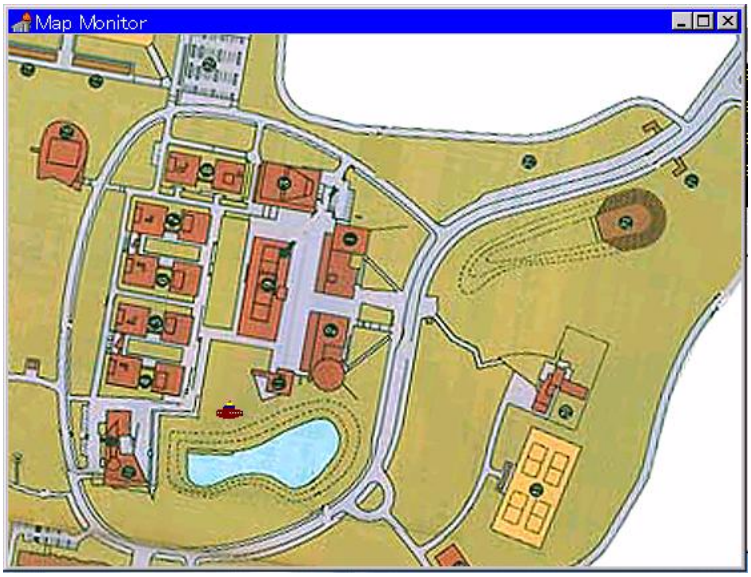

Figure 14: Map and Position Indicator

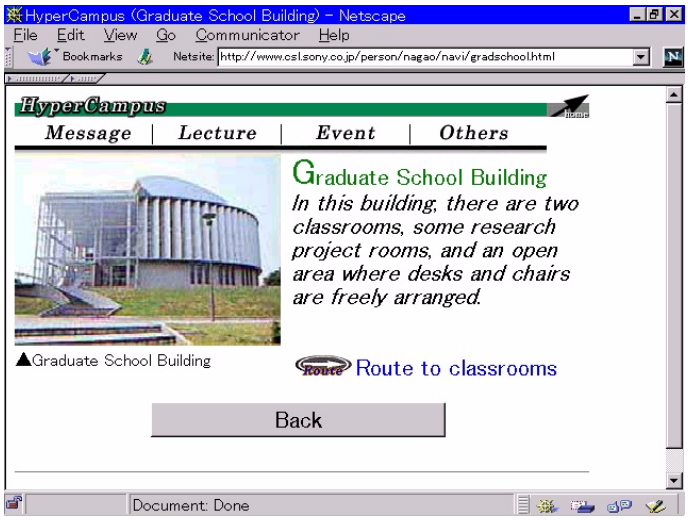

Figure 15: Information Shown on the Screen

3. Indoor information
   Floor plans, laboratory information, room information, information on indoor activities, etc.

4. Class information
   Timetable of classes, lecture room information, etc.

5. University information
   Other university related information

6. User behavior history
   Chronological list of places visited and information accessed

The system shows the user's current position using a campus map with a position indicator (Figure 14), and related information using a WWW browser that is connected with the location-awareness module (Figure 15).

7.3.2 Functions of the HyperCampus System

The GPS calculates global positions (latitude/longitude) and the system converts them into local map positions when the user is in outdoors. Inside, each room has an infrared transmitter at the entrance,

which transmits the room ID when the door opens. Therefore, the system knows its indoor positions by detecting room IDs. The Web browser is sensitive to each position and shows related information. Also, the Web browser depends on current time and changes contents according to time. So, when the user is close to a lecture room, for example, the system can present information on the class that is being held at that time. Of course, the user can retrieve information about other events regardless of time.

In order to relate information contents with real world situations, we prepared some scenarios that specify appropriate information resources that should be selected when the user is on a typical scene. However, it is impossible to predict every interaction that a user could perform at a certain scene. So, we also provided a mechanism that is flexible enough to reduce user confusion and to allow easy recovery of the previous status if the system's prediction function were to fail.

Because of varying light conditions, the small portable LCD, used for visual presentation, is hard to read when the user is outside. So, the system also gives information multimodally using sound.

Personalization of this system is done by a predescribed user profile that includes the user ID, subject, research theme, and interested items. History of the user's activities such as places visited and information accessed also contributes to this personalization. The learning mechanism involved will be described later. The system uses the profile and history to select some appropriate scenarios for guidance. The selection depends on the measure of relevance between information and predicted user interests. The relevance value is calculated with a distance of weighted feature vectors that abstractly represent the user's recent concerns.

7.3.3 Example Scenarios

To give an example: After identifying the user, the system accesses class information and searches for particular classes related to her subject and interests. Then, the system guides the user to the proper classroom of the next class by considering her current location and the time.

Another scenario demonstrates a laboratory tour. Some visitors and students are looking forward to learning about activities related to their studies and/or interests. The system can create a personalized plan of lab tour for each user. The system will also contact the staff of each lab included in the tour. The order of visits is decided based on relevance values and physical distances.

We are also preparing a library scenario. At a university library, if the user walks around a bookshelf, the system retrieves information on the available books there that are related to her interests. It also will indicate which books are reserved and which can be borrowed. This is an extension of a subscription system that provides its subscribers with information updated since their last visit on the Web. Physical location can be a filtering clue to select appropriate information and timing.

There is a scenario for uninterestedness. If the user has not accessed any information and not moved for a certain period, the system may decide that she is resting for a while. Then, the system will *push* information relevant to her current position and interests. This will encourage the user to become more interested in the campus and the university.

Other scenarios concern community support. An example would be the location notification of other people. If a friend of the user is accessing information in the user's neighborhood, the system informs the user of her friend's location and situation. This function might infringe on the privacy of another. So, the system must solicit agreement before notifying the user of the other person's current location and situation. The system can also support asynchronous communication via physical environments. The user can leave a message at some physical place and other users can read this message when they reach there. The creator of the messages can restrict those recipients by specifying user IDs or group IDs.

We are also designing a human memory supplement system that stores the user's activities connected with physical places. Like Forget-me-not [12], the system works as a memory cue by showing short descriptions of the user's accessed information and places visited in a chronological order.

7.3.4 A Learning Mechanism

We implemented a learning mechanism for user personalization of the HyperCampus system. The mechanism uses a weighted feature vector. The feature corresponds to an attribute of concepts related to the user's interests. The concepts include activities, facilities, and social events at the university. The attributes are related objects (e.g., 'book' is an attribute of 'library'), functions (e.g., 'to study' is an attribute of 'lecture'), characteristics (e.g., 'computerization' is an attribute of 'auditorium'), and so on.

Learning is roughly divided into data acquisition and model modification. In the HyperCampus system, user's behavioral data are acquired by detection of location change and information access. These data

include the time and duration that places were visited and information was accessed and the features related to the places and the information. Model modification is performed by calculating the relevance value between an input feature vector that corresponds to an action of the user and a model of the user's interests that is constructed from previous data, plus adjustment of weights of features in the constructed model.

First, an initial model is built based on the profile data of the user. The profile consists of the user ID, subject, research theme, and interested items. We defined an initial feature vector (user model) as a one-dimensional array of real number values that implicitly reflect the user's subject, research theme, and interested items. For example, if the user majored in history, features related to historical events and architectures have higher weights than other features.

The model modification algorithm is very simple, because we calculate the average value of all feature vectors. The reason is as follows: Let x be a model of interests, and $\{e_1, e_2, ..., e_n\}$ be a set of feature vectors. A relevance value between a feature vector e_1 and model x is given by an inner product of two vectors $e_i * x$. Then, in order to maximize the sum of relevance values $S(x) = \Sigma_i(e_i * x) = nE * x$, x should be αE, where E is the avarage of all feature vectors and α is a positive constant for normalization.

Considering the task, feature vectors have enough representation power and allow a simple calculation algorithm to be applied for parameter tuning.

The HyperCampus system is being evaluated by some freshmen at Keio University, in Japan. Since they are not very familiar with their campus, the system will prove helpful in finding good lectures, classes, and professors that will interest them.

8 Augmented Communication and Support for Community Activities

Agent augmented reality also works for the augmentation of human-human communication. Personalized real world agents can add a supplemental context about personal information and current status of users to their messages. This can help receivers of messages to understand their context better, causing the efficiency of communication to increase.

Agents can also help users make use of previous conversations within the context of the current conversation, by showing a summary of texts like the Remembrance Agent mentioned earlier. This will facilitate the conversation by reducing duplication.

An example application of this system is a social matchmaker that searches for a person who has the same interests as the user when they are participating in a meeting or at a party. Agents communicate with other participants' agents and inquire as to their users' interests. The user will be able to interface with the person selected by the agent very smoothly because they may have similar background and/or experiences.

Another example is an intelligent telephone operator that checks the current status of the potential callee before the user makes a call to that person. If the potential callee is very busy or talking to another person, the agent will recommend that the user not call at that time. Again, agents will always take care to protect the privacy of humans and will cooperate to satisfy their users' demands.

We are creating a community support system that integrates augmented communication via real world agents and asynchronous communication based on electronic mail and bulletin boards. Users can attach their agents to text messages for conveying some contextual information about their physical environments. Also, they can hide their messages inside agents by translating the messages into an agent language whose semantics is well defined by the agent's ontology. In this case, more advanced communication will be possible since agents can understand the contents of messages and will be able to convey nuances as well as physical contexts. Futhermore, interactions between the messenger agent and the receiver agent will result in intelligent filtering and summarization of messages.

Community activities will be supported by the agent-based intelligent message handling system. The communication among community members will overcome, not only the gap of space but also that of time. The communication will be customized for each member by her agent using her personal information. So, received messages will be filtered according to user preferences. Messages being sent will be personalized with the user's situational information (locations, physical environments, mental states, etc.).

Cooperation among agents will work as social filtering that is a type of information filtering based on recommendations given by other users. The agents exchange the users' recommendations with their interests about some specific topics. A recommendation includes a user or agent ID, keywords, an identifier

of the information resource (e.g., URL), an annotation of contents, and an evaluation value (normalized positive value). The evaluation value is decided considering a frequency of access to the information and counts of usage, reference, and recommendation of the information. The recommendations are generated semi-automatically and stored in a shared database. First, an agent searches for recommendations about a particular topic. Then, the agent tries to contact agents that are responsible for the selected recommendations. Next, the agent tries to assign a credit to each responsible agent contacted. The credit is determined through communication between the requesting agent and the responsible agent, because the credit depends on a task of the requesting agent and a career of the responsible agent. The agent repeats the credit assignment process for all selected recommendations. Then, it gives the user the recommendations that have higher credits.

Another important function of the community support system is semantic-content-based text processing. Agents read natural language texts and translate them into expressions in their own language that are unambiguous and used for content-based retrieval and summarization. Although converting natural language texts to agent language is not so easy in general, we make it simpler by employing a semi-automatic semantic tagging system that can help users disambiguate sentence meanings and identify objects linked with referring expressions. Semantic tags are annotated by users through an intelligent text editor. This editor is capable of natural language analysis using a case-based method and a semantic frame system tightly coupled with the agent ontology and the agent language construction.

For text-content-based tagging, we are proposing the Global Document Annotaion (GDA). The GDA Initiative aims at having many Internet authors annotate their HTML documents with some common standard tag sets in such a way that machines can automatically recognize the underlying structures (syntactic, semantic, pragmatic, and so on) of those documents much more easily than by analyzing plain HTML files. A huge amount of annotated data is expected to emerge, which should serve not just as tagged linguistic corpora but also as a worldwide, self-extending knowledge base, mainly consisting of examples showing how our knowledge is manifested.

GDA is comprised of the following three steps:

1. Propose an XML (Extensible Markup Language) tag set which allows machines to automatically infer the underlying structure of documents.

2. Promote development and spread of communication-aiding applications which can exploit those tags.

3. Motivate, thereby, the authors of WWW files to annotate their documents using those tags.

The tags proposed in Step 1 will also encode coreferences, the scope of logical/modal operators, rhetorical structure, the social relationship between the author and the audience, etc., in order to render the document machine-understandable to various degrees. Step 2 concerns AI applications such as machine translation, information retrieval, information filtering, data mining, consultation, expert systems, and so on. If annotation with such tags as mentioned above may be assumed, it is certainly possible to drastically improve the accuracy in such applications. For instance, translation using those tags will be almost guaranteed to produce intelligible outputs. New types of applications for communication aids may be invented as well.

The Internet has opened up an era in which an unrestricted number of people publish their messages electronically through their home pages. Step 3 encourages their commitment to present themselves to the widest and best possible audience by organized tagging. Those WWW authors will be motivated to annotate their home pages, because documents annotated according to a common standard can be translated, retrieved, etc., with higher accuracy, and thus have a greater chance to reach many, more targeted readers. Thus, tagging will make documents stand out much more effectively than decorating them with pictures and sounds. People tend to be unaware of the original spirit of SGML/XML and are using HTML almost solely for the sake of visual layout, but GDA will reintroduce that spirit to the HTML world.

Not only texts, but also images (photographs), can be shared with community members. Each member can annotate information on the shared images. By preparing multiple layers of annotation for each image, the community support system can combine or separate several people's annotation. Also, using annotator's information (e.g., interests, preferences), the agent can select more appropriate annotated information for the user. We think that these jointly annotated images will also work as a common memory of the community. They will be elaborated by the community and will become common knowledge. The knowledge will support our daily activities.

The real world agents can utilize real world information as well as online digital information. Considering physical environments, the agents can guide their users to meet in some place in the real world. The agents negotiate and decide the most appropriate time and place to meet. Since interagent communication includes information on physical environments and personal status/plans, mutual understanding for each other will be established more efficiently than by present-day e-mail or telephone calls.

Our community support also has a role of seamlessly integrating real and virtual community places. Recently, some electronic squares for chatting have been open to the public. There are also virtual cities where inhabitants have three-dimensional figures and can move around a virtual world and talk with one another using speech balloons. Our system tries to reconstruct real community activities on virtual environments. The users can remember their activities by interacting with the virtual environments and change/update their ideas and comments. Our system also supports each individual's memory and extends it to the common knowledge database of the community.

An example application of the integration of real and virtual community places is tracing and re-experiencing real world activities in a virtual environment. Using GPS, the system traces the user's walk through a real town and then shows it on a virtual town that imitates its correspondence. Later, the user can re-experience her previous walk using the virtual town and can go so far as to find her favorite items in the stores again. The agent will facilitate her remembrance by showing memory cues and guide her to her destination.

9 Final Remarks

In this chapter, we have discussed a new approach to introducing agent technologies that support everyday tasks such as walking and shopping with navigational aids and information. The main objective of this research is to integrate the real world and information worlds seamlessly, not just in a visual sense. We think that combining the technologies being pursued in the research areas of augmented reality and software agents is a very promising approach to this problem.

Interaction between humans and agents is one of the most important issues for software agents. Since real world agents detect their users' physical environments, the agents should be aware of users' potential desires and implicitly deduce their concrete intentions when they say or do something.

We have described three implemented prototype systems based on the concept of real world agents. Of course, they are not yet practical because of the lack of adequate technology and open experimentation.

Future work includes a more detailed analysis of situated conversation and agent learning and collaboration. Since we have implemented them in a rather ad-hoc way, a more general mechanism that better realizes the true functions of a real world agent also needs to be pursued.

Furthermore, the agents will take care of our mental/emotional situation by using such psychophysiological signal sensors as electroencephalograms (EEG), electrooculograms (EOG), and skin conductances (GSR) [8]. Then, agents will be able to judge the user's current mental and physical situations and respond to the user considering these situations. We are now creating such personalized agents with these mental-awareness capabilities.

By using psychophysiological information, we will be able to develop an automatic photographing system. In order to record a scene, we usually take a photograph. However, we sometimes want to remember the scene in a more mental mode. So, we are planning to combine the automatic photographing system with a real world agent, which can recognize the user's attentional states and takes a photograph incorporating the user's view. Human attentional states are measured by using EOG that is an electrical signal of muscles controlling eye movements and a brain wave that is related to special cognitive processes [13]. The camera will move according to the movement of the user's head. The system will tell the user that it has taken a photograph of the scene at which the user was just looking, and then prompt the user to annotate the photograph taken.

Agent augmented reality enables several applications, including an information provider for our daily life, a personal place-to-place navigator, a community support tool, and so on. We have a lot of work ahead of us to make this concept practical. For example, we have to establish an information and physical infrastructure, that is a basis of seamless connection between the real world and cyberspace. To implement agent technologies, we have to develop a secured interagent communication method, a multimodal human-agent interaction technique, and agent learning and adaptation techniques. We hope that many elegant solutions will be invented in the next decade and that agent augmented reality applications will make our personal life more creative and our community life more communicative.

Acknowledgments

The author would like to thank Mario Tokoro, Jun Rekimoto, and other colleagues at Sony CSL for their encouragement and discussion. Special thanks go to the researchers in the Speech Processing Group at the Electrotechnical Laboratory for their help in developing the speech recognition module. We also extend our thanks to Hiroko Inui, Yuki Hayakawa, and Yasuharu Katsuno for their contributions to the implementation of the prototype systems.

References

[1] Michael Bajura, Henry Fuchs, and Ryutarou Ohbuchi. Merging virtual objects with the real world: Seeing ultrasound imagery within the patient. *Computer Graphics*, 26(2):203–210, 1992.

[2] Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

[3] Eugene Charniak and Robert P. Goldman. A Bayesian model of plan recognition. *Artificial Intelligence*, 64(1):53–79, 1993.

[4] Oren Etzioni and Daniel Weld. A Softbot-based interface to the Internet. *Communications of the ACM*, 37(7):72–76, 1994.

[5] Steven Feiner, Blair MacIntyre, and Dorée Seligmann. Knowledge-based augmented reality. *Communications of the ACM*, 36(7):52–62, 1993.

[6] T. Finin, J. Weber, G. Wiederhold, M. Genesereth, R. Fritzson, D. McKay, J. McGuire, P. Pelavin, S. Shapiro, and C. Beck. Specification of the KQML agent-communication language. Technical Report EIT TR 92-04, Enterprise Integration Technologies, 1992.

[7] George W. Fitzmaurice. Situated information spaces and spatially aware palmtop computers. *Communications of the ACM*, 36(7):38–49, 1993.

[8] Kenneth Hugdahl. *Psychophysiology: The Mind-Body Perspective*. Harvard University Press, 1995.

[9] Katunobu Itou, Satoru Hayamizu, and Hozumi Tanaka. Continuous speech recognition by context-dependent phonetic HMM and an efficient algorithm for finding n-best sentence hypotheses. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP-92)*, pages I.21–I.24. IEEE, 1992.

[10] Yasuharu Katsuno, Ken-ichi Murata, and Mario Tokoro. A java front-end approach for programming mobile agents. In *Proceedings of OOPSLA '97 Workshop on Java-based Paradigms for Mobile Agent Facilities*, 1997.

[11] Henry Kautz. A circumscriptive theory of plan recognition. In Philip R. Cohen, Jerry Morgan, and Martha E. Pollack, editors, *Intentions in Communication*, pages 105–133. The MIT Press, 1990.

[12] Mik Lamming and Mike Flynn. Forget-me-not: Intimate computing in support of human memory. In *Proceedings of the FRIEND21 International Symposium on Next Generation Human Interface*, 1993.

[13] P. J. Lang, M. K. Greenwald, and M. M. Bradley. Looking at pictures: Affective, facial, viseral and behavioral reactions. *Psychophysiology*, 30:261–273, 1993.

[14] Pattie Maes. Agents that reduce work and information overload. *Communications of the ACM*, 37(7):30–40, 1994.

[15] Katashi Nagao. A preferential constraint satisfaction technique for natural language analysis. In *Proceedings of the Tenth European Conference on Artificial Intelligence (ECAI-92)*, pages 523–527. John Wiley & Sons, 1992.

[16] Katashi Nagao. Abduction and dynamic preference in plan-based dialogue understanding. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, pages 1186–1192. Morgan Kaufmann Publishers, Inc., 1993.

[17] Katashi Nagao and Jun Rekimoto. Ubiquitous Talker: Spoken language interaction with real world objects. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*, pages 1284–1290. Morgan Kaufmann Publishers, Inc., 1995.

[18] Katashi Nagao and Akikazu Takeuchi. Social interaction: Multimodal conversation with social agents. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, pages 22–28. The MIT Press, 1994.

[19] Rosalind W. Picard. Affective computing. Technical Report 321, MIT Media Laboratory, 1995.

[20] Jun Rekimoto. Augmented interaction: Interacting with the real world through a computer. In *Proceedings of the 6th International Conference on Human-Computer Interaction (HCI International '95)*, pages 255–260, 1995.

[21] Jun Rekimoto and Katashi Nagao. The world through the computer: Computer augmented interaction with real world environments. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST'95)*, pages 29–36, 1995.

[22] Paul Resnick and Hal R. Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.

[23] Bradley J. Rhodes and Thad Starner. Remembrance Agent: A continuously running automated information retrieval system. In *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi Agent Technology (PAAM'96)*, pages 487–495, 1996.

[24] Thad Starner, Steve Mann, Bradley Rhodes, Jeffrey Levine, Jennifer Healy, Dana Kirsch, Rosalind W. Picard, and Alex Pentland. Augmented reality through wearable computing. *Presence*, 1997.

[25] E. Tulving, editor. *Elements of Episodic Memory.* Clarandon Press, 1983.

[26] Marilyn Walker, Masayo Iida, and Sharon Cote. Japanese discourse and the process of centering. *Computational Linguistics*, 20(2):193–232, 1994.

[27] Roy Want, Andy Hopper, Veronica Falcao, and Jonathan Gibbons. The active badge location system. *ACM Transactions on Information Systems*, 10(1):91–102, 1992.

[28] Mark Weiser. Some computer science issues in ubiquitous computing. *Communications of the ACM*, 36(7):74–85, 1993.

[29] Pierre Wellner. Interacting with paper on the Digital Desk. *Communications of the ACM*, 36(7):86–96, 1993.

[30] Jim White. The foundation for the electronic marketplace. *General Magic White Paper*, 1994.

[31] Massimo Zancanaro, Oliviero Stock, and Carlo Strapparava. Dialogue cohesion sharing and adjusting in an enhanced multimodal environment. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93)*, pages 1230–1236. Morgan Kaufmann Publishers, Inc., 1993.