LGURJCSIT
Volume No. 1, Issue No. 2 (April- June) pp. 41-51

Multimodal Interaction Recognition Mechanism by Using Midas Featured By Data-Level and Decision-Level Fusion

Muhammad Habib, Noor-Ul-Qamar

Abstract:- Natural User Interfaces (NUI's) dealing with gestures is an alternative of traditional input devices on multi-touch panels. Rate of growth in the Sensor technology has increased the use of multiple sensors to deal with various monitoring and compatibility issues of machines. Research on data-level fusion models requires more focus on the fusion of multiple degradation-based sensor data. Midas, a novel declarative language to express multimodal interaction patterns has come up with the idea of developers required patterns description by employing multi-model interaction mechanism. The language as a base interface deals with minimum complexity issues like controlling inversion and intermediary states by means of data fusion, data processing and data selection provisioning high-level programming abstractions.

Keywords: *Degradation-based sensor data, Multimodal interaction patterns, MIML (Multimodal Interaction Markup Language), temporal combinations.*

1. INTRODUCTION

Starting in the late sixties, human-computer interaction has shifted from Command Line Interfaces (CLI) to Graphical User Interfaces (GUI). In recent years this trend is taken one step further by expanding human-computer interaction beyond the typical keyboard and mouse setup in attend called Natural User Interfaces (NUI) [1]. A

Muhammad Habib, MS(CS), Ncba&E, Faislabad,
Lecturer (CS), Lahore Garrison University, Lahore, Pakistan, e-mail ch.muhammadhabib@gmail.com
Noor ul Qamar, BS(CS) Lahore Garrison University, Lahore Pakistan
T.A. (CS), Lahore Garrison University, Lahore Pakistan, e-mail , noorulqamar@lgu.edu.pk

NUI is an interaction methodology which incorporates human skills such as touch, sight and body movement to enable human-computer interaction. Many NUIs rely on interaction patterns that are also used in everyday life. For example a virtual deck of cards can be dealt by swiping towards players, as if performed with real cards on a table.

a. *In a similar way, a baseball game with a NUI interface enables players to hit the ball by swinging their arms.*
b. *New commodity hardware facilitates the expansion of NUI applications. Devices such as Apple's iPad*
c. *HP's Sprout*
d. *Microsoft's Pixel Sense*
e. *Add a new dimension to human-computer interaction because one can touch, move and manipulate virtual digital objects in a natural way. Moreover, physical objects, in the form of tangibles, can be placed multi-touch table to initiate interaction.*

This paper is positioned at the crossroad of two major computer science domains, namely Human-Computer Interaction and Programming Language Design. This is due to the fact that the complexity to extract information coming from input devices is increasingly higher. Today's software-related limitations severely hinder the ability to experiment with novel interaction techniques. Additionally, the robustness of every day's use of these devices compared to traditional input methods decreases due to the inability to program user interface code at an adequate abstraction level. The goal of this research is to reduce the accidental complexity, the approach must facilitate the implementation of multimodal interaction patterns through high-level programming abstractions.

2. RELATED WORK

The World Wide Web Consortium’s (W3C) Multimodal Architecture and Interfaces (MMI Architecture) standard serves as an architecture and communication protocol that enables a wide variety of independent modalities to be integrated into multimodal applications. By encapsulating the functionalities of modality components and requiring all controlling information for the Interaction Manager, the MMI architecture

simplifies the task of components integration through various sources [2].

According to the physical medium of communication the computer's ability to create prototypes of other media and their accumulation into a single technology was recognized to bring an end to any substantial differentiation. Digitization opens existing media and their previously distinct modes recombination with second-order manipulation under the technological environment of the computer with unique methods of encoding [3].

An NUI system has the capability to combine its events from other connected NUI devices which produce a standard output. The conversion of low level input data into high-level NUI events being stored in repository is possible through NUI adaptors. A behavioral analysis engine uses the stored events of NUI products by combining their output for users according to behavioral pattern rules. This engine makes use of NUI events for preparing the system to create and utilize new pattern rules [4].

A feature level fusion technique named Discriminant Correlation Analysis (DCA) incorporating class associations feature sets into the correlation analysis. DCA through maximization of effective feature fusion performs the pairwise correlations across the two feature sets and eliminates the between-class correlations by restricting the correlations within the classes. DCA is the first technique that takes class structures in feature fusion. Moreover, it can be employed in real-time applications and has low computational complexity [5].

3. METHODOLOGY

To cope with the vast amount of low-level input events in soft real-time, the approach must facilitate an execution engine and react accordingly to the given multimodal descriptions. This includes dealing with segmentation (i.e. the process of extracting meaningful bits from continuous streams), and supporting overlapping matches (i.e. where input data can be shared between multiple multimodal descriptions).

To fuse low-level data with high-level data, the approach must facilitate cross-level multimodal fusion. Therefore it is stimulating due to the fact that low-level data and high-level data operate at die rent frequency rates. Additionally, the approach must integrate with

existing fusion processes, such as feature extractors, in order to reuse existing specialized methods with a small amount of development. The paper explains a quick development method for multi-modal dialogue system by applying MIML (Multimodal Interaction Markup Language). The MIML describes the dialogue patterns communicating between interactive agents and human. This language is featured by three-layered description of agent-based interactive systems as "device dependent realization" "task level description", and "interaction description". Several advantages of describing model extension, high-level interaction and compatibility are also a provision of this language [6].

3.1. Multimodal Concerns:

In their research, Leanne et al. express four important concerns a fusion engine needs to deal with probabilistic input, multiple and temporal combinations, adaptation to context, tasks and users, and error handling. In the following, we elaborate on these four concerns.

Probabilistic input Traditional applications rely on deterministic events such as keystrokes or mouse clicks. However, in most multimodal solutions developers rely on sensor information that should be interpreters. This interpretation needs to deal with a high degree of uncertainty due to sensor noise, such as background noise or convoluted image frames. Multimodal frameworks aim to reduce this uncertainty by ignoring noise, allowing relaxation parameters (such as spatial approximation via inter-19 Multimodal Interactional) or by embedding probabilistic information within the events. Dealing with probabilistic input is therefore a key concern for multimodal fusion engines.

3.2. Multiple and Temporal Combinations:

The CASE model is used to classify multiple and temporal combinations from a machine point of view. The CASE model categorizes die rent usage of modalities in concurrent, alternate, synergistic and exclusive. The concurrent category is dined as the parallel use of modalities without the need for time synchronization between the two modalities. If this information is combined such as the fusion of audio input with lip information to increase recognition rates the fusion process is synergistic. In the alternate category,

modalities are used in a sequential manner and information will be fused when the interaction is completed. An example is shown by Kuiper's et al. the user can enter text in natural language with word References (for instance the user types \what is the size of this disk?"), then clicks with the mouse on the graphical object representing a physical disk. Like users cannot turn the page via a swipe gesture while actively writing notes with the pen. Without exclusiveness, a moving palm on the touch interface might accidentally trigger the swipe gesture.

From a human perspective, the CARE properties characterize and assess the usability and fusion in multimodal interaction. Complementary modalities are used when an interaction is best achieved with two or more combined modalities. This allows for a more natural interaction such as using speech and pointing at the same time. The assignment property indicates the absence of a choice. The user is forced to use a single modality to reach their goal. On the other hand, when two modalities with the same expressiveness are required to be used to achieve one goal, we talk about redundancy. This can be used to limit unintentional actions, for example to delete a movie on a smart TV requires a voice command and of the head nodding. Fusion engines should support the CASE and CARE properties to deal with various multimodal scenarios, however this requires the ability to express advanced temporal relations between multiple input sources. Adaptation to context, tasks and users Adaptation to context can be an important factor to increase recognition rates.

3.3. Multimodal Fusion Levels:

The problem in the modeling of multimodal interaction using markup languages is mentioned keeping in view of the analysis of the multimodal interaction description languages. Four various roles including configuration, teaching, communication and modeling targeted by this language along with nine guidelines implied at multimodal interaction description are specified in the research [7].

Gesture might be different, clues might be used (such as the gaze direction) or a lower threshold can be used to activate commands when the user is increasingly nervous. In order to deal with these adaptations, fusion engines

require access to application-level information and need to react to inferred cues from other modality input.

3.4. Multimodal Interaction for Language Processing:

Error handling Multimodal frameworks aid developers to handle input containing noise and missing information. However, obtaining error-free results will be very hard. This is also the case for human-to-human interaction. Therefore multimodal solutions should provide mechanisms to correct mistakes and learn from them. The previously mentioned multimodal concerns span the entire fusion process. However, the transformation of low-level input data to high-level semantic information is complex and typically happens in multiple stages. Sharma et al. distinguish three levels of abstraction to characterize multimodal input data fusion: data-level fusion, feature-level fusion decision-level fusion. In this section, we present these different levels with some classic use cases.

3.5. Data-Level Fusion:

Data-level fusion focuses on the fusion of identical or tightly linked types of multimodal data. The goal is to (1) remove the excess of noise (2) provide feature candidates (3) provision in a real-time manner. This is challenging due to the fact that information continuously arrives at a high frequency. Data-level fusion requires to initiate raw data processing of noise leering and it rarely deals with the data semantics. Fusion of two streamed videos to extract the depth map of the scenes by limiting it from different angles. Some efforts on developing decision and feature-level and prognostic fusion methodologies, the development of “data-level” fusion models has a low scale contribution in the research and modelling [8].

3.6. Decision-Level Fusion:

Decision-level fusion focuses on deriving interpretations based on semantic information. It is the most versatile kind of multimodal fusion, as it can correlate information coming from loosely coupled modalities, such as speech and gestures. The well-known put that there example by Bolt fuses speech input such as \that" and \there" with pointing information to identify an object and a new location. Decision-level multimodal fusion

includes merging high-level information obtained by data- and feature-level fusion as well as modelling human-computer dialogues. Decision-level fusion is assumed to be highly resistant to noise and failures by relying on the quality of previous processing steps. Therefore, the information that is available for decision-level fusion algorithms may be incomplete or distorted.

3.7. Criteria for Expressing Multi-Level:

In this dissertation we focus on a fusion framework with both high-level programming language and architectural support. Multimodal programming languages are designed to support developers in specifying their multimodal gesture interaction requirements more easily than with general purpose programming languages. General purpose programming languages such as Java often require an excessive amount of code to express a developer's intention which makes them hard to read and maintain. A domain-specific language (DSL) [11] might help to reduce the repetitive boilerplate that needs to be written in existing languages as described by Van Custom. Van Custom argues that languages can shape the thought (earlier attributed to \The limits of my language means the limits of my world", Ludwig Wittgenstein). For instance, interaction patterns can be declaratively described by its requirements versus an imperative implementation with manual state management. This impacts the way of thinking during design and implementation.

3.8. Multimodal Interaction:

A number of criteria is used that characterize (1) the choice of a particular framework, (2) the implementation of the multimodal interaction and (3) the open issues in the multimodal engineering do-main. These criteria combine features proposed by different approaches, including domains such as machine learning, multimodal architectures, multimodal languages and template matching. They are compatible but on a more ne-grained level, than with previously established fusion criteria .Together with our experiments and the reuse of core criteria of existing work, these requirements view of multimodal fusion frameworks. We split the criteria up in four main categories: language features, multimodal processing,

multimodal and accessibility as well as tooling.

3.9. Criteria for Expressing Multi-Level Multimodal Fusion:

Customization is concerned with the effort a developer faces to modify a multimodal interaction gesture specification for use in a different context. How easy is it, for example, to adapt the existing definition of an interaction when an extra condition is required or the order of events should be changed? For graphical programming toolkits, the customization aspect is broadened to how easy it is to modify the automatically generated code (if possible at all). Note that in many machine learning approaches customization is limited due to the lack of a decent external representation [9].

The integration of application information in the fusion process allows developers to inspect the application state before executing certain functionality. For instance, a multi-touch scroll gesture can only happen when both are inside the GUI region that supports scrolling, the procedure is known as Application Symbiosis

3.10. Multimodal Interaction:

Whenever a multimodal interaction is recognized, an action is typically executed. In some cases the developer may want to provide a more detailed activation policy such as trigger only once or trigger when entering and leaving a particular state. Another example is the sticky bit option that activates the gesture for a particular GUI object. A shoot-and-continue policy denotes the execution of a complete gesture followed by online gesture activation. Multimodal processing concerns impact the design of a multimodal fusion engine. For example, does the engine need to be reactive such that it executes code for every input event or can it wait for certain characteristic events?

Criteria for Expressing Multi-Level Multimodal Fusion tractions can be supported in a framework by allowing concise definitions or by providing con-struts (i.e. high-level features) that advanced callback mechanisms to monitor the progress of a larger interaction. Multimodal Interaction matches is a complex mechanism that is supported by several approaches and intentionally blocked or ignored by others.

3.11. Segmentation:

Typically, streams of sensor input events do not contain explicit hints about the beginning and the end of a gesture. This is known as segment-action or gesture spotting. Segmentation gains importance given the trend towards the continuous capturing and free-air interaction in which a single event stream can contain many potential begin events. The difficulty of gesture segmentation manifests itself when one cannot know beforehand which potential begin" events should be used until a middle" or even an end" candidate event is found to form decisive gesture trajectory. It is possible that potential begin and end events can still be replaced by better future events. For instance, how does one decide when a free-air swipe right gesture begins or ends without reasoning about past (i.e. starting poses) or the future (i.e. trigger a narrow swipe right or wait for a wider swipe right gesture)? This lack of explicit begin- and end-points generates a lot of gesture candidates and increases the computational complexity.

Several approaches tackle this issue by means of a velocity heuristic with a slack variable (i.e. a global constant by the developer) or by programming incremental processing code. Many solutions make use of a garbage gesture model to increase the accuracy of the gesture segment-action process. Other segmentation problems include the distinction of background noise versus conversations to invoke the speech recognizer at the correct time[10].

4. CONCLUSION

In this paper we presented Midas, a novel declarative language to express multimodal interaction patterns. The core idea is that developers can focus on the essential complexity of describing their interaction patterns and have to deal with less accidental complexity such as handling irrelevant events, storing intermediate state and dealing with inversion of control. Primitive entities: templates (and facts), modules, rules, attempts and functions. An input event is translated into a fact and stored in the fact base. Rules can then try to a combination of facts that matches their conditions in a reactive manner.

Modules, attempts and functions modularize the multimodal description logic into small reusable parts such that

they can be composed to form more complex interaction descriptions, without requiring a developer to have a deep knowledge of all particular details. We have explored how complex gestural interaction can be described in 2D and 3D space using a technique called control points .This allows developers to automatically segment candidate gestures from a continuous stream of data while retaining full expressive control over the recognition process (which is normally lost in existing machine learning-based solutions).

Furthermore, we have shown how developers can seamlessly express cross-level fusion using Midas. There is no manual chaining of composition boxes required (such as required by data stream-oriented solutions), and there is no loss of expressiveness compared to semantic differencing solutions. Additionally, shadow facts, alternating between conditions and modifiers and the application symbiosis provide adequate language constructions to integrate declarative multimodal interaction patterns with application logic.

REFERENCES

[1] B Loureiro, R Rodrigues - Information systems and technologies (cisti), 6th iberian conference on, IEEE, pp. 1-6, 2011.

[2] DA Dahl - Journal on Multimodal User Interfaces, Springer Berlin Heidelberg, pp. 171-182, 2013.

[3] L Shackelford- a journal for the interdisciplinary study of literature, Volume 47 Number 4, December, - Mosaic, pp. 167-170, 2014.

[4] RB Suraparaju, A Jappani, K Bharat - US Patent 9, Google Patents, pp.189,736, 2015.

[5] M Haghighat,M Abdel-Mottaleb… - IEEE Transactions on Information Forensics and Security, IEEE, pp. 1984-1996, 2016.

[6] M Araki, K Tachibana - 06 Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue, Association for Computational Linguistics, pp. 88-95, 2009.

[7] B Dumas, D Lalanne, R Ingold - Journal on multimodal user interfaces, Springer Berlin/Heidelberg, pp. 237-247, 2010.

[8] K Liu, NZ Gebraeel, J Shi - IEEE Transactions on Automation Science and Engineering (T-ASE), IEEE, pp. 191-207, 2013.

[9] T Renaux, L Hoste, C Scholliers - Proceedings of the 2nd Workshop on Programming for Mobile & Touch, ACM, pp. 9-16, 2014.

[10] Xiong Juntao, Liu Zijian, Sun Baoxia, et al. The gesture tracking and action recognition algorithm based on visual technology [J], Computer and modernization, pp. 75-79, 2014.

[11] R Membarth, O Reiche, F Hannig, A Scheduler for Parallel Soft Real-Time Applications in Virtualized, IEEE, pp. 841 – 854, 2016.