Evaluating the Future of HCI: Challenges for the Evaluation of Emerging Applications

Ronald Poppe[1], Rutger Rienks[1] and Betsy van Dijk[1]

[1] University of Twente
Human Media Interaction Group
{poppe,rienks,bvdijk}@ewi.utwente.nl

Abstract. Current evaluation methods are inappropriate for emerging HCI applications. In this paper, we give three examples of these applications and show that traditional evaluation methods fail. We identify trends in HCI development and discuss the issues that arise with evaluation. We aim at achieving increased awareness that evaluation too has to evolve in order to support the emerging trends in HCI systems.

Keywords: Human computing, human-computer interaction, evaluation

1 Introduction

The field of Human-Computer Interaction (HCI) is concerned with the research into, and design and implementation of systems that allow human users to interact with them. Traditionally, the goal of HCI systems is to aid human users in performing an explicit or implicit task. Currently, there is a shift in emphasis towards interfaces that are not task-oriented but rather focused on the user's experience. More subjective factors such as the beauty, surprise, diversion or intimacy of a system are important [1; 2].

A vast body of literature deals with evaluation of traditional HCI systems. These evaluation methods are widely used. However, given the new directions of HCI, it is unlikely that these evaluation methods are appropriate.

Recently, the term Human Computing (HC) has been introduced. In this paper, we discuss Human Computing and the differences with Human Computer Interaction in Section 2. We outline new trends in HCI systems in Section 3. Section 4 presents three examples that illustrate the need for new evaluation methods. In Section 5, we discuss common evaluation methods, argue why these are inappropriate and identify challenges for evaluation of emerging HCI systems.

2 Human Computing and HCI

There is clearly an overlap between Human Computing (or, alternatively, Human-Centered Computing (HCC) [3]) and Human-Computer Interaction. Both deal with humans who interact with computers or machines. The role of the user is central in both paradigms, but there is a difference in the extent. In Human Computing, the user and its contexts are not only observed, but the user's intentions and motives are estimated from the observed behavior. In turn, the system is to display behavior that informs the user about the intentions and motives of the system. For both the observation and presentation of intentions, models of interaction are required. For successful, natural, interaction, these models should be as close to human-human interaction models as possible. Due to their importance, these interaction models are explicitly part of the Human Computing paradigm. The concept is illustrated in Figure 1.

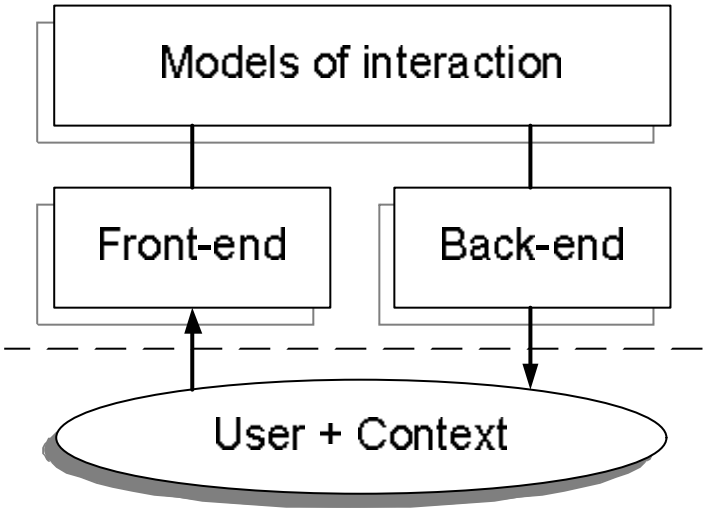

Figure 1: Schematic model of Human Computing

In Figure 1, there are clearly three distinct parts. The *front-end* deals with observing the user and its context. Aspects like the activities, affective state, but also the context such as the environment and other users are taken into account. Pantic *et al*. [4] discuss the front-end, with a focus on recognition of the user's affective state. The *back-end* is concerned with the presentation of information to the user and the control of actuators in the environment of the user. Different from HCI are the *models of interaction*, that are used both in the front-end to understand user behavior, and in the back-end to generate appropriate behavior in turn. Dialog management models, for example turn-taking and argumentation, are part of the interaction models.

In the remainder of this paper, we discuss and analyze the current state of the art in HCI systems and the evaluation thereof. However, the challenges that we are facing when dealing with evaluation of emerging HCI applications are discussed in

the context of Human Computing since we feel that proper models of interaction are essential for these applications.

3 HCI systems

3.1 Traditional HCI systems

Traditional HCI systems allow human users to input commands using keyboards, mice or touch screens (e.g. ATM machines, web browsers, online reservation systems). These input devices are reliable in the sense that they are unambiguous. Traditionally, systems are single-user, task-oriented and the place and manner in which the interaction takes place are largely determined by the projected task and expected users. This allows system designers to specify the syntax and style of the interaction. Since both input and output interfaces are physical, an explicit dialogue between the user and the computer can be established. This dialogue is more even more facilitated when only the user can initiate the interaction.

If we look at the ATM, the user that interacts with the system clearly wants to perform a task: withdrawing money or viewing the account balance. The interaction devices are physical: the buttons and card reader for input into the system, the screen, money slot and ticket printer for output to the user. ATMs are intended to be single-user, and the user always initiates the interaction. The dialogue between user and system is explicit, and highly standardized.

3.2 Emerging HCI systems

Emerging HCI systems and environments have a tendency to become *multi-modal* and *embedded* and thereby allowing people to interact with them in natural ways. In some cases, the design of computer interfaces is merging with the design of everyday appliances where they should facilitate tasks historically outside the normal range of human-computer interaction. Instead of making computer interfaces for people, people have started to make people interfaces for computers [5].

The nature of applications is changing. Looking beyond traditional productivity-oriented workplace technologies where performance is a key objective, HCI is increasingly considering applications for everyday life. HCI design now encompasses leisure, play, culture and art. Compared to traditional HCI systems, we can identify four main trends in HCI systems:

1. **New sensing possibilities** New sensing technologies allow for the design of interfaces that go beyond the traditional keyboard and mouse. Automatic speech recognition is common in many telephone applications. The current state of video tracking allows not only for localization of human users, but also to detect their actions, identity and facial expressions [4]. This opens up possibilities to make interfaces more natural. Humans will be able to interact

in ways that are intuitive. However, this comes at a cost of having to reconsider the syntax of the application. When using speech or gestures, the vocabulary is almost infinite. Moreover, many of the 'behaviors' that we recognize, must be interpreted in relation to the context. Context aware applications employ a broad range of sensors such as electronic tags, light sensing and physiological sensing. However, integration and the subsequent interpretation of these signals is hard, and context aware systems are likely to consider contexts differently than users do [6]. For example, when a user decides to watch a movie at home and closes the blinds to make the room darker, the system may automatically switch on the lights. Clearly, there user and system have a different view of the current situation.

Related to the use of a multiplicity of sensors is the trend that sensors are moving to the background [7; 8]. This moves interfaces away from the object-oriented approach that is traditionally considered [9]. This trend has large implications for interaction design since it restricts the traditional dialog-oriented way of interaction, and effort must be paid to the design of implicit interactions [10].

2. **Shift in initiative** Traditional HCI systems embrace the explicit way in which the dialog with the user is maintained. Moreover, the user is virtually always the one who initiates the interaction. Consequently, traditional HCI systems are responsive in nature. Nowadays, pro-active systems are more common. Ju and Leifer [10] define an initiative dimension in their framework for classifying implicit interactions. They state that, when regarded more generally, there is direct manipulation at the one end, and autonomy at the other. They argue that for HCI, neither of these states are appropriate. Instead, the interaction is likely to be mixed-initiative. This implies that there must be a way to coordinate the interaction, which should be the focus of interaction design. For applications where multiple users can engage in the interaction, there's also a mixed-initiative among the different users.

3. **Diversifying physical interfaces** The physical forms of interfaces are diversifying [11], as was foreseen by Mark Weiser [8]. One movement is to make interfaces bigger, such as immersive displays and interactive billboards. Another movement is to make interfaces smaller, such as wearable and embedded displays. This last movement is largely motivated by the popularity of mobile devices. The market for mobile phones is still growing, and so is the number of applications. With the increased connectivity and bandwidth, it is possible that people interact remotely with the same application. The trend of diversifying physical interfaces is most visible for general purpose desktop computers. These are increasingly often replaced by more purpose-designed and specialized appliances [11].

4. **Shift in application purpose** There is a shift in application purpose for HCI systems. This shift is partly a consequence of new technology, and partly motivates the development of new technology. Whereas traditional systems

are, in general, task-based, new applications are more focused on everyday life [11], thus on the user. User Experience (UX), although associated with a wide variety of meanings [12], can be seen as the countermovement of the dominant task and work related ‘usability’ paradigm.

UX is a consequence of a user’s internal state (e.g. predispositions, expectations, needs, motivation and mood). The literature on UX reveals three major perspectives [13]: human needs beyond the instrumental; affective and emotional aspects of interaction; and the nature of experience. Hassenzahl and Sandweg [14] argue that future HCI must be concerned about the pragmatic aspects of interactive products as well as about hedonic aspects, such as stimulation (personal growth, increase of knowledge and skills), identification (self-expression, interaction with relevant others) and evocation (self maintenance, memory). The task is no longer the goal, but rather the interaction itself (e.g. [15]).

Typical UX applications are focused on leisure, play, culture and art. Consequently, this focus affects the interface. Factors as pleasure, aesthetics, expressiveness and creativity play an increasingly important role in the design of both interface and interaction. Video games are a clear example of UX applications.

Another aspect is that interfaces are not only more centered on the user and the interaction, but also show a trend towards product integration. Domestic technology is becoming increasingly complex [16]. Our microwaves function also as stoves, we can listen to music, take pictures and exchange media with our mobile phones and our washing machines can also dry the laundry for us. Ubiquitous computing (UC), although radically different from traditional HCI on a number of criteria, is one extreme example where functionality is integrated. Of course, this influences the choice of physical interface.

4 Stressing the need for evaluation: three examples of emerging HCI applications

In this section, we discuss three examples of emerging HCI systems. These serve to demonstrate the observed trends in HCI system development, and allow us to pinpoint the difficulties with traditional evaluation methods in Section 5.3.

4.1 Groupware systems

One example of an area where a lot of money has been invested into the development of a product because of its expected scenario gains is the area of group support systems (GSS) or groupware. De Vreede *et al*. [17] conclude from extensive research that 15 years after the introduction of the first group support system, these systems indeed provide added value to meetings. They are said to provide savings, and

increase efficiency. It was a rather complex and non-straightforward process to come to this conclusion.

One of the reasons that it took so long was the fact that people were facing difficulties when using the system, as they were not familiar with the changes in work practice that were introduced by them [18]. People were forced to use novel tools during meetings and had to abandon their common meeting practice. As a consequence, also the benefits proved hard to measure as people objected to the use of these tools.

GSS are clear examples of systems that establish a *shift in application purpose*. Although Grudin [19] already noted that adequate understanding of the political and social factors at work were to be considered in the design and implementation phases in order to avoid an initial reject from the public, the task of supporting the meeting process (e.g. facilitate brainstorming) was considered more important than how these systems were used in practice. It was therefore not strange that people found it difficult to understand what the system was supposed to do for them and their group [20]. Design for intuitive interaction with the user as focal point would have facilitated its adoption, without any doubt.

4.2 Smart homes

Smart home systems are typical examples of ubiquitous systems, characterized by their pervasive nature. Users are observed in their homes using a large number of sensors, ranging from cameras and microphones to pressure and heat sensors. See Figure 2(a) for an example of a smart home setting. From a user point of view, ubiquitous systems do not necessarily have a task. They can be anywhere between responsive and pro-active. An example that lies somewhere in between responsive and pro-active is for instance the smart home described in Intille *et al*. [6] where the system *suggests* users which clothes to wear given the outside temperature, or suggests measures to save energy. From a system point of view, smart homes have the task to maintain the homeostasis of the environment, and to support the users that are living in it. One example is a smart home that supports elderly people in order to allow them to live (semi-)independently. These homes not only take care of lighting and heating issues, but also facilitate communication in case of emergencies.

Figure 2: (left) Philips' vision of smart homes: the environment is adjusted to make patients feel more comfortable in hospitals. (right) User interacting with the Virtual Dancer.

When the environment itself becomes the interface, people go about their daily lives and perform their tasks while the computing technologies are there to support them transparently [8]. People start to implicitly interact with computers and technology that have moved to the background. Despite being written over 10 years ago, many aspects of Mark Weiser's vision of ubiquitous computing appear as futuristic today as they did in 1991 [21].

As Davies and Gellersens [22] mention there are many aspects that need to be resolved before ubiquitous interfaces really will break through. They mention, amongst others, the need for fusion models and context awareness. Due to the lack of an explicit interface, users are required to communicate naturally with the system. This requires fusion of multiple communication channels. The system must be aware of the context, and interpret the user's actions in this context. On the other hand, the user must be familiar with the system's abilities, and system's state.

The complexity and black box characteristics of smart homes make them even more difficult to evaluate. They do not only introduce a *shift in application purpose*, but also employ *new sensing possibilities*. There is a radical *change in physical interface* since the smart home has become the interface itself. Some smart homes are pro-active, which presents a clear *shift in initiative*.

4.3 Virtual dancer

Fun and entertainment are becoming increasingly important in almost all uses of information technology [23]. *Ambient entertainment* is the field of research that deals with applications that are centered on this theme. One example of an ambient

entertainment application is the Virtual Dancer, as described in Reidsma *et al*. [15]. It is an interactive installation where users can dance together with a virtual character. The virtual character reacts to the observed movements of the user, and tries to influence the movements of the user in turn. The movements are observed using a dance mat and a camera. The camera recognizes global movement features that are mapped onto a database of prerecorded movements for the Virtual Dancer.

The camera and dance mat provide *new sensing possibilities*. Also, during the dance, there is a constant *shift in initiative*. The goal of the application is to entertain the user, without the provision of an explicit task. Instead, the interaction itself is the goal of the application, a clear *shift in application purpose*.

In this so-called taskless interaction, not the task but the interaction itself and the user experience need to be evaluated. Attempts so far to evaluate the interaction have been limited to analyzing video recordings of the user in order to determine engagement in the interaction. This does not allow for reliable assessment of aspects that improve the user' s experience during the interaction, let alone which parts of the system should be improved. One important aspect is that the responses of the user to certain actions of the systems have to be measured. This requires the knowledge of system states, i.e. the context. While this information proves valuable in the assessment of the participation level of the user, it does not provide much information about the actual user experience. Instead, this information could be collected using questionnaires or by employing biosensors that measure heart rate and the respiratory level.

5 Evaluation

Evaluation is broad concept. In the domain of HCI, Preece *et al*. [24], page 602 defined this concept in 1994 as follows:

> *Evaluation is concerned with gathering data about the usability of a design or product by a specific group of users for a particular activity within a specified group of uses or work context.*

In 2007 they have expanded this definition [25], page 584:

> *It [evaluation] focuses on both the usability of the system, e.g. how easy it is to learn and to use, and on the users' experience when interacting with the system, e.g. how satisfying, enjoyable, or motivating the interaction is.*

The use of evaluation methods for the assessment of the suitability of HCI systems has become a standard tool in the design process. Many HCI systems are designed iteratively, where in each cycle design issues of the previous one are addressed. These issues are identified in an evaluation step. We discuss the design criteria of HCI systems first in Section 5.1. We then focus on current evaluation practice in the HCI field in Section 5.2. Section 5.3 discusses issues that appear when dealing with evaluation for emerging HCI applications.

5.1 Design criteria in HCI

Much has been written about the design of HCI systems (e.g. [26; 27]). Designed well, interactive systems can allow us to reap the benefits of computation and communication away from the desktop, assisting us when we are physically, socially or cognitively engaged, or when we ourselves do not know what should happen next. Designed poorly, these same devices can wreck havoc on our productivity and performance, creating irritation and frustration in their wake [10]. Good practice is to explicitly formulate design choices.

Norman [28] identifies a number of principles for good interaction design. Often used are the principles visibility, feedback, constraints, consistency, recovery, and affordance. If things are *visible,* people can see what functions are available and what the system is currently doing. Then they are more likely to know what to do next as a consequence of the psychological principle that it is easier to recognize things than to recall them. The *feedback* principle is related to the visibility principle and refers to sending back information from the system to the users so that they know what effect their actions have had. Timely feedback provides the necessary visibility for user interaction and will enhance the feeling of control. *Constraints* should prevent people from making errors through properly constraining allowable actions. For instance by deactivating certain menu options people are restricted to choose only permissible actions. *Consistency* is a design principle that emphasizes the importance of uniformity in the placement, appearance and behavior of screen elements and operations to make systems easier to learn and use. Recovery refers to the principle that a system should enable users to recover from actions, particularly errors and mistakes, quickly and effectively. Finally, the principle of *affordance* refers to the fact that things should be designed in a way that it is clear what they are for and how to use them. For instance buttons afford pressing and should look like buttons to invite people to press them.

Traditionally, HCI systems are designed for a certain task, in a given context, and with a certain user profile in mind. Key point is that the HCI system must be useful, usually referred to as usability. There are many different approaches to making a product usable and there is no generally accepted definition. Nielsen [29] identifies five components of usability that are commonly used: efficiency, learnability, memorability, errors, and satisfaction. Three of these criteria can be used as quantitative indicators to asses the usability of a system by measuring respectively time to complete a task, time to learn a task and the number of errors made when carrying out a specific task. Other important usability goals are effectiveness and utility referring to how good a system is at doing what it is supposed to do and to what extent the system provides the right kind of functionality [25].

In addition, usability can be regarded from three distinct viewpoints [30; 31]: product-oriented, user-oriented and user performance-oriented. The product-oriented view can be measured in terms of ergonomic attributes of the product. The user-oriented view in terms of mental effort and attitude of the user and the user performance-view by examining how the user interacts with the product with emphasis on either the ease of use or the acceptability of the product in the real world.

The above views are complemented by the contextual view, which tells us that usability of a product is a function of a particular user class of users being studied, the application at hand and the environment in which they work.

Besides usability, in the interaction between the human and the computer also the user interface and user experience come into play. To stress the necessity of shifting the focus from what computers can do to what users can do, Shneiderman introduced the term ”the new computing” [32]. The broadening of the user community to include almost everyone, even technology resisters, puts a challenge on system designers who realize now that understanding the user is important. This forms the basis of User-Centered Design (UCD). UCD is a multidisciplinary design approach based on the active involvement of users to improve the understanding of user and task requirements, and the iteration of design and evaluation [33]. It has been mentioned that this approach is the key to product usefulness and usability and overcomes the limitations of traditional system-centered design [32].

One view of UCD is to design HCI as close as possible to natural human-human interaction [34]. The rationale is that users do not have to learn new communication protocols, which leads to increased interaction robustness. This aids the user experience and provides guidelines for designing the user interface. A drawback is that one should be familiar with the application to know what to expect from it. Shneiderman [27] argues that most designs for natural interaction do not provide users with available task actions and objects. In terms of the design principles treated before they violate visibility. For knowledgeable and frequent users who are aware of available functions this will not be a problem but for them a precise, concise command language is usually preferable. Hence Shneiderman claims that natural interaction will only be effective for intermittent users who are knowledgeable about specific tasks and interface concepts but have difficulties remembering syntactic details.

5.2 Current evaluation practice in HCI

As stated before, evaluation is nowadays common practice in the field of HCI. The use of evaluation methods is motivated by the reported increased return on investments.

In general, we can identify two broad classes of evaluation methods: expert-based evaluation (e.g. cognitive walkthrough, heuristic evaluation, model based evaluation) and user-based evaluation (e.g. experimental evaluation, user observation, use of questionnaires, monitoring physiological responses). The bulk of early HCI designers and evaluators were cognitive psychologists. Cognitive models like GOMS [35] were very influential, as were laboratory experiments. Nielsen [29] took a more pragmatic approach, stating that full-scale evaluation of usability is too complicated in many cases, so that ’ discount’ methods are useful instead. His work has been very influential, partly due to the ease of application, partly due to the relative low cost. His vision has lead to an enormous number of different methods in regular use for the evaluation of usability.

Since its early days, HCI research focused almost exclusively on the achievement of behavioral goals in work settings. The task that had to be performed

by the user was the pivotal point of user centered analysis and evaluation. Rengger [36] defined four classes of performance measures:

1. Goal achievement (accuracy and effectiveness)
2. Work rate (productivity and efficiency)
3. Operability (function usage)
4. Knowledge acquisition (learning rate)

Though satisfaction has been one of the components of usability since the early days, in the last few years there is an increased focus on user experience. The expansion of the definition of evaluation in the beginning of Section 5 is not typical for the book of Sharp, Rogers and Preece [25]. Similar changes can be found in most literature on HCI. This is one of the answers of the HCI community to the shift in application purpose of systems mentioned in Section 3.2. But as we discussed before, emerging HCI systems require other measures, and other evaluation practice. In the next section, we identify challenges for evaluation of emerging HCI systems, and use the examples in Section 4 as an illustration.

5.3 Challenges for evaluation of emerging HCI systems

The characteristics of emerging HCI systems imply that traditional approaches to usability engineering and evaluation are likely to prove inappropriate to the needs of its users. As a result of the trends that we discussed in Section 3.2, problems emerge in the design and evaluation of HCI systems. We start by discussing the front-end of Human Computing, using the examples of Section 4.

Human sensing
The use of keyboards, buttons and mice for interaction with HCI systems is found to be inconvenient since these devices do not support the natural ways in which humans interact. Although debated, the use of natural communication is often considered more intuitive and therefore expected to be more efficient from a user's point of view. Voice, gestures, gaze and facial expressions are all natural human ways of expression. In natural contexts, humans will use all these channels, one to enhance and complement another. To make truly natural interfaces, this implies that all these channels should be taken into account. This, however, is difficult for at least three reasons:

1. The recognition is error-prone
2. The lexicon of expression is much larger than with 'artificial input'
3. Integration of multiple channels often leads to ambiguities

Error-prone recognition When using natural channels, the data obtained from sensors (microphone, camera) needs to be analyzed. From the streams of data, we need to recognize the communicative acts (e.g. words, gestures, facial expressions). Although much research is currently devoted to making automatic recognition more accurate, these systems will never be error-free. Another aspect is that automatic recognition is

probably less fine-grained than what human observers are able to perceive [37]. Subtleties might easily go unnoticed.

Reduction of errors is probably the most convenient way of improving the usability. However, as recognition will never be error-free, repair mechanisms need to be present. Feedback and insight in the system state are useful because they give the user insight in how the input is recognized and interpreted. Still, there are many challenges in how to present the feedback or system state [38]. One approach in speech recognition tasks is to feed the recognition back to the user. The can be done, for example, in applications where tickets can be ordered via telephone. After the user has specified the date, the system could ask "How many tickets do you want to order for this Tuesday?" Although this kind of mechanism can improve the recognition performance, attention must be paid to how users can correct the recognition.

Assessment of the input reliability is an important aspect of usability evaluation. One way to do this is by applying standard benchmark sets. Well-known benchmark sets are the NIST RT sets [39] for automatic speech recognition or FRVT and FRGC for face recognition [40]. These sets are specific for a given context and task. Since they contain ground truth and the error metrics are known, they allow for good comparison of recognition algorithms. However, they still evaluate only the reliability of the input. In addition to this, the system must be evaluated together with the (unreliable) input.

Large lexicon In natural human-human interaction, humans use a large lexicon of speech, and eye, head and body movements, both conscious and unconscious. When allowing humans to communicate with HCI systems in a natural way, the input devices should be able to recognize the whole range of signals. This poses severe requirements on the recognition.

Two factors are important when evaluating the lexicon. First, the lexicon should be sufficiently large to allow for the recognition of all foreseen (and unforeseen) actions. For a system such as the Virtual Dancer (see Section 4.3), this implies that the whole range of dance movements that a user can make, should be included in the lexicon. Alternatively, only a subset of all communication signals can be considered. However, it should be acknowledged to the user whether the signals are recognized and interpreted.

Second, the choice of the lexicon should be intuitive. In many cases, an *ad hoc* lexicon is chosen, often to maximize the recognition. Ideally, the lexicon should contain signals that users naturally make when interacting with the HCI system. Note that, although this interaction is natural, the lack of a clear interface might prove that it is also not intuitive [41]. A preliminary investigation should be conducted to see what these movements and sounds are, for example by conducting Wizard of Oz experiments. An example of such an investigation is described in [42].

When dealing with attentive or pro-active systems, not only the communicative actions are of importance. These systems require awareness of things as user state and intentions, which generally can be deducted from behavior that is non-communicative.

Integration of channels Human behavior is multi-modal in nature. For example, humans use gestures and facial expressions while speaking. Understanding of this

behavior does not only require recognition of the input of individual channels, but rather the recognition of the input as a whole. Despite considerable research effort in the field of multi-modal fusion (see e.g. [43]), our knowledge about how humans combine different channels is still limited. When dealing with multi-user systems, the problem is even harder since also the group behavior needs to be understood. For remote participation in the interaction, it might be very difficult since the interaction mechanisms for human-human interaction are probably not applicable. Furthermore, due to the disappearing interfaces, the lack of explicit turn-taking will cause users to employ many alternate sequences of input, and requires HCI systems to be more flexible in handling these in turn [9].

Similar to the performance evaluation of single communication channels, the recognition of the fused channel information need to be assessed. Integration of multiple channels can lead to reduction of signal ambiguity, provided that the context is known. Therefore, accurate assessment of the context is needed.

Context awareness

It is often mentioned that human behavior is to be interpreted in a given context. For example, a smile in a conversation can be a sign of appreciation, whereas, during negotiation, it can show disagreement. So for reliable interpretation of the human behavior, it is important to be aware of the context of the situation. To date, there is no consensus of what context is precisely, and how we should specify this [44]. Without a good representation for context, developers are left to develop *ad hoc* and limited schemes for storing and manipulating this key information [37]. This is acceptable for small domains, but is inappropriate for larger and more complex applications.

Usually, the context is specified as the identity and location of the users, and the characteristics and timing of the action performed. Ideally, even the intentions of the user should also be taken into account. This is particularly difficult since these cannot be measured. These components of context are referred to as the 5 Ws [37; 4]: who, what, where, when, why. These basic components are limited, and one might include the identity and locations of all objects of interest, as well as the current goal of the user. Also, the history of all environment changes and user actions are considered important for reasoning about the context.

It difficult to assess the right values for all these properties, and context aware systems are likely to consider contexts differently than users do. Intille *et al.* [6] observe that, for smart homes (see Section 4.2), the user naturally considers contexts that the system has not, and propose to use suggestive systems, rather than pro-active ones.

Reference tasks

Whittaker *et al.* [45] observed that many developed HCI systems can be considered radical inventions. They do not build further on established knowledge about user activities, tasks and techniques but rather push the technology envelope and invent new paradigms. Although we lack basic understanding of current users, tasks and technologies, the field of HCI is encouraged to try out even more radical solutions, without pausing to do the analysis and investigation required to gain systematic

understanding. The absence of shared task or goal information makes it difficult to focus on research problems, to compare research results and to determine when a new solution is better, rather than different. This prevents proper consolidation of knowledge.

When the users are not familiar with the task or goal the application supports, users are likely to use the system in a different way. This makes evaluation of the fitness of the system difficult. For example, interfaces that support creative thinking are designed for a specific task that is new to the users. Without proper familiarization, these interfaces are less effective (see for example the Groupware example in Section 4.1).

The lack of reference tasks can be seen as a challenge for the development of proper interaction models. Now we move to the back-end of Human Computing.

Performance metrics

In contrast to Rengger [36], as discussed in Section 5.2, emerging HCI applications often do not have well-defined tasks, which asks for novel measures. There are many factors in HCI that have a substantial impact on the success of applications but are not easily quantified. Amongst them are user experience [16], fun [46], ethical issues [47], social relationships [48] and aesthetical issues [1]. For example, for the Virtual Dancer (see Section 4.3), it remains a challenge to define proper measures to evaluate the success of the interaction. These critical parameters are also required in order to compare similar applications [49]. When the application supports multiple users, these measures might be shared among the users.

Learnability

Given the increasing complexity of HCI systems, it is to be expected that the time needed to learn to work with a system grows along. Currently, evaluation of these systems focuses on ‘snap shots’, but fail to focus on the learning [50]. Longitudinal studies that assess how the use of a system develops from the first encounter are needed to gain insight in what kind of barriers users encounter when using the system, and how they solve these.

Context of authentic use

HCI systems should be evaluated in a context as close as possible to the context of authentic use [37]. The context is often difficult to realize, especially for multi-user applications. Evaluating HCI systems in laboratory settings is likely to cause unnatural behavior of the users. This makes proper evaluation of the system difficult, if not impossible.

Another drawback of using laboratory testing is that parameters can be controlled (background noise, lightning conditions) that cannot be controlled in the context of authentic use. As a consequence, there is a difference in how these systems perform in reality.

As an example, the live-in laboratory PlaceLab [51] has been built to ensure that assumptions about behavior in the lab correspond to behavior in more realistic (and complex) situations in real smart homes.

6 Conclusions

New HCI systems are emerging that differ from traditional single-user, task-based, physical-interface HCI systems. We identify four trends: new sensing possibilities, a shift in initiative, diversifying physical interfaces, and a shift in application purpose. Traditional evaluation practice does not suffice for these new trends.

The use of more natural interaction forms poses problems when the input is ambiguous, the communication lexicon is potentially large, and when interpreting signals from multiple communication channels, ambiguities might arise. Identifying the context of use is important because interpretation of input is often dependent on the context. For complex systems, sensing the context is increasingly difficult. Evaluation of context aware systems is consequently difficult.

There is no consensus about appropriate performance metrics for emerging HCI systems. Task-specific measures are useless for evaluation of task-less systems. Related to this is the lack of common reference tasks. The ‘radical invention’ practice in the field of HCI prevents proper consolidation of knowledge about application tasks and goals, and user activities. Therefore, it is difficult to compare HCI systems.

As HCI systems are becoming more complex, the learning process of users is more and more important. This is currently a neglected part of evaluation. The introduction of longitudinal evaluation studies is needed to gain insight in the learning mechanisms. A final practical issue is the lack of authentic usage contexts. Many systems are only evaluated in a laboratory setting, instead in their projected context.

We summarized trends in HCI systems and pointed out where problems appear. We discussed three examples of complex HCI systems, and argued the need for appropriate evaluation. With this paper, we aimed at achieving increased awareness that evaluation too has to evolve to support the emerging trends in HCI systems.

Acknowledgements

This work was partly supported by the European Union 6th FWP IST Integrated Project AMI (Augmented Multi-party Interaction, FP6-506811, publication **TODO**, and is part of the ICIS program. ICIS is sponsored by the Dutch government under contract BSIK03024. The authors wish to thank Anton Nijholt for his valuable comments.

References

1. Lauralee Alben. Quality of experience: defining criteria for effective interaction design. Interactions, 3(3):11–15, 1996.
2. Bill Gaver and Heather Martin. Alternatives: exploring information appliances through conceptual design proposals. In Proceedings of the conference on Human factors in computing systems (CHI' 00), The Hague, The Netherlands, pages 209–216, 2000.
3. Alejandro Jaimes, Nicu Sebe, and Daniel Gatica-Perez. Human-Centered Computing: A Multimedia Perspective. In Proceedings of the ACM international conference on Multimedia, pages 855–864, Santa Barbara, CA, 2006.
4. Pantic, M., Pentland, A., Nijholt, A., Huang, T.S.: Machine Understanding of Human Behavior: A Survey. In: Artificial Intelligence for Human Computing. Springer-Verlag (this issue)
5. Michael H. Coen. Design principles for intelligent environments. In Proceedings of the National Conference on Artificial Intelligence (AAAI' 98), pages 547–554, Madison, WI, 1998.
6. Stephen S. Intille, Emmanuel M. Tapia, John Rondoni, Jennifer Beaudin, Chuck Kukla, Sitij Agarwal, Ling Bao, and Kent Larson. Tools for studying behavior and technology in natural settings. In Proceedings of the International Conference on Ubiquitous Computing (Ubicomp' 03), volume 3869 of Lecture Notes in Computer Science, pages 157–174, Seattle, WA, 2003.
7. Norbert Streitz and Paddy Nixon. Introduction: The disappearing computer. Communications of the ACM, 48(3):32–35, 2005.
8. Mark Weiser. The computer of the 21st century. Scientific American, 265(3):66–75, 1991.
9. Jakob Nielsen. Noncommand user interfaces. Communications of the ACM, 36(4):83–89, 1993.
10. Wendy Ju and Larry Leifer. The design of implicit interactions. Design Issues, Special Issue on Design Research in Interaction Design, to appear.
11. Steve Benford, Holger Schndelbach, Boriana Koleva, Rob Anastasi, Chris Greenhalgh, Tom Rodden, Jonathan Green, Ahmed Ghali, Tony Pridmore, Bill Gaver, Andy Boucher, Brendan Walker, Sarah Pennington, Albrecht Schmidt, Hans-Werner Gellersen, and Anthony Steed. Expected, sensed, and desired: A framework for designing sensingbased interaction. ACM Transactions on Computer-Human Interaction, 12(1):3–30, 2005.
12. Jodi Forlizzi and Katja Battarbee. Understanding experience in interactive systems. In Proceedings of the conference on Designing Interactive Systems (DIS' 04), pages 261–268, Cambridge, MA, 2004.
13. Marc Hassenzahl and Noam Tractinsky. User experience a research agenda. Behaviour & Information Technology, 25(2):91–97, 2006.
14. Marc Hassenzahl and Nina Sandweg. From mental effort to perceived usability: transforming experiences into summary assessments. In Extended abstracts on Human factors in computing systems (CHI' 04), pages 1283–1286, Vienna, Austria, 2004.
15. Dennis Reidsma, Herwin van Welbergen, Ronald Poppe, Pieter Bos, and Anton Nijholt. Towards bi-directional dancing interaction. In International Conference on Entertainment Computing (ICEC' 06), volume 4161 of Lecture Notes in Computer Science, pages 1–12, 2006.

16. Peter Thomas and Robert D. Macredie. Introduction to the new usability. ACM Transactions on Computer-Human Interaction, 9(2):69–73, 2002.
17. Gert-Jan de Vreede, Douglas R. Vogel, Gwendolyn L. Kolfschoten, and JeroenWien. Fifteen years of GSS in the field: A comparison across time and national boundaries. In Proceedings of the Hawaii International Conference on System Sciences (HICSS' 03), page 9, Big Island, HA, 2003.
18. Jay F. Nunamaker Jr., Robert O. Briggs, and Daniel D. Mittleman. Electronic meeting systems: Ten years of lessons learned. In David Coleman and Raman. Khanna, editors, Groupware: Technology and Applications. Prentice Hall, Englewood Cliffs, NJ, 1995.
19. Jonathan Grudin. Groupware and social dynamics: Eight challenges for developers. Communications of the ACM, 37(1):93–105, 1994.
20. Robert O. Briggs, Gert-Jan de Vreede, and Jay F. Nunamaker Jr. Collaboration engineering with thinklets to pursue sustained success with group support systems. Journal of Management Information Systems, 19(4):31–64, 2003.
21. Albrecht Schmidt, Matthias Kranz, and Paul Holleis. Interacting with the ubiquitous computer: towards embedding interaction. In Proceedings of the joint conference on Smart objects and ambient intelligence (sOc-EUSAI' 05), pages 147–152, Grenoble, France, 2005.
22. Nigel Davies and Hans-Werner Gellersens. Beyond prototypes: Challenges in deploying ubiquitous systems. IEEE Pervasive Computing, 2(1):26–35, 2002.
23. Charlotte Wiberg. Usability and fun: An overview of relevant research in the HCI community. In Proceedings of the CHI Workshop on Innovative Approaches to Evaluating Affective Interfaces, Portland, OR, 2005.
24. Jenny Preece, Yvonne Rogers, Helen Sharp, and David Benyon. Human-Computer Interaction. Addison-Wesley Longman Ltd., 1994.
25. Helen Sharp, Yvonne Rogers, and Jenny Preece. Interaction Design: Beyond Human Computer Interaction. 2nd edn. John Wiley and Sons, 2007.
26. Alan Dix, Janet Finlay, Gregory Abowd, and Russell Beale. Human Computer Interaction, third edition. Prentice Hall, 2004.
27. Ben Shneiderman, and Catherine Plaisant. Designing the User Interface: Strategies for Effective Human-Computer Interaction. 4th edn. Addison Wesley, 2005.
28. Donald A. Norman. The Design of Everyday Things. MIT Press, 1998.
29. Jakob Nielsen. Usability Engineering. Academic Press, Boston, MA, 1993.
30. Nigel Bevan, Jurek Kirakowski, and Jonathan Maissel. What is usability? In Proceedings of the international Conference on HCI, pages 651–655, Stuttgart, Germany, 1991.
31. Matthias Rauterberg. Quantitative measures for evaluating human-computer interfaces. In Proceedings of the International Conference on Human-Computer Interaction, pages 612–617, Orlando, Florida, 1993.
32. Ben Shneiderman. Leonardo's Laptop: Human Needs and the New Computing Technologies. MIT Press, 2002.
33. Ji-Ye Mao, Karel Vredenburg, Paul W. Smith, and Tom Carey. The state of user-centered design practice. Communications of the ACM, 48(3):105–109, 2005.
34. Leah M. Reeves, Jennifer Lai, James A. Larson, Sharon L. Oviatt, T. S. Balaji, Stéphanie Buisine, Penny Collings, Phil Cohen, Ben Kraal, Jean-Claude Martin, Michael McTear, TV Raman, Kay M. Stanney, Hui Su, and Qian YingWang. Guidelines for multimodal user interface design. Communications of the ACM, 47(1):57–59, 2004.
35. Stuart K. Card, Allen Newell, and Thomas P. Moran. The Psychology of Human-Computer Interaction. Lawrence Erlbaum Associates, Mahwah, NJ, 1983.

36. Ralph E. Rengger. Human Aspects in Computing: Design and Use of Interactive Systems with Terminals, chapter Indicators of Usability based on performance, pages 656–660. Elsevier, Amsterdam, The Netherlands, 1991.
37. Gregory D. Abowd and Elizabeth D. Mynatt. Charting past, present, and future research in ubiquitous computing. ACM Transactions on Computer-Human Interaction, 7(1):29–58, 2000.
38. Victoria Bellotti, Maribeth Back, Keith Edwards, Rebecca E. Grinter, Austin Henderson Jr., and Christina V. Lopes. Making sense of sensing systems: Five questions for designers and researchers. In Proceedings of the Conference on Human Factors in Computing Systems (CHI' 02), pages 415–422, Minneapolis, MN, 2002.
39. Jonathan G. Fiscus, Nicolas Radde, John S. Garofolo, Audrey Le, Jerome Ajot, and Christophe Laprun. The rich transcription 2005 spring meeting recognition evaluation. In Steve Renals and Samy Bengio, editors, Revised Selected Paper of the Machine Learning for Multimodal Interaction Workshop 2005 (MLMI' 05), volume 3869 of Lecture Notes in Computer Science, pages 369–389, Edinburgh, United Kingdom, 2006.
40. Jonathon Phillips, Patrick J. Flynn, Todd Scruggs, Kevin W. Bowyer, and William Worek. Preliminary face recognition grand challenge results. In Proceedings of the Conference on Automatic Face and Gesture Recognition 2006 (FGR' 06), pages 15–24, Southampton, United Kingdom, 2006.
41. Anton Nijholt, Thomas Rist, and Kees Tuijnenbreijer. Lost in ambient intelligence? In Extended abstracts on Human factors in computing systems (CHI' 04), pages 1725–1726, Vienna, Austria, 2004.
42. Johanna Höysniemi, Perttu Hämäläinen, Laura Turkki, and Teppo Rouvi. Children' s intuitive gestures in vision-based action games. Communications of the ACM 48(1): 44–50, 2005.
43. Sharon L. Oviatt. The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications, chapter 14: Multimodal interfaces, pages 286–304. Lawrence Erlbaum Associates, 2003.
44. Arthur H. van Bunningen, Ling Feng, and Peter M.G. Apers. Context for Ubiquitous Data Management. In International Workshop on Ubiquitous Data Management (UDM' 05), pages 17–24, Tokyo, Japan, 2005.
45. Steve Whittaker, Loren Terveen, and Bonnie A. Nardi. Let' s stop pushing the envelope and start addressing it: a reference task agenda for HCI. Human Computer Interaction, 15(2-3):75–106, 2000.
46. Mark A. Blythe, Kees J. Overbeeke, Andrew F. Monk, and Peter C. Wright. Funology: From Usability to Enjoyment, volume 3 of Human-Computer Interaction Series. Kluwer Academic Publishers, 2003.
47. Bonnie A. Nardi, Allan Kuchinsky, Steve Whittaker, Robert Leichner, and Heinrich Schwarz. Video-as-data: Technical and social aspects of a collaborative multimedia application. Computer Supported CooperativeWork, 4(1):73–100, 1995.
48. Jonathan Grudin. Why CSCW applications fail: problems in the design and the evaluation of organizational interfaces. In Proceedings of the Conference on Computer Supported Cooperative Work (CSCW' 88), pages 85–93, New York, USA, 1988.
49. William M. Newman. Better or just different? On the benefits of designing interactive systems in terms of critical parameters. In Proceedings of the conference on Designing interactive systems: processes, practices, methods, and techniques, pages 239–245, Amsterdam, The Netherlands, 1997.
50. Marianne G. Petersen, Kim H. Madsen, and Arne Kjær. The usability of everyday, technology-emerging and fading opportunities. ACM Transactions on Computer-Human Interaction, 9(2):74–105, 2002.

51. Stephen S. Intille. The goal: smart people, not smart homes. In Proceedings of the International Conference on Smart Homes and Health Telematics, pages 3–6, Belfast, United Kingdom, 2006.