



Title	Implicitly influencing the interactive experience
Authors(s)	O'Grady, Michael J., Dragone, Mauro, Tynan, Richard, O'Hare, G. M. P. (Greg M. P.), Wan, Jie, Muldoon, Conor
Publication date	2010-05-10
Publication information	O'Grady, Michael J., Mauro Dragone, Richard Tynan, G. M. P. (Greg M. P.) O'Hare, Jie Wan, and Conor Muldoon. "Implicitly Influencing the Interactive Experience," 2010.
Conference details	Presented at the International Workshop on Collaborative Human/AI Control for Interactive Experiences (CHACIE 2010), at the Ninth International Conference on Autonomous Agents and Multiagent Systems (AAMAS2010), Toronto Canada, May 10 2010
Item record/more information	http://hdl.handle.net/10197/2104

Downloaded 2024-04-16 05:19:35

The UCD community has made this article openly available. Please share how this access benefits you. Your story matters! (@ucd_oa)



© Some rights reserved. For more information

Implicitly Influencing the Interactive Experience

Michael J. O'Grady
CLARITY: Centre for Sensor
Web Technologies
School of Computer Science &
Informatics
University College Dublin
michael.j.ograde@ucd.ie

Mauro Dragone
CLARITY: Centre for Sensor
Web Technologies
School of Computer Science &
Informatics
University College Dublin
mauro.dragone@ucd.ie

Richard Tynan
CLARITY: Centre for Sensor
Web Technologies
School of Computer Science &
Informatics
University College Dublin
richard.tynan@ucd.ie

Gregory M.P. O'Hare
CLARITY: Centre for Sensor
Web Technologies
School of Computer Science &
Informatics
University College Dublin
gregory.ohare@ucd.ie

Jie Wan
Department of Computing &
Networking
Institute of Technology Carlow
Carlow, Ireland
jessie.wan@yahoo.co.uk

Conor Muldoon
Computing Laboratory
Oxford University
Oxford, UK
conor.muldoon@comlab.ox.ac.uk

ABSTRACT

Enabling intuitive interaction in system design remains an art more than a science. This difficulty is exacerbated when the diversity of device and end user group is considered. In this paper, it is argued that conventional interaction modalities are unsuitable in many circumstances and that alternative modalities need be considered. Specifically the case of implicit interaction is considered, and the paper discusses how its use may lead to more satisfactory experiences. Specifically, harnessing implicit interaction in conjunction with the traditional explicit interaction modality, can enable a more intuitive and natural interactive experience. However, the exercise of capturing and interpreting implicit interaction is problematic and is one that lends itself to the adoption of AI techniques. In this position paper, the potential of lightweight intelligent agents is proposed as a model for harmonising the explicit and implicit components of an arbitrary interaction.

Categories and Subject Descriptors

H.4 [Information Systems Applications]: Miscellaneous

General Terms

Design

Keywords

Implicit interaction, Intelligent agents

1. INTRODUCTION

A laudable objective of many computing applications and services is the provision of seamless and intuitive interaction. Ubiquitous computing is a case in point. The vision articulated by the proponents of this paradigm envisages a world saturated with electronic infrastructures, with the objective of making computing services available everywhere such that it may be accessed in an as-needed fashion. Indeed, the late Mark Weiser, the father of ubiquitous computing, likened ubiquitous computing to a common every-

day signpost, both in its pervasiveness in the environment, as well as the ease, intuitiveness and lack of effort associated with its use. However, how such intuitiveness was to be achieved in practice was not stated. To address this, the Ambient Intelligence (AmI) [1] concept was proposed. This explicitly acknowledged the interaction problem of practical pervasive or ubiquitous computing environments and proposed the adoption of Intelligent User Interfaces (IUIs) [5] as a means of addressing this. Again, the pragmatic issues of how such interfaces may be realised in practice remains unanswered.

It is well known that in commercial software, one prerequisite to success is ensuring that the user experience is a satisfactory one. For conventional workstation environments, many useful heuristics have been constructed pertaining to the effective design of interfaces and management of interactions. Given that such environments have been studied since the 1960s, it would indeed be disappointing if significant progress had not been made in this time interval. However, given the many form factors that computing frequently utilises, as well as the multitude of domains in which it is applied, it is questionable as to what degree conventional good practice HCI principles apply for non-workstation environments. In the case of mobile computing, one reason why such principles may not be applicable is that the nature of the context in which a mobile interaction occurs may differ radically from that of conventional interactions. For example, it has been demonstrated that there may be up to eight fold differences between the attention span that users give to tasks under both laboratory conditions and mobile contexts [10].

In this paper, it is argued that successfully harnessing the implicit interaction modality may offer significant potential for augmenting the interaction experience. By incorporating implicit interaction, the potential for sharing control of an application need be not seen as the exclusive preserve of either the human operator or the application in question. Rather it can be regarded as a collaborative effort. In circumstances where intelligent agents have been adopted as the software construct for capturing and interpreting im-

licit interaction, this collaboration may be one shared between the human and the agent.

2. INTERACTION MODALITIES

A number of modals of interaction have been proposed, those of Beale [3] and Norman [7] being well documented examples. For the purposes of this discussion, interaction is considered, albeit briefly, from both a unimodal and multimodal perspective.

2.1 Unimodal Interaction

Unimodal interaction refers to interactions that occur when only one modality is used, for example speech. Conventional interactions with computational devices of various genre are almost inherently unimodal. Though the prevalent approach, it is instructive to note that this is almost diametrically opposite to human communication, which is inherently multimodal.

2.2 Multimodal Interaction

Multimodal interaction involves a number of modalities being used in parallel, for example voice and gestures. This may be regarded as a human centric view of interaction. An alternative interpretation is based on the role of the computational artefact, making this central to the definition. Sebu [12] considers that the computational equivalent of the human senses is what makes a system multimodal. For example, the use of voice recognition using a microphone, and gesture recognition using camera would constitute multimodality. Alternatively, using a suite of cameras, for example to identify gestures and facial expressions, would be regarded as unimodal. For this discussion, the human centric view is adopted.

Multimodal interaction is perceived as being more natural and intuitive [11]. However, this comes at a price: complexity and timeliness. Taken individually, gesture and voice recognition demand sophisticated complex solutions. In parallel, the difficulty is aggravated. Both modalities must be interpreted separately. Then the result must be considered in combinations such that a semantic meaning can be attributed to the interaction. All of this requires significant computational resources if the interaction is to be interpreted correctly, and, importantly, responded to, in a timely manner. Frequently, Artificial Intelligence (AI) techniques, for example, machine learning, are used to facilitate the process of interaction identification.

3. EXPLICIT & IMPLICIT CONTROL

An alternative interpretation of interaction is to consider intent as the key parameter. Indeed, if intent was known with certainty in all circumstances, the issue of intuitive interaction would be less problematic, although its realisation in practice so as to meet user expectations might still raise particular difficulties. When considered in the light of user intent, we can consider the interaction, or by extension the control of the system, as being expressed in either an explicit or implicit fashion. However, an understanding of human communications is necessary before these can be considered.

3.1 A Reflection of Human Communications

For the most part, the vocal channel is the predominant one used everyday. However, this is usually accompanied by

a variety of non-verbal cues that people interpret subconsciously. Thus, it might be concluded, irrespective of the particular utterance, that the speaker is sad, happy, busy or just indifferent. Though contentious, it has been estimated that nonverbal cues have up to four times the effect of verbal cues [2]. Indeed, a number of classifications of such cues exist, for example Ekman & Friesen [4] have identified 5 categories–

1. Emblems: actions that carry meaning of and in themselves, for example a thumbs up.
2. Illustrators: actions that help listeners better interpret what is being said, for example, for example, finger pointing;
3. Regulators: actions that help guide communication, for example head nods;
4. Adaptors: actions that are rarely intended to communicate but that give a good indication of physiological and psychological state;
5. Affect: actions that express emotion without the use of touch, for example, sadness, joy and so on.

These give a flavour of the kind of cues that, if captured correctly, would lead to significant enhancements to the interactive experience.

3.2 Explicit Interaction

Explicit interaction is the normal method of interacting with and controlling software, and may be regarded as unimodal in nature. It is event or stimulus driven. For example, a button is pressed, and the system responds, ideally in some meaningful way. Its simplicity makes it easily understood by all, at least in principle. From a software engineering perspective, most programming languages make the managing of event handling relatively easy for software developers, for example through the Model-View-Controller pattern.

3.3 Implicit Interaction

Implicit interaction is a more subtle construct, and computationally challenging to implement. It is modelled on how humans communicate, and is essentially multimodal in character. The challenge is to capture the cues that invariably contribute to the human communication process, leading to a more complete understanding of the interaction. Even a simple explicit interaction, such as clicking a mouse, takes place within a context. For example, a mouse might be clicked in anger. Assuming this anger can be detected, and developments in affective computing make this increasingly likely, then the software can adapt. How? this will depend on the domain in question. Similar to multimodal interaction, the challenge with implicit interaction is to capture and interpret it in a meaningful fashion.

A question to be considered is whether implicit interaction occurs on its own, or should be regarded merely as augmented explicit interaction. In some cases, the lack of an explicit interaction event, if expected or available, may signify an implicit interaction. In many e-commerce WWW sites, users usually ignore the multitude of advertisements on offer. This they do subconsciously, for the most part, unless of course their attention is obtained and they explicitly decide to click on some advertisement.

4. THE LOCUS OF CONTROL

Ultimately, control of any interactive application or service must rest with the user. However, in many cases, the application will perform in an autonomous or semi-autonomous fashion, resulting in control being shared. Thus the user is nominally in control, but is happy to remain outside the control loop as long as the application is performing to their satisfaction. This may happen in systems of all hues. Indeed, the objective of autonomic computing is to actually remove the human operator from the command chain as much as possible. Human time is perceived as a scarce and expensive commodity, as such must be used wisely.

With interactive entertainment systems in contrast, the motivation is different with the user being the key actor, as it were. In this case, the user (usually) wants to minimise their cognitive load, and with minimum interaction, leave the application follow its own cycle. Any significant intervention must be motivated. However, most entertainment systems react in a stimulus /response manner. No effort is made to monitor the user's reaction or perception to what is happening.

Considering the previous discussion on implicit interaction, it can be seen that there is significant opportunity available for enhancing the user experience, provided effort is expended to capture this interaction. By transparently observing the user as they interact, the potential for a more fulfilling experience emerges. In short, an arbitrary system can dynamically adapt its behaviour in response to observed cues. Obviously, the strategies and policies used for adaptation will be domain, and maybe user, dependent. However, how such adaptivity should be designed for remains an open question. It is envisaged that successfully adapting the behaviour of the application to the user is likely to minimise their need to explicitly take control of the application, and increase the likelihood that they will cede control to the application even if they should need to make an intervention occasionally.

A final issue that needs to be addressed is the characteristics of a software architecture for realising such adaptivity.

5. SOFTWARE ARCHITECTURE

Realising an adaptive application demands that the software inherently possesses certain traits. Autonomy is essential, as a capability to react to external events. For a dynamically adaptive solution, this is not sufficient. A capacity to act proactively and to plan ahead is also essentially. Ideally, some capacity to learn would also be supported. Such characteristics immediately suggest the harnessing of the agent paradigm as this encapsulates these characteristics. However, not all agent architectures are sufficiently endowed with such capabilities. However, those that subscribe to the BDI model for example, could be reasonably expected to be capable of forming the basis of adaptive applications. In selecting an agent framework to support implicit interaction and realise an adaptive solution, it is essential that the attributes necessary for its realisation be kept in mind.

In our own research, the viability of lightweight embedded agents [9] have been demonstrated in mobile computing contexts for managing and interpreting interactions, both explicit and implicit. In the mobile tourism domain, information has been adapted to tourists' contexts [8] while in the e-commerce domain, agents have negotiated deals for

items on users' shopping lists [6]

6. CONCLUSIONS

In this paper, the potential of the implicit interaction modality as a means of augmenting the interactive experience was advocated. However capturing and interpreting such interaction is computationally complex. The harnessing of agent frameworks of sufficient power and complexity is suggested as a basis for realising applications and services that can dynamically adapt to implicit user input.

7. ACKNOWLEDGMENTS

This work is supported by Science Foundation Ireland under grant 07/CE/I1147

8. REFERENCES

- [1] E. Aarts and F. Grotenhuis. Ambient intelligence 2.0: Towards synergetic prosperity. In *Aml '09: Proceedings of the European Conference on Ambient Intelligence*, pages 1–13, Berlin, Heidelberg, 2009. Springer-Verlag.
- [2] M. Argyle, V. Salter, H. Nicolson, and N. W. P. Burgess. The communication of inferior and superior attitudes by verbal and nonverbal signals. *British Journal of Social & Clinical Psychology.*, 9:221–231, 1970.
- [3] A. Dix, J. Finlay, G. D. Abowd, and R. Beale. *Human Computer Interaction*. Pearson, Harlow, England, 3. edition, 2003.
- [4] P. Ekman and W. Friesen. The repertoire of nonverbal behavior: Categories, origins, usage, and coding. *Semiotica.*, 1:49–97, 1969.
- [5] K. Höök. Designing and evaluating intelligent user interfaces. In *IUI '99: Proceedings of the 4th international conference on Intelligent user interfaces*, pages 5–6, New York, NY, USA, 1999. ACM.
- [6] S. Keegan, G. M. O'Hare, and M. J. O'Grady. Easishop: Ambient intelligence assists everyday shopping. *Information Sciences*, 178(3):588–611, 2008.
- [7] D. A. Norman. *The Design of Everyday Things*. Basic Books, New York, reprint paperback edition, 2002.
- [8] M. J. O'Grady, G. M. P. O'Hare, and C. Donaghey. Delivering adaptivity through context-awareness. *Journal of Network and Computer Applications*, 30(3):1007–1033, August 2007.
- [9] G. M. P. O'Hare, M. J. O'Grady, C. Muldoon, and J. F. Bradley. Embedded Agents: A Paradigm for Mobile Services. *International Journal of Web and Grid Services (IJWGS)*, 2(4):379–405, December 2006.
- [10] A. Oulasvirta, S. Tamminen, V. Roto, and J. Kuorelahti. Interaction in 4-second bursts: the fragmented nature of attentional resources in mobile hci. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 919–928, New York, NY, USA, 2005. ACM.
- [11] S. Oviatt. Multimodal interactive maps: designing for human performance. *Hum.-Comput. Interact.*, 12(1):93–129, 1997.
- [12] N. Sebe. Multimodal interfaces: Challenges and perspectives. *Journal of Ambient Intelligence and Smart Environments*, 1(1):23–30, 2009.