Paper 1

Representing multimodal input in a unification-based system: the Staging project

Patrizia Paggio and Bart Jongejan
patrizia@cst.ku.dk, *bart@cst.ku.dk*
Center for Sprogteknologi (CST)
Copenhagen, DK

1 Introduction

The integration of multimodal input is a key research issue in Staging, a Danish research project the aim of which is to investigate the nature and applications of multimodal, 3D virtual worlds. A crucial aspect related to the use of such worlds is the way in which users can interact with them. Working on the assumption that an interface allowing the user to exploit several modalities of input, is a step towards more natural man-machine interaction, the project is developing a methodology for the integration of multimodal input in a prototype implementation. The prototype is a virtual world application implementing at the moment a farm scenario (Madsen and Granum (2000)). The agents are animals as well as a humanoid figure playing the role of a helper on the farm. We are developing an interface where the user can talk to animals and helper, and combine speech with (a limited number of) deictic, iconic and turn-taking gestures (see Cassell and Prevost (1996) and Cassell (2000)). The modality devices used by the system are a speech recogniser and a data glove. The multimodal input is sent to a parser that attempts to build a combined representation of the two inputs. The meaning representation thus obtained is then sent to the virtual world that decides how to react to it. Later on, a repair phase that further refines or corrects the parse results, and a communication manager capable of dealing with a simple dialogue model, will be added.

In this paper, first a few previous approaches to multimodal integration are discussed in Section 2; then the use of multimodality in Staging is briefly described in Section 3; in Section 4 our approach to modality integration is dealt with in some detail; and finally in Section 5 a conclusion is drawn.

2 Discussion of previous approaches

Many sources argue that users generally prefer multimodal systems, and indicate a preference for one or the other modality (see e.g. Chapelier et al. (1994), Rudnicky (1993), and Oviatt (1999)). More recently, it has also been argued that at least for certain tasks (map-based tasks) multimodal interfaces are more efficient than

Proceedings of the Workshop on Integrating Information from Different Channels in Multi-Media-Contexts at ESSLLI MM
K. Holmqvist, P. Kühnlein & H. Rieser (eds).

patrizia@cst.ku.dk, *bart@cst.ku.dk*
Center for Sprogteknologi (CST)
Copenhagen, DK.

graphical user interfaces, even considering how quickly the user can recover from an error situation (Cohen et al. (2000)). However, users' preferences are conditioned by many factors, e.g. the layout of the experiment, the task, response time, previous computer experience, the modalities available and so on. Therefore, we have chosen to focus on a methodology which allows us to combine multimodal inputs irrespective of the particular modalities generating the information, and which may be reused across different applications. We have chosen to parse multimodal inputs in an integrated fashion, thus taking advantage of unification to treat redundant, complementary as well as ambiguous and even contradictory inputs.

Our approach is thus different from frame-based multimodal systems like the one presented in Vo and Wood (1996), where input from different modalities is integrated by adding information in semantic frames tailored to the domain. The use of a multimodal grammar, and the fact that unification is the only operation used to merge inputs coming from different devices, makes it more similar to the framework adopted in Johnston et al. (1997). Johnston *et al.* describe a multimodal (pen/voice) interface for map-based tasks where the integration of spoken and gestural information is driven by typed-feature structures and carried out via unification. In their system, multimodal integration takes place after speech and gestures have been independently parsed into separate, partial typed-feature structures. A multimodal integration agent then merges the two representation fragments into a unified whole.

Our approach resembles theirs in the basic use of unification as the merging operation. However, at the moment we are experimenting with a strategy where speech and gestural inputs are parsed together.

3 Multimodality in Staging

The Staging software consists of several components running on different machines with different operating systems. Some of the components are commercial software, other components are made by Staging collaborators, often reusing existing software. To handle communication between these components we use CORBA (OmniORB 2.8.0) and to a lesser degree, Microsoft COM.

User input to the Staging platform is multimodal. Here, we will consider input from a microphone and from a data glove. The speech input is sent to an off-the-shelf speech recogniser. Currently, we are using Dragon NaturallySpeaking Professional, which provides word-based, continuous speech analysis trainable to different speakers' voices and accents. Dragon supports the Microsoft Speech Application Programming Interface (SAPI), which can be exploited to integrate the program with other components. The gestural input is sent to a custom-made component still under development, which will be able to recognise at least the following types of gesture:

- Pointing gestures (deictic)
- Gestures indicating the relative size of objects (iconic)
- Gestures indicating a change of turn-taking

Interestingly, these three gesture types interact with speech in different ways, as shown by the following domain-relevant examples:

(1) “Please, feed the animals/the pigs/those/them”
(pointing at a group of pigs)
(2) “Move the smaller cow”
(with a gesturing meaning “small”)
(3) “No, I meant the other animal”
(moving the hand to attract the agent’s attention)

In example (1), the pointing gesture is associated either with a full nominal phrase or a pronominal place-holder. It adds referential information which must be compatible with the semantics of the rest of the speech act. However, considering the fact that some degradation of information is inevitable during the first phase of multimodal processing, ensuring such compatibility may not be trivial. In fact, the data glove component may compile a list of objects that are all in the direction in which the user has pointed. The same uncertainty may hold for the speech recognition software. In the Staging project, we have chosen to let the speech recogniser output only one interpretation of the speech input, but for what follows it is helpful to think of the speech recogniser’s output as a set of alternatives, the best of which has to be picked out in one way or another.

In example (2), the gesture adds a different kind of semantic information, which can usefully be represented as a feature contributing to the semantics of the noun “cow”. Again, situations may be envisaged in which the system is faced with contradictory or ambiguous input. Furthermore, the gesture may add content which is not overtly expressed in the utterance.

As we shall see, in the approach adopted in this paper these two types of modality interaction are dealt with at the level of parsing. Example (3), however, is different, in that it does not directly contribute meaning to the utterance. It is rather to be interpreted as a metaexpression at the level of dialogue managing. Therefore, the gesture must be interpreted as an independent speech act which sent to the virtual environment by the communication manager, will cause the agent to listen to the user’s next utterance.

4 Modality integration and parsing

4.1 When to integrate?

In the Staging architecture, several possibilities for doing modality integration can be considered. Figure 1.1 shows that gestural information could potentially be integrated at all stages of speech processing, and also that speech processing could drive gesture recognition. Clearly, a system allowing for all the possibilities shown would be very complex and difficult to control. So we want to discard at least some of them.

The earliest stage in which speech and gestural input may be merged is prior to parsing. For example, the system might decide that the user said *horse* and not *house* based on the fact that one of the candidate objects being pointed at is a horse, while a house is not a candidate. However, this kind of early integration runs the risk of being tuned to specific modalities, and lies outside our focus of interest.

To take a radically different view, integration may be delayed until after parsing, and be performed by the communication manager. It was mentioned earlier that it is the job of the communication manager to react to turn-taking gestures, but the possibility exists of also integrating pointing gestures at this stage. Communication management is the first process in the pipeline that has a working memory of speech acts. Therefore, a strategy could be that of looking for a deictic reference expressed by a gesture only if preceding dialogue did not provide a reference for some object mentioned by the user. For example if the user said: *Give the cow an apple*, the

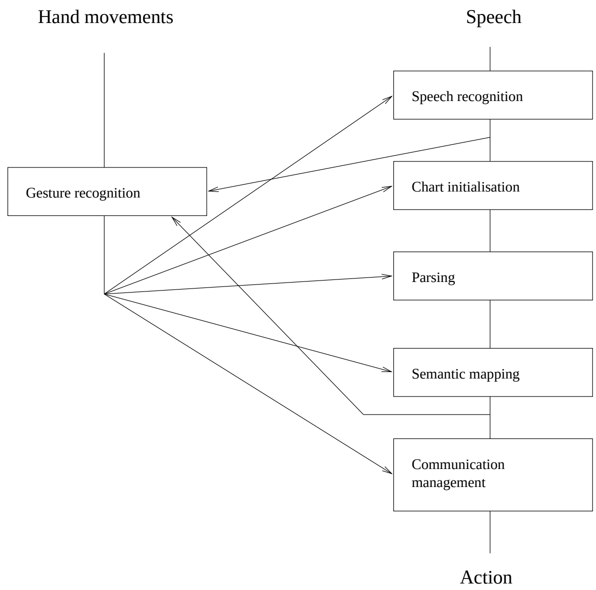

Figure 1.1: Integration possibilities

communication manager would first try to find a reference to a cow in the dialogue. If that failed, the communication manager would try to match *the cow* with a deictic movement. The fact that integration of gestures would become speech-driven can be seen as an advantage: a (possibly costly) attempt at integration is only undertaken if there is a clear need for it. On the other hand, speech analysis may not always be (and we know that it often is not) correct. Furthermore, this approach assumes that dialogue history has precedence over hand movements. This may be wrong: it can be argued that deictic reference is stronger than anaphoric reference.

A solution which is more suitable for our goal of generality, and which allows us to combine modalities without having one of them drive the process, is that of accounting for information coming from both channels during parsing. This can be achieved in several ways, e.g. by reading the gestural information into the initial chart built by the parser over the speech input, and writing grammar rules which attempt to unify the two information streams; or alternatively by using up gestural information during rule application, on the basis of features stated in the grammar rules. The solution we are experimenting with at the moment relies on integration during chart initialisation, and on the use of specific gestural features in both lexicon and grammar rules.

Figure 1.2 shows the integration choices we have made.

4.2 Integration during parsing

The parser used in the project is an adaptation of Njal (Music and Povlsen (1993)), a bottom-up left-corner parser, which is being adapted to receive input from several modalities. Njal is a good candidate for the Staging prototype since it is fast, and is able to produce partial parses in case the input is ill-formed or outside the coverage of the grammar. Another adaptation of Njal has in fact also been used with success in a multimodal application (Brøndsted et al. (1998)).

To build a representation of the input, Njal applies lexicon and grammar rules expressed in an augmented context-free formalism, where category symbols such as S (sentence), NP (nominal phrase) etc. are associated with other syntactic and

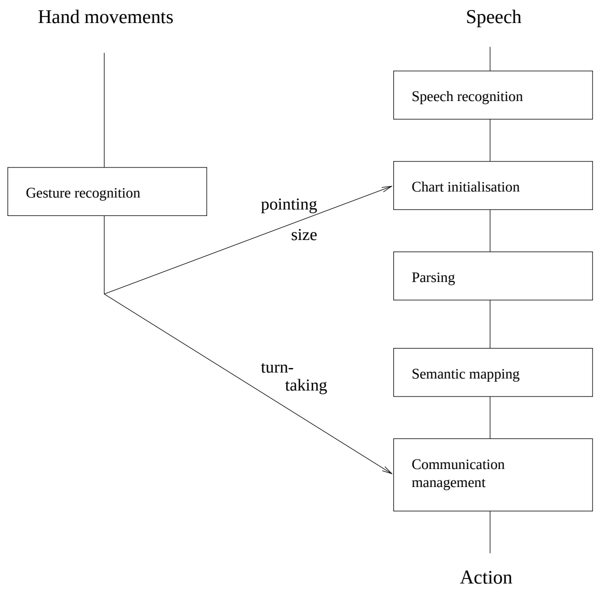

Figure 1.2: Integration choices

semantic features. The result of parsing is a parse tree associated with a feature structure containing the semantics of the sentence or phrase analysed. Before discussing the way in which gestural input is accommodated in the analysis, it is also useful to note that the parser applies two different types of rule, grammar rules to build the syntactic parse trees, and so-called semantic mapping rules to build the semantic representation.

For Njal to be able to process speech and gestures together, the speech signal and the signal from the data glove are time-stamped and synchronised. We are still considering which strategy to adopt for pairing gestures with the relevant words. We know from the literature that gestures often start before the corresponding speech segment is spoken. But it is also to be considered whether a single gesture should always be paired with a single word or with a sequence of words. Currently, we assume that a gesture is paired with the noun which either directly overlaps with it, or follows it with a maximum of some specified interval.

Two different types of gesture can be handled by Njal at the moment. Pointing gestures are represented as a string the final part of which is a unique identifier, e.g. **cow$1**. When read into the chart, such an input will be stripped of the identifier to yield what we call the object type, in this case **cow**. If the object type matches exactly the word it is paired with, then all the readings of the word, after being read from the lexicon, will be enriched with an *objectid* feature, the value of which is the full pointing gesture string including the identifier. Furthermore, a feature *resolvednoun=yes* will be added. The input will then be parsed in the usual way. Semantic mapping rules will add the reference information in the semantic representation. For example, the final semantic representation of the utterance *Move the cow* paired with the gesture represented as **cow$1**, will yield the following interpretation:

```
(4) {act=request,
     predicate=move,
     arg2={reln=cow,
           semtype=animal,
           objectid=cow$1,
```

```
                resolvednoun=yes}}
```

To sum up, identity of semantic types between word and corresponding gesture is treated in terms of a simple identity of lexicon lookup keys, and the information contributed by the gesture is added as a feature.

If object type and word do not match, on the other hand, the object type will be used to retrieve an alternative lexical entry. A relevant example here is the combination *Feed it* plus the gesture **pig$4**. In this case, the gesture complements the speech input by providing not only a reference, but also the semantic type of the object. The parser has two alternative word readings in the same chart position, one corresponding to *it* and the other to *pig*. After having applied syntax rules and semantic mapping rules, two analyses will be produced:

```
(5.a) {act=request,
       predicate=feed,
       arg2={objectid=unresolved}}

(5.b) {act=request,
       predicate=feed,
       arg2={reln=pig,
             semtype=animal,
             objectid=pig$4}}
```

Note that in this case, the feature *resolvednoun=yes* is not assigned, as the gesture is not "replacing" a noun, but a pronoun. It is important for the analysis containing the unresolved reference to survive. Although in this specific example it can be ignored in further processing, there will be cases in which the data glove component either provides a reference that doesn't fit in with the semantics of the utterance, or no reference at all. The communication manager must then try to resolve the anaphora by looking at the dialogue history.

Now let us consider a simple case of contradictory information, which can also be treated quite elegantly by following the same strategy. The input is taken to consist of the utterance *Feed the house* (this can be due to an error in speech analysis) paired with the pointing gesture **cow$1**. The parser has again two alternative lexicon entries to parse – one corresponding to the *house* reading, and the other to the *cow* reading. Further analysis, however, will reject *house* as the object of the verb *feed* since it is not of the required type "animal".

The same method applies to examples where either the speech signal or the gestural input are ambiguous, e.g. a case where the utterance *Feed that* is paired with a list of alternative gestures containing reference to an animal and a different kind of object. Again, the gestural interpretation which fits the selectional restrictions of the predicate would be selected directly via unification. However, semantic restrictions will not always be fine-grained enough to be able to disambiguate fully. Therefore, strategies for the virtual environment to cope with ambiguous interpretations will have to be devised.

The other type of gesture which can input information to the chart is exemplified by gestures indicating size. In this case, however, the gesture is not used to look up words in the dictionary, but to add a value to a feature *gesture* in the lexical readings of the noun it is paired with. It is then up to the grammar, or to the virtual world, to find out whether the characterisation added by the gesture can be made sense of. For example, the utterance *Feed the cow* paired with the gesture **small** will yield the analysis shown below. Note that the size specification has become the value of a *mod* feature associated with the object (*arg2*).

```
(6) {act=request,
     predicate=feed,
     arg2={reln=cow,
            semtype=animal,
            objectid=cow$1,
            mod={reln=small}}}
```

4.3 Multimodal constraints in grammar rules

As shown in the preceding section, in many cases in which there is an ambiguity either in the speech or the gesture stream or both, the parser will not be able to decide immediately which reading of a noun to select, or which syntactic analysis to assign to the input. Modality integration constraints can be stated in the grammar to solve at least some of these situations.

A relevant example here is the analysis of the utterance *Feed that cow* paired with the reference **cow$2**. The sentence can be assigned two different structures:

(7) (Feed (that) (cow))
(8) (Feed (that cow))

Note that in both cases, the noun *cow* will have been assigned the feature *resolvednoun=yes*. The former analysis (analogous to what one would have in *Give the animals water*, however, can be avoided by making use of gestural information in the grammar[1]. This can be done easily by specifying in the grammar rule responsible for the analysis in (7) that for a single noun to be analysed as an NP, it cannot bear the feature *resolvednoun=yes*:

```
(9)  np_nodet = {cat=np} [ {cat=n, resolvednoun=no} ].
```

The rule will thus not build an NP interpretation for single nouns such as *cow* when they are paired with a pointing gesture. However, it can still build NPs consisting of nouns which are not associated with gestures, or gestures not associated with nouns as in example (5b).

5 Conclusion and future work

We have presented a methodology for the treatment of multimodal input (speech and hand movements) where the inputs are parsed together and represented in a uniform way in a feature-based formalism. We have shown how the use of unification as the fundamental merging operation allows us to accommodate complementary, but also ambiguous and contradictory inputs by relying on the semantic and syntactic constraints expressed in the lexicon and the grammar.

We believe this approach is an interesting and fruitful one in its simplicity. However, a lot still has to be accomplished to have a smoothly running system. Firstly, the data glove component must be fully developed, and the algorithm for synchronising speech and gesture streams must be tested and refined. Secondly, we know that the representations built by the parser are not always complete or unique. More research is needed to specify in what way the subsequent repair phase can reduce remaining ambiguities, as well as remedy failures of the rule-based modality integration or the parsing proper. An interesting approach is that described in Rosé and Lavie (1997), where parse fragments are combined on the basis of the "fitness" of the resulting representation. In our case, fitness would be determined by the

[1] A different approach could be that of exploiting the distinction between countable and uncountable nouns to rule out (7).

number of operations in the repair hypothesis, and the number of semantic features instantiated in the resulting representation.

Finally, the interface between multimodal analysis (parsing and communication managing) and the virtual environment is object of intensive study. In particular, the CORBA interfaces, which are rather unwieldy at the level of programming languages such as C++, are specified at a much more abstract level using IDL (interface definition language). We intend to look at the IDL's potential for expressing the semantic domain accessed by the virtual world component, the data glove software, the multimodal parser and the communication manager.

Bibliography

Brøndsted, T., Larsen, L. B., Manthey, M., Kevitt, P. M., Moeslund, T., and Olsen, K.: 1998, The Intellimedia WorkBench – an environment for building multimodal systems, in *Proceedings of the second international Conference on Cooperative Multimodal Communication (CMC '98)*, pp 166–170, Tilburg, The Netherlands

Cassell, J.: 2000, Embodied conversation: Integrating face and gesture into automatic spoken dialogue systems, in S. Luperfoy (ed.), *Spoken Dialogue Systems*, Cambridge MA: MIT Press, (to appear)

Cassell, J. and Prevost, S.: 1996, Distribution of semantic features across speech & gesture by humans and machines, in *Proceedings of the Integration of Gesture in Language and Speech*

Chapelier, L., Fay-Varnier, C., Roussanaly, A., and Saint-Dizier, V.: 1994, Recueil et analyse d'un corpus d'interactions multimodales homme-machine, in R. Patesson (ed.), *Proceedings of ERGO-IA*, pp 128–39, IDLS, - Biarritz, - CRIN 94-R-131

Cohen, P., MacGee, D., and Clow, J.: 2000, The efficiency of multimodal interaction for a map-based task, in *Proceedings of the 6th Applied Natural Language Conference and the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pp 331–337, Seattle, Washington, USA

Johnston, M., Cohen, P. R., McGee, D., Oviatt, S. L., Pittman, J. A., and Smith, I.: 1997, Unification-based multimodal integration, in *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, Madrid, Spain, ftp://cse.ogi.edu/pub/chcc/Publications/acl97.ps

Madsen, C. B. and Granum, E.: 2000, Aspects of interactive autonomy and perception, in L. Q. et al. (ed.), *Inhabited 3D Virtual Spaces*, Chapt. 3, p. (32), Springer Verlag, In press

Music, B. and Povlsen, C.: 1993, The NLP module of a spoken language dialogue system for Danish flight reservations, in *Proceedings of the 3rd European Conference on Speech Communication and Technology*, pp 1859–62

Oviatt, S. L.: 1999, Mutual disambiguation of recognition errors in a multimodal architecture, in *Proceedings of the Conference on Human Factors in Computing Systems*, pp 576–583, ACM Press: New York

Rosé, C. P. and Lavie, A.: 1997, An efficient distribution of labor in a two stage robust interpretation process, in *Proceedings of EMNLP*, [http://xxx.lanl.gov/abs/cmp-lg/9706021 accessed 980623]

Rudnicky, A. I.: 1993, Factors affecting choice of speech over keyboard and mouse in a simple data-retrieval task, in *Proceedings of Eurospeech*, pp 2161–4

Vo, M. T. and Wood, C.: 1996, Building an application framework for speech and pen input integration in multimodal learning interfaces, in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, Atlanta, GA