Intelligent Interactions Based on Motion

Mathieu Barnachon, Mathilde Ceccaroli, Amélie Cordier,
Erwan Guillou, and Marie Lefevre

Université de Lyon, CNRS
Université Lyon 1, LIRIS, UMR5205, F-69622, France
firstname.lastname@liris.cnrs.fr
http://liris.cnrs.fr/iibm

Abstract. This paper introduces a project called IIBM (Intelligent Interactions Based on Motion). In this project, we address the question of how to design and develop intelligent software systems controllable by gestures. This work is related to the field of natural interactions, but we are not only interested in the motion capture issues. We focus on human-computer interaction and we seek to support negotiation of meaning and co-construction of knowledge between the user and the system. The goal is to develop easy to use, intelligent and adaptive systems, through gestures by end users. For that purpose, we exploit the CBR paradigm. CBR enables us to capture users' experiences and to reuse them in order to provide relevant feedback and assistance. This position paper presents the main ideas of our approach and illustrates them with a first experiment made within our test bed environment, with the application IIANNOTATE. The paper also discusses the benefits and risks of applying our approach to video games.

1 Introduction

The imagination of video game designers is limitless to provide gamers with new, more creative forms of interactions. We moved from the "button system" (keyboards, pads) to that of the mouse, and then very quickly to motion-based interactions. Devices such as the *Eyetoy*, and more recently, the *WiiRemote* and the *PS Move*, offer the player a new approach to the game. The goal is no longer to click on a button to perform an action, but to actually perform an action. Naturally, the trend aims at totally eliminating any control device between the user and the game. This is the case for example with *Kinects*, which provides us with a capture device as well as a middleware for natural interactions. Thus, we see a real paradigm shift. The former level of indirection "a button triggers an action" is no longer used. Now, an action in the real world matches a similar action in the game.

This revolution in the way of designing the game play also causes a kind of revolution among players. Although some pure gamers criticize these new approaches, it is clear that new interactions attract the attention of different people. Notably, video games are now open to new users who are better acquainted with this new and intuitive way of playing. The interactive game market is booming. To give an idea, 50 million Wii were sold by the end of 2009, and more than 64 million in January 2010. In comparison, in 2010, only 36 million XBox (Microsoft) and 31 million PS3 (Sony) were sold. More recently, 10 million Kinect were sold in less than one year. This is the "Fastest-Selling Consumer Electronics Device"[1], with around $133,333$ units per day.

We thus see the emergence of new players and, with them, new expectations are emerging as well. Players expect richer, more interactive and intuitive. One way of designing such games is to make them able to adapt themselves to the gamers. Introducing motion as a control for the game is a first step in response to this challenge. Indeed, the freedom of control offered by motion and gestures increases the possibilities of interactions with the video game. Controls are no longer limited to the combination of available buttons on the joystick. The universe of possibilities extends considerably. Video game designers cannot anticipate every situation. In consequence, gamers are sometimes frustrated because they would like to go beyond what is already planned in the game. The new goal to reach is to design games able to adapt themselves to unforeseen situations. With such new interaction opportunities, game designers now can imagine new ways of playing and can design creative and innovative games. Nevertheless, despite all, interactivity is too often limited to what is "expected" in the game, and video games lack ability to interact with gamers.

The potential of new modalities of interaction is not efficiently exploited, especially since video games have no ability to adapt to users. Games are designed to interpret typical gestures and are not able to cope with unforeseen user actions. We believe that developing these abilities and integrating them in the game design process is the next stage of game development and will undoubtedly enhance the game design.

This paper takes a step in that direction. This is a position paper in which we describe our early work focusing on intelligent interactions based on movements. For that purpose, we go further than the classical concept of "action, reaction" and investigate that of "action, interpretation, adapted reaction". We seek to develop systems that rely on dynamic knowledge to interpret (and not only to identify) user actions and propose appropriate responses accordingly. We argue that in the field of video games (and more generally for any application where the graphical user interface plays a leading role), the development of intelligent interaction mechanisms is a key step for the improvement of systems, game play and game design. In addition, it is also an important factor to ensure the appropriation of such systems by end users.

To develop this concept of intelligent interactions, we combine two directions of research: motion interpretation and artificial intelligence (specifically case-based reasoning). Motion interpretation techniques allow us to have access to rich interaction modalities. Case-based reasoning (CBR) enables the system to have an appropriate behavior in response to users' actions. The CBR process

[1] http://community.guinnessworldrecords.com/_Kinect-Confirmed-As-Fastest-Selling-Consumer-Electronics-Device/blog/3376939/7691.html

relies on a knowledge base that is automatically updated as the user uses the system. Interactions between the user and the system are also recorded and stored in a trace base for future reuse.

This paper is organized as follows. In section 2, we introduce the IIBM project (Intelligent Interactions Based on Motion) which is the framework of this research. Then, we describe the two main aspects of this work: motion interpretation approach, and case-based reasoning for intelligent interactions. In section 3, we present an experiment in a test bed environment called IIANNOTATE. Application of our approach to the field of video games is briefly discussed in section 4. Section 5 outlines the agenda for future studies.

2 The IIBM project

The IIBM project (IIBM stands for Intelligent Interactions Based on Motion) aims at combining motion interpretation and artificial intelligence in order to develop natural interactions [1]. A common definition says that natural interaction is based on human senses (vision and hearing). In our context, we develop an application that the user can control remotely with bare hands. However, we do not work only on how to capture gestures. We focus on how to exploit gesture-based interactions in a smart way. Our ultimate goal is to build gesture-based intelligent, interactive and adaptive applications.

This project brings together researchers from two different fields: motion capture and artificial intelligence. It explores new ways of supporting human-computer interactions (HCI) by trying to provide an answer to the following question: How can artificial intelligence contribute to enrich interactions between a user and a software system? The merger of these two approaches should lead to the development of solutions enabling users and systems to negotiate in order to co-construct a shared knowledge on how to interact together. The main challenge is to orchestrate interactions in order to support this negotiation of meaning [2].

Challenges are both in the field of motion capture and in that of case-based reasoning. Regarding motion capture, the goal is to develop algorithms for fast and efficient gesture recognition (see section 2.1). Case-based reasoning is used to facilitate reuse of users' experiences as well as to enable experience sharing between users. The challenge here is to achieve capture of this experience, to find dynamic ways of structuring it in cases, and to organize these cases for future reuse (see section 2.2). Expected results are applications capable not only to recognize pre-defined user actions and trigger new actions in consequence (retrieval phase in CBR), but also to discover new actions performed by users and to associate a meaning and a suitable reaction accordingly (adaptation phase).

Naturally, the targeted application domain is that of video games. For video games, taking into account the specificities of the players would add tremendous value to the game play. Such an approach would also allow game designers to add more creativity in their games. However, this approach could be useful for other applications where user interactions play an equally important role. As we shall see in section 3, the example we take does not come from the game domain.

As we said before, our approach is related to researches in the field of Human-Computer Interactions, and more precisely, to motion-based interactions [3]. However, to our knowledge, there are few works combining artificial intelligence and motion capture to build intelligent and adaptive systems. Some approaches draw their inspiration from techniques such as K-Nearest Neighbors (KNN), Classifiers or Hidden Markov Model (HMM) to improve motion capture [4, 5]. Other applications use artificial intelligence to implement intelligent behaviors in applications [6]. Furthermore, CBR has been used for several purposes in the games domain [7–9]. In these approaches, CBR is used either to learn behaviors from user and reproduce them in the game or to choose appropriate strategies depending on the context. Here, we use CBR to support interactions, and in particular motion-based interactions, between the user and the game. Yet, we think that there is no work considering a coupling of motion capture and CBR in an integrated approach.

2.1 Motion Capture and Motion Interpretation

Motion capture aims at being used in our everyday life by providing solutions in areas such as entertainment [10], medical applications [11] or intelligent video-surveillance [12], just to name a few. One particular and futuristic application of computer vision is to achieve a Human-Computer Interaction of the same level as the human-human one. Ideally, humans will be able to interact with computers as if they were interacting with other humans. Although many researchers have proposed new solutions to improve HCI, most of them are still simplistic as they are usually limited to a small number of basic interactions.

Simple applications of tracking and/or matching are solved by markers. Typically, Zhang *et al.* [13] have used an ordinary piece of paper as a marker and they were able to point, click, draw and turn. Their main drawback is the use of a marker, even though unique and simple, that has limited the number of recognizable actions. Some others solutions have overcome the marker [14]. They are more adapted to an HCI context, but actions are very limited (and not necessarily well adapted to the action targeted). If most of these solutions are sufficient to be a substitute for a mouse or for simple keyboard shortcuts, they are far from the ideal target of a natural HCI.

Motion interpretation has mostly been addressed by activity monitoring [15–17]. Many of these methods are successful at recognizing very different activities [15]. These methods are based on machine learning techniques, requiring very large databases for recognizing human actions. They are tedious to create and not flexible, *i.e.*, adding a new action requires the database to be re-created.

The interested reader could consult the recent extensive survey on the topic by [18].

2.2 Case-based reasoning for intelligent interactions

As we said before, to implement more intelligent interactions, it appears necessary to take into account the experiences of users. Therefore, it seemed natural to use CBR and more specifically the TBR (Trace-Based Reasoning) [19, 20]. TBR is an approach similar to CBR with the difference that cases are no longer stored in a case base but extracted on the fly from interaction traces recorded by the system. Indeed, in our approach, all interactions between the user and the system are captured and stored in traces. Then, these traces constitute our dynamic case base. This enables us to overcome issues related to the dynamic aspect of our application. The question of how to extract cases (*i.e.* past situations) from traces when they are needed amounts to a problem of extracting patterns from sequential data [21].

In the IIBM project, we use trace-based reasoning at two levels. First, we use it to support the negotiation of meaning between a user and the system [22]. Next, we use it to enable sharing of experiences among different users [23]. In both cases, the knowledge acquired by the system is about both the gestures and the actions that the user wants to perform.

In trace-based reasoning, the storage of the cases is very different from the way it is done in CBR. However, the reasoning steps are quite similar. The retrieval step looks for a similar situation in the trace, *i.e.* a sequence of actions performed by a user. The adaptation step, meanwhile, is iterative. It is during this step that the negotiation process takes place. Depending on the context, the system reuses previous experiences to provide assistance and recommendations to the user who can accept them or not.

The scenario described in section 3 shows how CBR is implemented in IIBM.

3 A case study with IIANNOTATE

In order to conduct our first experiments, we have implemented a gesture-controlled application called IIANNOTATE. IIANNOTATE is an application for annotating (tagging) photos. The main interface of the application is displayed in figure 1. It is made up of a viewer that displays:

- one or more photos;
- a thumbnail bar allowing users to view the photos;
- a tag bar allowing users to handle tags that can be used to annotate photos.

The functionalities of the application are numerous (annotation of a photo or group of photos, creation of a tag or meta-tag, user management, etc.). One of the main characteristics of this application is that it has been designed so that it can be completely manipulated by gestures. Acquisition of movements is made through a middleware relying on Kinect.

The application has a mechanism that records all interactions between the user and the system in the form of "interaction traces". Interaction traces are then exploited to improve motion interpretation, which should lead to a better usability of the application. In order to experiment with this idea, we have designed several experimental scenarios. One of these scenarios is detailed below to give a clearer view of the benefits of the application. The scenario highlights the contribution of the CBR in the interpretation of user actions.

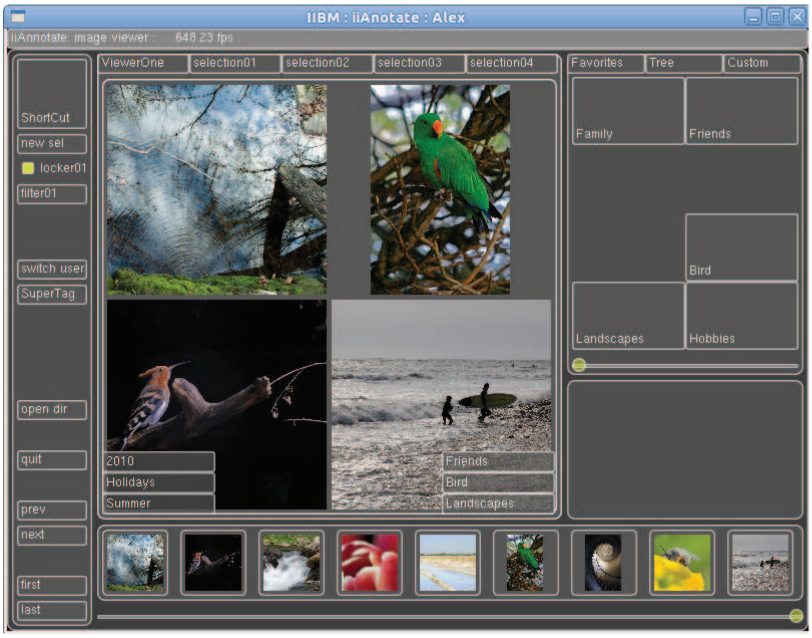

Fig. 1. IIANNOTATE: Interface of the application allowing for annotating photos.

Alex uses the IIANNOTATE application to annotate his photos. He annotates a first folder of photos by assigning the following tags:

– *holiday*, *summer*, *2010* and *bird* to the first two photos;
– *holiday*, *summer*, *2010* and *friends* to the third photo;
– *holiday*, *summer*, *2010* and *landscape* to the fourth photo.

When he starts to annotate the fifth photo with *holiday*, *summer*, *2010*, the system identifies that he has performed similar actions in the past (retrieval). Relying on its knowledge base, the system recommends a suitable action in this situation, which is "creating a meta-tag" (a meta-tag is a shortcut for a group of tags). Alex accepts this recommendation and creates a meta-tag for the group of tags *holiday + summer + 2010*. Alex is now able to annotate his photos with three tags in a single action. He can even associate a personal gesture to this meta-tag. In this first part of our scenario, CBR is used for two different purposes. First, it enables the system to identify a repetitive action from the user and to provide him with appropriate assistance. Next, it is used to learn the gesture that the user wants to associate to a specific action (namely, annotating a picture with a specific meta-tag).

Then, Alex annotates a second folder containing photos taken at Christmas. All photos are tagged with *Christmas* and *holiday*, and five of them with the tag *Christmas Tree*. A second user, John, uses the same application to annotate his photos. It must be noted that, even if John has his personal space with his own photos and his own tags, he has access to data and knowledge belonging to other users and can reuse them if he wants. John annotates his holiday photos taken during the summer 2010. He annotates the first photo with *holiday*, *summer*, *2010* and *daddy*. When he annotates the next photo using *holiday*, *summer*, *2010* for the second time, the system immediately proposes to create a meta-tag. This is made possible because iiAnnotate supports experience sharing between users. Actions performed by John were analyzed on the fly, and the system identified a similar action in its case-base. In a previous situation (case of Alex), the system provided a user with assistance and this assistance was successful. Therefore, the system provides the same assistance again. Let's assume that John is also happy with the proposed assistance and accepts to create the recommended meta-tag. The system provides him with a second form of assistance. It shows him the gesture learned from Alex and asks if John wants to use the same gesture or create another one. This second part of the scenario demonstrates how CBR enables experience sharing between users and improves efficiency of the annotation process by providing relevant assistance.

Then John annotates a second folder of photos. He begins by using *Christmas Tree* and *holiday* tags. The system recommends to complete these tags with the tag *Chrismas* because every time Alex has used *Christmas Tree* and *holiday* tags, he has completed with the *Chrismas* tag. In this case, the system uses the experience of another user to provide additional annotations.

John annotates a third folder using the three consecutive tags: *Savoie*, *mountain* and *summer*. The system uses the experience of Alex for suggesting, faster than the first time, the creation of a meta-tag and a gesture. Indeed, iiAnnotate is able generalize learnt experiences. In this case, the action of recommending the creation of a meta-tag when three tags are often used together has been generalized from the recommendation which has been made to Alex beforehand.

iiAnnotate is still under development, and some parts of this scenario are not implemented. Our initial results demonstrate that the CBR approach that we applied provides relevant results and helps users with the annotation task. Further developments will allow us to get more results that will be published on the iibm project web site[2] as soon as they are available.

4 Towards Intelligent Interactions in Video Games

Our main objective with the iibm project is to apply intelligent interactions to video games. However, for purely technical reasons (development time), our first test application is not a video game. Yet, we want to investigate this issue in the near future. In this section, we present a quick summary of our thoughts about the application of the iibm approach to video games.

Video games need richer interactions. We believe that the potential of intelligent interaction should be taken into account while designing games. This should enable game designers to invent new ways of playing and to be more creative in their way of designing game play.

[2] iibm project: `http://liris.cnrs.fr/iibm`

We thought of using intelligent interactions in several kinds of games. One of the first applications we had in mind was sport games (dance, soccer...). In our laboratory, since 2009, a webcam-based demonstration shows a proof of concept in a fun way[3]. In 2010, this demonstration was adapted with a *time-of-flight* camera[4] (before the Kinect release). A similar game is now commercialized [24]. Nevertheless, this game is really impressive from a technical standpoint, it is actually very classical in terms of game play. Our ambition is to go a step further in the game play by developing smart and truly interactive games.

Another natural application is First Person Shooter (FPS). Indeed, even if hard core gamers are more interested by efficiency, casual gamers want to have fun. Including more interactivity in such games would enable players to identify themselves to their avatar. Sniping, sword fights or boxing will be enriched from motion interpretation and negotiation.

These applications are not the most interesting ones. Role Play Games (RPG) are created from dreamlike vision and gamers play with metaphors. More intelligent interactions would give gamers more freedom to create their own metaphors. For example, a druid can co-learn a spell with the system. A novice druid can learn a new spell from an initial pattern. His own way of performing the gesture (intensity, speed, frequency) can lead to different consequences on the game. The same thing can be done with a warrior (sword fight), the rogue (stole a purse), *etc.*

RPG are attractive targets, but other applications such as Real Time Strategy Games (RTS) can be envisioned. Traditionally, these games are designed to be controlled through keyboard and mouse. These games are not very popular on consoles because they are very hard to control with a pad. Nonetheless, if we could control such games with gestures, we could invent a new way of playing. One idea is to design a game where the user can dynamically create his own gestured shortcuts to complex actions (for example: select all the units in a specific zone), and where the game is able to *understand* this shortcut. Such a new interface could provide us with the opportunity to control gaming universes as never before.

In all these scenarios, we have to combine artificial intelligence and motion interpretation in order to design new intelligent and usable interfaces for game play and game design.

5 Conclusion and future work

In this paper, we argue that combining motion acquisition techniques and artificial intelligence methods can lead to richer interactions between users and systems. Richer interactions could bring a considerable gain not only for video games but also for everyday systems. Our approach belongs to the field of natural interactions. We develop motion-based interactive systems able to negotiate

[3] 2^{nd} video on `http://liris.cnrs.fr/~bmichoud/english/index.shtml`

[4] Mesa Imaging, SR4000, `http://www.mesa-imaging.ch`

with their users in order to co-construct knowledge about good practices of interactions. Our goal is to make simple the lives of users and to facilitate the appropriation of the systems by making them self-adaptive.

This research is only at its beginning, but we believe in the potential of the approach. In the near future, we plan to continue the implementation of our test bed environment with the application iiAnnotate. Simultaneously, we are going to apply this approach to other application domains, and our first project is to design an experiment with a video game. We are interested in investigating the impact of intelligent interactions on the game play as well as on the game design. This objective is ambitious and difficult. Our goal is to improve the game by providing even richer opportunities for interactions, while taking care not to spoil the interest of the game.

This project opens new research issues in both fields. Results are expected in the field of gesture recognition. We hope that taking into account knowledge about users will improve the quality of motion acquisition. Regarding the CBR field, we hope that this starting work will provide us with new sources of inspiration and new successful examples of the benefits of CBR for experience reusing and experience sharing.

References

1. OpenNI: The Open Natural Interaction Organization. `www.openni.org`
2. Stuber, A.: Co-construction de sens par négociation pour la réutilisation en situation de l'expérience tracée. Thèse de doctorat en informatique, Université Lyon 1 (December 2007)
3. Sears, A. and Jacko, J.A.: The human-computer interaction handbook: fundamentals, evolving technologies, and emerging applications (2nd Edition). Human factors and ergonomics. Lawrence Erlbaum Associates (2008)
4. Krüger, Björn and Tautges, Jochen and Weber, Andreas and Zinke, Arno: Fast local and global similarity searches in large motion capture databases. In: Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation. SCA '10, Aire-la-Ville, Switzerland, Switzerland, Eurographics Association (2010) 1 – 10
5. Zheng, Yue and Hicks, Yulia and Marshall, Dave and Cosker, Darren: Real-Time Generation of Interactive Virtual Human Behaviours. In Ranchordas, AlpeshKumar and Araújo, Hélder J. and Pereira, João Madeiras and Braz, José, ed.: Vision and Computer Graphics. Theory and Applications. Volume 24 of Communications in Computer and Information Science. Springer Berlin Heidelberg (2009) 70 – 82
6. Stumpf, Simone and Rajaram, Vidya and Li, Lida and Wong, Weng-Keen and Burnett, Margaret and Dietterich, Thomas and Sullivan, Erin and Herlocker, Jonathan: Interacting meaningfully with machine learning systems: Three experiments. Int. J. Hum.-Comput. Stud. **67** (August 2009) 639–662
7. Molineaux, M.; Aha, D., Moore, P.: Learning Continuous Action Models in a Real-Time Strategy Environment. In: Proceedings of the Twenty-First International Florida Artificial Intelligence Research Society Conference (FLAIRS 2008). Volume Coconut Grove, USA, May 15-17. AAAI Press. 257–262
8. Puga, G.F.; Díaz-Agudo, B., González-Calero, P.: Experience-Based Design of Behaviors in Videogames. In: Proceedings of the European Conference on Case-Based

Reasoning (ECCBR 2008). Volume Trier, Germany, September 1-4. Springer. 180–194
9. Hulpus, I.; Fradinho, M., Hayes, C.: On-the-Fly Adaptive Planning for Game-Based Learning. In: Proceedings of the International Conference on Case-Based Reasoning (ICCBR 2010). Volume Alessandria, Italy, July 19-22. Springer. 375–389
10. Freeman, W.T and Tanaka,K. and Ohta, J. and Kyuma, K.: Computer vision for computer games. Automatic Face and Gesture Recognition, IEEE International Conference on **0** (1996) 100
11. Perini, E. and Soria, S. and Pratiand Rita Cucchiara, A.: FaceMouse: A Human-Computer Interface for Tetraplegic People. In: Computer Vision in Human-Computer Interaction. 2006. (2006) 99–108
12. Kilambi, P. and Masoud, O. and Papanikolopoulos, N.: Crowd Analysis at Mass Transit Sites. In: IEEE International Conference on Intelligent Transportation Systems. (2006) 753–758
13. Zhang, Zhengyou and Wu, Ying and Shan, Ying and Shafer, Steven: Visual panel: virtual mouse, keyboard and 3D controller with an ordinary piece of paper. In: PUI '01: Proceedings of the 2001 workshop on Perceptive user interfaces, New York, NY, USA, ACM (2001) 1–8
14. Shanqing Li and Jingjun Lv and Yihua Xu and Yunde Jia: EyeScreen: A Gesture Interface for Manipulating On-Screen Objects. In: Human-Computer Interaction. HCI Intelligent Multimodal Interaction Environments. (2007) 710–717
15. Ayers, D., Shah, M.: Monitoring human behavior from video taken in an office environment. Image and Vision Computing **19**(12) (2001) 833–846
16. Cuntoor, N., Yegnanarayana, B., Chellappa, R.: Activity modeling using event probability sequences. IEEE Trans. Image Processing **17**(4) (April 2008) 594–607
17. Ivanov, Y.A., Bobick, A.F.: Recognition of Visual Activities and Interactions by Stochastic Parsing. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 852–872
18. Poppe, R.: A survey on vision-based human action recognition. Image and Vision Computing **28**(6) (2010) 976 – 990
19. Mille, A.: From case-based reasoning to traces-based reasoning. Annual Reviews in Control **30**(2) (October 2006) 223–232 Journal of IFAC.
20. Cordier, A., Mascret, B., Mille, A.: Dynamic Case Based Reasoning for Contextual Reuse of Experience. In Marling, C., ed.: Provenance-Awareness in Case-Based Reasoning Workshop. ICCBR 2010. (July 2010) 69–78
21. Cram, D., Cordier, A., Mille, A.: An Interactive Algorithm for the Complete Discovery of Chronicles. Technical Report RR-LIRIS-2009-011, LIRIS UMR 5205 CNRS/INSA de Lyon/Université Claude Bernard Lyon 1/Université Lumière Lyon 2/École Centrale de Lyon (April 2009)
22. Stuber, A., Hassas, S., Mille, A.: Language games for meaning negotiation between human and computer agents (October 2005)
23. Cordier, A., Mascret, B., Mille, A.: Extending Case-Based Reasoning with Traces. In: Grand Challenges for reasoning from experiences, Workshop at IJCAI'09. (July 2009)
24. MTV Game: Dance Central. `http://www.dancecentral.com`