Patterns Architecture for Fusion Engines

Ahmad Wehbi[1,2], Manolo Dulva Hina[1,2], Atef Zaguia[1,2], Amar Ramdane-Cherif[2], and Chakib Tadj[1]

[1] MMS Research Group, Université du Québec, École de technologie supérieure
1100, rue Notre-Dame Ouest, Montréal, Québec, H3C 1K3
[2] LISV Laboratory, Université de Versailles-Saint-Quentin-en-Yvelines
10-12, avenue de l'Europe, 78140 Vélizy-Villacoublay, France
{ahmad.wehbi,manolo-dulva.hina,atef.Zaguia}.1@etsmtl.ca,
ctadj@ele.etsmtl.ca, rca@prism.uvsq.fr

Abstract. This paper presents a design of an architecture that facilitates the work of a fusion engine. The logical combination or merging of input streams invoked by the fusion engine is based upon the definition of a set of patterns and its similarity with previously collected data from various modalities. Previous fusion engine designs had many weaknesses, among them their being specialized on a specific domain of application. The proposed architecture addresses such weakness and provides additional features, namely its ability to handle large number of modalities and due to its using knowledge base and standardization characteristics; it becomes suitable to various types of multimodal systems. The techniques used to achieve these features are discussed in this paper.

Keywords: modality, patterns, XML, data interpretation, knowledge base, normalization.

1 Introduction

As per W3C specification, a multimodal interaction activity seeks to allow users to dynamically select the most appropriate mode of interaction for their current needs, including consideration of any disability whilst enabling developers to provide an effective user interface for whichever modes the user selects. Various multimodal applications [1] are conceived and are effective solutions for users who have constraints, disabled or are on the go [2 - 5]. These applications are integrated with web services which are application functions or services [6]. Application service can be implemented as an autonomous application or as a set of applications. Engel et al [7] proposed an approach for processing modalities in a system called SmartKom [8]. Its idea is to generate all meaningful combinations after considering all hypotheses and afterwards selects the n best results which are passed to the intention analyzer. The disadvantage of this approach is that under adverse circumstances, the generation of all meaningful combinations takes too much time. Sonntag et al [9] proposed an ontological solution for a system called SmartWeb. It is based on question answering technology that combines different kinds of domain ontologies into an integrated and

B. Abdulrazak et al. (Eds.): ICOST 2011, LNCS 6719, pp. 261–265, 2011.

modular knowledge base. For this purpose, they defined an upper model ontology based on SUMO and DOLCE and integrated each domain ontology on it. The solution presented in this work is very limited. The architecture of HephaisTK system developed by Dumas et al [10] is based on software agents that are dispatched to manage individual modality recognizers, receive and encapsulate data from the recognizers, and send them to an individual central agent named the “postman”. This postman agent centralizes all data coming from the dispatched recognizers agents in the database, and distributes the data to other interested agents. However, this architecture needs a configuration file to be specified for describing the human-machine multimodal dialog desired for the client application, and for the specification to which recognizers need to be used. Having taken these weaknesses into account, we proposed an architecture that addresses these issues. This is done through modeling patterns that deal with different modalities, and by creating a knowledge base that contains these patterns. The adoption of this architecture will facilitate the work of a fusion engine by giving it the most meaningful combinations of data.

The paper is organized as follows. In section 2, we present the architecture itself and describe its components and finally we conclude this paper in section 3.

2 Pattern Architectural Design

In synopsis, the proposed architecture provides features that allow it to handle large number of modalities, support different types of multimodal systems and facilitate the work of a fusion engine by providing it the most probable matching of data. In this section, we describe our architectural design with focus on the use of patterns as a solution to the described problems of previous systems architectures.

2.1 General Schema and Approach

A general overview of the architectural framework of multimodal fusion is shown in Figure 1. It illustrates how a user supplies input streams and the interaction of various system components leading to the merging process by the fusion engine. These system components and/or processes are as follows:

- Modalities: These are modes of human-computer interaction. Media devices can be used to receive data input from the user (e.g. cellular phone, microphone, etc.)
- Recognition: The framework identifies and recognizes data provided by different modalities by transforming them to XML or EMMA.
- Patterns: These are predefined models that describe a modality; it is composed of data format, events and parameters. Patterns are used to check a match with data associated with modalities.
- Interpretation: The framework measures the match between a given data against a registered pattern and generates the most probable combinations of data.
- Models of rules patterns: These are predefined models that describe rules needed by the fusion engine for data merging.
- Fusion: Framework logically merges data to obtain an output.
- Output: Final meaningful result obtained after data streams have been merged.

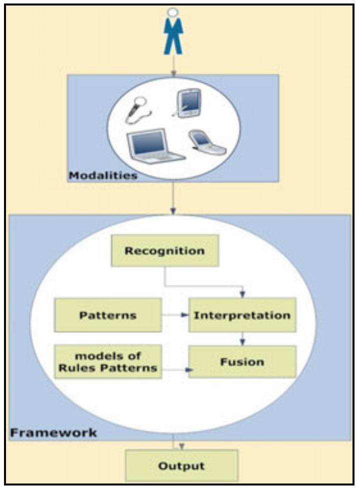

Fig. 1. Architectural framework of multimodal fusion

2.2 Overview of the Pattern's Architectural Design

The previous diagram in Figure 1 is further refined in Figure 2 showing the whole concept of the architecture. Aside from the recognition, interpretation, and fusion processes that were described earlier, the refined architectural design shows new components, namely the knowledge base and the database. The remaining system components and the description of various steps involved in the fusion process are described below:

- **Knowledge Base:** It is a container that stores the patterns as ontology concepts.
- **Database**: It is a container that stores all probable meaningful combinations of interpreted data. These combinations are used by the fusion engines.
- **Step 1 – User - Modality:** The user begins his interaction with the system, using various modalities such as voice, facial expression, etc. By performing an action, the user automatically launches/activates modalities.
- **Step 2 – Modality – Recognition of Modalities:** This module receives as input XML files containing data concerning modalities, usually captured using sensors. From each XML file, this module extracts some tag data needed for fusion. Afterwards, it creates a resulting XML containing the selected modalities and each modality's corresponding parameters. In conformity with W3C standard on XML tags for multimodal applications, we use EMMA. EMMA [11] is a generic tagging language for multimodal annotation.
- **Step 3 – Patterns – Knowledge base:** Patterns are stored as concepts in the knowledge base which contains data format, parameters and modality events. These patterns are used by the interpretation module to check a similarity match between the data provided by modalities against the patterns themselves. These concepts are semantically represented inside the knowledge base using the Web Ontology Language (OWL).

- **Step 4 – Interpretation of Modalities:** This module is responsible for checking if there is a match between the data associated with modalities and that of the patterns. Match checking is done by determining the following conditions: (1) the data format - the data format from a special modality is checked with the data format of the pattern that concerns that modality, (2) parameters – these are also checked to verify the most suitable selection of modalities, and (3) the events – this is done by comparing the present event. If the three conditions are true, two types of result are produced, namely (1) the generation of probable meaningful combinations of data, and (2) the detected modality used by the user is correct. The generated meaningful data are stored in a database which will be used later by the fusion engine.
- **Step 5 – Fusion Engine:** This module is responsible for the merging of data by gathering different input streams coming from various modalities and obtaining a meaningful combinational result. Input signals are intercepted by the fusion agent and then combined taking into account some given semantic rules.

2.3 Context Pattern

A pattern is defined as an idea that has been proven to be useful in one practical context and therefore will probably be useful in others [12]. Patterns are often defined as something strictly described and commonly available. In our work, a pattern is composed of data format, events and parameters. Patterns are used to check a match of a user action involving a modality against the pre-defined data associated with modalities. Patterns are modeled, taking into account important characteristics, namely, the format of the pattern, its parameters and the modality events. Patterns are stored in a knowledge base.

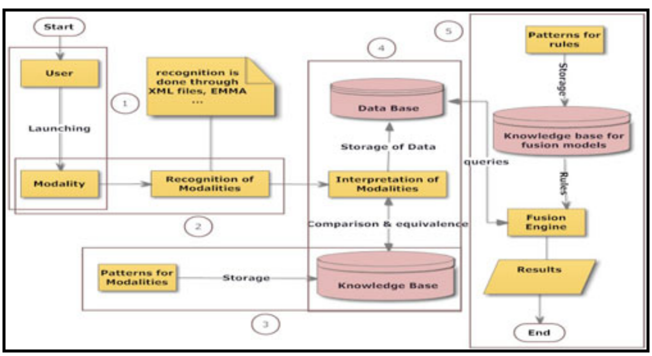

Fig. 2. Architecture overview

2.4 Models of Rules

Another type of pattern is needed in our architecture. This pattern contains the *Models of rules* needed for data merging and it is stored in a knowledge base. A multimodal fusion rule consists of constraints and result construction rules. Patterns are modeled, taking into account important characteristics, namely, Content, Category, Constraints, Time, Probability and Modality.

3 Conclusion

In this paper, we presented an architecture that is very useful in a multimodal system. The architecture proposes a solution that standardizes the interpretation of modalities. Such standardization is implemented by defining a set of patterns that are stored in a knowledge base; these patterns, implemented using OWL semantic web language, take into account special data formats, parameters, events and models of rules that are associated to modalities. This paper proposes a new solution by modeling an architecture that facilitates the work of a fusion engine, using predefined patterns stored in a knowledge base or ontology. These patterns will be a reference that measures the similarity between data that was previously taken from different modalities and the pattern itself. This technique will offer the fusion engine the most meaningful combination of data that can be used in the fusion process.

References

1. Yuen, P.C., et al.: Multimodal Interface for Human-Machine Communication, vol. 48. World Scientific Publishing Co., Pte. Ltd., Singapore (2002)
2. Shin, B.-S., et al.: Wearable Multimodal Interface for Helping Visually Handicapped Persons. In: 16th Int. Conf. on Artificial Reality and Telexistence, Hangzhou, China (2006)
3. Raisamo, R., et al.: Testing usability of Multimodal Applications with Visually Impaired Children, vol. 13, pp. 70–76. IEEE Computer Society, Los Alamitos (2006)
4. Lai, J., et al.: Examining Modality Usage in a Conversational Multimodal Application for Mobile E-mail Access. Int. Journal of Speech Technology 10, 17–30 (2007)
5. Debevc, M., et al.: Accessible Multimodal Web Pages With Sign Language Translations for Deaf and Hard of Hearing Users. In: 20th Int. Workshop on Atabase and Expert Systems Application, Linz, Austria (2009)
6. Li, Y., et al.: An Exploratory Study of Web Services on the Internet. In: IEEE Int. Conf. on Web Services, Salt Lake City, USA (2007)
7. Engel, R.P., Norbert, J.: Modality Fusion.SmartKom: Foundations of Multimodal Dialogue Systems. Springer, Berlin (2006)
8. Norbert Reithinger, J.A., et al.: SmartKom - Adaptive and Flexible Multimodal Access to Multiple Applications. DFKI GmbH – German Research Center for Artificial Intel. (2003)
9. Daniel Sonntag, M.R.: A Multimodal Result Ontology for Integrated Semantic Web Dialogue Applications. DFKI GmbH German Research Center for Artificial Intel. (2006)
10. Dumas, B., et al.: Description Languages for Multimodal Interaction: a Set of Guidelines and its Illustration with SMUIML. Journal on Multimodal User Interfaces 3, 237–247 (2010)
11. Desmet, C., et al.: <emma>: re-forming composition with XML. Literary & Linguistic Computing 20, 25–46 (2005)
12. Fowler, M.: Analysis Patterns – Reusable Object Models. Addison-Wesley, Reading (1997)