Eye gaze assisted human-computer interaction in a hand gesture controlled multi-display environment

Tong Cha
Fraunhofer Institute of Optronics, System Technologies and Image Exploitation IOSB
Fraunhoferstr. 1
76131 Karlsruhe, Germany
tong.cha@iosb.fraunhofer.de

Sebastian Maier
Fraunhofer Institute of Optronics, System Technologies and Image Exploitation IOSB
Fraunhoferstr. 1
76131 Karlsruhe, Germany
sebastian.maier@iosb.fraunhofer.de

ABSTRACT

A special human-computer interaction (HCI) framework processing user input in a multi-display environment has the ability to detect and interpret dynamic hand gesture input. In an environment equipped with large displays, full contactless application control is possible with this system. This framework was extended with a new input modality that involves human gaze in the interaction. The main contribution of this work is the possibility to unite any types of computer input and obtain a detailed view on the behaviour of every modality. Information is then available in the form of high speed data samples received in real time. The framework is designed with a special regard to gaze and hand gesture input modality in multi-display environments with large-area screens.

Categories and Subject Descriptors

H.5.2 [**INFORMATION INTERFACES AND PRESENTATION**]: User Interfaces

General Terms

Human Factors

Keywords

gaze based interaction, hand gesture interaction, multimodal interfaces

1. INTRODUCTION AND RELATED WORK

Results of explorative user studies show that natural gaze as an additional input modality can improve human-computer interaction [1]. In one experiment, the participants were instructed to drag-and-drop several virtual figures to a new position by performing hand gestures while recording gaze behaviour continouosly. This experiment was accomplished in a multi-display environment with extensive displays and an active gesture recognition system so that gaze behaviour combined with hand gesture input in target-oriented interaction and other gaze-based aspects could be analyzed. The surveys show that natural gaze behaviour applied in interactive environments contain useful information which can be gainfully used to design multimodal interaction techniques [3]. Based on this work [3], this paper constitutes a disquisition of the question how detailed gaze behaviour and any other modality in HCI can be analyzed and how to deal with the specific features of various input modalities, both in real time. Further more, gaze behaviour has valuable information about the attention of the user while interacting with computer systems. This information is useful in individual and prioritized visualization of information [5]. In *Gaze-Augmented Manual Interaction* techniques are investigated in which ways the combination of gaze and gesture input can improve interaction: "Gaze data may provide clues about an intention even before the corresponding action is carried out" [6]. What does this fact mean to the interaction in multi-display environments?

This paper discusses the aspect of integrating eye gaze and hand gesture input in a multi-display environment. The main focus is on the technical solution of harmonizing these two input modalities in an existing system designed in our laboratories.

2. THE INPUT MODALITIES

2.1 Gesture recognition

The detection of hand gestures is accomplished by a special video-based recognition system [4]. A stereo infrared camera set detects the hands and their single fingers in a particular area of the multi-display environment. In addition to finger and hand positions, the system detects and classifies 16 possible hand configurations which are shown in Figure 1. From every frame of the video sequences data is generated and then sent to the framework for processing.

Figure 1: Hand configurations [4]

4th Workshop on Eye Gaze in Intelligent Human Machine Interaction.
October 26, 2012, Santa Monica, California, USA.

2.2 Gaze detection

To obtain the gaze information, an eye tracker is used as shown in Figure 2. The device features two cameras: one is directed to the user's field of view (scene cam) and the other camera records one eye (eye cam).

Figure 2: Head unit of the eye tracker system [8]

The scene is equipped with up to 16 special markers (Figure 3). Once the user looks to the scene, the result of the detection is a datastream of 25 samples per second. Each data sample contains coordinates (*marker coordinates*) describing the detected pupil of the user as a point in relation to the left upper corner of the marker that is detected by the scene cam.

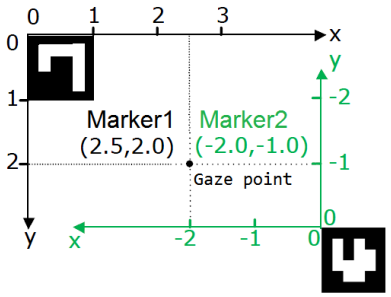

Figure 3: Gaze point relative to markers

3. THE INTERACTION FRAMEWORK

3.1 The framework components

The interaction framework is a distributed system designed to handle multimodal input in a multi-display environment. It can easily be extended with more displays and integrate various input modalities by uniting arbitrary input data (Figure 4).

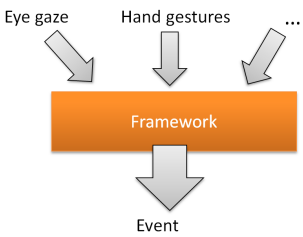

Figure 4: Uniting heterogeneous input data

All detection processes of the framework must be based on a fixed reference point in the scene. Moreover, all displays and input modalities are referenced to this point. This allows to combine different input modalities to create a multimodal HCI. As a result, the framework produces output events which can be mapped to traditional mouse and keyboard events on a display or be used to communicate directly with adapted software.

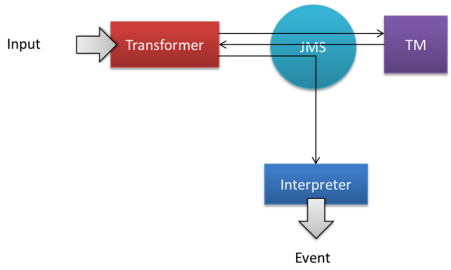

Figure 5: General framework architecture

There are three basic sections that constitute the framework (Figure 5):

Topology Manager (TM): This section holds a model of each registered display and input modality. It provides information of every item about its location in the scene (related to the common reference point). Concerning the displays in the scene, the TM provides information about the display size and resolution.

Transformer: This part of the framework is responsible for the transformation of coordinates from an input device (detector of hand-gestures, head-pose, eye gaze etc.) into *pixel coordinates* for every display in the scene. This can be done using the information provided by the TM. Every input modality owns a specific Transformer.

Interpreter: All transformed data comes together in the Interpreter in form of separate datastreams. After time synchronization of the streams, this module examines the datastreams according to predefined input patterns and triggers events which are directly sent to the output of the framework.

The framework is using the Java Message Service (JMS) to communicate via network.

3.2 Transforming sensor data

GestureTransformer

The specific transformation unit for the data coming from the gesture recognition system is the *GestureTransformer*. It calculates pixel coordinates (also called *display coordinates*) from the gesture detection data by correlating world and display coordinate systems (Figure 6). In addition to the currently performed hand configuration (see 2.1), information about hand gesture in relation to every registered display is then available in the Interpreter.

GazeTransformer

To detect the gaze position on a display (pixel coordinates) when the user looks at it, another specific transformer unit has to be involved in the framework. The *GazeTransformer* handles information applied by the eye tracking system. Relating to the position data of every marker and display which is all stored in the TM, the GazeTransformer calculates at which (pixel) point of the display the user is currently loo-

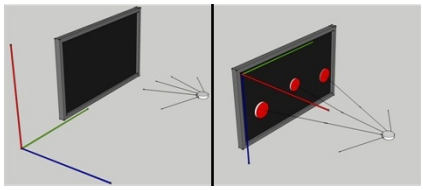

Figure 6: Principal of gesture transformation

king at. In this process, head moving does not affect the calculated results because the focus is on the gaze point relative to fixed markers in the scene.

3.3 Interpreting user input

During the examination of the datastreams, the Interpreter has all information about the currently used input modalities (gesture, gaze, touch etc.) and in which area or on which point of the displays the interaction takes place.

Recognizing predefined gesture patterns

The Interpreter uses a logic based approach based on predefined hand gesture patterns. Each of these patterns are directly related to determined data sequences in the streams. The patterns can be freely combined, for example, if two hand gesture patterns, *Hand Is Near Display X* as one and *Hand Perfoms A Pointing Gesture* as another pattern, are detected in the datastream, one event is created and sent to the output of the framework.

Detecting gaze fixation

The Interpreter was implemented to detect if the user has performed a gaze fixation on a display. Therefore, an approach was chosen which is based on the *Dispersion-Threshold Identification Algorithm* (I-DT) for identifying gaze fixations by Dario D. Salvucci and Joseph H. Goldberg [7].

4. THE MULTIMODAL APPROACH

As a proof of concept, an experimental implementation induced the Interpreter to detect if the user performs a pointing gesture and simultaneously fixes a point on the screen with the gaze. To accomplish this, the Interpreter analyzes both, the incoming datastreams of the GestureTransformer and the GazeTransformer and triggers an event when both patterns lie inside the same time of period (Figure 7). The system feedback is a popup on the display announcing that this pattern was detected.

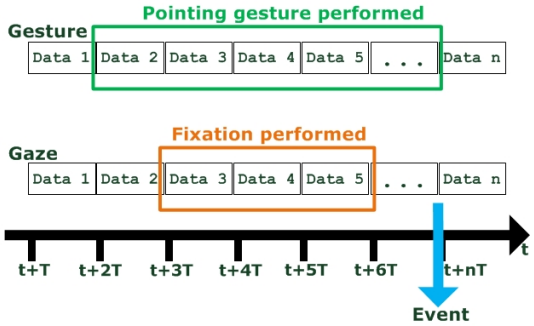

Figure 7: The multimodal approach

5. DISCUSSION AND CONCLUSION

First of all, it is important to mention that the implementations (especially pattern detection) of gaze interaction in this work are designed with the awareness that using gaze in an inappropriate manner (e.g. using gaze for explicit commands) might be more exhausting than using it in its natural form [2]. For now, any usage of gaze as an explicit input method will be avoided in this issue. The multimodal approach in this work can be extended in many ways. As an example, gaze could be used to anchor focus on one display in a multi-display environment. The anchoring would be disposed only by looking on the desired display then all gesture input are applied to this one.

In this paper, gaze input in a multimodal context, especially in a multi-display environment was discussed from a practical view. The question if gaze information can be technically integrated in an environment with large displays and gesture control can be answered positively.
So far, the application tests while designing the framework modules verified the capability of the Interpreter to detect pattern combinations of gaze and gesture, in spite of the challenging fact that the features of both input modalities, gaze and hand gesture, are completely different. To confirm the results and improve the multimodal approach, experimental evaluations are pending in the future.

6. REFERENCES

[1] T. Bader and J. Beyerer. Influence of user's mental model on natural gaze behavior during human-computer interaction. *2nd Workshop on Eye Gaze in Intelligent Human Machine Interaction*, Palo Alto, California, USA, February 2011.

[2] T. Bader and J. Beyerer. Putting Gaze into Context: A Framework for Analyzing Gaze Behavior in Interactive and Dynamic Environments. *International IUI 2010 Workshop on Eye Gaze in Intelligent Human Machine Interaction*, Hong Kong, China, February 2010.

[3] T. Bader, M. Vogelsang and E. Klaus. Multimodal Integration of Natural Gaze Behavior for Intention Recognition During Object Manipulation. *International Conference on Multimodal Interfaces & the Workshop on Machine Learning for Multimodal Interfaces*, Cambridge, Massachusetts, USA, November, 2009.

[4] T. Bader, R. Räpple and J. Beyerer. Fast Invariant Contour-Based Classification of Hand Symbols for HCI. *CAIP 2009, volume 5702 of LNCS, pages 689-696*, Springer, 2009.

[5] R. Vertegaal. Designing Attentive Interfaces. *Proceedings of the 2002 symposium on Eye tracking research & applications*, New Orleans, USA, 2002.

[6] H. Bieg. Gaze-Augmented Manual Interaction. *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, Boston, USA, 2009.

[7] D. D. Salvucci and J. H. Goldberg. Identifying Fixations and Saccades in Eye-Tracking Protocols. *Nissan Cambridge Basic Research, Dept. of Industrial and Manufacturing Engineering*, Pennsylvania, USA, 2000.

[8] Image courtesy of Ergoneers GmbH. Ergoneers – Ergonomic Engineers. *www.ergoneers.com*, 2011.