Aerospace Science and Technology 13 (2009) 423–430

Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

Increasing robustness, reliability and ergonomics in speech interfaces for aerial control systems

Javier Ferreiros [a,*], Rubén San-Segundo [a], Roberto Barra [a], Víctor Pérez [b]

[a] Departamento de Ingeniería Electrónica, ETSI Telecomunicación, Universidad Politécnica de Madrid, Spain
[b] Boeing Research and Technology Europe, Spain

ARTICLE INFO

Article history:
Received 24 January 2008
Received in revised form 2 June 2009
Accepted 9 June 2009
Available online 13 June 2009

Keywords:
Aerial control systems
Aerial vehicles
Spontaneous speech
Robustness
Reliability and ergonomics of aerial control systems
Speech human–computer interfaces

ABSTRACT

This paper proposes several speech technology improvements for increasing robustness, reliability and ergonomics in speech interfaces for controlling aerial vehicles. These improvements consist of including a statistical language model for increasing the robustness against spontaneous speech, incorporating confidence measures for evaluating the performance of on-line the speech engines (better reliability), and a flexible response generation for improving the interface ergonomics. This paper includes a detailed description of the speech control interface developed as a result of the collaboration between the GTH (Grupo de Tecnología del Habla or Speech Technology Group) at Universidad Politécnica de Madrid (UPM) and the company Boeing Research and Technology Europe under the contract No. 206/05. This interface includes modules that perform speech recognition, natural language understanding and response generation via a speech synthesizer. In the system evaluation, the final results reported a 96.4% Word Accuracy and a 92.2% Semantic Concept Accuracy. This paper also provides a state-of-art review of using Speech Technology for controlling aerial vehicles, comparing the main initiatives carried out. A significant conclusion of this work is that Speech Technology is now ready enough to be considered as a new modality (in parallel with traditional ones) for introducing high level commands while the controller is carrying out others actions when interacting with these control systems. In critical applications (such as this) the best performance of this technology is achieved when all the configuration possibilities of the speech engines are accessible and the speech interface is designed in collaboration with Speech Technology experts.

1. Introduction

Speech Technology has now reached a high level of performance, making it applicable in many critical systems. Relevant improvements have been achieved on the basis of the greatly increased effort into research carried out by aeronautical companies and regulatory institutions such as: Eurocontrol and AENA in collaboration with Speech Technology expert groups. As a result of this effort, large speech and text databases have been generated and new speech and text processing models have been developed and adapted to the specific field requirements. These technological advances have been supported by the significant increase in speed obtained both by the hardware that executes these algorithms and the improvements within the algorithms themselves that exhibits properties of intelligent search for the best solution. An important area of critical applications supported by these new capabilities is aerial traffic control. In 1993 a big project was developed at LIMSI (CNRS – France) for training air traffic controllers in their tasks by using speech recognition and synthesis, creating a so-called automatic "pseudo-pilot" [17,18]. From 1993 on, there have been significant efforts in integrating this technology into air traffic control applications [4,12,13,20,22]. In Spain, a special mention must be made to the INVOCA project (Vocal Interfaces for Air Traffic Control) as a cooperation of AENA (Spanish Airports and Air Navigation) [2,8] with the Grupo de Tecnología del Habla at the Universidad Politécnica de Madrid (hereinafter referred as GTH-UPM).

Another important area has been the use of speech technology for controlling aerial vehicles. In 2001, the first results of the WITAS project were presented [16] for controlling a UAV (Unmanned Aerial Vehicle). During the next 4 years, the WITAS evolution produced important results on new interfaces for aerial vehicle control [10,14,15,21]. This project is a very good reference, also including the integration of speech technology into the aerial control interface. Other research efforts into integrating speech recognition into aerial control systems are [1,5,19].

* Corresponding author at: Grupo de Tecnología del Habla, Dpto. Ingeniería Electrónica, ETSI Telecomunicación, Ciudad Universitaria s/n, 28040 Madrid, Spain. Tel.: +34 914533542; fax: +34 913367323.
E-mail address: jfl@die.upm.es (J. Ferreiros).

1270-9638/$ – see front matter
doi:10.1016/j.ast.2009.06.002

The paper provides a detailed state-of-art review and proposes several speech technology improvements for tackling reliability and ergonomic issues with robustness; in speech interfaces for controlling aerial vehicles in general (the experiments have been carried out considering a UAV but the conclusions are general for controlling any aerial vehicle). The target is not to propose speech as an alternative modality for controlling all the possibilities of an aerial vehicle, the main objective is to introduce speech as a new modality (in parallel with traditional ones: joysticks, keyboards, etc.) for allowing the controller in parallel to introduce high level commands and to look at the screen or use a joystick, for example. Under these circumstances, the requirements for the speech interface are:

- To have a performance better than 90% for command recognition.
- To provide a confidence measure for every recognized command.
- To work in real time.
- To provide a flexible response with several levels of verbosity.

The structure of this paper is as follows: Section 2 includes the state-of-art review of speech technology as applied to aerial control systems. Section 3 describes an overview of the system developed by the GTH-UPM. The speech recognition, natural language understanding and response generation modules, including the new proposals for increasing robustness, reliability and ergonomics, are described in Sections 4, 5 and 6 respectively. Section 7 presents the system evaluation results. Finally, Section 8 summarizes and compares the different initiatives highlighting the main conclusions of this work.

2. State of art in speech technology for aerial control systems

When developing the interface for an aerial control system, it is necessary to include the following Speech Technology modules: a Speech Recognizer for converting natural speech into a sequence of words (text), a Natural Language Understanding module that extracts the main semantic concepts from the text (the commands to be carried out and their corresponding data for the aerial control system), and a Response Generation Module for creating a natural response to the user that will be converted into speech by a speech synthesizer. This section is structured into these modules.

2.1. Speech recognition

All speech recognizers developed so far are based on two sources of knowledge: the characterization of phone acoustics, and language structure. Related to the acoustic modeling, all current speech recognition systems are based on Hidden Markov Models (HMMs). These models are very common in several recognition problems [6,7]. For each allophone (a characteristic pronunciation of a phoneme), one HMM model is calculated as a result of a training process carried out using a speech database. A speech database consists of several hours of transcribed speech (made up of files with speech and text combined, where is possible to correlate the speech signal to the words pronounced by the person). There is a very important link between the size of the database used to train the HMMs, and the versatility and robustness of the speech recognizer. Database acquisition is a very costly process because it requires linguistics experts to transcribe the speech pronounced by different speakers by hand. Because of this, only important companies such as IBM, Microsoft, Dragon Systems, Nuance, Telefónica or important research centers such as MIT (Massachusetts Institute of Technology), CMU (Carnegie Mellon University), CU (Cambridge University), GTH-UPM (Universidad Politécnica de Madrid) with wide-ranging experience in this technology can offer speech recognition systems with the highest guarantee of having enough robustness and flexibility to be incorporated into a critical application such as aerial control. In the speech community, there are two main associations that sell valuable speech databases for research and development: they are LDC (Linguistic Data Consortium: http://www.ldc.upenn.edu/) and ELRA (European Language Resources Association: http://www.elra.info/). In [11], there is a very good review of the state of art focusing on acoustic modeling for speech recognition.

The second source of knowledge included in a speech recognizer is the language modeling. This model complements the acoustic knowledge with the information on the most probable word sequences. There are several techniques for language modeling: grammar-based language modeling and statistical language modeling (N-gram). The first type consists of defining all the possible sentences that the system can recognize. Any other word sequence, not foreseen in these sentences, is rejected. This model is easier to generate by a non-expert but it is very strict and does not deal well with the spontaneous or stressed speech found in live utterances.

Statistical language modeling consists of computing the probability of one word, given the $N-1$ previous words. For example, a 3-gram model consists of the probabilities of every word preceded by any combination of two words. The statistical model is generated automatically from an application oriented text (set of sentences), considering a smoothing process for unseen sequences. This smoothing allows all word sequences to be permitted to some extent (there are no forbidden word sequences), playing the roll of a fundamental robustness factor. This fact is very important when modeling spontaneous speech: word repetitions, doubts, etc.

So far, all speech recognition systems incorporated in aerial control systems are commercial programs: Microsoft [19] or Nuance [5,10,14–16,21] Speech Recognizers. These recognizers are integrated by the aerial control interface developer, typically an expert in aerial control task assignment but not necessarily a speech technology expert. Although the speech recognition systems (especially the commercial ones) are evolving to more robust and user-friendly software engines, there are still significant limitations in their configuration that drastically affect the performance of the speech recognizer. One important aspect is the language modeling: the commercial recognition engines offer the possibility to define grammar-based models (easy to define by a non-expert) by this configuration but they are not flexible enough for spontaneous or stressed speech as could easily appear in these control interfaces.

The performance of a speech interface not only depends on the speech recognizer, but the language understanding module must also be considered (as will be shown in the evaluation section). When using a commercial speech recognizer, the limitations in the flexibility can affect the correct integration between both modules and impair the possibility of using other sources of information which could make the recognition vocabularies and/or language models variable depending on the system state.

2.2. Spoken language understanding

This process consists of extracting the semantic information or "*meaning*" (within the specific application domain) from the speech recognizer output (sequence of words). The semantic information is represented by means of a frame containing a number of semantic concepts. A semantic concept consists of an identifier or attribute, and a value. For example: one concept could be WAYPOINT_CODE while the value is "A01". The natural language understanding is mainly carried out using rule-based techniques: the relationships between semantic concepts and sequences of words or other concepts are defined by hand by an expert. The rule-

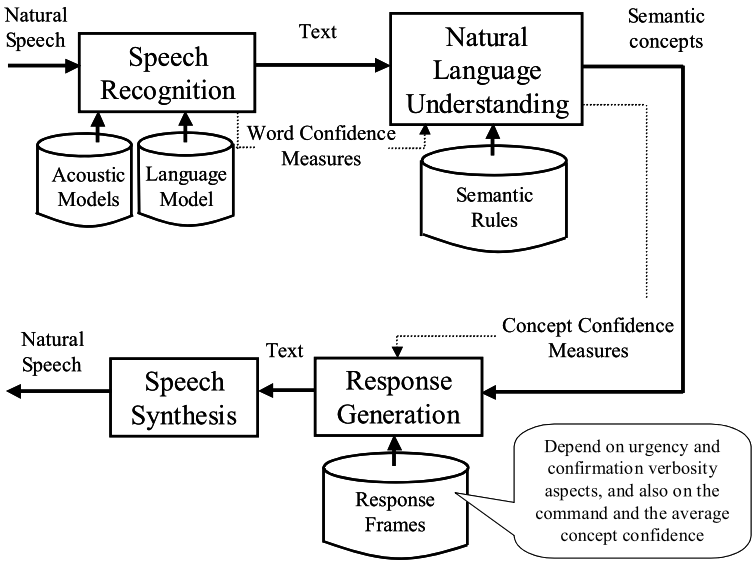

Fig. 1. System module diagram.

based techniques can be classified into two types: top–down and bottom–up strategies:

- *Top–down strategy*. In this case, the rules try to obtain the semantic concepts from an overall analysis of the whole sentence, aiming to find a unique axiom for it. This strategy tries to match all the words in the sentence, to a sequence of semantic concepts. Considering that a speech recognizer can produce errors in the word sequence, this technique is not flexible and robust enough to deal with these errors: one error in the word sequence causes the semantic analysis to fail. In the speech interfaces for aerial control reviewed in the literature (except the WITAS project), all the natural language understanding are rule-based techniques with a top–down strategy.
- *Bottom–up strategy*. In this case, the semantic analysis is carried out by starting from each word individually and extending the analysis to neighboring context words or other already built conceptual islands. This extension is made to find specific combinations of words and/or concepts (blocks) that generate a higher level semantic concept. The rules implemented by the expert define these relations. This strategy is more robust against speech recognition errors and is necessary when a statistical language model is used in the recognizer. Depending on the scope of the block relations defined by the rules, it is possible to achieve different compromises between the reliability of the concept extracted (higher with higher lengths) and the robustness against recognition errors (higher with smaller lengths). The WITAS project uses the SRI GEMINI parser [23]. It is a natural language processing engine that applies a set of syntactic and semantic grammar rules to a word string using a bottom–up parser to generate a logical form. A logical form is a structured representation of the context-independent meaning of the string.

This paper proposes a new natural language understanding module that generates semantic confidence measures, targeting the goal of increasing the reliability and robustness of the speech interface.

2.3. Natural response generation

The response generation module translates the understood concepts into a natural language sentence used to confirm the information back to the user. These sentences can be fixed or built using templates with some variable fields. These fields are filled in with the information obtained from the semantic interpretation of the previous sentence. In the literature, both kinds of response generation modules appear: fixed sentences [19] or template-based modules [21]. Finally, the natural language sentence is converted into speech by means of a text to speech conversion system that ends up with a speech synthesizer.

A more flexible template-based response module is presented in this paper. This flexibility is modulated by the urgency and confirmation verbosity desired by the user, significantly increasing the interface ergonomics, by adapting its behavior to the relevant external conditions of the system.

3. Overview of the control interface developed by GTH-UPM

Fig. 1 shows the module diagram of the interface developed by the GTH-UPM and the company Boeing Research and Technology Europe for aerial control systems. The main modules are as follows:

- The first module, the speech recognizer, converts natural speech into a sequence of words (text). One important characteristic of this module is the statistical language model that has been trained for increasing the robustness against spontaneous speech. Another relevant characteristic is the confidence estimation: every recognized word is tagged with a confidence value representing the belief of the recognizer on the correctness of its own work: a value between 0.0 (lowest confidence) and 1.0 (highest confidence). In critical applications, the confidence measures report the reliability of the word sequence obtained from the recognizer.
- The Natural Language Understanding module extracts the main semantic concepts (commands and their corresponding data in our application) from the text, using semantic rules defined by an expert. This module also generates an estimation of confidence for every semantic concept extracted. In the literature review, there is no example of a language understanding module with this characteristic applied to speech interfaces for aerial vehicle control.
- The third module is the Response Generation Module. In this implementation, it uses several response templates to create a natural language sentence as confirmation for the under-

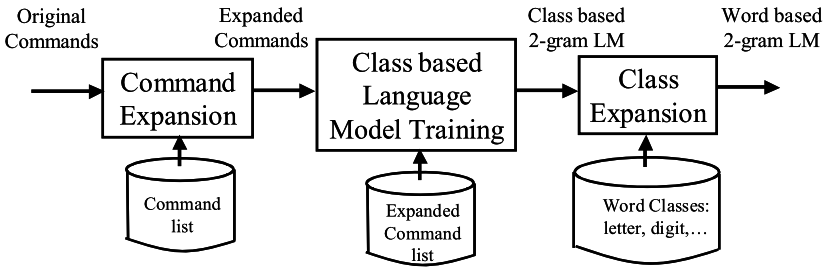

Fig. 2. Language model generation process.

stood command. The solution proposed in this paper is more flexible and powerful than any proposed before: the response templates are variable and depend on the semantic concept confidences, on the control urgency status and on the confirmation verbosity desired by the controller.

4. Speech recognition

The speech recognizer used in this prototype is a state of art speech recognition system developed at GTH-UPM. It is an HMMs-based system (Hidden Markov Models) with the following main characteristics:

- It is a Continuous Speech recognition system: it recognizes utterances made up of several continuously spoken words. In this application, the vocabulary size is 93 Spanish words.
- Speaker independence: the recognizer has been trained using a large database, making it robust against a great range of potential speakers without further training by actual users.
- The recognition system can generate one optimal word sequence (given the acoustic and language models), a solution expressed as a direct acyclic graph of words that may compile different alternatives, or even the *N*-best word sequences sorted by similarity to the spoken utterance.
- The recognizer provides one confidence measure for each word recognized in the word sequence. The confidence measure is a value between 0.0 (lowest confidence) and 1.0 (highest confidence) [9]. This measure is important because the performance of the speech recognizer can vary depending on several aspects: level of noise in the environment, non-native speakers, more or less spontaneous speech, or the acoustic similarity between the different words contained in the vocabulary. Nowadays, the commercial recognition engines do not provide this characteristic because it is difficult to manage when designing a speech interface for critical applications.

4.1. Acoustic modeling

The speech recognizer uses 5200 triphone HMMs for modeling all possible allophones and their context. The system also has 16 silence and noise HMMs for detecting acoustic effects (non-speech events like background noise, speaker artifacts, filled pauses, etc.) that appear in spontaneous speech. It is important to detect and process them in order to avoid these noises affecting the recognition performance.

In our case, the recognition system uses continuous HMMs: this means that the pdfs (probability density functions) used in every state of each model are continuous functions (multi-Gaussian). This modeling has been shown to be the most powerful strategy for implementing HMMs. The acoustic HMMs have been trained with a very large database, containing more than 20 hours of speech from 4000 speakers. The size of the database and the variability of the speakers provide the acoustic models with a significant recognition power and robustness.

4.2. Language modeling

The second source of knowledge included in a speech recognizer is the language model. This model complements the acoustic knowledge with the information on the most probable sequences of words. In this system, the recognition module includes a statistical language modeling: 2-gram. This type of model computes the probabilities of every word preceded by one word. As was commented on in previous sections, this kind of language modeling has the best robustness when modeling spontaneous speech (word repetitions, doubts, etc.), because it does not prohibit any word sequence. On the other hand, it needs a more complicated configuration of the automatic tools for language model generation, requiring expert intervention. This is why commercial recognizers offer limited options for adapting and smoothing the language model and only strict models are adopted by them.

Fig. 2 shows the process carried out to train a word-based 2-gram language model from the original command description provided by the control experts. This process consists of 3 steps:

- In the command expansion, every command description is appropriately replicated considering its defined structure. Some examples of expansion are as follows:
  - Optional parts: the command is expanded by considering all possible structures. For example, ASCEND [AND HOLD] {$HEIGHT} is expanded in two: “ASCEND {$HEIGHT}” and “ASCEND AND HOLD {$HEIGHT}” (the words between catches are optional).
  - List of elements: when a list of possible values is defined, copies of the same command are generated by choosing one value for each instance. For example, “(SHORT|MEDIUM|LARGE) RADIUS”, three examples with different values are generated (the list elements are expressed between parenthesis and separated by vertical lines).
  - Macro expansion: every macro is expanded by reproducing its structure. For example, {$HEIGHT} can be expanded to several structures: “{$DIGIT}{$DIGIT}{$DIGIT}{$DIGIT} FEET”, “FLIGHT LEVEL {$DIGIT}{$DIGIT}{$DIGIT}”, where {$DIGIT} is another macro containing the words for the basic numbers from “ZERO” TO “NINE”. Another example may be {$WAYPOINT_CODE} that could be expanded to “{$LETTER} {$DIGIT} {$DIGIT}”.

  This command expansion has a significant limitation. There are several cases where it is not possible to expand all the possible values (letters or digits). For example, if we wanted to expand all possible values for a waypoint (considering it would be made up of latitude *digit digit* degrees *digit digit* minutes *digit digit* seconds plus longitude *digit digit digit* degrees *digit digit* minutes *digit digit* seconds) considering all possible “digit” values, there would be $10 \times 10 \times 10 \times 10 \times 10 \times 10 \times 2 \times 10 \times 10 \times 10 \times 10 \times 10 \times 10 = 2 \times 10^{11}$ possibilities. In order to avoid this situation, two word classes have been considered: letter and digit, training a class-based language model (LM).

- In the class-based LM training, a class 2-gram LM is generated by computing the probabilities of any word/class followed by any word/class, considering the command partially expanded in the previous step. In our case, there are two word classes: "letter" (with all the possible letters), and "digit" (with all possible digits). During this process a smoothing is applied for providing some probability to sequences of words/classes not seen in the expanded commands. This smoothing can be controlled and has been adjusted for this task.
- In the last process the class LM is converted into a word LM. This process is carried out by replacing the estimated probabilities for any class ("digit", for example) by the probabilities for the words belonging to this class ("cero, uno, dos,..."). The word probabilities are computed by considering the class probabilities (obtained in the previous step) and the total number of words belonging to this class. At the end of the process, the 2-gram word LM is saved as the one that can be directly used by the speech recognizer.

5. Natural language understanding

This process is responsible for the extraction of the semantic information or "meaning" (within the specific application domain) from the speech recognizer output (sequence of words). The semantic information is conveyed by a frame containing semantic concepts. A semantic concept consists of an identifier and a value. For example: the concept VELOCITY has VELOCITY as identifier/attribute while a possible value is "140 knots". In this system, we have identified 33 main concepts: 22 commands and the corresponding data associated to them. Internally, the system manages other intermediate concepts that carry the semantic information when it is developed from the input (exclusively made up of words) through intermediate representations with a mixture of words and concepts (both internal and main concepts).

The language understanding module has been implemented by using a rule-based technique considering a bottom–up strategy. In this case, the relationships between semantic concepts and word and/or concept sequences are defined by hand using an expert. In a bottom–up strategy, the semantic analysis is carried out by starting from each word individually and extending the analysis to neighboring context words or already-formed concepts. This extension is carried out to find specific combinations of words and/or concepts that generate another concept. Not all the words contribute (or with other wording, need to be present) to the formation of the final interpretation. The rules implemented by the expert define these relations. This strategy is more robust against speech recognition errors and is frequently preferred when a statistical language model is used in the recognizer. Depending on the scope of the word relationships defined by the rules, it is possible to achieve different compromises between the reliability of the concept extracted (greater with greater lengths) and the robustness against recognition errors (greater with shorter lengths).

The understanding process is carried out in two steps (Fig. 3). In the first one, every word is mapped onto one or several syntactic-pragmatic tags. For example: ZERO, ONE, TWO, etc. are assigned the "DIGIT" tag (the same as for ALPHA, BRAVO, CHARLIE,... mapped to "ALPHABET_ITEM" tag). An example of multiple tags is the words "FLIGHT PATTERN". They are tagged with the labels COMMAND13 (to establish the predefined flight pattern) and COMMAND14 (to establish a specific flight pattern). Later on through the understanding process and depending on the data detected, only one of these tags is selected.

The understanding module works by applying different rules that convert the tagged words into semantic concepts and values by means of grouping words (or concepts) and defining name concepts. In order to show the process let see an example: detecting MISSION_CODE, PATTERN_CODE and WAYPOINT_CODE. These three concepts have the same structure: letter–digit–digit. There is a rule that detects these patterns in the word sequence (cat_letra cat_digito cat_digito) and replaces them with an internal concept GENERAL_CODE with a code value developed through the concatenation of the blocks with the specified tags. This new GENERAL_CODE tag is used at this level where more information is necessary to determine fully the actual nature of this value. Let's look at an example in Fig. 4. The rule also converts the word "bravo" to the letter B and the same for "cero", "tres" words to more convenient forms. At the end of the process a GENERAL_CODE concept is renamed depending on the command detected using other rules. If the command is "mission activation", the GENERAL_CODE chunk is converted to a MISSION_CODE, and so on.

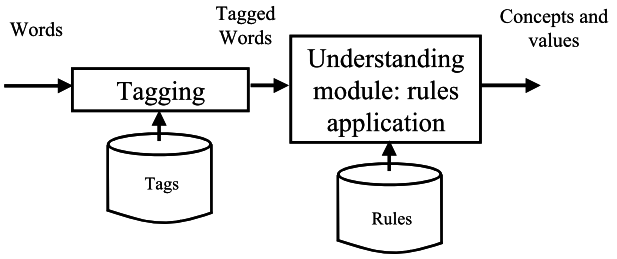

Fig. 3. Structure of the natural language understanding module.

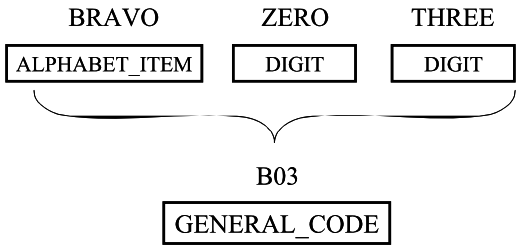

Fig. 4. Example the application rule.

5.1. Conceptual confidence estimation

The developed system generates one confidence value for every concept obtained: a value of between 0.0 (lowest confidence) and 1.0 (highest confidence) [9]. This confidence for the concepts is computed by an internal procedure that is coded within the proprietary language interpreter that carries out each rule. In this internal engine, there are "primitive functions", responsible for the execution of the rules written by the experts. Each primitive has its own way of generating the confidence for the elements it produces. One common case is for the primitives that check the existence of a sequence of semantic blocks to generate new ones, where the primitive usually assigns the average confidence of the blocks to the newly created blocks, upon which it has relied. For example in Fig. 4, the confidence measure of the concept GENERAL_CODE is the average of the word confidence values for "BRAVO", "ZERO" and "THREE". After that, MISSION_CODE will have the same confidence value as the GENERAL_CODE concept. In other more complex cases, the confidence for the new blocks may be dependent on a combination of confidences from a mixture of words and/or internal or final concepts.

In the literature review, there is no language understanding module (for aerial control interfaces) that generates a confidence measure for semantic concepts. In critical applications such as aerial control interfaces, it is very important to reach a high level of performance but it is also very useful to have a confidence

Table 1
Relationship between the different levels of urgency status and desired confirmation verbosity and the size of the response system currently in effect.

		Urgency		
		LOW	MEDIUM	HIGH
Confirmation Verbosity	LOW	**SO**	**VSO**	**VSO**
	MEDIUM	**LO**	**SO**	**VSO**
	HIGH	**LO**	**LO**	**SO**

Table 2
Sentence examples for the different response and understanding confidence levels.

Verbosity level	Confidence in the understanding GREATER than threshold	Confidence in the understanding LESS than threshold	
		Incorrect command data	Correct command data
LO	"Activando misión A01" (*activating mission A01*)	"Reconocido comando C2 Activa misión, pero existe incongruencia de datos" (*command C2, activate mission, understood but there is incoherence in the data*)	"Disculpe no le he entendido, vuelva a introducir un comando" (*I am sorry, I didn't understand you. Please say the command again*)
SO	"Misión A01" (*mission A01*)	"Comando incompleto, no lo ejecuto" (*Incomplete command, I won't carry it out*)	"Disculpe no le he entendido" (*I am sorry, I didn't understand you*)
VSO	"OK!"	"No entiendo!" (*I don't understand*)	"No entiendo!" (*I don't understand*)

measure providing information on the reliability of the semantic information obtained. These measures avoid carrying out control actions with the possibility of misunderstood information, increasing the reliability of the whole system.

6. Response generation and speech synthesis

In this implementation, the response generation module uses response templates to create a natural language sentence as confirmation of the understood command. The solution proposed in this paper is more flexible and powerful than previous ones. In this solution, the response templates are variable and depend on the confidence of the semantic concepts, on the urgency of the aerial system and on the verbosity of the confirmation. In this module, three kinds of response templates (three levels of verbosity) have been defined. These are as follows:

- LO (Long Output): the system generates the longest sentence including all the information understood.
- SO (Short Output): in this case, the sentence is shortened and part of the information is omitted: the larger and more tedious parts, those for which the speech interfaces are the least justified, such as full specified longitudes or latitudes, that could be better confirmed in a textual or graphical form.
- VSO (Very Short Output): in this case, the system only asserts the command understanding, without any specification on what it is actually understood.

The actual level of verbosity is modulated through the specification of two parameters: urgency system status and desired confirmation verbosity. A higher level of urgency implies less verbosity in itself, while a higher level of confirmation verbosity increases the size of the response. There are three levels of urgency status (high, medium and low) and three levels of confirmation verbosity (high, medium and low). Table 1 shows the mapping between their corresponding settings and the overall system verbosity.

Apart from the level of verbosity, the action carried out by the aerial control interface and the actual contents of the response will depend on the confidence in the understanding obtained for the current utterance (very important in aerial control), as compared to a confidence threshold. Additionally, when the confidence in the understanding is greater than the confidence threshold, the system provides a different output depending on the command structure (if the command contains the correct data for carrying it out or not). Table 2 shows examples of the output content for the three verbosity levels depending on both the confidence in the understanding and the completeness of the command data.

Eventually, the natural language sentence is converted into speech by means of a speech synthesizer. The speech synthesis module is a male voice text to speech system developed in by GTH-UPM (BORIS [3]). This module uses a diaphone unit concatenating algorithm, able to modify the speaking rate and speaker pitch. The speaking rate and the speaker pitch have been adjusted for every level of verbosity defined:

- LO: at this level, the default values were considered: 180 syllables/minute (speaking rate) and approx. 130 Hz (speaker pitch).
- SO: at this level, the speaking rate is increased to/by 10% and pitch is also increased to/by 10% to generate a faster and more dynamic voice.
- VSO: the speaking rate is augmented to/by 25% and the pitch is increased to/by 20% from the default values.

7. Speech interface evaluation

In this project, the system has been evaluated with 93 utterances containing 819 words and 214 semantic concepts. The speech recognition module has been evaluated by computing the percentages of correct words, inserted words (those that were not spoken but were written by the recognizer), deleted words (spoken but not recognized) and substituted words (those cases in which the system recognized a wrong word). From these percentages, it is possible to compute the Word Error Rate (WER) or the Word Accuracy (WA). The following Eqs. (1), (2), (3) define these metrics.

$$\text{Sub}\ (\%) = 100 \times \frac{N_S}{N_T} \qquad \text{Del}\ (\%) = 100 \times \frac{N_D}{N_T}$$
$$\text{Ins}\ (\%) = 100 \times \frac{N_I}{N_T} \tag{1}$$

$$\text{World Error Rate}\ (\%) = \text{Sub}\ (\%) + \text{Inser}\ (\%) + \text{Del}\ (\%) \tag{2}$$

$$\text{World Accuracy}\ (\%) = 100\% - \text{WER}\ (\%) \tag{3}$$

Table 3
Speech recognition module evaluation.

WA (%)	Sub (%)	Ins (%)	Del (%)
96.40	1.56	0.60	1.44

Table 4
Understanding module evaluation.

CA (%)	Sub (%)	Ins (%)	Del (%)
92.24	5.48	0.00	2.28

where N_S is the total number of substitutions, N_I insertions, N_D deletions, and N_T the total number of words in the correct target sentences (labeled reference). In order to compute the number of substitutions, insertions, deletions or correct words, the utterance recognized is compared to the reference using a dynamic programming algorithm, which considers equal costs for any kind of error. With this evaluation, the system presents a very high WA: 96.4% (Table 3). Another evaluation parameter considered is the percentage of perfectly recognized sentences: sentences in which all of the words were correct and obtaining 88.2% of correct sentences.

Regarding the processing time, the speech recognizer works in 0.80 RT (Real Time) on a Pentium III: the speech recognizer needs (for decoding) only 80% of the time the user needs to pronounce the sentence in the worst case. The recognition process starts as soon as the speaker begins to speak and continues in parallel while the user is speaking; so the actual delay as the speech recognizer is very low: just a few milliseconds latency time.

Paralleling the metrics proposed in the speech recognition module, it is possible to compute the percentage of correct semantic concepts, inserted concepts, deleted concepts and substituted concepts. In the same way as considered previously, Concept Error Rate (CER) and the Concept Accuracy (CA) are calculated. In order to calculate the number of substitutions, insertions, deletions or correct concepts, the set of concepts (attribute and value, e.g. VELOCITY [140 knots], HEIGHT [1200 feet]) in the semantic frame obtained is compared to the reference with a dynamic programming algorithm. Table 4 presents these results with a CA greater than 92.2%. If there is an error in one of the velocity digits, the whole concept is considered as incorrect.

From these results, it is possible to conclude that in 92.24% of cases, the human operator could introduce a new order to the aerial vehicle without the need for any correction. We want to emphasize again that the purpose is to allow the controller to order high level commands while carrying out other actions at the same time, so minimum interference with the controllers main duties is achieved. This value summarizes the performance of the speech interface (including speech recognition and language understanding performances).

In order to evaluate the confidence measures provided by recognition and understanding modules, it is necessary to compute the Correct Rejection (CR: percentage of incorrect words/concepts that were rejected correctly: they had a confidence value lower than the threshold) and the Incorrect Rejection (IR: percentage of correct words/concepts that were rejected incorrectly: they were correct but they had a confidence value lower than the threshold).

In a speech interface, it is usually considered that the Incorrect Rejection (IR) rate must be lower than 5%: it is unfriendly to reject when the system recognized or understood the utterance spoken by the user correctly. In our case, once this condition ($IR < 5\%$) is considered, the system correctly rejected (Correct Rejection) 33.3% incorrect words and 16.7% incorrect concepts. The confidence measure avoided a wrong command in these cases being executed, thus increasing the system reliability.

The processing time required by the understanding module is less than 1 millisecond. For response generation and speech synthesis modules, the time is less than 100 milliseconds (in all cases considering the system working on a Pentium III computer). With these numbers, it is possible to report an overall response time of less than 200 milliseconds from the moment the user finishes speaking until the system gives the spoken response.

8. Comparison of aerial vehicle control interfaces including speech technology as one of the modalities and main conclusions of this work

Table 5 summarizes and compares the main aerial control interfaces including speech technologies. From this table, it is possible to conclude that, so far, WITAS has been the project that has incorporated the highest number of speech techniques in aerial control interfaces. This paper presents significant improvements in the robustness, reliability and ergonomics of speech technology for an aerial vehicle control interface:

- In order to increase the flexibility and robustness when dealing with spontaneous speech, the speech recognizer has incorporated special acoustic models (dealing with effects like speaker artifacts, filled pauses, etc.), and a smoothed statistical language model: 2-gram model (for permitting a high level of flexibility in the word sequences).
- Confidence measures for speech recognition and language understanding modules. These measurements provide valuable information on the reliability of these modules. It is a very important aspect in critical applications such as aerial control: confidence management adds up a new and very relevant feature to the speech interface: the ability to reject ill-formed utterances that could cause catastrophic consequences if not avoided.
- A new response generation module is proposed considering several levels of verbosity and selected depending on two parameters: the urgency of the situation and the confirmation verbosity desired by the user. Considering these specifications the speaking rate and pitch of the speech synthesizer is also modified. This new response generation module has significantly increased the ergonomics of the interface.

Apart from the aforementioned improvements, this paper provides a detailed state-of-the-art review of incorporating the speech technology into aerial control systems from a speech technology expert point of view. This paper also describes the speech interface for aerial control developed by GTH-UPM and the Boeing Research and Technology Europe Company. In the evaluation of the system, the final results reported a 96.4% Word Accuracy and a 92.2% Semantic Concept Accuracy. Additionally, the confidence measures were very useful in avoiding the carrying out of wrong commands and the flexibility of the response generation module made the interface more agile, ergonomic and adapted to the situation and user preferences. The system performance achieves the requirements stated in the introduction section.

From the work presented in this paper, one important conclusion is that the speech technology is ready enough to be considered as a new modality (in parallel with traditional ones such as joysticks or keyboard) for introducing high level commands while the controller is carrying out other main duties for commanding aerial vehicles. But the speech technology has to be improved in order to become the only interaction modality.

It is important to highlight that the best performance is reached when all the configuration possibilities of the speech engines are accessible (in some cases the commercial engines have explicitly

Table 5
Summary of speech technology applied to aerial control interfaces.

	WITAS [10,14–16,21]	Morgan Quigley et al. [19]	Mark Draper et al. [5]	GTH-UPM
Speech recognition	Nuance (speaker independent continuous speech recognition)	Microsoft (speaker independent continuous speech recognition)	Nuance (speaker independent continuous speech recognition)	Proprietary speaker independent continuous speech recognition with **Confidence Measures and special acoustic models**
Language modeling	Grammar-based LM (Nuance)	Grammar-based LM (Microsoft)	Grammar-based LM (Nuance)	2-gram statistical Language Model: robustness against **Spontaneous Speech**
Natural language understanding	SRI Gemini parser (Rule-based technique: bottom–up strategy)	(Rule-based technique: top–down strategy)	(Rule-based technique: top–down strategy)	Proprietary Rule-based module: bottom–up strategy with **Confidence Measures**
Response generation	Proprietary Template based technology	Prefixed sentences	No	Proprietary Template-based technology with **three levels of verbosity**
Speech synthesis	Festival 1.4.1. (Free software for English)	Microsoft Speech API	No	Boris: **speaking rate and speaker pitch modulation depending on the level of verbosity**
Main conclusion of the work	They are very optimist using speech technologies in aerial control interfaces. Good results in clean environments	Speech is a very ergonomic modality but is slow and the performance is not very good. Vocabulary of 50 words but it was evaluated in noisy conditions	Very good results for speech interfaces (WA > 90% clean environment with 160 words)	The speech technology is ready to be used in critical applications such as aerial control but the best performance of this technology required the collaboration of speech technology experts. (**WA > 95%** with 93 words)

set limits in order to increase the robustness against non-expert user handling) and the speech interface is designed in collaboration with speech technology experts.

Acknowledgements

Authors want to thank discussions and suggestions from the colleagues at GTH-UPM and Boeing Research and Technology Europe Company. Some of the developments have been partially supported by the following projects: SD-TEAM (MEC ref: TIN2008-06856-C05-03) and ROBONAUTA (ref: DPI2007-66846-C02-02). Authors also want to thank Mark Hallett for the English revision.

References

[1] E. Craparo, E. Feron, Natural language processing in the control of UAV, AIAA Guidance, Navigation, 2004.
[2] R. de Córdoba, J. Ferreiros, R. San-Segundo, J. Macías-Guarasa, J.M. Montero, F. Fernández, L.F. D'Haro, J.M. Pardo, Air traffic control speech recognition system. Cross-task & speaker adaptation, IEEE Aerospace and Electronic Systems Magazine (ISSN 0885-8985) 21 (9) (August 2006) 12–17.
[3] R. de Córdoba, J.M. Montero, J. Gutiérrez-Arriola, J.A. Vallejo, E. Enríquez, J.M. Pardo, Selection of the most significant parameters for duration modeling in a Spanish text-to-speech system using neural networks, Computer Speech & Language (ISSN 0885-2308) 16 (2) (2002) 183–203.
[4] M. Dora, H. Waage, E. Thora, Language Technology in Air Traffic Control, IEEE, ISBN 0-7803-7844-X, 2003.
[5] Mark Draper, Gloria Calhoun, Heath Ruff, David Williamson, Timothy Barry, Manual versus speech input for Unmanned Aerial Vehicle control station operations, in: Proceedings of the Human Factors & Ergonomics Society's 47th Annual Meeting, 2003.
[6] M. Faundez-Zanuy, Signature recognition state-of-the-art, IEEE Aerospace and Electronic Systems Magazine 20 (7) (July 2005) 28–32.
[7] M. Faundez-Zanuy, E. Monte-Moreno, State-of-the-art in speaker recognition, IEEE Aerospace and Electronic Systems Magazine 20 (5) (March 2005) 7–12.
[8] F. Fernández, J. Ferreiros, J.M. Pardo, V. Sama, R. Córdoba, J. Macías-Guarasa, J.M. Montero, R. San-Segundo, L.F. D'Haro, M. Santamaría, G. González, Automatic understanding of ATC speech, IEEE Aerospace and Electronic Systems Magazine 21 (10) (October 2006) 12–17.
[9] J. Ferreiros, R. San-Segundo, F. Fernández, L. D'Haro, V. Sama, R. Barra, P. Mellén, New word-level and sentence-level confidence scoring using graph theory calculus and its evaluation on speech understanding, in: Interspeech 2005, Lisboa, Portugal, September 2005, pp. 3377–3380.
[10] Alexander Gruenstein, Conversational interfaces: A domain-independent architecture for task-oriented dialogues, M.S. Project Symbolic Systems Program, Stanford University, 2002.
[11] T. Hain, P.C. Woodland, G. Evermann, M.J.F. Gales, Xunying Liu, G.L. Moore, D. Povey, Lan Wang, Automatic transcription of conversational telephone speech, IEEE Trans. on Speech and Audio Processing 13 (2005) 1173–1185.
[12] H. Hering, Comparative experiments with speech recognizers for ATC simulations, EEC Note No. 9/98, Eurocontrol Experimental Centre, Eurocontrol, Bretigny, France, 1998.
[13] A. Lechner, P. Mattson, K. Ecker, Voice recognition: Software solutions in real-time ATC workstations, IEEE AESS Systems Magazine (November 2002) 11–15.
[14] Oliver Lemon, Alexander Gruenstein, Collaborative activities and multi-tasking in dialogue systems, TAL 43 (2002).
[15] Oliver Lemon, Alexander Gruenstein, Multithreaded context for robust conversational interfaces: Context-sensitive speech recognition and interpretation of corrective fragments, 2003, ACM Transactions on Computer–Human Interaction 11 (3) (September 2004) 241–267.
[16] Oliver Lemon, et al., The WITAS multi-modal dialogue system I, 2001.
[17] F. Marque, S. Bennacef, F. Neel, S. Trinh, Parole: A vocal dialogue system for air traffic control training, in: ESCA Workshop "Applications of Speech Technology", Lautrach, Germany, September 16–17, 1993.
[18] F. Marque, F. Neel, PAROLE. Aide à la formation et l'entraînement des contrôleurs de trafic aérien, in: 6th Aerospace Medical Panel Meeting, Symposium Virtual Interfaces: Research and Applications, Lisbon, October 18–22, 1993.
[19] Morgan Quigley, Michael A. Goodrich, Randal W. Beard, Semi-autonomous human–UAV interfaces for fixed-wing mini-UAVs, 2002.
[20] J. Rankin, P. Mattson, Controller interface for controller–pilot data link communications, in: Proceedings of the 16th Digital Avionics Systems Conference, October 1997.
[21] Erik Sandewall, Patrick Doherty, Oliver Lemon, Stanley Peters, Words at the right time real time dialogues with the WITAS unmanned aerial vehicle, 2003.
[22] Dirk Schäfer, Context-sensitive speech recognition in the air traffic control simulation, Universität Der Bundeswehr München Fakultät Für Luft- Und Raumfahrttechnik, PhD Thesis, 2001 and Eurocontrol Experimental Centre EEC Note No. 02/2001.
[23] http://www.ai.sri.com/natural-language/projects/arpa-sls/nat-lang.html.