Année académique 2008-2009

Working paper 09/15

Fusion engines for multimodal input : A survey

Denis Lalanne, Laurence Nigay, Philippe Palanque,
Peter Robinson, Jean Vanderdonckt,
Jean-François Ladry

LOUVAIN
School of Management

Site de Louvain-la-Neuve - Place des Doyens, 1 - 1348 – Louvain-la-Neuve

Fusion Engines for Multimodal Input: A Survey

Denis Lalanne
University of Fribourg, Switzerland
denis.lalanne@unifr.ch

Laurence Nigay
LIG, University of Grenoble, France
Laurence.Nigay@imag.fr

Philippe Palanque
IHCS, University of Toulouse, France
palanque@irit.fr

Peter Robinson
Computer Laboratory,
University of Cambridge, UK
peter.robinson@cl.cam.ac.uk

Jean Vanderdonckt
Université catholique de Louvain
(UCL), Belgium
vanderdonckt@isys.ucl.ac.be

Jean-François Ladry
IHCS-IRIT, University of Toulouse,
France
ladry@irit.fr

ABSTRACT
Fusion engines are fundamental components of multimodal interactive systems, to interpret input streams whose meaning can vary according to the context, task, user and time. Other surveys have considered multimodal interactive systems; we focus more closely on the design, specification, construction and evaluation of fusion engines. We first introduce some terminology and set out the major challenges that fusion engines propose to solve. A history of past work in the field of fusion engines is then presented using the BRETAM model. These approaches to fusion are then classified. The classification considers the types of application, the fusion principles and the temporal aspects. Finally, the challenges for future work in the field of fusion engines are set out. These include software frameworks, quantitative evaluation, machine learning and adaptation.

Categories and Subject Descriptors
D2.2 [**Software Engineering**]: Design Tools and Techniques – *Modules and interfaces; user interfaces*. H.1.2 [**Information Systems**]: Models and Principles – *User/Machine Systems*. H5.2 [**Information interfaces and presentation**]: User Interfaces – *Prototyping; user-centered design; user interface management systems (UIMS)*. I.6.5 [**Model Development**]: modeling methodologies.

General Terms: Design, Experimentation, Human Factors, Reliability

Keywords: Fusion engine, multimodal interfaces, interaction techniques.

1. INTRODUCTION
Interactive systems featuring multimodal interfaces are becoming widespread. They now cover many different application domains and support a wide variety of users in the performance of their tasks. While originally multimodal interfaces focussed on speech as a central modality, they now encompass a wide range of modalities including eye gaze, gestures, touch interaction, two-handed interaction as well as modalities based on multiple inputs on one device (e.g. multitouch).

ICMI-MLMI'09, November 2–4, 2009, Cambridge, MA, USA.

Multimodal interactive systems enable users to interact with computers through various input modalities (e.g. speech, gesture, eye gaze) and output channels (e.g. text, graphics, sound, avatars, synthesized speech). This type of user interface is not only beneficial for enhanced accessibility (e.g. visually or motor impaired people), but also for greater convenience (e.g., natural input mode recognition) as well as flexibility (e.g. adaptation to context of use, to tasks or to users' preferred interaction modalities).

In multimodal interactive systems, multimodal fusion is a crucial step in combining and interpreting the various input modalities. This survey paper is organized as follows: Section 2 introduces our terminology and some key features of fusion engines. Section 3 presents the history of the work in the field of fusion engines while section 4 presents a classification of existing approaches for fusion engines based on criteria including tools, types of applications, fusion principles and temporal aspects. Section 5 presents an agenda for future research in the field of fusion engines for interactive systems featuring multimodal interfaces.

2. FUSION ENGINES: TERMINOLOGY
According to the classification by Nigay & Coutaz [27], multimodal interfaces can handle inputs in different ways in order to make sense of a set of information provided by the various modalities. The vertical columns of Table 1 represent how modalities may be used by the users of the multimodal interface, while the lines represent the fact that information provided by several modalities may be combined or may be kept independent. One way of increasing the bandwidth between the user and the system (i.e. the rate of transmission of information from the user to the interactive system) is to allow the users to use several modalities at the same time. This corresponds to the *parallel* column in Table 1. If the information received in parallel from the modalities is combined then the multimodal interaction technique is called *synergistic* [27].

Table 1. Types of multimodal interfaces: two dimensions from the classification space presented by Nigay & Coutaz [27].

		Use of modalities	
		Sequential	**Parallel**
Fusion	Combined	Alternative	Synergistic
	Independent	Exclusive	Concurrent

2.1 Terminology on fusion engines

The mechanisms used for combining information (whether it is received in a sequential or parallel way) have received different names in the past.

For instance Cubricon [26] uses the word *combining* (e.g. in "combining Natural Language and Gesture") while Martin et al. [22] use the word *cooperation of modalities*. In a similar context, Quickset [9], Pfleger [33] have used the word *integration* (e.g. in "integration of input from different mode" (Quickset) and "context-based multimodal integration" (Pfleger)) while Latoschik [21], Johnston et al. [18] and Shiker et al. [36] report research work on *multimodal integration*.

More widely, the word *fusion* has been used for describing such a mechanism. However, different qualifications have been used together with fusion depending on which aspect of the mechanism was concerned. For instance, Flippo et al. [14], Nigay et al. [27] and Portillo et al. [34] talk about *fusion process*, Milota [25] about *fusion strategies*, Nigay et al. [28], as well as Dumas et al. [12][13], about *fusion mechanisms*. In some work, the emphasis has been put on the information element of the multimodality rather than the process of combining the information as in [37] where the authors use the words *data fusion*. Distinction between input and output combination of modalities has also been made explicit: For instance, Nigay et al. [28] and Melichar et al. [24] define this concept as *multimodal input fusion* while Mansoux et al. [23] explicitly focus on a component-based framework for output multimodality that is adapted from an approach for multimodal input fusion. Many authors have recently been using the name *multimodal fusion* as a way to address these concepts in a generic way [11][20][17].

In this survey paper, we will use exclusively the term *fusion engine* to refer to the computational element in charge of combining the information produced through user actions captured by input devices into meaningful commands.

2.2 Levels of fusion engines

Fusion engines are often classified by the level at which fusion takes place. Multimodal fusion can operate at the data level (directly on the input streams), at the feature level (patterns and characteristics extracted from the data), or at the decision level (e.g. recognized tasks). Based on Nielsen's model of interaction [32], seven layers are identified from a user's mental goal (e.g., delete a paragraph from my letter) to physical actions. As shown in Table 2, Nielsen's model of interaction decomposes a goal (e.g., delete a paragraph from my letter) into seven layers, refining it into simpler units of interaction at each level.

Fusion can be performed at any level of Table 2. Extending a combination of Tables 1 and 2, Vernier & Nigay [39] define a comprehensive design space for multimodal fusion, based Allen's relationships [1] applied to the levels of Nielsen's model of interaction. A fusion engine may incorporate several fusion mechanisms which correspond to different sections of the design space presented.

2.3 Important features of fusion engines

One major problem tackled by multimodal fusion engines resides in the various types of inputs being manipulated. Moreover fusion engines manipulate temporal combinations of deterministic inputs as well as non-deterministic inputs (whose meaning can vary according to the context e.g. user and task and require interpretation that remains uncertain until additional information is provided):

- *Probabilistic inputs* - In a standard GUI, mouse movements and keyboards strokes are used to control the machine. They correspond to deterministic events. But input streams can also correspond to natural human communication means such as speech or gesture. They have to be first interpreted by probabilistic recognizers (HMM, GMM, SOM, etc.) and thus their results are weighted by a degree of uncertainty.
- *Multiple and temporal combinations*- Time synchronicity between input modalities is particularly problematic since the interpretation might vary depending on the time at which modalities are used.
- *Adaptation to context, task and user*- A multimodal command can be interpreted differently depending on the context of use (e.g. home, car, work), the task being performed, the application's state, or the user's preferences,.
- *Error handling*- With probabilistic inputs that can be combined in different ways over time and that should be interpreted by considering contexts, tasks and users' preferences, errors will be difficult to avoid. Fusion engines should provide mechanisms for users to correct the machine's answers and for it to learn from its errors.

Fusion engines should address these features in order to enable robust and real-time multimodal interactions with interactive systems.

Table 2. Seven levels of Nielsen's linguistic model of interaction [27].

Level	Title	Units	Definition	Example	World
1	Goal	Concepts of real world	Mentalization of a goal, a wish in the user's head	Delete a paragraph from my document	Conceptual
2	Pragmatic	Concepts of system	Translation of a goal into system concepts	Delete 6 lines of the current paragraph in the edited text	
3	Semantic	Detailed functions	Real world objects translated into system objects manipulated by functions	Delete several lines	
4	Syntactic	System sentences	Time & space sequencing of information units	DELETE 6	Perceptual
5	Lexical	Information units	Smallest elements transporting significant information: word, figure, screen coordinates, icon	[DELETE] command, [6] number	
6	Alphabetic	Lexems	Primitive symbols: letter, numbers, columns, lines, dots, phonems, ...	D, E, L, E, T, E, 6	Physical
7	Physical	Physically coded information	Light, sound, physical moving	Pressing [CTRL] + [D] followed by [6]	

3. HISTORICAL PERSPECTIVE ON FUSION ENGINES

This section presents how research in the field of fusion engines has evolved over the years. The seminal paper from Bolt [4] presented a prototype of an interactive system featuring a multimodal interface using both speech and gesture modalities in 1980, but it took another 13 years for the first paper addressing details of fusion engine design to be published [27].

Since that early work, several scientific studies have focussed on fusion engines. Figure 1 presents the number of such contributions over the years. They are organised by year and by conference. The red part corresponds to the number of scientific papers at CHI conferences, the blue part corresponds to ICMI conferences while the green part corresponds to other HCI related conferences. We can see that there was a peak of about 10 publications per year in 2003-04, and that several contributions on the topic of fusion engines have been made every year since then.

It is interesting to note the impact of the ICMI conference on publications related to fusion engines that are at the core of multimodal interfaces. Indeed, the creation of the conference has boosted the number of published contributions significantly.

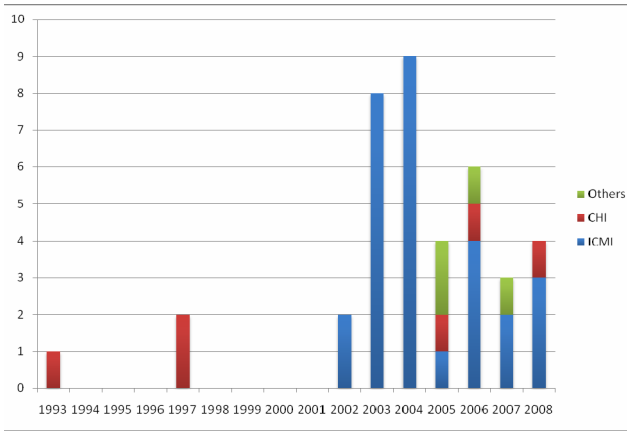

Figure 1. A Publication History of Fusion Keyword

We now examine these contributions using Brian Gaines' model of technological development and diffusion [15]. This model called BRETAM defines six phases namely **B**reakthrough, **R**eplication, **E**mpiricism, **T**heory, **A**utomation and lastly **M**aturity. Nigay et al. [29] used the BRETAM model in their study of software tools and architecture models for multimodal interaction: They show that such tools are now crucial for the **R**eplication phase. While generic tools and platforms (e.g., toolbox, UIMS) for developing multimodal interaction are crucial for the Replication phase of the "multimodal interface" research field, we argue in this section that research work carried out in the field of fusion engines has now reached the **M**aturity level according to the BRETAM model

3.1 Breakthrough Phase

According to that framework, each Technology begins with a breakthrough. In the field of fusion engines, the breakthrough came from Bolt's *Put that there* paradigm [4]. However, the notion of fusion engine was neither introduced nor discussed in that paper, even though merging of information produced by the two modalities (gesture and speech) had to be addressed at the implementation level.

Since then the multimodal interfaces have started to be designed and implemented which move the research field into the Replication phase.

3.2 Replication Phase

Further research work has now moved the research field from Breakthrough to the Replication phase. As far as fusion engines are concerned, the work in the Replication phase has identified issues raised by fusion engines but remain at a very high level of abstraction more focusing on the identification of problems rather than proposing solutions.

CUBRICON [26] uses speech with deictic gestures and graphical expressions in a map application. The system combines the input streams into a single compound stream having temporal order of the tokens. The parser corresponds to a state-based model represented by a generalized augmented transition network. CUBRICON contains a set of rules for inferring the intended referent in case of ambiguity. This is done by either selecting the closest object that satisfies the criteria or by issuing an advisory statement describing the inconsistency. These disambiguation rules (in addition to the input stream fusion) can be considered as the first explicit representation of fusion engine behaviour.

Xtra [40] (eXpert TRAnslator) is an interactive multimodal system based on keyboard for Natural Language and mouse pointing as input modalities. The underlying idea of Xtra is to exploit a multimodal interfaces in order to increase the bandwidth between the user and the underlying tax declaration system.

CUBRICON and Xtra can be considered as first steps towards the engineering of fusion engines. However, CUBRICON and Xtra only focus on the sequential usage of modalities. For example, in CUBRICON the user must stop to speak before pointing. As a consequence, these first descriptions of fusion engines only address part of the design space presented in Figure 1 (the "sequential" column).

3.3 Empiricism Phase

In Gaines' Empiricism phase, lessons are drawn from experience and formulated as empirical design rules that have been found useful.

We can find four significant contributions that can be assigned to that phase: the integration of speech, gaze and hand gestures by Koons et al. [19], the PAC-Amodeus architecture and its fusion engine [28], the Quickset platform [9] and the fusion engine by Johnston and Bangalore [18].

Koons et al. [19] study three modalities: gaze, speech and hand gestures in the "blocks world", a graphical 3D system. Modalities are first parsed individually; the parsers then produce the information in a *common frame-based format* for fusion. All the information is received in parallel and is time-stamped.

PAC-Amodeus [28] is a software architecture model for multimodal interactive systems. It clearly belongs to the Empiricism phase as the architecture aims at providing guidelines for the structure of the code of such systems. The architecture is illustrated by considering a system called MATIS. MATIS (Multimodal Air Traffic Information System) is a multimodal interface to a database. It offers several modalities such as natural language input (through speech and text via keyboard) and graphical input via a mouse and direct manipulation interaction technique. It is possible for the user to use any modality for triggering commands in the system as the modalities are "equivalent" according to the properties defined in [10]. In MATIS, fusion is made at a high-level of abstraction (in a component called

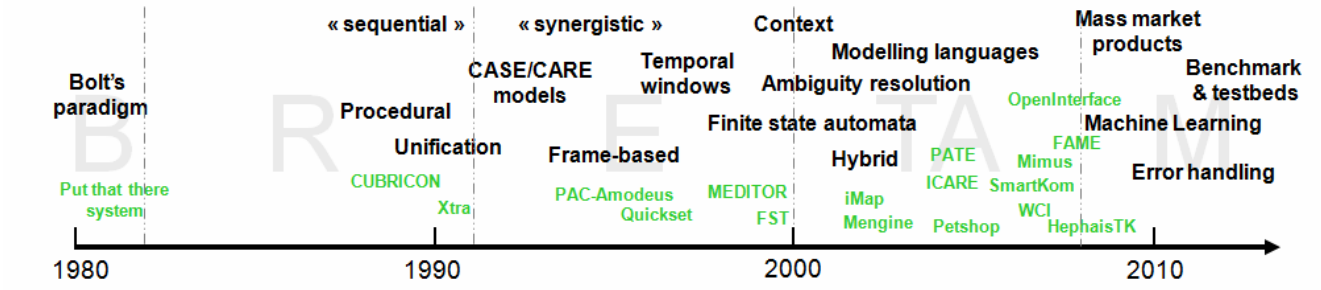

Figure 2 – Evolution of multimodal fusion concepts, mechanisms and systems following the BRETAM model. Although the market is now ready for multimodal applications, performance evaluation of fusion engines and a better error handling is still required towards reliable and usable systems

Dialogue Controller defined in the PAC-Amodeus architecture model) using a generic fusion engine based on a common representation, a Melting Pot. In the melting pot (which is a *two dimensional structure* representing an event with both structural parts and temporal information) fusion obeys to three principles: complementarity, near time and context rules in the case of a long delay without resolved fusion.

Quickset [9] is an interactive system featuring a multimodal interface with two modalities: graphical (using a pen-based interaction) and speech (using a voice recognition system). These modalities are used to control Leathernet which is a simulation system for training of US Marine Corps platoon leaders. Quickset has also been used with MIMI a search engine for finding health care facilities. Fusion in Quickset is done by means of unification which checks the consistency of two pieces of partial information (from the two modalities). If the information received is consistent then Quickset combines them into a single result. Fusion is done by means of *feature structures* [8], which involve the parsing of the two event streams to determine potential integration, the tagging of the speech and gesture events as complete or partial and the examination of timestamps. Gesture can compensate speech errors.

Johnston & Bangalore [18] present a multimodal user interface for a corporate directory and messaging interactive systems. The system features two modalities: a pen-based and a speech-based one. The two recognizers (in charge of receiving the events produced by the input devices) send to the integration part (i.e. the fusion engine) a lattice representing the possible recognized strings and the possible recognized gestures. The fusion is described by means of a set of *finite state automata* representing a context-free grammar (one automaton for each modalities plus one for the fusion engine).

All these systems support synergistic usage of two or more modalities according to the classification of Figure 1. Moreover the user can use the modalities in a concurrent way i.e. at the same time. This usage of modalities increases the complexity of the fusion engine and adds issues such as (for instance) processing the time differences between the interpretation processes related to modalities such as speech and unambiguous direct manipulation using the mouse. The fact that the modalities can be used in a synergistic way has required the use of notations to describe the behaviour of the fusion engines: Each of these contributions has proposed a different representation on which relies the fusion engine.

3.4 Theory and Automation Phases

The two phases following the Empiricism one are called Theory and Automation. When the technology reaches this phase, hypotheses are formed about the causal systems underlying experience and developed as theories. In the automation phase, theories are accepted and used automatically to predict experiences and to generate design rules. We gather theses two phases together as, in the field of multimodal interfaces and more precisely fusion engines, each theoretical contribution is immediately integrated into a system that plays the role of demonstrator.

The first group of work presented in this section is a set of five contributions building on the Empiricism phase presented in the previous section.

Latoschik [21], for example, extends the Johnston and Bangalore work with tATN (temporal augmented transition network) in order to represent quantitative temporal aspects in the fusion engine. The need for a quantitative representation of time was already identified in MATIS [27] but Latoschik introduces a formal notation to address this issue. Flippo [14] and Portillo [34] combine techniques from Quickset and PAC-Amodeus to create a hybrid fusion engine exploiting both time-frame and unification mechanisms for solving ambiguities. Bouchet & Nigay [5] [6] extend their early fusion work based on the PAC-Amodeus architectural model by defining a set of micro fusion engines as reusable and composable software components.

In parallel to that research work, new approaches have been proposed to address unsolved issues in the engineering of fusion engines. These new approaches are presented with more details in Section 4 where we propose a classification of fusion engines.

3.5 Maturity Phase

According to Gaines' schema, at the Maturity phase theories have been assimilated and are used routinely without questions. One way to assess this characteristic for a technology is when it starts being deployed in large practical applications or in the field of safety critical systems.

Multimodal interfaces and their accompanying fusion engines have reached such a maturity phase. Indeed, new worldwide mass market products such as the Wii game console [35] and the iPhone [38] feature a native multimodal interaction either by means of several input devices (two or more wii-motes for instance) or the multitouch tactile interaction on the iPhone. In the field of safety critical systems, the introduction of KCCU (Keyboard Cursor Control Units) in the cockpits of large aircrafts such as the Airbus 380 or the Boeing 787 makes it implicit the necessity to handle synergistic use of multiple input devices (even though one is managed by the pilot and the other one by the first officer) [31].

4. FUSION ENGINE CLASSIFICATION

We now propose a classification of fusion engines using the criteria shown in Table 3. The systems are first sorted according to the BRETAM model which splits the table in 4 clusters of lines: B for Breakthrough, R for replication, etc. The first two columns provide the reference of the work and the names that are used in the bibliography to refer to that work. The last column of the table provides the types of application that were described in the corresponding publications. It does not mean that the contribution is not able to deal with other application types but it provides a perspective on the kind of problems that the authors were trying to solve. The two remaining groups of columns named *Fusion* and *Time Representation* directly address the characteristics of the fusion engines. In the column *fusion* the first element is **notation**. The notation is the language for representing the behaviour of the fusion engine. When that information is not given in the contribution (it usually means that the fusion engine was directly implemented in a programming language) the keyword None is used. It is clear that the notation used has an impact on the fusion type which is the second element of the fusion column.

In the column *fusion,* the second element is **fusion type**. The possible values belong to the following set {Frame-based, Unification, Procedural and Hybrid}. This value corresponds to the way fusion is performed either in a tabular form (Frame-based), using rule-based constructions of valid commands (Unification), constructing algorithmic management of input events to be combined usually by means of explicit representation of the state space (Procedural) or by merging the previous types Frame-based and Unification (Hybrid).

The third element of the *fusion* column deals with the **level** at which the fusion is performed. Instead of considering the seven levels of Table 2, we only consider two coarse-grain levels corresponding to the architectural description of interactive systems as provided in Arch [2]. The possible values are Low-level or Dialog. When fusion is called Low-level, it means that the fusion engine is able to perform fusion of raw data events provided by the input devices to produce higher-level events. When the level is Dialog, it means that the fusion of events can immediately be used to trigger application commands. Most of the fusion engines presented here focus primarily on the Dialog level. This does not mean that low-level fusion is not possible but that the authors were more interested in showing the expressivity of their approach at that level. One reason for that interest for the Dialog-level fusion is also that most of the contributions target at speech as a primary modality. In such cases, speech information typically refers to domain objects that are only available at the Dialog level.

The fourth element of the *fusion* column deals with the set of **input devices** that have been used as inputs for the fusion engine. The column defines what is presented in the contribution and does not mean that the fusion engine is not able to handle other input devices but that capability has not been demonstrated.

The last element of the *fusion* column deals with **Ambiguity Resolution**. Indeed, when several input events have to be fused, there is always the possibility that some information is missing, that there is too much information or that the information received is not compatible. Two main policies for ambiguity resolution have been found: the first one is based on defining priorities amongst the modalities which appear as "S/G" or "G/S" in the table. Iterative testing after the production of a list of possible fusion is the second policy and is called "N-best" in the table. Other resolving policies exist and are usually embedded in the behavioural description of the fusion engine. For instance, Latoschik [21] uses fuzzy constraints in the temporal augmented network to address ambiguity.

The last but one column deals with the notion of **Time Representation** in the description of the fusion engine behaviour. Time is a key concept to be represented in order to produce commands from several events received from multiple input devices. The temporal behaviour can be defined at two different levels: Quantitative and Qualitative. Quantitative time allows us to represent behavioural temporal evolutions related to a given amount of time (usually expressed in milliseconds) or at a precise moment in time (at 10.00 am for instance). Qualitative time addresses the issue of ordering of actions such as precedence, succession, simultaneity. Qualitative time approaches can be used for representing temporal evolutions between events such as ev1 before ev2, ev1 after ev2, ev1 in any order with ev2.

The expressive power of the underlying notation for describing the fusion engine has a direct impact on the temporal representation. For instance, finite state automata or Augmented Transition Network do not allow for the representation of concurrent behaviour and thus do not make it possible to express constraints such as ev1 and ev2 can occur at the same time.

5. RESEARCH AGENDA FOR FUSION ENGINES

Multimodal fusion engines exhibit a large potential for wide applications in various domains including authentication, video search, affective cues, augmented reality, user interface adaptation, animation, and mobile user interfaces. Research in this domain has reached a mature phase in terms of concepts and know-how with many implementation solutions in the last decade. It is now necessary to consolidate findings and build *common evaluation procedures*, with the associated *testbeds* and metrics, to compare fusion engines at a performance level. Further, in order to build useful testbeds, fusion engine *interpretation errors* should be better characterized.

Table 3. Characteristics of Fusion Engines

	Reference	Tool/ language/ program	Fusion					Time Representation		Application types
			Notation	Type	Level	Input Devices	Ambiguity Resolution	Quantitative	Qualitative	
B	Bolt [4]	Put that here system	None	None	Dialog	Speech gesture	?	N	?	Map manipulation
R	Wahlster **Error! Reference source not found.**	XTRA	None	Unification	Dialog	Keyboard Mouse		N	Y	Map manipulation
	Neal [26]	Cubricon	Generalized Augmented Transition Network	Procedural	Dialog	Speech Mouse Keyboard	Proximity-based	N	Y	Map manipulation
E	Koons [19]	No name	Parse tree	Frame-based	Dialog	Speech, Eye gaze, Gesture	First solution	Y	Y	3D World
	Nigay [28]	Pac-Amodeus	Melting Pot	Frame-based	Dialog + low level	Speech, Keyboard, Mouse	Context-based resolution	Y	N	Flight Scheduling
	Cohen [9]	Quickset	Feature Structure	Unification	Dialog	Pen Voice	S / G & G / S & N best	Y	N	Simulation System training
	Bellik [3]	MEDITOR	None	Frame-based	Dialog + low level	Speech Mouse	History Buffer	Y	Y	Text Editor
	Martin [22]	TYCOON	Set of processes – Guided Propagation Networks	Procedural	Dialog	Speech Keyboard Mouse	Probability-based resolution	Y	Y	Edition of graphical user interfaces
	Johnston [18]	FST	Finite State Automata	Procedural	Dialog	Speech pen	Possible (N best)	Y	Y	Corporate Directory
T & A	Krahnstoever [20]	iMap	Stream Stamped	Frame-based	Dialog	Speech gesture	Not given	Y	N	Crisis Management
	Dumas [12]	HephaisTK	XML Typed (SMUIML)	Frame-based	Dialog	Speech Mouse Phidgets	First one	Y	Y	Meeting assistants
	Holzapfel [17]	No Name	Typed Feature Structure	Unification	Dialog	Speech gesture	N Best list	Y	N	Humanoid Robot
	Pfleger [33]	PATE	XML Typed	Unification	Dialog	Speech pen	N Best list	Y	Y	Bathroom design Tool
	Milota [25]	No Name	Multimodal Parse Tree	Unification	Dialog	Speech Mouse keyboard Touch-screen	S / G & G /S	Y	N	Graphic Design
	Melichar [24]	WCI	Multimodal Generic Dialog Node	Unification	Dialog	Speech Mouse Keyboard	First One	?	?	Multimedia DB
	Sun [37]	PUMPP	Matrix	Unification	Dialog	Speech gesture	S / G	N	Y	Traffic Control
	Bourguet [7]	Mengine	Finite State machine	Procedural	Low level	Speech Mouse	Not given	N	Y	No example
	Latoschik [21]	No Name	Temporal Augmented Transition Network	Procedural	Dialog	Speech gesture	Fuzzy constraints	Y	Y	Virtual reality
	Bouchet [5] [6] Mansoux [23]	ICARE (Input/Output)	Melting pot	Frame-based	Dialog + low level	Speech, Helmet visor HOTAS, Tactile surface, GPS localization, Magnetometer, Mouse, Key-board	Context-based resolution	Y	N	Aircraft Cockpit, Authentication, Mobile Augmented Reality systems (Game, Post-it), Augmented Surgery
	Navarre [30]	Petshop	Petri nets	Procedural	Dialog + low level	Speech mouse Keyboard Touchscreen	***	Y	Y	Aircraft Cockpit
	Flippo [14]	No Name	Semantic tree	Hybrid	Dialog	Speech Mouse Gaze gesture	Feedback for missing data	Y	N	Collaborative Map
	Portillo [34]	MIMUS	Feature Value Structure (DTAC)	Hybrid	Dialog	Speech Mouse	Knowledge-able agent	Y	N	
	Duarte [11]	FAME	Behavioral Matrix	Hybrid	Dialog	Speech Mouse Keyboard	Not given	?	?	Digital talking Book

Table Key for Ambiguity Resolving column: S / G: Speech resolving gesture ambiguity ---- G / S: Gesture resolving speech ambiguity ---***: Possible: Ambiguity resolving is embedded in the procedural model

Evaluation of multimodal systems has mainly focused so far on user interaction and user experience evaluation. These evaluations offer important insights about a given user interface, and on the way it is being used, but, considering the complexity of the processing chain associated with multimodal interactive systems, analysis of what to correct and how, is problematic. To open the fusion engines' black box and quantitatively evaluate them, their evaluation should be properly decoupled from the evaluation of input recognizers. Further, practitioners should reflect on the most critical issues associated with fusion at the decision level and on difficult cases or combination of events that generate interpretation errors or ambiguities. Issues related to fusion engines' adaptation to context (environment and also applications), as well as users' favorite usage patterns or repetitive errors, should also be considered. We feel that, by providing a common measurement of the efficiency and effectiveness of fusion engines and by testing them against a set of problematic and common cases, strengths and weaknesses of these different mechanisms should clearly be identified.

For a reliable quantitative evaluation of fusion engines, it is important to identify precisely the different types of interpretation errors they can generate. This is particularly important to build relevant testbeds on which to run performance evaluations. A testbed for multimodal fusion processes should also pay attention to user and context. In particular concerning user adaptation, it has been shown that, if integration patterns differ largely from one user to another, a given user tends to keep the same integration patterns and remain persistent throughout a same session. For this reason, testbeds should also supply a set of sequences of consistent uses for a given use case, so that the adaptability to users' integration patterns can also be evaluated.

Aside from performance evaluation and errors handling that will contribute to consolidate the domain on fusion engines, we also identify extensions of fusion engines that require further studies.

First the *dynamic adaptation (adaptivity) of fusion engines* to usage patterns and preferences should be further studied. For example machine learning techniques could enable fusion engines to adapt to users, as well as to detect context and user's behaviour patterns and changes. Machine learning has been already applied to multimodal interfaces, mainly modality recognition (e.g. speech, gesture recognition). The goal would be to define adaptive fusion engines that are reliable and usable.

Second, *engineering aspects of fusion engines* must be further studied, including the genericity (i.e., engine independent of the combined modalities), software tools for the fine-tuning of fusion by the designer or by the end-users as well as tools for rapidly simulating and configuring fusion engines to a particular application by the designer or by the end-users.

6. CONCLUSION

The article surveys the technical challenges associated with the design, implementation and evaluation of fusion engines and their evolution since the seminal work of Bolt [4]. Reviewing the major system implemented over the last 25 years, the various fusion types they implement are presented, as well as their temporal properties, notations and ambiguity resolution features. Finally, the article proposes a research agenda for future works in the domain, such as issues related to software frameworks, quantitative evaluation, machine learning and adaptation.

However, the proposed research agenda is not set in stone. Do we clearly know what problems are susceptible to be solved by multimodal fusion? Are there different classes of problems and a set of associated technologies? How will machine learning techniques affect fusion engines? These questions should be properly addressed by practitioners in the field in order to characterize better the applications and problems that fusion engines try to address.

7. ACKNOWLEDGMENTS

This research was partly financed by the CNES R&T Tortuga project, R-S08/BS-0003-029, the Network of Excellence ReSIST IST-4-026764-NoE (www.resist-noe.org), the ITEA2 Call 3 project UsiXML, DGA (French Army Research Dept.) under contract #00.70.624.00.470.75.96 INTUITION, and by the EU project OpenInterface FP6 STREP FP6-035182 on MM interaction.

8. REFERENCES

[1] Allen, J.: Maintaining Knowledge about Temporal Intervals. *Communications of the ACM, Vol. 26, No. 11*, (1983) 832-843.

[2] Bass, L., Pellegrino, R., Reed, S., Seacord, R., Sheppard, R., and Szezur, M. R. The Arch model: Seeheim revisited. proceeding of the User Interface Developpers' workshop. 91.

[3] Bellik, Y. (1997) Media integration in multimodal interfaces. *Proceedings of IEEE Workshop on Multimedia Signal Processing*, Princeton, NJ.

[4] Bolt, R. A. 1980. "Put-that-there": Voice and gesture at the graphics interface. In *Proceedings of the 7th Annual Conference on Computer Graphics and interactive Techniques*. SIGGRAPH '80. ACM, New York, NY, 262-270.

[5] Bouchet, J., Nigay, L. ICARE: A Component-Based Approach for the Design and Development of Multimodal Interfaces. In *Extended Abstracts CHI'04 (2004)*. ACM Press, 1325-1328.

[6] Bouchet, J., Nigay, L. ICARE Software Components for Rapidly Developing Multimodal Interfaces. . In *Proceedings of the 6th international Conference on Multimodal interfaces* (State College, PA, USA, 2004). ICMI '04. ACM, New York, NY, 251-258.

[7] Bourguet, M. L. A Toolkit for Creating and Testing Multimodal Interface Designs. *Proceedings of UIST'02*, Paris. 2002, pp. 29--30.

[8] Carpenter, R. 1990. Typed feature structures: Inheritance, (In)equality, and Extensionality. In W. Daelemans and O. Gazdar (Eds.), Proceedings of the 1TK Workshop: Inheritance in Natural Language Processing, Tilburg University, pp. 9-18.

[9] Cohen, P. R., Johnston, M., McGee, D., Oviatt, S., Pittman, J., Smith, I., Chen, L., and Clow, J. 1997. QuickSet: multimodal interaction for distributed applications. In *Proceedings of the Fifth ACM international Conference on Multimedia* (Seattle, Washington, United States, November 09 - 13, 1997). MULTIMEDIA '97. ACM, New York, NY, 31-40.

[10] Coutaz, J., Nigay, L., Salber, D., Blandford, A., May, J. and Young, R. Four Easy Pieces for Assessing the Usability of Multimodal Interaction: The CARE properties. Proceedings of INTERACT'95 conference, 1995, pp. 115--120.

[11] Duarte, C. and Carriço, L. 2006. A conceptual framework for developing adaptive multimodal applications. In *Proceedings of the 11th international Conference on intelligent User interfaces* (Sydney, Australia, January 29 - February 01, 2006). IUI '06. ACM, New York, NY, 132-139.

[12] Dumas, B., Lalanne, D., Guinard, D., Koenig, R., and Ingold, R. 2008. Strengths and weaknesses of software architectures for the rapid creation of tangible and multimodal interfaces. In *Proc. of the 2nd int. Conf.*

on Tangible and Embedded interaction (Bonn, Germany, 2008). TEI '08. ACM, 47-54

[13] Dumas B., Lalanne D. & Oviatt S. Multimodal Interfaces: A Survey of Principles, Models and Frameworks. D. Lalanne and J. Kohlas (Eds.): Human Machine Interaction, LNCS 5440, pp. 3–26, 2009. © Springer-Verlag Berlin 2009

[14] Flippo, F., Krebs, A., and Marsic, I. 2003. A framework for rapid development of multimodal interfaces. In *Proceedings of the 5th international Conference on Multimodal interfaces*. ICMI '03. ACM, New York, NY, 109-116.

[15] Gaines, B. R., Modeling and Forecasting the Information Sciences. *Information Sciences* 57-58, 1991, p. 3-22.

[16] Jaimes, A. and Sebe, N. 2007. Multimodal human-computer interaction: A survey. *Comput. Vis. Image Underst.* 108, 1-2 (Oct. 2007), 116-134.

[17] Holzapfel, H., Nickel, K., and Stiefelhagen, R. 2004. Implementation and evaluation of a constraint-based multimodal fusion system for speech and 3D pointing gestures. In *Proceedings of the 6th international Conference on Multimodal interfaces* (State College, PA, USA. ICMI '04. ACM, New York, NY, 175-182.

[18] Johnston, M. and Bangalore, S. Finite-state multimodal integration and understanding. *Nat. Lang. Eng.* 11, 2 (Jun. 2005), 159-187

[19] Koons, D. B., Sparrell, C. J., and Thorisson, K. R. 1993. Integrating simultaneous input from speech, gaze, and hand gestures. In *intelligent Multimedia interfaces*, M. T. Maybury, Ed. American Association for Artificial Intelligence, Menlo Park, CA, 257-276.

[20] Krahnstoever, N., Kettebekov, S., Yeasin, M., and Sharma, R. 2002. A Real-Time Framework for Natural Multimodal Interaction with Large Screen Displays. In *Proceedings of the 4th IEEE international Conference on Multimodal interfaces - Volume 00* (October 14 - 16, 2002). International Conference on Multimodal Interfaces. IEEE Computer Society, Washington, DC, 349.

[21] Latoschik, M.E., 2002. Designing transition networks for multimodal VR-interactions using a markup language. Multimodal Interfaces, 2002. In Proceedings of the Fourth IEEE International Conference on Multimodal Interfaces. IEEE, 411-416.

[22] Martin, J.C., Veldman, R., & Beroule, D. 1998. Developping multimodal interfaces: a theoretical framework and guided propagation networks. Multimodal Human-Computer Communication. Springer Verlag LNAI 1374.

[23] Mansoux, B., Nigay, L., Troccaz, J. Output Multimodal Interaction: The Case of Augmented Surgery. In *Proceedings of HCI'06, Human Computer Interaction, People and Computers XX, the 20th BCS HCI Group conference*, (London, UK,11-15 September, 2006). Springer Publ, 177-192.

[24] Melichar, M. and Cenek, P. From vocal to multimodal dialogue management. In *Proceedings of the 8th international Conference on Multimodal interfaces* 2006. ICMI '06. ACM, New York, 59-67.

[25] Milota, A. D. Modality fusion for graphic design applications. In *Proceedings of the 6th international Conference on Multimodal interfaces*. ICMI '04. ACM, New York, NY, 167-174.

[26] Neal, J. G., Thielman, C. Y., Dobes, Z., Haller, S. M., and Shapiro, S. C. 1989. Natural language with integrated deictic and graphic gestures. In *Proceedings of the Workshop on Speech and Natural Language*. Human Language Technology Conference. Association for Computational Linguistics, Morristown, NJ, 410-423

[27] Nigay, L. and Coutaz, J. A design space for multimodal systems: concurrent processing and data fusion. In *Proceedings of the INTERCHI '93 Conference on Human Factors in Computing Systems*. S. Ashlund, A. Henderson, E. Hollnagel, K. Mullet, and T. White, Eds. IOS Press, Amsterdam, 172-178.

[28] Nigay, L. and Coutaz, J. A generic platform for addressing the multimodal challenge. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Denver, Colorado, United States, May 07 - 11, 1995). Conference on Human Factors in Computing Systems. ACM Press, New York, NY, 98-105.

[29] Nigay, L., et al. Software Engineering for Multimodal Interactive Systems. In *Multimodal user interfaces: from signals to interaction*, Chapter 9, Springer, Lecture Notes in Electrical Engineering, ISBN13: 9783540783442, 201-218.

[30] Navarre, D., Palanque, P., Bastide, R., Schyn, A., Winckler, M.A., Nedel, L., Freitas, C. A Formal Description of Multimodal Interaction Techniques for Immersive Virtual Reality Applications. In: INTERACT 2005: IFIP TC13 International Conference, Rome, Italy, Springer-Verlag LNCS 3585, p. 170-185, 2005.

[31] Navarre, D., Palanque, P., Basnyat, S. Usability Service Continuation through Reconfiguration of Input and Output Devices in Safety Critical Interactive Systems. The 27th International Conference on Computer Safety, Reliability and Security (SAFECOMP 2008), LNCS 5219, pp. 373–386, 2008.

[32] Nielsen, J.: A Virtual Protocol Model for Computer-Human Interaction. International Journal of Man-Machine Studies 24,3 (1986) 301–312

[33] Pfleger, N. 2004. Context based multimodal fusion. In *Proceedings of the 6th international Conference on Multimodal interfaces* (State College, PA, USA, October 13 - 15, 2004). ICMI '04. ACM, New York, NY, 265-272.

[34] Portillo, P. M., García, G. P., and Carredano, G. A. 2006. Multimodal fusion: a new hybrid strategy for dialogue systems. In *Proceedings of the 8th international Conference on Multimodal interfaces* (Banff, Alberta, Canada, November 02 - 04, 2006). ICMI '06. ACM, New York, NY, 357-363

[35] Schlomer, T., Poppinga B., Henze N. & Boll S. Gesture Recognition with a Wii Controller, Proceedings of the 2nd international conference on Tangible and embedded interaction, ACM, 2008

[36] Shikler, T. S., Kaliouby, R. and Robinson, P. 2004. Design Challenges in Multi-modal Inference Systems for Human-Computer Interaction. In Proceedings of International Workshop on Universal Access and Assistive Technology, Fitzwilliam College, University of Cambridge, United Kingdom, 22nd-24th March, 2004

[37] Sun, Y., Chen, F., Shi, Y. and Chung, V. 2006. A novel method for multi-sensory data fusion in multimodal human computer interaction. In *Proceedings of the 18th Australia Conf. on Computer-Human Interaction: Design: Activities, Artefacts and Environments*. . OZCHI '06, vol. 206. ACM, New York, NY, 401-404.

[38] Jobs, S. P. et al. (2008). Touch Screen Device, Method, and Graphical User Interface for Determining Commands by Applying Heuristics. United States Patent Application 20080122796. Kind Code A, May 29, 2008.

[39] Vernier, F. and Nigay, L. A Framework for the Combination and Characterization of Output Modalities. *In Proceedings of DSV-IS 2000,* (Limerick IR, 2000). LNCS 1946, Springer-Verlag, 32-48.

[40] Wahlster, W. 1991. User and discourse models for multimodal communication. In *intelligent User Interfaces*, J. W. Sullivan and S. W. Tyler, Eds. ACM, New York, NY, 45-67.