Deliverable 3.4

"Limited Language and Speech Recognition and Synthesis System"

Contract number**: FP7-215554 LIREC**

LIving with Robots and intEractive Companions

Start date of the project: 1st March 2008 Duration: 54 months

The research leading to these results has received funding from the European Community's Seventh Framework Programme (*FP7/2007-2013*) under *grant agreement* n° 215554.

Identification sheet

Project ref. no.	**FP7-215554**
Project acronym	LIREC
Status & version	[Draft] / [Final] "D3.4"
Contractual date of delivery	
Actual date of delivery	
Deliverable number	D3.4
Deliverable title	
Nature	"report "
Dissemination level	**PU** Public **PP** Restricted to other programme participants (including the Commission Services) **RE** Restricted to a group specified by the consortium (including the Commission Services) **CO** Confidential, only for members of the consortium (including the Commission Services)
WP contributing to the deliverable	WP3
WP / Task responsible	"WP3/T3.1.1"
Editor	David Martins de Matos
Editor address	david.matos@inesc-id.pt
Author(s) (alphabetically)	Alberto Abad, David Martins de Matos, Luís Caldas de Oliveira, Isabel Trancoso
EC Project Officer	Pierre-Paul Sondag
Keywords	
Abstract (for dissemination)	Automatic Speech Recognition, Speech Synthesis

CONTENTS

1 Introduction

This report focuses on the tools available for speech interaction in environments with limited resources, especially portable devices or platforms such as those envisioned as embodiments for the agents in the LIREC scenarios. In addition, language issues and how they impact the project are also discussed. These include, but are not limited to, the choice and availability of resources for a given natural language, for instance, European Portuguese or Scottish English.

Many levels of understanding can be considered in an interaction such as the ones defined in the LIREC scenarios. These levels range from simple morpho-syntactic analysis (e.g., to identify the constituents of an utterance), to full semantic processing in which world objects are considered when processing an utterance. In this document, though, we only discuss the issues concerning speech recognition and synthesis. These two steps are important because they represent two windows to the human world. The first of these, speech recognition, allows the agent to hear human speech and act on it. The second, speech synthesis, sometimes also referred to as text-to-speech (because it is often used for that purpose), allows humans to listen to what the agent has to say.

This document is organized as follows: section 2 presents the speech recognition task considering the LIREC context, discussing alternatives for LIREC scenarios. Section 3 presents the speech synthesis task, also considering the LIREC context. In both cases, off-the-shelf solutions were considered. Where appropriate, the nature of the necessary language resources is also discussed.

2 Speech Recognition

2.1 Introduction

In a previous report [D3.1], three basic types of speech recognition systems were considered: isolated word recognizers (IWR) of limited vocabulary, large vocabularies continuous speech recognizers (LVCSR) with sub-word acoustic models, and intermediate keyword spotting (KWS) approaches, which aim at recognizing a keyword in the middle of a continuous audio stream.

Addressing robots such as AIBO or PLEO like pets calls for a keyword spotting approach targeting a limited vocabulary, typically less than 200 words.

On the other hand, the type of human-machine interaction that one may look for in a scenario such as robot-house may range from simple Command and Control type of interaction (e.g. drive 3 meters forward) to a very complex one requiring natural language understanding (e.g. *what's on my schedule for tomorrow afternoon?* Or, *who else has already arrived?*).

The need for in-domain training data and the target of using off-the-shelf developed systems made us constrain our concerns to the first type of recognition systems. KWS approaches are broadly classified into two categories (Szöke et al. 2005): one based on the acoustic match of the audio data with keyword models in contrast to a background model, and the other one based on LVCSR. The acoustic solution can be based on either word or sub-word models. The LVCSR solution typically takes advantage of the lattice of recognized words, containing several hypotheses in parallel, thus allowing improved performances compared to searching in the raw output result.

Whenever possible, the speech recognition system will be activated by the user, in the sort of push-to-talk technique widely used in PDA interactions.

In the sort of interaction envisaged in LIREC, it may make sense to use acoustic models adapted to the user/environment, bootstrapping from speaker independent acoustic models on the basis of some previously recorded training data from the user in that particular environment.

A particular concern in verbal interaction with moving robots is the noise caused by the mechanical movements. Acoustic noise cancelling approaches may be used to reduce this problem.

Another major concern is microphone placement: speech captured by distant microphones in any enclosure will inevitably contain noise and reverberation components (Nakatani et al. 2010). Although many successful techniques have been proposed for dealing with noise (particularly with uncorrelated noise with simple spectral shape), sound reverberation remains a very hard problem. Ideally, all verbal interaction in the LIREC framework should take place via close-talking microphones, head-mounted, lapel or handheld, to minimize these problems.

2.2 Speech Recognition Engine

Our first speech recognition experiments within the scope of this project were made using the in-house speech recognition engine. Audimus is a hybrid recognizer that combines the temporal modelling capabilities of Hidden Markov Models (HMMs) with the pattern discriminative classification capabilities of multi-layer perceptrons (MLPs). The acoustic modelling combines phone probabilities generated by several MLPs trained on distinct feature sets: PLP (Perceptual Linear Prediction), Log-RASTA (log-RelAtive SpecTrAl) and MSG (Modulation SpectroGram). Each MLP classifier incorporates local acoustic context via an input window of 13 frames. The resulting network has two non-linear hidden layers and N softmax output units (N=40 for European Portuguese, corresponding to 38 phones plus silence and breath noises). Our decoder is based on the Weighted Finite-State Transducer (WFST) approach to large vocabulary speech recognition (Mohri et al. 2000). In this approach, the search space is a large WFST that maps HMMs (or in some cases, observations) to words. This WFST is built by composing various components of the systems represented as WFSTs. In our case, the search space integrates the HMM/MLP topology transducer, the lexicon transducer and the language model one.

The most significant deployment of this engine, in its LVCSR mode, is for the fully automatic subtitling system of the Portuguese TV news shows (daily, since March 2008). Although Audimus has been fine-tuned for European Portuguese, versions were developed for other varieties of Portuguese (Brazilian, African), and also for Spanish and English. The common metric used to evaluate such systems is the WER (Word Error Rate). For Broadcast News recognition, Audimus achieves a WER of 18.4% for European Portuguese, 21.6% for Brazilian Portuguese, 27.9% for African Portuguese (small amount of training data), 15.7 for European Spanish, and 22.0%/20.6% (on Eval 97/ Eval 03) for American English.

We have compared the performance of this hybrid speech recognizer with a public domain recognizer (Sphinx), distributed by Carnegie Mellon University (http://cmusphinx.sourceforge.net/), for which there are already trained acoustic and language models. The WER achieved by Sphinx in the test corpora that we had access to (Eval 97 / Eval 03) is about 3% higher than the one achieved for Audimus. However, more expertise with the Sphinx system is needed to achieve better results with the system.

2.2.1 Acoustic-based keyword spotting

A general schema of acoustic based keyword spotting systems is shown in Figure 1. Recognition of selected keywords in competition with a background model is performed. The better a background model is, the highest likelihood obtains for out-of-vocabulary words and the lower for keywords.

Two acoustic matching approaches to keyword spotting have been assessed depending basically on how the background or filler model was modelled: one based on a phoneme loop network in the recognition grammar and the other one based on the explicit computation of a posterior probability for a background unit based on the average of the most probable phonemes at each time instant.

2.2.1.1 Background modelling: Phoneme loop network

The most straightforward way of tackling the problem of background modelling in hybrid recognition systems is the use of a loop of all the possible phonemes in the recognition grammar as the background/filler model.

However, this approach presents some practical problems. When the audio presented to the recognizer is previously segmented based on an speech/non-speech, the kind of solution in Figure 1 fails, since an unrestricted sequence of phonemes is more likely to present a higher likelihood than any restricted sequence of phonemes corresponding to a target keyword (at least in the actual recognition configuration). In other words, there is an unacceptable amount of misses.

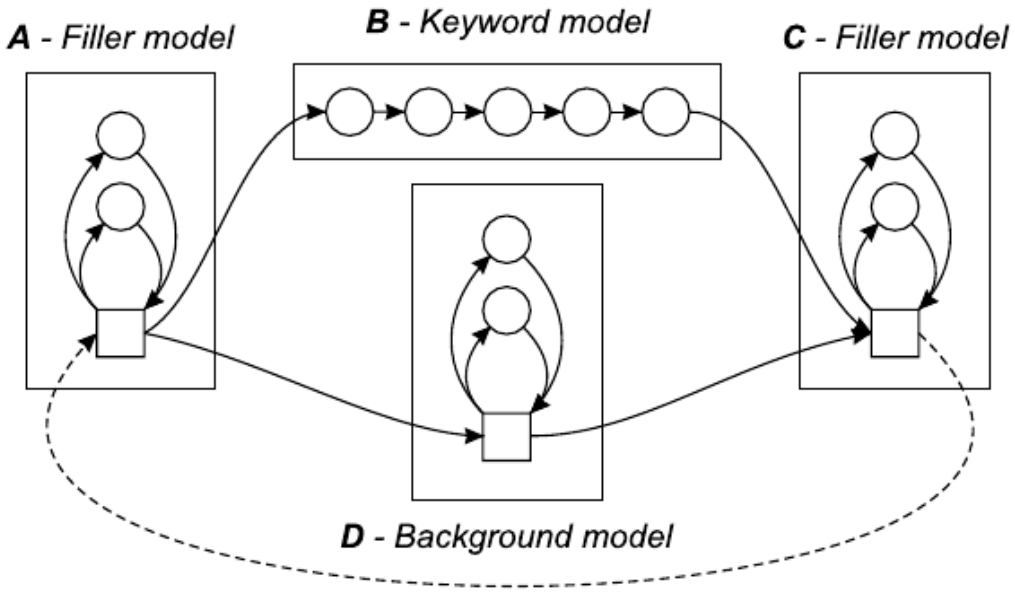

Figure 1: Acoustic KWS.

This approach was evaluated on the test set and showed a very poor performance as it can be seen in the results Table 1 in the row named ACoustic KeyWord Spotting with Phone Loop unrestricted (*AC-KWS-PL unrestricted*). As commented previously, errors come from deletions/misses.

Note that this result does not completely invalidate the approach since it is possible to apply this solution in an alternative on-line configuration. When using KWS in real-time operation,

we can skip a previous segmentation stage and define a grammar that is composed of a filler model followed by a keyword model. That is, the right filler model and the alternative background model is removed from the scheme of Figure 1 and the keyword detection also acts somehow as a kind of segmentation.

Since we cannot assess this approach in an off-line evaluation, we propose to validate it in a slightly different configuration. Basically, the right-filler model is kept but the alternative background model is removed. That is, we force to detect at list one keyword in each testing utterance. The results of this experiment are labelled as *AC-KWS-PL restricted* and show a much reasonable performance for a task that the one proposed.

2.2.1.2 Background modelling: Average Posterior Probability

An alternative to phone loop network for background modelling that will be referred to as ACoustic KeyWord Spotting with Average Posterior Probability estimation (AC-KWS-APP) has been also assessed.

The basic idea is to build a phoneme classification network that in addition to the classical phoneme units, also models the posterior probability of a background unit representing "general speech". This is usually done by using all training speech as positive examples for background model. Unfortunately, this approach requires re-training of the acoustic networks in order to add an additional output and it is an option that is not considered in this work.

Alternatively, in some works the posterior probability of the background unit is estimated based somehow on the posterior probabilities of the other phones (Pinto et al. 2007). In this work, we propose to estimate the posterior of a background unit as the mean probability of the five most likely outputs of the phonetic network at each time frame. This simple approach allows estimating posterior probabilities for specific background recognition unit without the need for network re-training.

Table 1 shows results for both unrestricted and restricted conditions described in the previous section. In addition to performing better than AC-KWS-PL approach, it can be seen that it is possible to use this approach under the unrestricted condition, that is, a background model is competing against keyword models without forcing at least one keyword detection. As expected, the unrestricted condition results are worse than the restricted ones (again mainly because of the deletions) but the loss in performance is not so huge as in the AC-KWS-PL method.

System	#words	#insertions	#deletions	#substitutions	Error %
AC-KWS-PL unrestricted	558	0	475	0	85.12
AC-KWS-PL restricted	558	5	83	64	28.14
AC-KWS-APP unrestricted	558	36	47	9	16.48
AC-KWS-APP restricted	558	39	15	20	13.26
LVCSR	558	9	141	2	27.24

Table 1: Results for the systems compared in this work.

2.2.2 LVCSR keyword spotting

The use of searching methods on the LVCSR result has been shown to be a successful alternative for keyword spotting. It is usually preferred performing search in the recognition lattices since they contain more information that can be useful in the search of target words than using the raw recognition result.

In this work, for the sake of simplicity we evaluate KWS based on LVCSR by searching the selected keywords only in the recognition result. For that purpose, we have used the same acoustic models than in the previous sections with a language model with an active vocabulary of 100k words for Broadcast News transcription.

The last row of Table 1 (LVCSR) shows the results for this method, which can be considered quite promising, especially if we take into account that the search has been done in the text result and that it has not been done any tuning of the recognition configuration/parameters. It must be noticed that one of the selected keywords ("rechamar") is not in the vocabulary of LVCSR system and it can never be detected by this system. As already commented in the Introduction, this is one of the limitations of these methods.

It seems that acoustic methods based on average posterior probability estimation are the ones achieving better performances with a relative low complexity. However, methods based on LVCSR seem also a promising alternative, particularly if we take into account that no lattice search has been used.

A number of improvements can be done in the near future. For instance:

- No parameter optimization was done at this stage, which may be a critical issue in keyword spotting. By tuning acoustic scale weight, word insertion penalty and other typical configuration parameters it would be possible to reduce deletion problems in some of the systems presented.
- The use of confidence measures and its estimation is also a very important issue in keyword spotting. No work/experiment has been done in this sense yet.
- In the particular case of the acoustic based methods, it would be interesting to make the comparison between sub-word based models (such as the ones presented in this report) and word-based models. In this last case, an HMM-based system should be developed with an HMM modelling each different keyword and the background.
- In the particular case of the LVCSR methods, the use of lattices may result in a considerable improvement.

2.2.3 ASR Toolkits

2.2.3.1 ATK

ATK (http://htk.eng.cam.ac.uk/develop/atk.shtml) is an API designed to facilitate building experimental applications for the widely used HTK (Hidden Markov Model Toolkit). As the name implies, the latter is a portable toolkit for building and manipulating hidden Markov models. HTK consists of a set of library modules and tools available in C source form. The tools provide sophisticated facilities for speech analysis, HMM training, testing and results analysis. The software supports HMMs using both continuous density mixture Gaussians and discrete distributions and can be used to build complex HMM systems.

ATK consists of a C++ layer sitting on top of the standard HTK libraries. This allows novel recognizers built using customized versions of HTK to be compiled with ATK and then tested

in working systems. Like HTK itself, it is portable across the main Unix platforms and Windows.

ATK features include:

- Multi-threaded to allow efficient and responsive real-time operation.
- Synchronised audio input/output with barge-in support.
- Support for finite-state grammars and trigram language models.
- Ability to return recognition results word-by-word as they are recognised to reduce latency
- N-best recognition output
- Support for HLDA
- Integrated Flite speech synthesis.
- Make files for single-build under Linux and Windows

ATK supports both Linux and Windows in a single set of sources so like HTK itself, there are no platform specific distributions.

2.2.3.2 Pocketsphinx

Pocketsphinx (http://cmusphinx.sourceforge.net/) is a lightweight speech recognizer specifically designed for portable devices. It is a branch from Carnegie Mellon University's (CMU) famous Sphinx speech recognizer. Pocketsphinx as well as the other versions of the Sphinx speech recognizer has an in-depth tutorial explaining how to compile, train and install the speech recognizer. Even though Sphinx is a continuous speech recognizer, this limited-resources version is a semi-continuous speech recognizer.

We are currently working on this stage. We were able to use Pocketsphinx to successfully recognized wave sound files in a batch procedure. However, we were unable to use the microphone to process real-time continuous (semi-continuous actually) speech recognition.

2.2.3.3 Julius

Julius (Kawara et al. 1998, Lee et al. 2001) is a portable open-source real-time large vocabulary speech recognizer.

Julius employs N-grams as a statistical language model (LM), though, as a toolkit for various tasks, grammar-based LM is suitable for small tasks, where easy-to-use and easy-to-customize LMs are preferable. In order to provide such a grammar-based recognition engine as a functional module of the toolkit, "Julian" has been developed. Julian can change more than one grammar sets on demand, and it can output incremental speech recognition results.

We were able to do real time speech recognition using the computer's microphone with Julius in Japanese. However the models used were very limited ones. We are currently testing very small dictionaries and grammars and using predefined untrained models.

3 Speech Synthesis

3.1 Introduction

Several of the scenarios envisaged in the project include the need to produce speech as one of the forms of interaction with the user. Although many of them only are only able to produce a limited number of phrases, the use of pre-recordings or the development of a limited domain synthesizer would seriously reduce the flexibility of the development process. For this reason the project opted for the use of speech synthesis systems of unlimited vocabulary commonly referred to as text-to-speech systems. The increased flexibility results in the reduction of the naturalness of the generated speech and greater computational requirements.

Since the development of speech synthesis systems is not included in the objectives of the LIREC project, it is required to evaluate existing off-the-shelf systems, both commercial and non-commercial, to see if they can be used in the project. As we shall see, no system meets the requirements for all scenarios, and thus the selection of the synthesizer to be used will depend on the scenario.

To help the selection process we will start by listing the most relevant speech synthesis systems available. Next we will define the most relevant requirements that the system should have. Finally we will evaluate the different systems in regard to these requirements.

3.2 Available Text-to-Speech Systems

3.2.1 Non-Commercial Systems

There are some non-commercial speech synthesis systems available for download on the internet. These are mostly originated in research projects in the speech technology area for which the researchers opted by this form of dissemination instead of licensing them for commercial exploitation. In selecting these systems we choose the ones that generate speech from text and that are available for several languages.

3.2.1.1 eSpeak (http://espeak.sourceforge.net/)

This system includes a parametric formant speech synthesizer. The control of the synthesizer parameters is done by manually written rules developed by experts. It is base on the synthesis model developed in the 80's by Dennis Klatt and is team at MIT and that resulted in one of the first commercially available text-to-speech systems: the DECTalk. This synthesis model produces speech with limited naturalness but can achieve high intelligibility, particularly for users that use those systems regularly. For this reason they are very commonly used together with screen readers to provide access to computers to users with visual impairments, since they usually speed-up the speech output rhythm to unnatural levels. The eSpeak system has the additional advantage of not requiring much memory and of being easily ported to a new platform. However, its use for synthetic or robotic characters is not usually accepted by the majority of the users since they expect a human like voice.

3.2.1.2 Festival (http://www.cstr.ed.ac.uk/projects/festival/)

The Festival system was developed with the aim of being a development platform for the evaluation of the different modules that compose a text-to-speech system. For this reason there are available several versions of the system with different technologies and for

different languages. In its base version, it is a concatenation-based synthesis system having the ability to use different voices, some of them with a very reasonable size. However, being a research system, the voices are mostly from non-professional speakers and segmentation of the recordings does not have enough quality when compared with commercial systems. On top of that, the system was developed with minimal efficiency requirements, using too much memory and time to process each sentence. For this reason, a more efficient version was developed, named Flite (Festival Lite) written in the C programming language. This version is much more efficient, but less flexible, and it is still limited by the quality of the voices. The Festival and Flite systems were used as based for commercial systems like the AT&T Natural Voices and Cepstral.

3.2.1.3 FreeTTS (http://freetts.sourceforge.net/)

The FreeTTS system is a port of the Flite system to the Java programming language, to increase its flexibility and to make it more platform-independent. One of the difficulties of Flite is the interface with the audio device that must be adapted for each platform. The Java environment offers an abstraction for the audio device, making the porting trivial. The FreeTTS system uses the same voices of Festival and thus the same quality limitations.

3.2.1.4 Mary TTS (http://mary.dfki.de/)

The Mary text-to-speech system was originated in a German research project involving DFKI and the universities of Saarbrucken and Saarland. It is currently maintained by DFKI that regularly produces new releases of the system. The project received a large grant from the German government that allowed the recording of high quality speech databases for German and English. For this reason it is the non-commercial system with the best quality offering also the flexibility of the Java programming language.

3.2.2 Commercial Systems

3.2.2.1 AT&T Natural Voices (http://www.wizzardsoftware.com/speech_overview.php)

Like all the commercial systems in this review, the AT&T Natural Voices system uses synthesis by concatenation of variable length segments of pre-recorded speech. All of the available voices are of high-quality. However, the system has not evolved in the past few years, the last version being from 2008.

3.2.2.2 Nuance RealSpeak (http://www.nextup.com/nuance.html)

The Nuance speech synthesis products are mostly based on the technology acquired by the company after the bankruptcy of Lernout & Hauspie in 2001. Although it is continuously producing new voices and new languages, some of the voices and languages that it sells are still the same that were used in L&H products and not all of them benefited from the technological progresses made by the company. By acquiring most of its competitors, Nuance placed itself has the leader in the speech technology market. The company is particularly focused in offering solutions integrating different speech technologies in products that use a large number of licences. This way the only way to acquire single user licences is through small commercial partners like NextUp.

3.2.2.3 Acapela (http://www.acapela-group.com/speech-synthesis-voices.html)

Acapela focuses exclusively on speech synthesis products, offering solutions in 25 languages. The company is the result of the merging of three European companies in this area: Babel Technologies (Belgian), Infovox (Swedish) and Elan Speech (French). The company's technology originates mostly in the MBROLA system developed by the Faculté Polytechnique de Mons and owned by Babel Technologies. The University still maintains the MBROLA project but the encoding of the recording for a new voice has to be performed internally. Also, the synthesis software is only publicly available in executable format and not the source code. Acapela voices are, in general, of very good quality.

3.2.2.4 Cepstral (http://cepstral.com/)

The Cepstral synthesis system is based on the Flite system. The strategy of the company is to offer a wide number of different voices for the English language, including character voices. The voices are smaller than the ones used in competing systems at the cost of naturalness of the synthetic speech. It is the least expensive commercial system.

3.2.2.5 SVOX (http://www.svox.com/TTS-Technology.aspx)

SVOX was created in 2000 as a spin-off of the do Swiss Federal Institute of Technology in Zurich (ETH Zurich). The company has been focused in the development of speech synthesis for embedded systems, mostly for the automotive industry. These systems are characterized by limitations in both memory and computing power imposing limitations that affect the quality of the synthesized speech. The company has recently acquired the speech technology division of Siemens A.G. The SVOX products are mostly solution for systems integrators with difficult access for researchers.

3.2.2.6 Loquendo TTS (http://www.loquendo.com/en/demos/demo_tts.htm)

The technology used by Loquendo originated in CSELT, the research laboratories of Telecom Italia. The company is the market leader in Europe and Nuance's main competitor. It offers a wide range of voices and most of them have the ability to include expressive cues in between the synthesized speech. Most of the voices have naturalness at the level of the best in industry. Like with Nuance, individual licences have to be bought through commercial partners.

3.2.2.7 IVONA TTS (http://www.ivona.com/)

IVO Software is a company originated in Poland working exclusively on the development of speech synthesis systems and applications. It offers 9 voices in 4 languages and sells their products directly on the Internet with individual licences. The naturalness of the voices was recognized in an international assessment involving mostly research systems and not the main actors of the industry.

3.2.2.8 CereVoice (http://www.cereproc.com/)

CereProc is a company created in 2005 exclusively dedicated to the development of text-to-speech systems. Its main product, CereVoice, is available for servers, desktops, mobile and

embedded devices. Part of its team was previously with Rethorical Systems that was acquired by ScanSoft (currently Nuance). Being a small company associated with the University of Edinburgh, it has an academic licensing program that allows the use of the system development kit with no cost for research purposes. The naturalness of the voices is aligned with the best offer in the market. In addition, it has the particular feature of having multi-lingual voices with the ability to produce some emotional variation.

3.3 Parameters for the Assessment of the Systems

In this section we describe the most relevant requirements for the use of a speech synthesis system in the scenarios of the LIREC project.

3.3.1 Naturalness

A detailed assessment of the naturalness of synthetic speech produced by a text-to-speech system is a complex task involving a panel of evaluators. Since a detailed classification is not need for our purposes, we will simply classify the systems in 4 broad classes:

A – High naturalness with almost no concatenation distortions, at the level of the best systems available.

B – Some naturalness with occasional concatenation distortions, but at an acceptable level.

C – Frequent concatenation distortions seriously affecting the naturalness of the resulting speech.

D – Clearly synthetic voice with low naturalness, only usable in specific situations.

3.3.2 Research License

This parameter indicates the availability of special features just for research purposes.

3.3.3 Cost of a Single License

Some of the technology providers do not have individual licenses except for a limited time evaluation. The scenarios of the project require very few licenses but for unlimited time.

3.3.4 Windows

Support for the Windows operating system: Windows XP, Windows Vista, or Windows 7.

3.3.5 Linux

Support for the Linux operating system.

3.3.6 Android

Support for Android operating system in mobile devices.

3.3.7 Southern UK English

Availability of voices with standard British English accent and which genders are available.

3.3.8 Scottish English

Availability of voices of English with Scottish accent and which genders are available.

3.3.9 European Portuguese

Availability of voices in European Portuguese and which genders are available.

3.3.10 Standard API

Support for a standard application programming interface (API). The most commonly used interfaces are currently SAPI 5.0 for Windows and JSAPI for applications written in Java.

3.3.11 Lipsync

Lipsync is the possibility of synchronizing the movement of the lips of a virtual character with the output speech. This requires information regarding the type and duration of each phonetic segment of the synthetic speech signal.

3.3.12 SSML

Support for the Speech Synthesis Markup Language (SSML). This allows the inclusion tags in the input text to control the expressiveness of the generated speech.

3.3.13 Memory

An estimate of amount of system memory required for a single voice.

3.3.14 Sampling Rate

The output signal sampling rate limits the bandwidth of the speech signal.

3.4 Assessment of the Speech Synthesis Systems

Table 2 summarizes the characterization of the different systems in regard to the selected parameters. It is easily noticeable that no single system fulfils the all the requirements to be used in all scenarios. When the need for access to the source code is mandatory, the best solution is to use the Mary TTS for its higher naturalness when compared with the remaining open source systems. In terms of flexibility of integration, the CereVoice system combines voices with very high naturalness with the capability to insert SSML tags to control its expressiveness. The access to the software development kit, through an academic license, also helps the integration process. The only limitation is the lack of a Portuguese European voice. To meet this requirement a commercial solution must be used: Nuance Real Speak, Acapela and Loquendo TTS. Having the three of them equivalent quality, the licensing cost of the Nuance system is cheaper than the others.

	System	Naturalness	Research Licence	Single License	Windows	Linux	Android	Southern UK English	Scottish English	European Portuguese	Standard API	Lipsync	SSML	Memory	Sampling Rate
Non-commerci	eSpeak	D	GPL	0 €	Y	Y	Y	M	M	M	SAPI 5.0	Y	NA	1 MB	16 KHz
	Festival	C	BSD	0 €	NA	Y	NA	M	M	NA	NA	Y	NA	?	16 KHz
	FreeTTS	C	BSD	0 €	Y	Y	NA	M	NA	NA	JSAPI	Y	NA	?	16 KHz
	Mary	B	LGPL	0 €	Y	Y	NA	M/F	NA	NA	NA	Y	Y	?	16 KHz
Commercial	AT&T Natural Voices	A	N	40 €	Y	Y	NA	M/F	NA	NA	SAPI 5.0	Y	Y	?	16 KHz
	Nuance RealSpeak	A	N	36 €	Y	NA	NA	M/F	NA	F	SAPI 5.0	Y	Y	100 MB	22 KHz
	Acapela	A	N	154 €	Y	NA	NA	M/F	NA	F	SAPI 5.0	Y	Y	200 MB	22 KHz
	Cepstral	B	N	24 €	Y	Y	NA	M/F	M	NA	SAPI 5.0	Y	N	50 MB	16 KHz
	SVOX	B	N	NA	Y	Y	Y	F	NA	F	?	?	?	?	16 KHz
	Loquendo	A	N	150 €	Y	Y	Y	M/F	NA	M/F	SAPI 5.0	Y	Y	30-60 MB	48 KHz
	IVONA	A	N	30 €	Y	NA	NA	M/F	NA	NA	SAPI 5.0	Y	NA	?	16 KHz
	CereVoice	A	Y	37 €	Y	Y	Y	M/F	M/F	NA	SAPI 5.0	Y	Y	?	22 KHz

Table 2: Summary of TTS systems features

4 References

Nakatani, T.; Kellermann, W.; Naylor, P.; Miyoshi, M.; Juang, B. H. (F.). 2010. Introduction to the Special Issue on Processing Reverberant Speech: Methodologies and Applications. IEEE Transactions on Audio, Speech, and Language Processing. v.18, n.7. p.1673–1675. ISSN 1558–7916.

Kawahara, T., Kobayashi, T., Takeda T., Minematsu, N., Itou, K., Yamamoto, M., Utsuro, T., and Shikano, K. 1998. Sharable software repository for Japanese large vocabulary continuous speech recognition. In ICSLP-98, p. 3257–3260.

Lee, A., Kawahara, T., and Shikano, K. 2001. Julius — an open source real-time large vocabulary recognition engine. In European Conf. on Speech Communication and Technology, p. 1691–1694.

Mohri, M., Pereira, F., Riley, M. Design principles of a weighted finite-state transducer library, Theoretical Computer Science, v.231 n.1, p.17–32, Jan. 17, 2000.

Pinto, J., Lovitt, A. and Hermansky, H. 2007. “Exploiting Phoneme Similarities in Hybrid HMM-ANN Keyword Spotting”, Proceedings of Interspeech'07.

Szöke, I., Schwarz, P., Matějka, P., Burget, L., Karafiát, M., Fapšo, M. and Černocký, J. 2005. “Comparison of Keyword Spotting Approaches for Informal Continuous Speech”, Proceedings of Interspeech'05, p. 633–636.