Functional Gestures for Human-Environment Interaction

Stefano Carrino[1,2], Maurizio Caon[1], Omar Abou Khaled[1],
Rolf Ingold[2], and Elena Mugellini[1]

[1] University of Applied Sciences of Western Switzerland, Fribourg, Switzerland
{Stefano.Carrino,Maurizio.Caon,Omar.AbouKhaled,
Elena.Mugellini}@Hefr.ch
[2] University of Fribourg, Fribourg, Switzerland
{Stefano.Carrino,Rolf.Ingold}@unifr.ch

Abstract. In this paper, we describe an opportunistic model for human-environment interaction. Such model is conceived to adapt the expressivity of a small lexicon of gestures through the use of generic functional gestures lowering the cognitive load on the user and reducing the system complexity. An interactive entity is modeled as a finite-state machine. A functional gesture is defined as the semantic meaning of an event that triggers a state transition and not as the movement to be performed. An interaction scenario has been designed in order to evaluate the features of the proposed model and to investigate how its application can enhance a post-WIMP human-environment interaction.

Keywords: natural interaction, functional gestures, pervasive computing, human-computer interaction.

1 Introduction

This paper presents a model for the design of an opportunistic system for natural human-environment interaction. Natural interaction approaches aim to facilitate the control of technological devices through the use of the communication modalities typical of the human-human interaction [1]. Gestures, speech, gaze are few examples of typical modalities. Although natural paradigms have been conceived to improve learnability, several gesture-based applications 1) lack of expressivity (small lexicon) or 2) have a significant cognitive load for the user caused by the big number of gestures.

In this paper, we address these issues proposing an opportunistic context-aware model conceived to augment the expressivity of a small lexicon of gestures. The small size of the lexicon reduces the impact on the user cognitive load, whereas the opportunistic approach augments the vocabulary expressivity, with the results of an increased expressivity and a reduced cognitive load. A reduced number of gestures lowers the complexity of the system improving also the accuracy rate when using machine learning techniques. In fact, a classifier trained on a small number of gestures generally outperforms in terms of recognition accuracy the same system trained on a larger number of gestures [2]. Section 2 presents the works related to this project

M. Kurosu (Ed.): Human-Computer Interaction, Part IV, HCII 2013, LNCS 8007, pp. 167–176, 2013.

in the field of gesture recognition and gesture vocabulary design. Section 3 defines the main concepts of the presented model. Section 4 details our model. Section 5 validates the proposed model discussing an interaction scenario. Finally, Section 6 discusses the achieved results.

2 Related Work

Several studies have investigated the definition and design of a gesture vocabulary for the natural interaction with objects, the environment and invisible computers (post-WIMP era). The definition of specific and generic gesture taxonomies is an important preliminary step designing the features of the interaction between the human and the machine. Researchers in HCI have proposed conceptual frameworks mixing gestures physical expression and semantics in the taxonomy (such as [3], [4] and [5]). For instance, in [3] Quek et al. define manipulative and semaphoric gestures. The first class involves "a tight relationship between the actual movements […] with the entity being manipulated"; the second "any gesturing system that employs a stylized dictionary of static or dynamic hand or arm gestures". These classes emphasize the relation of the gesture with the entity of the interaction and the signification of the gestures for the user.

Most of the studies on gesture interaction define an ad-hoc or rule-based gesture vocabulary to be used in the interaction (such for example in [6], [7] and [8]). Stern et al. [9] propose the definition of a gesture vocabulary based on psycho-physiological and technical factors for one-way (a human that commands a machine) communication. Differently from the approach we propose in this paper, the authors limit their study to the assumption one gesture – one command. Several studies take into account co-verbal gestures [3] [10], in order to study the relation between the gestures and speech or to extract multimodal information. In contrast, we focus on the single gestural modality leaving multimodal aspects to further studies. Researchers in the cognitive science domain are studying how object shapes can evocate functional and volumetric gestural knowledge [11]. In their work, Bub et al. define functional gestures as "gestures associated with the conventional uses of objects". Starting from this definition, we extend it toward generic entities that can be real or virtual, associating the function evocation to the semantic meaning.

The gesture vocabulary design can be done following different approaches. Akers et al. [12] propose an observation-based design to reveal the optimal gestures for a given task. However, designing a gestures vocabulary implies taking into account many aspects. In fact, Prekopcsák et al. [13] identify four main design principles for everyday hand gesture interfaces in ubiquity, unobtrusiveness, adaptively and simplicity. Our approach takes into account these four aspects with a special attention to the adaptation parameter. Also Nielsen et al. [14] propose an interesting approach for developing gestural interfaces focusing on parameters as intuitiveness and ergonomics. The authors define some directives to follow in order to adhere to the most important principles in usability and ergonomics. They state that two are the possible approaches for the investigation of suitable gestures for HCI interfaces: bottom-up

and top-down. In particular, the bottom-up approach consists of taking the functions and finding the matching gestures. Our model affects this part of their procedure introducing the concept of functional gesture defined in the next section. Our procedure enhancement aims at providing a smaller gestures vocabulary in order to go beyond the learnability obtainable with standard approaches improving the recognition accuracy at the same time.

3 Definitions

3.1 Interactive Entities

Through gestures, users can interact with devices and tools that can be real or virtual. In this paper, we generically call these devices *interactive entities.*Our model classifies the interactive entities according to the following 2-elements taxonomy: *two-state entities* and *complex entities*. **Two-state entities** are simple entities that are characterized from having just the states ON and OFF. Lamps could typically belong to this category. **Complex entities** can be modeled by a finite-state machine representation with more than 2 states, in which each transition defines an action. It is convenient to distinguish the two-state entity class for the wide availability of devices that can fit this class and can be described with the same model.

On the other hand, the state machine representation of a complex entity is strictly linked to the functions of the device. Although automatic state-machine generation, configuration and deployment are not the focus of this paper research, solutions based on an ontological description of the interactive entities (such as presented in [15]) can help this process: ontologies can abstract heterogeneous devices as homogeneous resources.

3.2 Interaction Expressivity versus Cognitive Load

Combinations of multiple gestures to provide complex and rich commands to a system are a challenging mean of interaction. The effort required to learn or reproduce such language requires the users to make a remarkable effort. In common approaches, with the exclusion of sign language alphabets, it is rare to find gestural sentences composed of a concatenation of multiple gestures. And, consequently, the most diffused approach is to associate one gesture to one command. If such solution works well in systems and applications with reduced needs of expressivity, it can fall short to control complex interfaces (complex not complicate interfaces [16]).

On the other hand augmenting the vocabulary size is not always a viable solution. Previous studies assessed that a big number of commands can have a sensible impact on the user cognitive load and the associated information can be difficult to process. According to Miller's seminal work, seven "*plus or minus two"* appears to be the upper limit in the number of information that can still be processed with a not excessive load on the cognitive processing capacity of our brain [17]. Based on these results, we empirically limited the number of gestures for our interaction scenario to 8. These gestures are: select, turn on/off, next, previous, undo, increase, decrease and

exit (their functions and particularities are detailed in section 3.4 and Table 1). We can observe that there are not specifications or limitations on how to perform a gesture.

3.3 Functional Gestures

As mentioned before, we focus on gesture classification based on the function of a gesture and not the physical movements or posture. From this perspective, a gesture, or more in general a command, does not imply constraints about its physical realization, improving flexibility and user adaptation. From a more general, multimodal, point of view, a command can be provided using different communication channels. According to our classification, if a command function remains the same, the command belong to the same functional command class.

But what is a function? Representing an entity as a finite-state machine, a *function* is the semantic meaning of an event or condition that triggers a transition. An *action* is a specific transition. Examples of functions are select, next element, undo, etc. A *functional gesture* is linked to the concept of function instead of action (in conformity with the conscious gestures of semantic type descripted in [14]). The functional gestures are strictly connected to the functions of the entity that we are interacting with. For instance, the *next element* function has not meaning for a two-state entity.

The aim and the advantage of this approach is the abstraction between the physics of the gestures and its interpretation. A system configured to respond to functional gestures does not force the users to specific movements and can implement a one to many or many to one relation (i.e., one gesture for multiple actions).

In this research, we limit our lexicon to 8 functional gestures: select, turn on/off, next, previous, undo, increase, decrease and exit. Table 1 presents the 8 functional gestures integrated in our taxonomy; we emphasize in italic some of the main assumptions that should be taken into account designing an interface according to our model.

Table 1. Functional gestures: function-action association

Function	Action
Select	Select the *highlighted* element (for a specific entity in a certain state)
Turn on/off	Two actions: switch the highlighted entity condition: OFF ->ON or ON->OFF
Next	Highlight the next element in a *sorted* list
Previous	Highlight the previous element in a *sorted* list
Undo	Undo the last command
Increase	Increase the *main property* of the highlighted entity
Decrease	Decrease the main property of the highlighted entity
Exit	Exit from the current state

Highlighting implies a form of feedback to the user. It can be visual, acoustic, or multimodal.

Sorting implies the concept of order between the different elements.

Increase and decrease should be applied to a *main property* of an element. The degree of relevance of the property can change with the state of the entity. For example, interacting with a media center in the *play movie* state, the increase and decrease function can be associated to the loudness of the volume, whereas in the pictures browser slide-show they can be associated with the zoom level on the images.

Finally, the exit command implies to have a state-machine representation aware of the application interface (this can imply the need to memorize the historic of the interaction).

4 Model and Design Directives

The proposed model aims to enhance the interaction between the human and the environment finding a good balance between cognitive load and vocabulary expressiveness, in the context of gesture-based interaction. In smart environments the gesture interaction can be very varied. In order to address these challenging issues and focus on our research, we fixed some design directives.

4.1 Design Directives

We identified a number of rules and constraints that the gesture lexicon and the environmental interfaces of a smart home should respect in order to be modeled with the proposed approach.

Interaction lexicon should:

1. Have a moderate number of gestures to reduce the cognitive load for the user that have to recall the interaction to perform. E.g., seven more or less two is the range of numbers suggested by Miller in [17] and that we adopted in our scenario.
2. Define a set of semantic meanings, and not the cinematic and dynamic of the gesture itself that the designer can freely choose. Such meanings should be generic. Based on the research of Neßelrath et al. [18], in our scenario we propose 8 functions that we associated to the selected semantic meanings.
3. Be designed following the Nielsen et al. procedure [14] in order to adhere specific ergonomics and usability principles.

Environmental feedback interfaces should:

4. Be designed to be compatible with the generic meaning of the gesture vocabulary and increase the intuitiveness of the interaction. Section 4.4 discusses typical issues that should be taken into account designing the feedback for the user.

Functional gestures must be dynamically associated to precise *actions* on the entities present in the environment exploiting contextual information. In particular, we use two kinds of contextual information: the system state and the environmental data. The environmental data contain generic information such as lighting conditions, noise levels, user position and activity.

4.2 System as State Machine

The entity model we proposed is based on a finite-state machine representation of the devices present in the environment. Gestures are interpreted differently according to the current device state. Fig. 1 shows an example of state machine modeling a simple 5-states device. The figure illustrates an interaction with a media center system (used in the scenario presented in section 5).

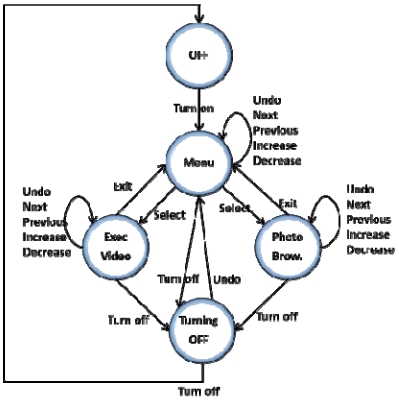

Fig. 1. Finite-state machine representation of a 5-state device (media center with video and photo browsing capabilities)

4.3 Contextual Information

Lighting conditions, noise levels, user position, user activity and the entity state are information that can be used in order to model the interaction properties from both interactive and technological perspectives.

From the point of view of the interaction, the context information can be used to propose the more appropriate gesture, modality or feedback (a seminal work on this subject is presented by Cheverst et al. [19]; we presented a prototype of context-based generation of multimodal feedback in [20]).

From a technological point of view, the contextual information can be used to improve the recognition task, for example selecting the most appropriate sensor in different conditions: low lighting conditions imply the use of sensors not based on RGB cameras, a highly noisy environment discourages the use of microphones or acoustic feedback, etc.

4.4 Feedback Design

In order to reduce the cognitive load on the user interacting with the environment the feedback design plays a crucial role. A highly dynamic environment should facilitate the interaction and help the user avoiding ambiguous interpretations of the system state. In fact, without an appropriate feedback the user can just assume, based on his

experience, the current state of the environment. Therefore, the interface should be able to adapt itself according to the user experience in the interaction. Novice users have greater needs of cues facilitating the interaction; on the other hand, expert users can feel such feedback as unnecessary, intrusive or distracting from the main task of the interaction [21]. The feedback design should take into account the following possibilities and needs:

- Visual (or multimodal) information suggesting the environment/entity state and the available interactions.
- Confirmation interface/state for important, long to undo actions. For example, the complete switch off of a device can imply a long time in order to return to a previous state (for instance, the turning off state in our scenario responds to this need).

5 Evaluation Scenario

We designed a scenario aiming at evaluating the features of our approach following the first two phases of the human-based experiment presented in [14]. We used this scenario in order to estimate the benefits and limits of the proposed approach. We use the user location (as proposed in [22]), the entity target of the interaction (as proposed in [23]) and its state as contextual information. The interactive entities are a media center, a radio, a lamp and a fan that can be remotely controlled (as depicted in Fig. 2). Our model uses the contextual information to dynamically adapt the link between the functional gestures and its related action.

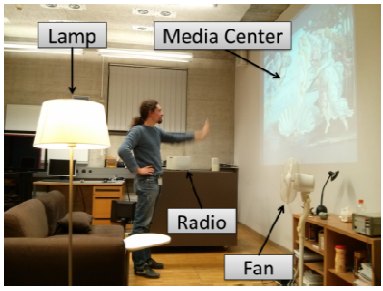

Fig. 2. Second scenario with a user and the interactive entities

In this section, we detail the interaction features related to the media center. For the other appliances we used a similar approach.

8 functional gestures are used in the interaction. The media center is modeled with a 5 finite-state machine: Off, Menu, Executing a video playlist, Photo browsing and Turning off (see Fig. 1). Table 2 details the dynamic, state-dependent links gesture-action. For instance, the functional gesture “turn on/off” is translated to the action “turn on” if the media center is in the Off state, “go to the turning off menu” if in the Menu, Executing a video playlist and Photo browser states, and “turn off” in the Turning off state.

Table 2. Functional gestures designed for a media center finite-state machine

Media center State / Gesture	Off	Menu	Executing a video playlist	Photo Browsing	Turning off
Select	-	Select field	-	-	-
Turn on/off	Turn On	Turning Off menu	Turning Off menu	Turning Off menu	Turn off
Next	-	Next element	Next video	Next pic.	-
Previous	-	Previous element	Previous video	Previous Pic.	-
Undo	-	Undo	Undo	Undo	Undo
Increase	-	Volume Up	Volume Up	Zoom in	-
Decrease	-	Volume Down	Volume Down	Zoom out	-
Exit	-	-	Go to Menu	Go to Menu	-

In this example, 8 functional gestures are needed in the interaction. A standard approach with a relation one-to-one between the gestures and the actions needs 15 gestures (select, turn on, turn off, next element, previous element, next picture, previous pictures, next video, previous video, volume up, volume down, zoom in, zoom out, go to menu, undo).

Table 3 compares our approach with classical methodologies. Each entity is characterized by a finite state machine and a set of *actions*. The radio features 6 actions: turn on, turn off, next channel, previous channel, volume up, volume down. The fan and the lamp are modeled as 2-state entities.

Table 3. Number of gestures needed per device per approach. The functional approach is our contribution.

Interactive entities	Simple approach	Entity-aware	Functional
Media center	15	15	8
Radio	6	6	5
Lamp	2	2	1
Fan	2	2	2
Total number of gestures	25	15	8

In the first approach, we called “simple approach”, each device requires specific gestures and there is a mapping one-to-one between gestures and actions. The user should learn 25 gestures in order to fully interact with all the entities: 15 for the media center, 6 for the radio, 2 for the lamp and 2 for the fan. The “entity-aware” approach exploits the context information to recognize the target of the interaction but it is unaware of the entity state. This allows introducing 6 *functions* that model all the actions

for the radio, lamp and fan entities, and a subset of the media center actions. Therefore, the media center requires other 9 supplementary gestures to model the remaining actions, for a total of 15 gestures.

Finally, the third column presents our contribution: a state-aware system design that we called "functional" approach. In this case, the same functional gestures have different meanings according to the device state. With such approach, we can maintain the interaction expressivity limiting the number of gestures. In fact, the media center can be represented as finite-state machine (Fig. 1) and this allows defining 8 functional gestures that are enough to richly interact with all the entities in the environment as previously explained (Table 2). Such advantages are achievable only thanks to the abstraction work done at design time defining a specific state machine for each device. As mentioned before, ontology based approaches can help reducing this limitation.

6 Conclusion

The presented paper shows a natural interaction, context-aware approach that maximizes the lexicon expressivity of a limited set of gestures based on functions reducing the cognitive load on the user. In addition, a small number of gestures lowers the complexity of the system improving the accuracy rate of a classifier. The introduced 8-gesture vocabulary represents a generalized instance of our model and can be adapted to several different contexts for human-environment interaction in the post-WIMP era.

We can observe that in a domestic environment, interactive devices are typically simple entities that are easy to model and categorize. On the other hand, a bigger effort is required to model complex entities and the feedback to the user. For this reason we provide some design directives that in conjunction with the work of Nielsen et al. [14] constitutes a complete guideline for natural gestural interfaces design.

References

1. Carrino, S., Mugellini, E., Abou Khaled, O., Ingold, R.: ARAMIS: Toward a Hybrid Approach for Human-Environment Interaction. In: Jacko, J.A. (ed.) Human-Computer Interaction, Part III, HCII 2011. LNCS, vol. 6763, pp. 165–174. Springer, Heidelberg (2011)
2. Wachs, J.P., Kölsch, M., Stern, H., Edan, Y.: Vision-based hand-gesture applications. Communications of the ACM 54, 60 (2011)
3. Quek, F., McNeill, D., Bryll, R., Duncan, S., Ma, X.-F., Kirbas, C., McCullough, K.E., Ansari, R.: Multimodal human discourse: gesture and speech. ACM Transactions on Computer-Human Interaction 9(3), 171–193 (2002)
4. Karam, M., Schraefel, M.C.: A taxonomy of gestures in human computer interactions (2005)
5. Aigner, R., Wigdor, D., Benko, H., Haller, M., Lindlbauer, D., Ion, A., Zhao, S., Tzu, J., Valino, K.: Understanding Mid-Air Hand Gestures.: A Study of Human Preferences in Usage of Gesture Types for HCI, Microsoft Research Technical Report (2012)

6. Baudel, T., Beaudouin-Lafon, C.: Remote Control of Objects using FreeHand Gestures. Communications of the ACM 36(7), 28–35 (1993)
7. Kjeldsen, R., Hartman, J.: Design Issues for Vision- based Computer Interaction Systems. In: Proc. of the Workshop on Perceptual User Interfaces, Orlando, Florida, USA (2001)
8. Abe, K., Saito, H., Ozawa, S.: Virtual 3-D Interface System via Hand Motion Recognition from Two Cameras. IEEE Trans. Systems, Man and Cybernetics, Part A 32(4), 536–540 (2002)
9. Stern, H.I., Wachs, J.P., Edan, Y.: Optimal Hand Gesture Vocabulary Design Using Psycho-Physiological and Technical Factors. In: 7th International Conference on Automatic Face and Gesture Recognition (FGR 2006), pp. 257–262 (2006)
10. Kettebekov, S., Sharma, R.: Toward natural gesture/speech control of a large display. Engineering for Human-Computer Interaction 2254, 221–234 (2001)
11. Bub, D.N., Masson, M.E.J., Cree, G.S.: Evocation of functional and volumetric gestural knowledge by objects and words. Cognition 106(1), 27–58 (2008)
12. Akers, D.L.: Observation-based design methods for gestural user interfaces. In: CHI 2007 Extended Abstracts on Human Factors in Computing Systems - CHI 2007, pp. 1625–1628. ACM (2007)
13. Prekopcsák, Z., Halácsy, P., Gáspár-Papanek, C.: Design and development of an everyday hand gesture interface. In: Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services - MobileHCI 2008, p. 479 (2008)
14. Nielsen, M., Moeslund, T., Storring, M., Granum, E.: A procedure for developing intuitive and ergonomic gesture interfaces for HCI. In: Proc. 5th Int. Workshop on Gesture and Sign Language based HCI, Genova, Italy (2003)
15. Sommaruga, L., Formilli, T., Rizzo, N.: DomoML - an Integrating Devices Framework for Ambient Intelligence Solutions. In: Proceedings of the 6th International Workshop on Enhanced Web Service Technologies - WEWST 2011, pp. 9–15 (2011)
16. Norman, D.A.: Living with complexity. MIT Press (2010)
17. Miller, G.A.: The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. The Psychological Review 63, 81–97 (1956), http://www.musanim.com/miller1956
18. Neßelrath, R., Lu, C., Schulz, C.H., Frey, J., Alexandersson, J.: A Gesture Based System for Context-Sensitive Interaction with Smart Homes. In: Ambient Assisted Living, 4. AAL-Kongress, pp. 209–219. Springer (2011)
19. Cheverst, K., Davies, N., Dix, A., Rodden, T.: Exploiting Context in HCI Design for Mobile Systems. In: First Workshop on Human Computer Interaction for Mobile Devices, pp. 1900–1901 (1998)
20. Perroud, D., Angelini, L., Khaled, O.A., Mugellini, E.: Context-Based Generation of Multimodal Feedbacks for Natural Interaction in Smart Environments. In: The Second International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT 2012), Barcelona, Spain, September 23-28 (2012)
21. Bau, O., Mackay, W.E.: OctoPocus. In: The 21st Annual ACM Symposium on User Interface Software and Technology - UIST 2008, p. 37. ACM Press, New York (2008)
22. Greenberg, S., Marquardt, N., Ballendat, T., Diaz-Marino, R., Wang, M.: Proxemic interactions. Interactions 18(1), 42 (2011)
23. Carrino, S., Péclat, A., Mugellini, E., Abou Khaled, O., Ingold, R.: Humans and Smart Environments: A Novel Multimodal Interaction Approach. In: Proceedings of the 13th International Conference on Multimodal Interfaces - ICMI 2011, p. 105 (2011)