EAI Endorsed Transactions
on Creative Technologies
Research Article EAI.EU

A Multimodal Interaction Framework for Blended Learning

N. Vidakis[1]

[1] Department of Informatics Engineering, Technological Educational Institute of Crete, Heralkion 71500, Greece

Abstract

Humans interact with each other by utilizing the five basic senses as input modalities, whereas sounds, gestures, facial expressions etc. are utilized as output modalities. Multimodal interaction is also used between humans and their surrounding environment, although enhanced with further senses such as equilibrioception and the sense of balance. Computer interfaces that are considered as a different environment that human can interact with, lack of input and output amalgamation in order to provide a close to natural interaction. Multimodal human-computer interaction has sought to provide alternative means of communication with an application, which will be more natural than the traditional "windows, icons, menus, pointer" (WIMP) style. Despite the great amount of devices in existence, most applications make use of a very limited set of modalities, most notably speech and touch. This paper describes a multimodal framework enabling deployment of a vast variety of modalities, tailored appropriately for use in blended learning environment and introduces a unified and effective framework for multimodal interaction called COALS.

Keywords: Multimodal Human-Computer Interaction, Blended Learning.

Received on 18 November 2016, accepted on 28 December 2016, published on 04 January 2017

doi: 10.4108/eai.4-9-2017.153057

1. Introduction

In everyday life, people constantly communicate with each other, through the use of diverse modalities such as speech, gestures, or vision. Thus, almost every natural communication amongst humans involve multiple, concurrent modes of communication **Error! Reference source not found.**. Considering this, it can be safely stated that multimodal interaction is present in ordinary human-to-human communication. This communication model is often desirable in human-computer interaction, since it provides a more natural way of communication, by means of gesture, speech or other modalities, as well as being a primary choice over unimodal interaction models by the users [3]. Moreover, this type of interaction has demonstrated better flexibility and reliability than any other human computer interaction model **Error! Reference source not found.**.

Multimodal interaction seeks to promote a more natural way of human-computer interaction. Despite studies proving that multimodal interfaces are not more efficient or quicker than standard WIMP interfaces [1], these interfaces were also proven to be more robust and stable [2]. Moreover, multimodal interfaces enjoyed greater acceptance from the vast majority of users. Multimodal interaction displays full support of naïve physics (multi-touch interaction), body awareness and skill (gesture and speech interaction), environment awareness and skills (plasticity), as well as social awareness and skills (collaboration and emotion based interaction).

Despite these benefits, multimodal interaction is quite demanding in terms of design and implementation. Multimodal interaction can make use of any type or number of modalities, requiring a wide range of skills in unrelated domains such as software engineering, human-computer interaction, artificial intelligence and machine learning. Moreover, a review of the literature reveals that few multimodal corpora currently exist, and are targeted in specific modules and components of a multimodal

interaction system. Moreover, most multimodal interaction systems, make use of a limited set of modalities, typically speech and gesture recognition.

Oviatt et al. [3] consider the following research directions: *new multimodal interface concepts*, such as blended interfaces that use both passive and active modes, *error handling techniques*, such as mutual disambiguation techniques, and dialogue processing techniques, *adaptive multimodal architectures*, that involve systems that automatically adjust to users and surroundings, and finally, *multimodal research infrastructures*, such as software tools that support the rapid creation of multimodal interfaces.

Garofolo [4] identifies another set of technological challenges, that is: *data resources and evaluation*, as the limited number of multimodal corpora in existence makes thorough evaluations unachievable, *core fusion research*, such as novel statistical methods and data representation heuristics and algorithms and, ultimately, *driver applications*, needed to guide research directions. Summarizing these findings, the following subset is derived, which is believed to be a representation of the most important issues in the field.

- Architectures for multimodal interaction: Because of the concurrent nature of these interaction types, tools that help in the rapid design and prototyping of multimodal interfaces are required, in order for multimodal interaction to become more mainstream.
- Modelling of the human-machine dialog: Because of the complex nature introduced by the large number of input and output modalities.
- Fusion of input modalities: A research domain, tightly coupled to human-computer dialog modeling, concerning effective fusion algorithms able to take into account multiple aspects of human-machine dialog.
- Time synchronicity: The ability to take into account, and adapt to multiple modal commands which can trigger different meanings, following their order, and delay between them.
- Plasticity/adaptivity to user & context: The capability of a human-machine interface to adapt both to the system's physical characteristics and environmental variables while preserving usability [5].
- Error Management: Being the weak link of multimodal interaction, since it is always assumed that users will behave in perfect accordance with the system's expectations of behavior, and that no unwanted circumstance will appear. Apparently, this is not the real life case, and error management will have to be carefully handled if multimodal interfaces are to be used broadly.
- User feedback is somewhat related to error management, in that the user is allowed to correct or adjust the behavior of the multimodal system in real time.

This paper presents a specific and efficient architecture and framework of multimodal interaction to be used in a blended learning environment. In the following Section, different interaction modalities will be investigated, while in Section 3 the main goals and principles of the blended learning approach will be presented. Section 4 presents the proposed multimodal interaction framework. Finally, section 5 draws the conclusions.

2. Interaction Modalities

According to Bellik and Teil [6] modality is "a concrete form of a particular communication mode" where mode is defined as the five human senses (sight, touch, hearing, smell and taste) which constitute the receiving information, and the multifarious ways of human expression (gesture, speech, etc.) which constitute the product information. Furthermore, Bellik and Teil's definition characterizes the nature of information of human communication as visual mode, sound mode, gestural mode, etc. For example, noise, music and speech are modalities of the sound mode.

Nigay and Coutaz [8] formally present the modality as: m=(d,r)|(m,r), where "d" denotes the physical I/O device, "r" an interaction language (representational system) and "m" an interaction modality. For example, the speech modality can be defined using the "Microphone" as a physical device and "Pseudo-natural language" as an interaction language.

A physical device is an artifact or an organ needed by the system or the user in order to acquire (input device) or deliver (output device) information. Examples include keyboard, microphone, ears and mouth.

An interaction language is a language used by either the user or the system in association with a physical device in order to commune information. The interaction language defines a set of all possible well-formed expressions, i.e., the conventional assembly of symbols that convey meaning for both parties. Examples include pseudo-natural language and direct manipulation language. Table 1 introduces some examples of Interaction Modalities where apart from the modality, the mode, the interaction language and the device are presented.

Table 1: Examples of Interaction Modalities

Modality	Mode	Interaction Language	Device
Acceleration	Gesture	Direct Manipulation	Accelerometer
Speech	Voice	Natural Language	Microphone
Hand Motion	Gesture	Specialized Sign Language	RGB Camera

Facial Expression	Gesture	Specialized Sign Language	RGB Camera
Pointing Gesture	Tactile	Direct Manipulation	Touch Screen
Orientation	Gesture	Direct Manipulation	Gyroscope
Speech Synthesis	Audio	Natural Language	Speaker

Modalities can also be classified according to the required user attention. A modality may be considered active if used consciously by the user, while it can be considered passive if used unconsciously. For example, using hand motions to control a specific element of the user interface is considered an active modality, while capturing the user location with a GPS is considered passive, since it does not need user attention. However, if the user moves on her own to go to a particular location by using the GPS location to create a path, the position using GPS modality may be then considered active.

2.1 Multimodal systems

A system is considered multimodal when it processes "two or more combined user input modes (such as speech, pen, touch, manual gesture, gaze and head and body movements) in a coordinated manner with multimedia systems output." [9]. Multimodal systems are endowed with multimodal capabilities for human/machine interaction and able to interpret data from various sensory and communication channels.

Multimodal interaction systems, provide a set of "modalities" to the user in order to allow them to interact with the machine. According to Oviatt Error! Reference source not found. : «Multimodal interfaces process two or more combined user input modes (such as speech, pen, touch, manual gesture, gaze, and head and body movements) in a coordinated manner with multimedia system output. They are a new class of interfaces that aim to recognize naturally occurring forms of human language and behavior, and which incorporate one or more recognition-based technologies (e.g. speech, pen, vision)». According to this definition, multimodal architectures display unique features. These features are the fusion of different data types, and real-time processing and temporal constraints imposed on information processing [10] [11].

Thus, multimodal interaction systems, represent a new type of human-computer interfaces that is not based on the WIMP (Window, Icon, Menu, and Pointer) paradigm. Multimodal interaction systems tend to emphasize on more natural ways of communication, usually speech and gestures, and the utilization of all the five senses. Therefore, the objective of multimodal interfaces is twofold: firstly to support and accommodate user's perceptual and communicative capabilities; and secondly to provide a wider range of communication means to humans.

3 Blended Learning

Over the years, pedagogical methods evolved and consistently improved compared to the educational systems of the past. A significant factor of this progress is unquestionably the increasing use of technology in teaching. Nowadays, most educators prefer to blend traditional teaching with interactive software, in order to achieve the maximum involvement of their students and to consolidate their learning.

A sufficient description of blended learning could be the following: "Blended learning is the process of using established teaching methods merged with Internet and multimedia material, with the participation of both the teacher and the students" [10]. Still, there are a few matters in question regarding the process of creating blended learning environments [11]:

- Firstly, there is the issue of the importance of face to face interaction. Several learners have stated that they are more comfortable with the part of live communication in merged teaching methods, considering it more effective than the multimedia based part. Others are of the exact opposite opinion, which is that face to face instruction is actually not as required for learning, as it is for socialization. There are also those who believe that both live interaction and online or software material are of the same significance and it is the learner's decision which of the two is more suitable for their educational needs.
- Secondly, there is the question of whether the learners opt for combined learning exclusively because of the flexibility and accessibility of the method, without taking into account if they are choosing the appropriate type of blended teaching. Also, there are doubts regarding the ability of the learners to organize their own learning without the support of an educator.
- Another matter, is the need for the instructors to dedicate a large amount of time to guide the learners and to equip them with the necessary skills in order to achieve their goals. In addition, the instructors need to continue being educated themselves to be able to meet the requirements of blended teaching.
- There is also the argument that schools which have integrated technology into the curriculum are mostly addressed and beneficial to people in a comparatively favorable position in terms of economic or social circumstances. This, however, is refuted by the fact that blended teaching methods are quite affordable and easily accessible. The quick distribution of the multimedia material used in this type of instructing, is considered an advantage, but the universality of the system raises the need to modify the provided

material, in order to make it more culturally appropriate for each audience.

- Lastly, a continuous effort is being made to proceed to the new directions given by the novelties of technology while maintaining a low-priced production of educational material. The uninterrupted evolution of communication and information technology is making this effort rather strenuous for the developers of such teaching models.

Despite the difficulties faced when designing blended learning techniques, the advantages of combining face to face instructing with technology are many. Some of enormous importance are [11]:

(i) The learner is intrigued by the procedure itself and as a result, the material being taught seems more interesting to them. This leads to the easier accomplishment of the educational goals that the teacher has set.
(ii) There is no limitation regarding the time or the place of the lesson. A student can attend remote classes being offered online, frequently having the opportunity to record and watch again the lesson.
(iii) The cost of the teaching process is greatly reduced. Every school unit that uses blended learning, has access to a variety of Internet and software material which would require a lot of time and effort to be independently produced, resulting in additional expenses.
(iv) In industrial applications of blended learning, it has been observed that the desired results have been achieved twice as fast as with the established instruction methods.

Blended learning should not be conceived solely as a method of enriching the class with technology or making the learning process more accessible and engaging. Its main purpose is to modify and adjust the teaching and learning interaction in order to upgrade it. It assists and enables the growth of critical thinking, creativity and cognitive flexibility [12]. In this context, the following section presents a framework of multimodal interaction to be used in a blended learning environment.

4 Multimodal Interaction Framework for Blended Learning

The Multimodal Interaction Framework for Blended Learning has to follow a number of requirements. According to this initial defined requirements, presented in Table 2, it was decided to implement a finite state machine-based architecture as a foundation for this framework based on the initial requirements proposed in [14]

Table 2: Initial Requirements [14]

Requirements	Description
Allow rapid creation of multimodal interfaces	The tool has to be expressive enough to describe a wide variety of potential multimodal applications, yet usable enough not to require tremendous knowledge in multimodal interaction.
Fusion of different input sources	One of the goals of the framework is to be able to manage and fuse different input sources, created by a multitude of devices. Thus, it has to provide facilities to integrate and manage different data representations, and different data semantics.
Fission of output based on context, user profiles and environmental variables	One of the main issues addressed by the framework is the ability to adapt to any given environmental and user context.
Has to have a small computational footprint	The framework must operate in a variety of devices, from computationally weak handheld devices to more advanced computer systems, thus it has to have a small computational footprint in order to be able to operate in handheld devices.
Be extensible	The Framework has to be extensible in a number of specific areas. First of all, as the framework has to be able to operate using different data representation. Second, as a framework allowing the creation of multimodal interfaces, it has to be able to accept input from a number of different input modality recognizers – and even offer the possibility to accept data from completely new types of input modalities. Third, fusion and fission results have to be shared with client applications

	in a way that will not constrict developers in the creation of their multimodal applications.

Based on the initial requirements expressed above and the goal for an efficient "educational platform for blended learning that enables multimodal player interaction" a framework has been developed that lays between the interaction devices deployed by the users and the engine used by the system platform, endowing the application with multimodal interaction capabilities. Figure 1 demonstrates the framework's architectural overview

.

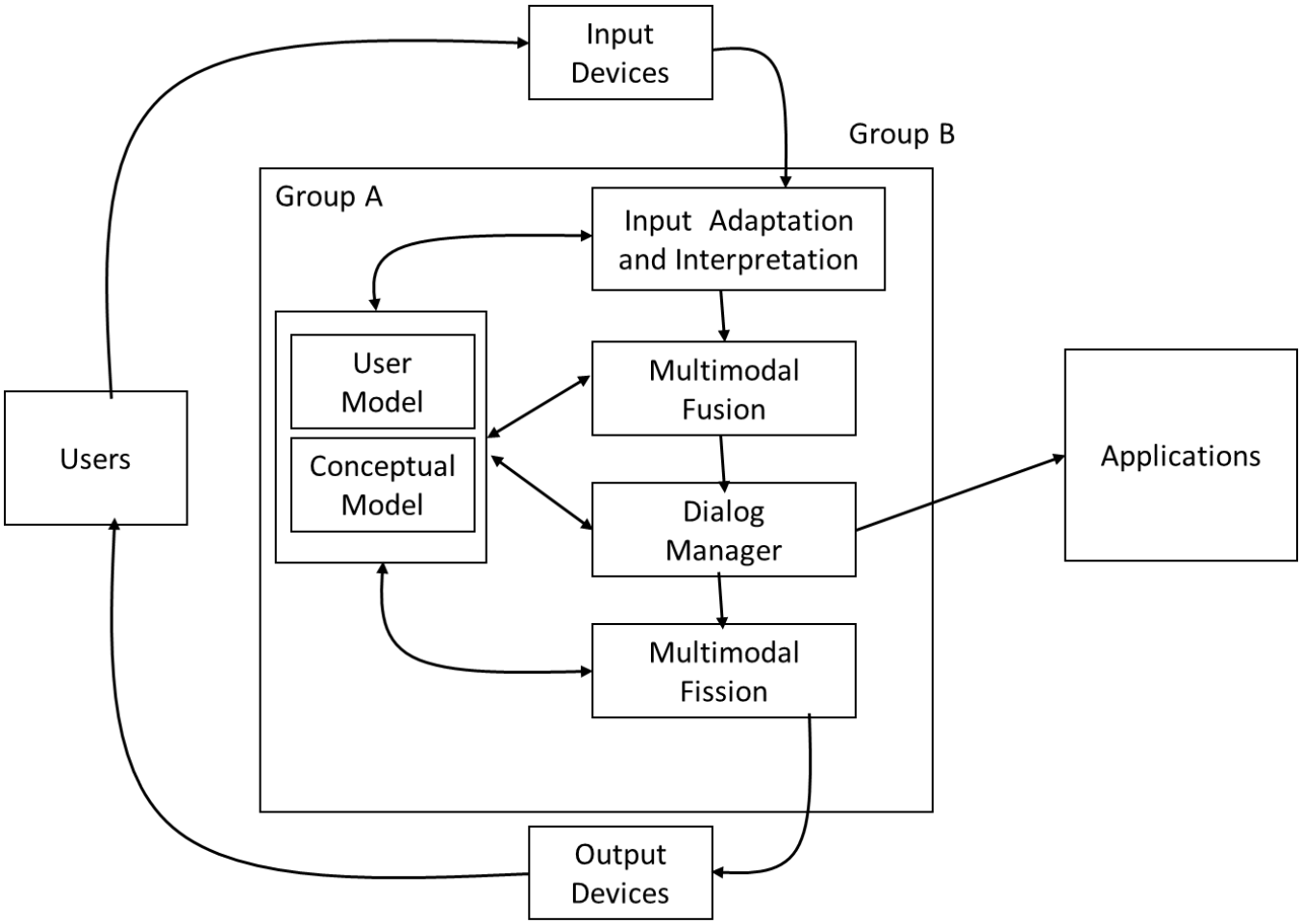

Figure 1: Framework's Architectural Overview.

The framework consists of two main modules namely the Context Management module (group A) and the I/O Management module (group B). In more detail the Context management module includes the "User Model" and the "Contextual Model" components and the I/O Management module includes "Input Adaptation and Interpretation", "Multimodal Fusion", "Dialog Management" and "Multimodal Fission" (see Fig. 1). In the next few paragraphs the framework modules and components are described in detail.

The Input Adaptation and Interpretation component, is responsible for the recognition of data provided by the user. This recognition can be both direct, e.g. speech recognition directly from the microphone, and indirect, e.g. gesture recognition from the Microsoft Kinect sensor.

The Multimodal Fusion component is responsible for interpreting data from the recognizers into meaningful commands. In a blended learning platform, which is deployed in a set-box environment, with often limited processing power, it is crucial to minimize the workload of each component framework. For that reason, the fusion component is centered towards decision level fusion, which assigns a major amount of responsibility to the various recognizers, which provide data that must be interpreted and merged to achieve a final interpretation. A frame-based strategy was chosen due to its simplicity and

the ease of augmentation. These augmentations, among others, include attribute constraints and modality prioritization.

The Dialog Manager component, manages changes in the application state, and is responsible for the communication of the framework with the application. It is also responsible for providing output information to the Fission Engine for communication between the framework and the user.

In our approach the dialog manager module uses a finite state machine to identify the command created by the user, and maps the results in a way that is understandable by a system engine. The data that trigger transitions in the finite state machine are generated by the Fusion Engine as described above. This approach is chosen due to the relatively small computational footprint.

The Dialog Manager component, uses a mapping between commands generated by the user (e.g. Command, Location, and Selection) and commands that are understandable by the system engine. This way the Dialog Manager can manipulate the commands communicated to the system engine in order to meet the commands issued by the user. Due to this architectural choice the blended learning platform can augment the functionality of the chosen system engine in order to support multimodal interaction and/or expand the device repertoire of the engine. Also this architecture allows the platform to function with any given system engine as long as a proper mapping of commands is created.

In addition, the dialog manager is responsible for generating data that are passed to the Fission Engine for the communication of messages generated either by the framework itself (e.g. an incomplete command) or the system engine (e.g. auditory notifications, for visually impaired players).

The Fission Engine component. The Fission Engine component is responsible for generating appropriate output messages directly to the user, in a format that is compliant with the user and application context, as well as the environmental variables.

4.1 Framework's Description Language

COALS stands for Command Object Attribute Location Selection. While not implied by its name, the language seeks to offer developers a language for describing multimodal interaction, expressing in an easy-to-read and expressive way the modalities used, the recognizers attached to a given modality, the human-machine dialog modelling, and the various events associated to this dialog. This section will give a broad view of the COALS language, beginning with the language structure, followed by a detailed view of the different levels described by the language.

Structure of the Language

The way an instance is split allows a clear separation between three levels necessary to describe multimodal human machine dialog. The recognizer is the lower, input level, and provides a mapping between recognizer data to COALS tokens. The actions level responsible for dialog transition forms the middle level, devoted to events management, and the upper level contains the dialog description, used for the mapping between dialog transitions with a specific application action. This separation in different abstraction layers can be observed in state of the art languages such as UsiXML [15] or MIML [16].

For a given client application three main sections form the description of the multimodal interaction scenario. The first part, dictionary, indicates how particular data will be tied to which COALS token. The second section transitions, lists the different events that will be of interest for the client application. The focus of this section is to model all events coming from the different recognizers. These events are described by means of a finite state machine. The final section, dialog, contains the mapping between a specific event, to a specific action or function of the client application.

```
{
    "COALS": {
        "dictionary": {
            "commands": [],
            "objects": [],
            "attributes": [],
            "locations": [],
            "selections": []
        },
        "transitions": {
            "states": [],
            "transitions": []
        },
        "dialog": {}
    }
}
```

Figure 2: General Framework's Layout

A complete COALS example is given in **Error! Reference source not found.**Figure 3. In this example the user can perform the classic "Put this there" sequence by using speech and gestures on a tactile surface.

Dictionary

At the dictionary level, the goal is to tie the COALS tokens with the actual recognizer data which is used by the developer for his application. In the context of COALS framework, all recognizers are treated as a continuous stream of data. For example, if a number of speech recognizers are available, each one of them will provide its respective stream of data, which then will be processed by the framework.

Commands represent any command that can be issued to the system in order to perform actions. These commands are application specific, in a game for example, appropriate commands would be the movement of the player's character in the game, the interaction of the player with the environment, etc.

Objects represent application entities that are either distinct, thus recognizable by an identifier, or entities selected by the user through a modality. For example a cube and an object located in specific coordinates of the screen, selected by the user are both treated as objects from the framework.

Attributes represent characteristics of entities of the application, which distinguish them from other entities. Attributes can be characteristics such as color, size or shape.

Locations are special tokens that allow the system to distinguish the target location of events. For example, when the user moves an object from a certain position to another, the endpoint of the trajectory is flagged by a location identifier.

Selections are another family of special tokens that allow the system to identify selections of entities in the application.

Transitions

Transitions are at the core of the transition mechanism of COALS. They describe a sub-set of interest from all the possible events coming from different recognizers.

A standard transition declaration is shown in Figure 3. In this example, four states and ten transitions are defined. This grants robustness to the system, since the user can usher the commands with any possible order, without the need of a specific syntax.

Dialog

The dialog element describes the communication between the framework and the application. In essence, the dialog is a finite state machine, with transitions and states represented in the previous section of the document. Callbacks define specific functions that have to be called once a specific transition occurs. Each callback has a unique name to identify it. The from and to elements define the transitions, upon which the action must be called. Finally the action element defines the application function to be called by the script.

4.2 COALS Language Interpretation

COALS language is mainly used as the way to script the COALS framework. In consequence, both the COALS framework and the COALS language share a strong bond. In fact, a good part of the way the COALS dialog manager represents information is based on the structure of the COALS language.

The COALS language consists of 3 main components, the language vocabulary of tokens that is described in previous sections, a grammar consisting of rules of how these tokens may be used, and a propositional structure, which places the tokens in linear structures.

```
{
      "COALS": {
            "dictionary": {
                  "commands": ["put"],
                  "objects": [],
                  "attributes": [],
                  "locations": ["there"],
                  "selections": ["this"]
            },
            "transitions": {
                  "states": ["put", "this",
"there", "point"],
                  "transitions": [{
                        "from": "put",
            "to": "there"
                  }, {
                        "from": "this",
                        "to": "put"
                  }, {
                        "from": "this",
                        "to": "there"
                  }, {
                        "from": "this",
                        "to": "point"
                  }, {
                        "from": "this",
                        "to": "there"
                  }, {
                        "from": "there",
                        "to": "put"
                  }, {
                        "from": "there",
                        "to": "this"
                  }, {
                        "from": "there",
                        "to": "point"
                  }, {
                        "from": "point",
                        "to": "this"
                  }, {
                        "from": "point",
                        "to": "there"
                  }]
            },
            "dialog": {
                  "callbacks": [{
                        "name": "on_put",
                        "from": "put",
                        "to": ["this", "there"],
                        "action": "move"
                  }]
            }
      }
}
```

Figure 3: An example of a COALS script

The framework's grammar can be represented with a finite state machine. The states of the machine consist of the tokens defined by the language's dictionary, and the transitions represent the way these tokens interact in order to generate valid sentences. The state machine diagram is presented in Figure 4 below.

The Start and End states represent the beginning and the end of the sentence respectively. As shown above, a valid COALS sentence can only be initialized as soon as there is a recognized Command. Also a valid sentence can contain one and only one Command. Each Location must be mapped with a point in the application plain of reference. Note that a Location can be preceding a Point and vice-versa. In the same manner, a Selection token must be mapped with a point. An interesting feature of the COALS grammar is that an Object is either generated through a Point (i.e. selected by the user via a selection token, or location token), but also be pre-processed if it is a point of interest for the application and its coordinates are predefined (e.g. certain interactive shapes in a game).

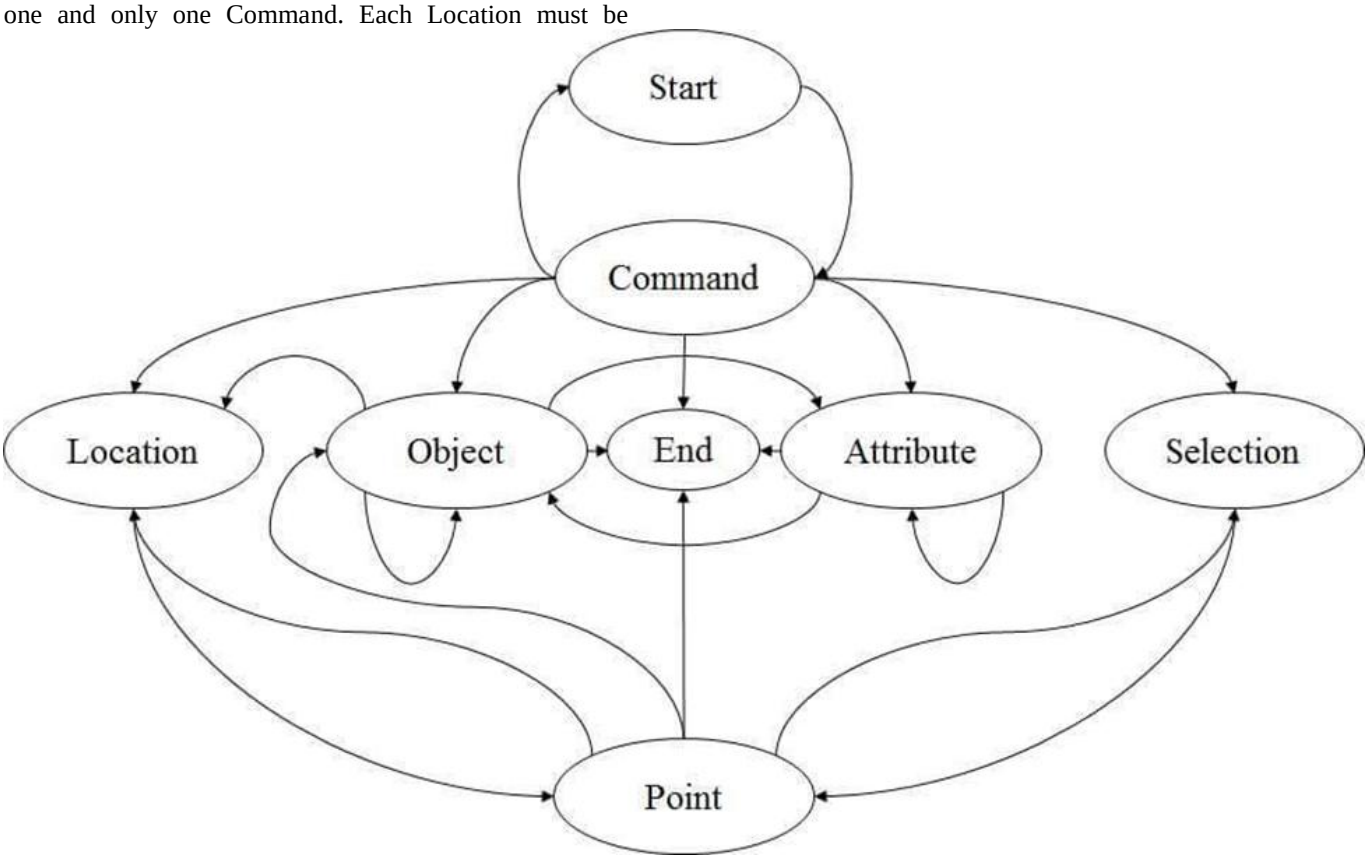

Figure 4: Framework's Description Language

5 Conclusion

In this paper we have presented a framework that supports multimodal interaction to assist blended learning. The beneficial use of an interactive multimodal framework is presented, in order to successfully handle a complicated system with various multimodal inputs, such as voice commands, hand gestures and touch gestures. Our primary design target is to set up a framework that supports multimodal interaction on educational games according to available I/O modalities, user needs, abilities and educational goals.

More precisely COALS framework is introduced to achieve the aforementioned stated goal, firstly by fusing – combining different modalities, secondly by interpreting the previous fusions while extracting semantic meaning from them and thirdly by providing the user the right feedback, according to the preceding procedures.

Ongoing work covers a variety of issues of both technological and educational engineering character. Some of the issues to be addressed in the future include: (a) Run various use cases in vivo with the guidance and involvement of users and (b) Elaborate further on the Multimodality Amalgamator module to involve more input and output modalities so that the roles between game player and machine are reversed and the player performs gestures, sounds, expressions etc. and the machine responds.

As any multimodal interface tool, a way to describe the dialog between the human and the machine has to be found. In the case of COALS framework, as the tool has to be able to describe at a high level the human-machine dialog, the choice is made to enable multimodal human-machine dialog description using a token based description

language, specifically created for use with the framework. This language should be defined and tested.

Research focus can be also given to the fusion engine in the COALS framework which should offer a unified way to handle input data. This method can be a combination of two different fusion techniques: Meaning Frame-based techniques and Finite State Machine-based techniques.

Another future work should be the detailed evaluation of the framework, employing a number of use cases in order to prove the capabilities of the tool and pinpoint its shortcomings.

References

[1] S. Oviatt, "Ten Myths of Multimodal Interaction," Commun. ACM, vol. 42, no. 11, pp. 74–81, Nov. 1999.

[2] S. Oviatt, "Advances in robust multimodal interface design," IEEE Computer Graphics and Applications, vol. 23, no. 5, pp. 62–68, Sep. 2003.

[3] S. Oviatt, P. Cohen, L. Wu, J. Vergo, L. Duncan, B. Suhm, J. Bers, T. Holzman, T. Winograd, J. Landay, J. Larson, and D. Ferro, "Designing the User Interface for Multimodal Speech and Pen-based Gesture Applications: State-of-the-art Systems and Future Research Directions," Hum.-Comput. Interact., vol. 15, no. 4, pp. 263–322, Dec. 2000.

[4] J. Garofolo, "Overcoming Barriers to Progress in Multimodal Fusion Research". In Multimedia Information Extraction: Papers from the 2008 AAAI Fall Symposium Arlington, Virginia, 2008. The AAAI Press, pp. 3-4.

[5] Thevenin, David, and Joëlle Coutaz. "Plasticity of user interfaces: Framework and research agenda." Proceedings of INTERACT. Vol. 99. 1999.

[6] N. Elouali, J. Rouillard, X. L. Pallec, and J.-C. Tarby, "Multimodal interaction: a survey from model driven engineering and mobile perspectives," J Multimodal User Interfaces, vol. 7, no. 4, pp. 351–370, Jun. 2013.

[7] Y. Bellik and D. Teil, "Définitions Terminologiques pour la Communication Multimodale," presented at the Proceedings of interface Hommemachine (IHM), 1992.

[8] L. Nigay and J. Coutaz, "Multifeature Systems: The CARE Properties and Their Impact on Software Design," in Multimedia Interfaces: Research and Applications, chapter 9, 1997.

[9] S. Oviatt, "Advances in robust multimodal interface design," IEEE Computer Graphics and Applications, vol. 23, no. 5, pp. 62–68, Sep. 2003

[10] Norm Friesen, "Report: Blended Learning.", 2012

[11] Bonk, C. J. & Graham, C. R. (Eds.), "Handbook of blended learning: Global Perspectives, local designs", San Francisco, CA: Pfeiffer Publishing, Chapter 1.1, Blended learning systems: Definition, current trends, and future directions

[12] Singh, Harvi, and Chris Reed. "A white paper: Achieving success with blended learning." Centra software 1, 2001.

[13] Garrison, D. Randy, and Heather Kanuka, "Blended learning: Uncovering its transformative potential in higher education.", The internet and higher education 7, no. 2, 2004, pp 95-105.

[14] B. Dumas, D. Lalanne, and S. Oviatt, "Multimodal Interfaces: A Survey of Principles, Models and Frameworks," in *Human Machine Interaction*, D. Lalanne and J. Kohlas, Eds. Springer Berlin Heidelberg, 2009, pp. 3–26.

[15] K. Katsurada, Y. Nakamura, H. Yamada, and T. Nitta, "XISL: A Language for Describing Multimodal Interaction Scenarios," in *Proceedings of the 5th International Conference on Multimodal Interfaces*, New York, NY, USA, 2003, pp. 281–284.

[16] M. Araki and K. Tachibana, "Multimodal Dialog Description Language for Rapid System Development," in *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*, Stroudsburg, PA, USA, 2006, pp. 109–116.