An Introduction to Computational Awareness

Qiangfu Zhao
Dept. of Computer and Information Systems
The University of Aizu
Aizuwakamatsu, Fukushima, Japan
qf-zhao@u-aizu.ac.jp

Abstract— In 2009, we organized the first International Workshop on Aware Computing. During the workshop, Goutam Chakraborty, Runhe Huang, Robert Kozma, Jianhua Ma, Tadahiko Murata, Hideyuki Takagi, Tomohiro Takagi, and I had a meeting and discussed seriously about the future of awareness computing. After that, we decided to establish the technical committee on awareness computing under the umbrella of IEEE SMC society. Since then, we have evolved our annual event from a workshop to a conference. Time passes quickly; even now many people still do not understand well what we are doing and what we should do. This paper serves as an introduction to computational awareness (CA). Although it is difficult to cover opinions of all people in a short conference paper, something is better than nothing. Hope this paper can help people to understand our activities better.

Keywords- Computational awareness, artificial awareness, artificial intelligence, awareness computing, awareness science, awareness engineering.

I. INTRODUCTION

Artificial intelligence (AI) has been a dream of researchers for decades. In 1982, Japan launched the 5th generation computer project, expecting to create intelligence in computers [1], but failed. Noting that logic approach alone is not enough; many people started to study soft computing in late 1980s. After many years, however, we have not got any system that is as intelligent as a human, in the sense of over-all performance.

Instead of trying to create high level intelligence directly, we may try to create some low level intelligence first. In fact, "awareness" was used in this sense when I planned to organize the first International Workshop on Aware Computing (IWAC2009). However, through discussions with many colleagues I found that awareness can have many different levels. On the one hand, awareness can be as simple as the function realized by a single neuron. A neuron can be activated when a certain input pattern is provided. In this sense, the neuron is aware of the existence of certain patterns. On the other hand, awareness can be as complex as finding the optimal way from Tokyo to Johannesburg. That is, high level awareness can be as intelligent as a human, or even beyond.

Briefly speaking, awareness is a mechanism for detecting some events. The detected events themselves may not make sense, but they can provide important information for awareness of other more complex events. Thus, creating different levels of awareness in a computer may lead to intelligence step-by-step, and this is one of the goals of our community.

This paper is an introduction to computational awareness (CA). The paper is organized as follows. Section II provides a brief review of existing aware systems and tries to classify them in a scientific way; Section III tries to define awareness and related terminologies so that we can have a common language in studying CA; Section IV proposes a general structure of awareness systems; Section V poses some problems to solve in CA; Section VI provides a possible future vision of awareness systems, and Section VII is the conclusion.

II. CLASSIFICATION OF AWARENESS SYSTEMS

The term "awareness computing" is not new. It has been used for more than two decades in the context of computer supported cooperative work (CSCW), ubiquitous computing, social network, and so on [2]-[14]. So far, awareness computing has been considered by researchers as a process for acquiring and distributing context information *related to what is happening, what happened, and what is going to happen* in an environment under concern. Thus, the main purpose of awareness computing or awareness engineering is to provide context information in a timely manner so that human users or computing machines can take actions or make decisions proactively before something really happens.

So far, many aware systems have been studied. The systems are often classified based on the event to be aware of. Examples include, context aware, situation aware, intention aware, preference aware, location aware, energy aware, risk aware, chance aware, and so on. This classification is not really scientific because it divides aware systems into too many categories and the boundaries between the categories are not clear. In this paper, we classify aware systems based on two factors only, namely 1) is the system aware of some event(s), and 2) does the system make some decision(s) based on awareness? Based on these two factors, existing aware systems can be divided into 3 types, namely NanD, AnD, and AmD. Detailed discussion is given as follows.

1. NanD (No aware nor Decision) systems

This is the simplest case in which the aware system just provides the context, and the human user must be aware of any useful information contained in the context, and make decisions based on the information. The "media space" developed in 1986 [15] is a NanD system. It can provide all kinds of background information for cooperative work. Users in different places can work together as if they are in the same room. In fact, most monitoring systems for traffic control, for nuclear power plant, for public facilities, and so on, are NanD systems. These systems are useful for routine cooperative work, but from the definition *they are not aware systems*.

2. AnD (Aware, but no Decision) systems

An AnD system is aware of the importance or urgency of different context patterns, so that critical information can be provided to the human user (or some other systems) in a more noticeable, comprehensible and/or visible way. Clearly, compared with NanD systems, AnD systems are more aware, and can enhance the awareness ability of human users significantly. The system may help human users to detect important clues for solving a problem, for detecting some danger, for seizing a chance, etc. Many decision supporting systems (DSS) developed (and used successfully in many areas) so far are AnD systems, although in many cases their developers did not intend to build an "aware system" at all.

3. AmD (Aware and make Decision) systems

An AmD system is aware of the meaning of the context patterns, and can make corresponding decisions for the user. For example, in an intelligent environment (IE) (e.g. smart home, smart office, smart nursing room, etc.)[16], the server may be aware of the contexts related to actions, locations, behaviors, and so on, of the human users; and can provide suitable services based on different context patterns. Examples of services may include: switching-on/off of light to keep the best lighting condition in a smart office; providing software or hardware resources to meet the requirement of a user in cloud-computing; sending a certain amount of electricity to a client in a smart grid power supply environment; and so on.

Generally speaking, AmD systems are more aware compared with AnD systems because it can also make decisions. Note that many existing AmD systems are not intelligent even if they are sometimes called intelligent systems for commercial purposes. This is mainly because in most cases the events to be aware of can be pre-defined, and the decisions can be made based on some simple manually defined rules. In fact, many context aware systems developed for mobile computing can be achieved simply by using some wireless sensors (known as smart sensors) connected to a server through a base-station (BS). These systems are not intelligent because they are just "programmed", although they are aware systems.

III. CLARIFICATION OF TERMINOLOGIES

According to Wikipedia [18], "*Awareness is a relative concept. An animal may be partially aware, subconsciously aware, or acutely aware of an event. Awareness may be focused on an internal state, such as a visceral feeling, or on external events by way of sensory perception. Awareness provides the raw material from which animals develop qualia or subjective ideas about their experience.*"

In CA, we may define awareness as a mechanism for obtaining information or materials which are useful for human users, for other systems, or for other parts of the same system, to make decisions. In general, awareness does not necessarily lead directly to understanding. Awareness may have many levels (see Fig. 1), and we believe that "understanding" and "intelligence" can be created at some of the awareness levels.

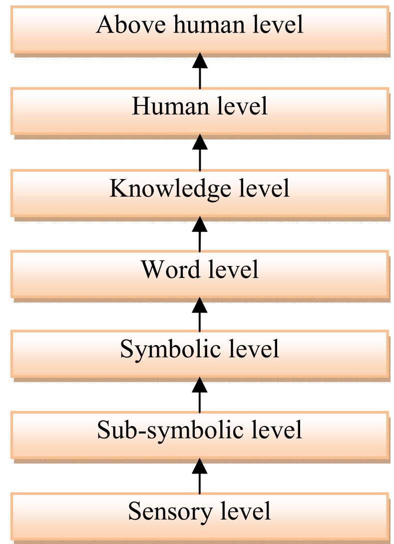

Fig. 1: Levels of awareness

The first level of Fig. 1 is related to the term perception. According to Wikipedia [19], "*perception is the process of attaining awareness or understanding of the environment by organizing and interpreting sensory information*". This concept has different meanings in psychology, neuroscience, and philosophy. In any case, perception is used mainly for brain related activities. In natural languages, perception may also mean "understanding".

In CA, we define perception as a mechanism for obtaining or receiving sensory data. That is, *perception is the sensory level awareness*, which is the first level in Fig. 1, and serves as an interface between the aware system body and the outside world. Some fundamental processing of the sensory data can be included in perception, but it is not necessary to produce understanding at this level.

Another related term is cognition. Again, according to Wikipedia [20], "*cognition refers to mental processes. These processes include attention, remembering, producing and understanding language, solving problems, and making decisions*". Compared with perception, cognition is often used for understanding more abstract concepts and for solving problems based on logic thinking. In CA, we can also follow this line, and define *cognition as the human level awareness*, which is the highest level awareness we human being can achieve.

Between sensory level and human level, there are several levels of awareness. From the bottom, we have sub-symbolic level, symbolic level, word level, and knowledge level. These terminologies have been used frequently in the context of machine learning and artificial intelligence. For example, to "symbolize" an object in a digital image, we may need to follow some rules taken from a "knowledge base". This is actually a kind of top-down approach, in which we have already obtained some knowledge related to object detection, segmentation, and symbolization. In Fig. 1, however, we use a bottom-up approach; although feedback from a higher level awareness is also possible (see discussion in the next section).

Specifically, sensory level awareness obtains different sensory data; sub-symbolic level awareness detects some useful patterns from the data; symbolic level awareness assigns some symbols to the patterns to make them more meaningful; word level awareness formulates some human understandable concepts based on the symbols; knowledge level awareness derives some re-usable rules from the concepts; and human level awareness solves different problems (including low level problems) based on the knowledge so obtained.

There are several other related concepts. For example, people are often confused about the relation between awareness and consciousness. "*Consciousness is a term that refers to the relationship between the mind and the world with which it interacts. It has been defined as: subjectivity, awareness, the ability to experience or to feel, wakefulness, having a sense of selfhood, and the executive control system of the mind*" (from Wikipedia [21]).

In CA, we may define consciousness as the wakefulness of an aware system to the current situation. A system is wakeful if it, given a certain situation, knows what to do and why. Thus, *consciousness is a relatively high level awareness*. For example, we are aware of heart-beating but subconsciously. Bees build their nests subconsciously, but human build their homes consciously. Usually, when we become conscious, we have already collected enough information for reasoning. Thus, consciousness should be somewhere above the word level awareness.

"Kansei" is used by Japanese researchers to mean something that can be understood in mind, but cannot be described well in language. "*Kansei engineering is a method for translating human feelings and impressions into product parameters*" (Wikipedia [22]). Roughly speaking, Kansei is the human feeling that cannot be described logically. In this sense, *Kansei should be a pattern level awareness*. The purpose of Kansei engineering is to transform Kansei to a high level awareness to make it more understandable.

In summary, awareness has many levels. Each level produces materials or information for the upper level awareness to produce more complicated materials or information. In CA, a*wareness above the word (not necessarily written) level can be defined as intelligence*.

IV. GENERAL STRUCTURE OF AWARENESS SYSTEMS

A conceptual structure of an aware unit (AU) is shown in Fig. 2. It can be mathematically described as follows:

$$y = R_3(R_2(R_1(x)))$$

where R_1 is a receptor, R_2 is a reactor, and R_3 is a relater (see Fig. 2). The input x and the output y are usually represented as real vectors. Each element of x can come from a physical sensor, a software sensor, or a lower level AU. Each element of y is a concept to be aware of or some information to be used by a higher level AU.

The purpose of the receptor R_1 is to receive data from outside, filter out irrelevant noises, enhance the signals, and normalize or standardize the inputs, so that inputs from different kinds of sensors can be treated in the same way. The purpose of the reactor R_2 is to take reaction to a given input, and extract/select important features. The purpose of the relater R_3 is to detect certain events, and make proper decisions based on features provided by R_2, and relate the detected events to other AUs. In fact, the receptor is a NanD system; and a receptor plus a reactor forms an AnD system.

Note that the data flow both forward and backward in an AU. An AU has two different modes, namely working mode and learning mode. In the working mode, the AU receives sensory inputs, and makes proper decisions. In the learning mode, the AU receives feedback from the higher level AUs, and sends feedback to lower level AUs. This happens also inside the AU itself. That is, the receptor receives feedback from the reactor; and the reactor receives feedback from the relater. System parameters can be adjusted based on the feedback.

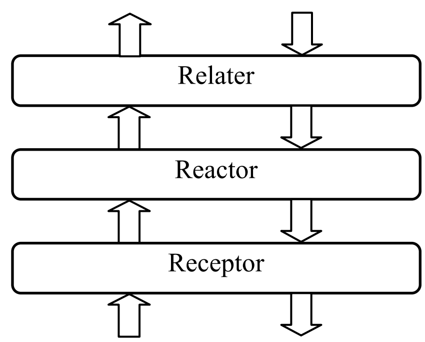

Fig. 2: Structure of an AU

An AU itself can be used as an aware system; or it can be used as a sub-system and form a larger system with other AUs. For example, we can arrange many AUs in the same way as a multi-layer perceptron (MLP). In this case, all AUs in the input (bottom) layer realize the receptor of the whole system, AUs in the hidden (internal) layer(s) together realize the reactor, and AUs in the output (top) layer realize the relater.

Note that most existing aware systems take the structure of Fig. 2. For example, in a typical context-aware system [6], the receptor may contain many sensors for collecting different information; the reactor may contain a context repository for storing important contexts selected from the input data; and the relater may contain a production system for sending proper contexts to proper users, or giving proper commands to proper actuators.

One defect of existing context-aware systems is that they are just "designed" or "programmed". When the contexts to be aware of are complex and dynamically changing, the system must be able to learn and become more and more aware autonomously. This is an important topic for further study in CA.

Based on the proposed AU model, any aware system can be connected through internet or intranet with other aware systems to form a larger system for providing a large variety of contexts. The larger system, of course, may not be owned by a single company or organization. This is not important for the users, as long as they can get proper services with proper prices. This is actually the true concept of cloud computing (i.e., any user can get any service anywhere and anytime, without knowing where is the provider), and should be promoted further.

We may consider an aware system formed through internet as a virtual AU. In this virtual AU, the nodes are correlated through the relaters of the nodes. The virtual AUs can form a still higher level AU through internet. That is, internet provides a flexible way to form higher and higher level AUs. This poses many related problems (e.g. security, privacy, ownership of resources, etc.), and these will be important topics for research in CA.

V. BASIC PROBLEMS TO SOLVE

Note that our aim is to create intelligence using CA, and at the same time, make aware systems more intelligent. For this purpose, ad hoc approaches so far used in the awareness computing community are not enough. We should understand the physical meaning of different problems first, and then propose different approaches for solving them. At the first glance, there are so many problems to solve. However, if we classify them properly, we may see that the problems actually belong to a very limited number of categories. Here, we try to consider some representative problems.

1. The "for what" problem: awareness for what?

The first problem we must consider is the purpose of awareness. Suppose that a system is aware of a piece of important information (context or concept). The system may use this information to help the user to avoid some danger, to avoid wasting time or money in doing something, to get more opportunity for success, etc.; or the system may help the producer to maximize its profit, to avoid attacks from malicious users, to get a good business chance, etc.

To solve the "for what" problem, it is necessary to build a correlation map between the input (context or concept) and the output (possible goal). It is relatively easy to find the correlation map if the number of possible outputs is small and the outputs can be derived directly from the input. In practice, however, there can be many unforeseen outputs (e.g. outputs not registered in the system, like the big Tsunami for destroying Fukushima nuclear power station), and the current input maybe the factor for many outputs to occur. In the former case, the aware system should be able to detect possible new outputs (novelty detection); and in the latter case, the system should be able to modify the correlation map dynamically, so that the scope of possible outputs can be narrowed when more information is acquired.

From the machine learning point of view, the "what for" problem can be solved by adding two functional modules in an aware system. The first one is a novelty detection module for detecting possible new outputs given some inputs; and the second is a reasoning module for predicting possible results given a sequence of inputs. For instance, support vector machine (SVM) can be used for the former, and Bayesian network (BN) can be used for the latter.

The "what for" problem can also be understood as follows. Suppose that the user has some goal (purpose or intention) when he/she uses an aware system. If the system is aware of the goal in an early stage, the system can provide services proactively, and the user can be more efficient in reaching the goal. In addition, if the user does not have a clear goal, the system can help him/her to formulate the goal. Goal or intention awareness can be achieved by asking user feedbacks, or the system may just record the history of the user, and guess the goal autonomously. Again, the system should have two function modules. One is to find possible new goals from the current situation; and the other is to modify the predicted goal based on a sequence of situations.

To be aware of the user goal, current situation alone is not enough. This is because different users may have different goals even under the same situation. Thus, goal awareness is closely related to user modeling. Specifically, the system should be aware of the physical and mental status of the user.

2. The "for whom" problem: awareness for whom?

User awareness is important to provide personalized services. For different users, they need different services even under the same situation. To be aware of the users, user modeling is important and has been studied extensively in the literature [23] [24] [25]. However, existing results are still not enough. In fact, user modeling is a very difficult task because it is related to several factors, namely the human factor, the social factor, the context factor, the spatial-temporal factor, and so on. The problem is difficult even if we focus on the human factor alone. For example, human emotion is difficult to be aware of because even for the same person, his/her emotion can be different if the situation or time is different.

If the user is a group (e.g. all persons holding the stocks of Toyota Company, all iPhone users, and all people interested in CA), it is more difficult to model the user. For example, it is difficult to predict Toyota stock price of next month; it is difficult to know what will happen next year for the market share of iPhone; and it is difficult to know what new results will be obtained next year in the CA community. In ubiquitous computing, it is important to predict the collective behavior of different group users. Thus, user modeling or user awareness will continue to be a hot topic in the coming years.

3. The "of what" problem: awareness of what?

So far, in the field of CA, the event to be aware of is often specified by the user, and the system just searches related information for the user from the database or captures the information from related sensors. In practice, however, it may be difficult to specify the event in advance. As an example, let us consider brainstorming. In brainstorming, when we see or hear something, we human being may be aware of some interesting information for producing good ideas. This kind of information cannot be pre-defined and must be captured in real time. As another example, we may consider the case when the system is aware of the goal of a user for doing something. The system may try to propose a plan for the user to reach the goal. The goal may be reached in several steps, and in each step, the system needs to dynamically define the event to be aware of.

Solving the "of what" problem is also important for making an aware system more comprehensible. Let us consider a multi-level aware system. As in a multilayer neural network, even if we can define or be aware of the concepts in the last (output) level, it is usually difficult to define those in the hidden level(s). If we can, we will be able to describe the input-output relation of each hidden unit using a symbolized concept, and a reasoning process can be provided for any decision made by the system. Thus, solving the "of what" problem can make the system more comprehensible or understandable for human.

4. The "with what" problem: Awareness with what?

Now suppose that we have already defined the event to be aware of. The next question is that what kind of inputs shall we collect? Even if the inputs are given, which inputs are the most informative? Without knowing the correct inputs, we can only design a system that uses all kinds of inputs, or part of the inputs. The former will not be efficient, and the latter will not be effective. Thus, an aware system should be able to learn to extract and select the most important features for making decisions.

The "with what" problem is partly related to the well-known feature selection problem that has been studied in the context of machine learning. However, existing results are not enough. In ubiquitous computing, since the working environment of an aware system changes constantly, we must consider the plasticity-stability problem seriously. That is, we cannot just select against features that are not important for the time being. We must consider the long-term performance of the system.

Another related problem is how to produce (extract) useful features. Remember that in an AU, it is the reactor that produces useful features for the relater to make decisions. The reactor not just select inputs and passes them to the relater, it also produces more useful features by combining existing inputs, linearly or non-linearly. This topic has been studied in the context of dimensionality reduction and feature extraction. The main purpose is to represent the input-output relation in a more compact way, so that the relater can make decisions more efficiently. If some *a priori* information is available to the reactor, the produced features can also enable the relater to make decision more effectively.

When we consider a multi-level aware system, the hidden level units as a whole can be considered as the "reactor" of the system. Thus, the "of what" problem is closely related to the "with what" problem. For the former, we are interested in how to symbolize the hidden units; and for the latter, we are interested in how to produce proper concepts using the hidden units. In a dynamically changing computing environment, both problems are subject to change, and there should be a mechanism for updating the produced concepts and the corresponding symbols, and at the same time for preserving the stability of the whole system.

VI. Future Awareness Systems

With the rapid progress of information and communication technology (ICT), many aware systems have been developed for providing different services. Most aware systems are connected and related to each other directly or indirectly through internet and/or intranet. We human being, as the most intelligent swarm ever appeared in the planet earth, is constructing a global-scale awareness server (GSAS) that may provide *any service to anyone, anywhere and anytime*. Aware systems developed so far are nothing but sub-systems of the GSAS. Although GSAS is also an AmD system, it is and will be much more intelligent than any existing AmD systems.

GSAS is still expanding, and becoming more and more intelligent. The main driven force for making GSAS more and more intelligent is actually the needs of users. Note that individual users nowadays are equipped with different computing devices (desktop, laptop, handtop and palmtop devices), and many different group users have been and will be created through interaction of individual users. As the user number and user type increase, the interaction between different users becomes more and more complex. To meet the needs of all kinds of users, GSAS must be a "general problem solver" (GPS).

In fact, to obtain a GPS has been a dream of many AI researchers for decades [17]. We may think that this dream can be easily realized now because computing technology of today is much more powerful than that of 1950s. However, this is not true, because most problems related to building a GPS are NP-complete, and increasing the computing power alone cannot solve the problems.

One heuristic method for building a GPS is divide-and-conquer (D&C). Briefly speaking, D&C first breaks down big problems into small ones, solves the small problems, and then puts the results together. Based on this concept, we can design many sub-systems for different problems first. Each sub-system is just a specialist in some restricted field. To solve big problems, solutions provided by the sub-systems must be integrated. There are two approaches for integration. One is "centralized" approach, and another is "decentralized" approach. To build GSAS, the latter is more useful. In the decentralized approach, all sub-systems are connected organically to form a network. This network as a whole can solve any given problems. This is actually the basic model of GSAS. In GSAS, many sub-systems have been and will be added for different applications. The system as a whole will soon or late become a GPS.

Note that no one can construct GSAS by him/herself. GSAS is being constructed by people around the world. All persons are divided into two classes, namely the users and the producers. Each user, individual or group, is an AmD sub-system in GSAS, and raises "problems" in the form of "requests". Each producer is an AmD sub-system, too, and provides "solutions" in the form of "services".

Constructing GSAS is a competitive and also cooperative task. The users and producers compete with each other. To win this competition, users cooperate with each other to form larger and larger social groups, and pose more and more difficult problems. On the other hand, producers may cooperate with each other to find new solutions. Both users and producers are driven by their desires to obtain more and more profits, and this is why they are contributing to the project enthusiastically. This "cold war" relation between users and producers can be considered as the true driven force for GSAS to be more and more intelligent, and finally become a GPS.

VII. CONCLUSION

In this paper, we proposed a new classification method of existing aware systems. Compared with the event-based classification, the proposed method is more general and more scientific, and can provide a unified way for studying different aware systems regardless where the system is applied. We then clarified several terminologies related to CA. We think that standardization of the terminologies is necessary for us to study different aware systems in a common framework.

We also proposed a general model of aware unit (AU). This model can be used to describe any existing aware systems, and at the same time, can be used to construct different systems in the future. Starting from the lowest level AU (a physical sensor), we can construct any aware systems of any size with any functions. Here, we do not provide mathematic proof for this, because an MLP is a special case of the proposed model, and its computing power has already been proved [26].

Based on the AU model, we then provided several problems related to CA. Although these problems have been studied in related fields to some extent, we think existing results are still not enough. To create intelligent using CA, or to make aware systems more intelligence, we should reconsider these problems, and propose more efficient and effective algorithms.

ACKNOWLEDGEMENT

The author would like to thank Goutam Chakraborty, Cheng-Hsiung Hsieh, Yong Liu, Jianhua Ma, Kurosh Madani, Tadahiko Murata, Incheon Paik, Hideyuki Takagi, Jhing-Fa Wang and many other colleagues for their fruitful discussions.

REFERENCES

[1] "Start of the 5th Generation Computer Project for Knowledge-Based Information Processing," Computer Museum, Information Processing Society of Japan. http://museum.ipsj.or.jp/en/computer/other/0002.html.

[2] P. Markopoulos, B. D. Ruyter and W. Mackay, *Awareness Systems*, Springer, 2009.

[3] S. Jajodia, P. Liu, V. Swarup, and C. Wang, *Cyber Situational Awareness*, Springer, 2010.

[4] M. Miraoui, C. Tadj and C. B. Amar, "Architectural survey of context-aware systems in pervasive computing environment," Ubiquitous Computing and Communication Journal, Vol. 3, No. 3, pp. 1-9, 2008.

[5] B. N. Schilit, N. Adams, and R. Want, "Context-ware computing applications," IEEE Workshop on Mobile Computing Systems and Applications, pp. 1-7, Dec. 1994.

[6] R. Schmohl and Uwe Baumgarten, "A generalized context-aware architecture in heterogeneous mobile computing environments," Proc. the 4th International Conference on Wireless and Mobile Communications, pp. 118-124, 2008.

[7] Singh and M. Conway, "Survey of context aware frameworks – analysis and criticism," UNC-Chapel Hill, Information Technology Services Version 1, 2006.

[8] H. L. Truong and S. Dustdar, "A survey on context-aware web service systems," Journal of Web Information Systems, Vol. 5, No. 1, pp. 5-31, 2009.

[9] T. Winograd, "Architectures for context," Human-Computer Interaction Journal, Vol. 16, Issue 2, Dec. 2001.

[10] M. Baldauf, S. Dustdar and F. Rosenberg, "A survey on context-aware systems," Int. J. Ad Hoc and Ubiquitous Computing, Vol. 2, No. 4, pp. 263-277, 2007.

[11] K. Barrett and R. Power, "State of the art: context management," State of Art Surveys, Release 2, May 2003.

[12] G. Biegory and V. Cahill, "A framework for developing mobile, context-aware application," Proc. 2nd IEEE Annual Conference on Pervasive Computing and Communications, 2004.

[13] C. Chien, S. Y. He, H. C. Tsai, and Y. K. Hsueh, "An extendible context-aware service system for mobile computing," Journal of Mobile Multimedia, Vol. 6, No. 1, pp. 49-62, 2010.

[14] Schmidt, Kjeld (2002): The problem with "awareness", Computer Supported Cooperative Work, The Journal of Collaborative Computing, Vol. 11, No. 3-4, pp. 285-298.

[15] http://en.wikipedia.org/wiki/Media_space (Aug. 2010).

[16] Diane J. Cook, and Sajal K. Das, "How smart are our environments? An updated look at the state of the art," Pervasive and Mobile Computing, Vol. 3, Issue 2, pp. 53-73. March 2007.

[17] Newell, A.; Shaw, J.C.; Simon, H.A. "Report on a general problem-solving program," Proceedings of the International Conference on Information Processing. pp. 256–264, 1959.

[18] Wikipedia, http://en.wikipedia.org/wiki/ Awareness (Mar. 2009).

[19] Wikipedia, http://en.wikipedia.org/wiki/Perception.

[20] Wikipedia, http://en.wikipedia.org/wiki/Cognition (June 2009).

[21] Wikipedia, http://en.wikipedia.org/ wiki/Consciousness.

[22] Wikipedia, http://en.wikipedia.org/wiki/Kansei_engineering (Aug. 2008).

[23] Santos, E. J., & Nguyen, H. "Modeling Users for Adaptive Information Retrieval by Capturing User Intent," In *Collaborative and Social Information Retrieval and Access: Techniques for Improved User Modeling*. (Edited by Max Chevalier, Christine Julien, Chantal Soulé) IGI Global, 88-118, 2009.

[24] Yan Chen, Hai Long Hou, and Yan-Qing Zhang, " A personalized context-dependent Web search agent using Semantic Trees," Fuzzy Information Processing Society, Annual Meeting of the North American , pp. 1 – 4, 2008.

[25] Zhongming Ma, Gautam Pant, and Olivia R. Liu Sheng, "Interest-based personalized search," ACM Transactions on Information Systems, Vol. 25, No. 1, Article 5, pp. 1-38, 2007.

[26] Kurt Hornik, "Approximation Capabilities of Multilayer Feedforward Networks", *Neural Networks*, 4(2), 251–257, 1991.