International Journal of Scientific & Engineering Research, Volume 8, Issue ƕƕƕ - ÖÝÌÔÉÌÙıĻŸƕƛ
ISSN 2229-5518

Embedded Real Time Speech Recognition System for Smart Home Environment

Leila Beltaifa – Zouari

Abstract— This paper describes an embedded real time speech recognition system developed, integrated and evaluated in a smart home local box. This work has been conducted in the framework of the project "Maison intelligente". The system work continuously and detect vocal commands coming from headset, webcam and smartphone microphones. Several realistic scenarios are defined, recorded, transcribed and tested. In order to improve the speech recognition rate with the respect to the box memory and speed limitations multiple acoustic models have also been considered.

Index Terms— Embedded systems, Hidden Marko Models, Real time, Semi continuous models, Smart home, Speech recognition, Vocal Activity Detection

1 Introduction

In the last decade huge effort has been done to improve Automatic Speech Recognition (ASR) techniques and effective systems have been developed. The most state-of-the-art ASR systems provide the performance quality (usually addressed by the recognition Word Error Rate WER and by the ratio of the processing time to the utterance duration), which affords the comfortable use of ASR in real applications [1], [16] ,[17],[18].

Recent mobile devices offer a large set of functionalities but their resources are still limited [2],[12],[13]. The strong increase of mobile devices in daily life has created a great demand for efficient and simple interfaces, in particular speech recognition being a key element of the conversational interface, there is a significant requirement for low-resource and accurate automatic speech recognition systems. However, the direct reproduction of the ASR algorithms is either not possible or mostly leads to unacceptable low performance on the mobile devices.

Smart home is a promising area. It has multiple benefits such as providing increased comfort, greater safety, and a more rational use of energy and other resources. This research application domain is important and will increase in future as it also offers powerful means for helping and supporting special needs of the elderly and people with disabilities [11],[15].

Due to the highly variable acoustic environment in the smart home and very limited resources available on the handheld terminals the implementation of ASR systems on the mobile devices necessitates special ajustment [6],[7],[10],[14],[16].

In this paper, we address the problem of development and integration of a low cost ASR system in smart home local box in order to control and monitor the home environment. Several constraints should be respected by this system:

1. Its memory size should not exceed 5 Mega bytes.
2. It has to run in real time on a MIPS 32 microprocessor which frequency is 500 MHz
3. The operation system is OpenWRT (a version of embedded Unix).

The different steps of this work that are mainly the vocal activity detection, the database development, and the speech decoding will be described.
As the performance and the speed of speech recognition systems are dependent on the number of HMM Gaussians [3],[4],[5],[8],[9], we will investigate different speech modeling techniques and compare their performance (WER, computing time and memory size).

2 System Architecture

Internet Of Things (IoTs) consists on connecting everyday objects like smartphones, internet televisions, sensors and actuators to the internet. These devices should be intelligently linked together to enable new forms of communication amongst people and themselves.
The significant advancement of IoTs over the last years has created a new dimension to the world of information and communication technologies. The IoTs technology can be used for creating new concepts and wide development space for smart homes in order to provide intelligence, comfort and improved quality of life.

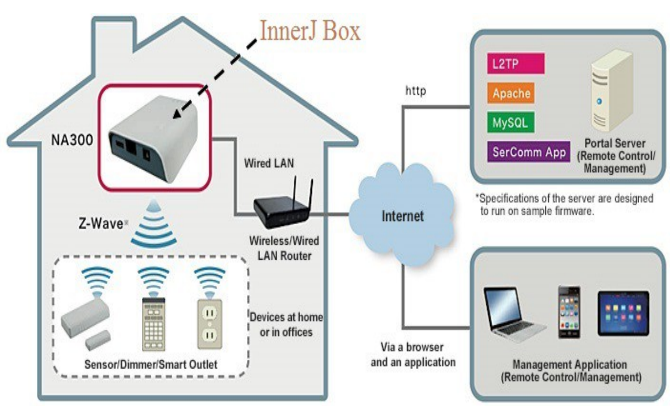

Figure 1. The smart home control system architecture

In this context, CHIFCO Company is developing the «InnerJ» solution (see Figure1). It is an energetic interactive platform that helps users to manage efficiently their energy consumption. By mean

of the «InnerJ» solution user can control his house office and job office and has statistics about his electric consumption.

CHIFCO implements the «InnerJ» solution in the box Sercom Na-300. The Secom NA-300 hardware is provided with a MIPS32 processor (with a frequency of 500 MHz) and 64 Mega bytes of memory (where only 5 mega bytes can be used by the ASR module). It is equipped with two outputs : USB and Ethernet.

Two protocols are used to communicate with the smart home devices: the « Z-Wave » wireless protocol and the Ethernet/WIFI protocol. Household equipments such as electric sensors and stove are commanded by the Z-wave protocol. Electronic peripherals like the computer or the smartphone communicate with the Ethernet and/or the WIFI protocol.

Nowadays, web services become the most open way of providing remote service access or enabling applications to communicate with each other. In particuler, for this application, the user can modify the settings by means of a web page. He can select the driver, he can remove or add a vocal command, etc. The drivers addressed by our application are the doors, the windows, the thermometers and the light switchs.

3 ASR SYSTEM OPERATION

.The ASR system should be implemented and integrated in the Sercom Na-300 box. It operates in three steps: speech acquisition, utterance recognition and command execution.

3.1 Speech input

The NA-3000 box has two data inputs/outputs: USB and Ethernet. A market study of the existing microphones shows that:

- For the USB connection, high and low sensibility microphones are available.

Meeting microphone and webcam are examples of high sensibility microphones.

Figure2. High sensitivity microphone examples

Headset and the snowflake are examples of low sensitivity microphones.

Figure3. Low sensitivity microphone examples

- For the Ethernet/WIFI connection, we find the tablet, the personnel computer and the smartphone.

In the light of these informations, the NA-3000 box can be connected to its peripherals as following:

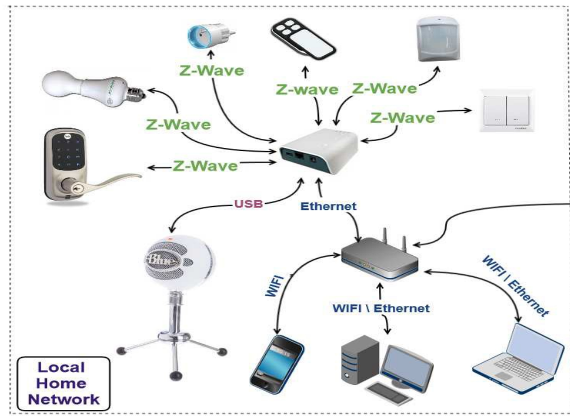

Figure4. Local home Network

3.2 Speech decoding

Several open source speech recognition toolboxes are available in the web: HTK[1], Julius[2], CMUSphinx[3], Kaldi[4], Simon[5], etc. We have chosen Sphinx Pocket because it has many advantages:

- It is compatible with the overspread audio cards such as Alsa, OSS, Jack, Pulse-audio,..
- It has a general public license
- It implements the semi-continuous models
- It implements the floating point algorithm in order to optimize the floats computing
- It is written in C language
- It has a make file close to the cross compilation

The ASR system is implemented locally in the box in order to prevent internet connection cut or ASR engines overhead or out of order.

3.3 Command execution

All the commands are written in a transcription file and a number is assigned to each sentence. The web services are written in another file and have the same numbering as the commands of the transcription file. The user can access both of these files and modify the commands. Figure6 show a sample of the transcription and the web sevices files.

After each utterance decoding, the program seek for the command number and performs the corresponding web service.

1 http://htk.eng.cam.ac.uk

2 http://julius.osdn.jp

3 https://cmusphinx.github.io

4 http://kaldi-asr.org

5 https://userbase.kde.org/Simon

```
activer reconnaissance
désactiver reconnaissance
allumer lumière
éteindre lumière
ouvrir porte
fermer porte
```

```
clear
clear
curl "http://IP_BOX:port/..........value=1"
curl "http://IP_BOX:port/..........value=0"
curl "http://IP_BOX:port/..........value=1"
curl "http://IP_BOX:port/..........value=0"
```

Figure6. Sample of command transcription and web Services files

4 Vocal Activity Detection

The ASR system should operate continuously. At the beginning, the source of the speech, which can be a USB microphone or a User Datagram Protocol Internet Protocol (UDP IP), is detected. This information is provided by the /proc/asound/cards file which contains the list of the connected cards. Then the program passes to a standby mode waiting for a vocal activity. If a sound is detected, the program switch to run mode in order to decode this speech. So at the beginning the command is recorded, then it is recognized and finally it is execued (Figure7).

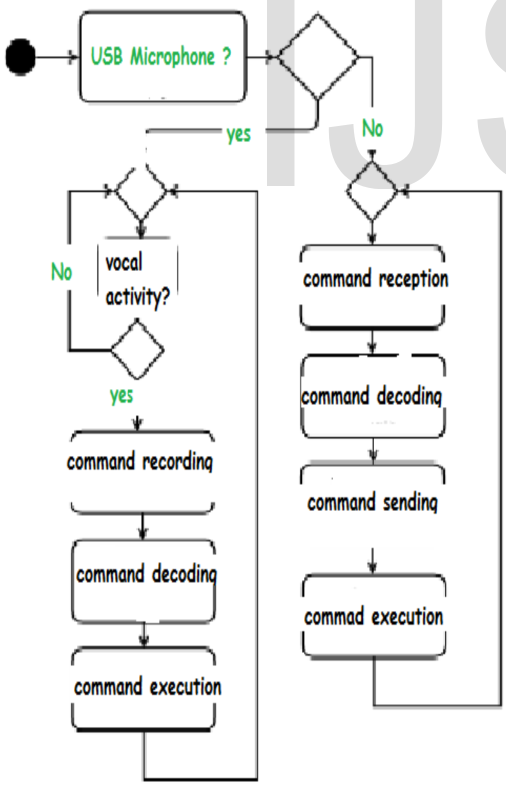

Figure7. Vocal Activity Detection diagram

4.1 USB based microphone

The USB microphone can be either high or low sensitivity.

- If the micophone is high sensitivity, the Vocal Activity Detection VAD module is unstable. In fact, the amplification gain vary to keep speech level constant. Another VAD can be implemented but it would be time consuming and could disturb the box operating (as it should manage the sensors outputs in the same time).

- If the micophone is low sensitivity, the existing VAD (which is integrated into the box) is used. In this case the distance between the user and the box should be limited (to about 1 meter). Using a Bluetooth based microphone or a USB connected smartphone can be considered as alternative solutions.

4.2 Wireless microphone

To make the system user friendly, we developed an android application that detects and records the vocal commands and send them to the smart home box by the UDP protocol (Figure8).

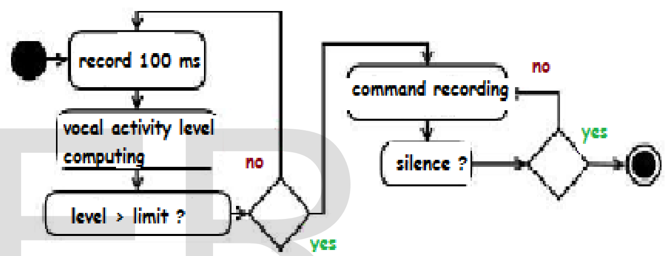

Figure8. Vocal Activity Detection algorithm

In order to ensure communication between the android and the box, this application needs the IP address of the box and the communication port.

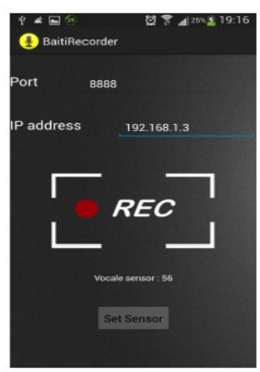

Figure9. Smartphone interface

The button “Set Sensor” makes it possible to modify the level of speech detection to be adequate to the environment. The vocal command is send to the smart home box by the UDP protocol. The smart home box response and the error messages are displayed in the smartphone “Dialog box”.

International Journal of Scientific & Engineering Research, Volume 8, Issue ŀŀŀŀ - ÖÝÌÔÉÌÙıŀŸŀÀ
ISSN 2229-5518

5 Experiments And Results

The development of an ASR system requires several resources that are the vocabulary, the grammar/language model and the acoustic models. For the following experiments we used Sphinx Train for the models training and Sphinx Pocket for the tests. The parameter vector contains 39 Cepstral coefficients (MFCC).
All the decoding experiments are performed with cross validation where 90% of the data is used for training and the rest for the tests.

5.1 Database development

Several realistic scenarios were defined. They concern the command of four devices:
- the doors (to open or close)
- the windows (to change the brightness)
- the thermostat (the temperature can be moved from 5 to 30 degrees)
- and the light switches.

The user is able to command the devices individually or to select a predefined scenario. He can also ask for the device status.
The commands are recorded then they are manually transcribed using Transcriber [13] and finally a grammar file is prepared (Figure10).

<res_mode> = (maison|thermostat|lumière)(mode(<mode>|<num_1_5>));
<num_1_5> = un | deux | trois | quatre | cinq;
<mode> = nuit | journée | matin;
<equipement> = (lumière | télévision | thermostat);

Figure10. Sample of the grammar file

Three French databases have been produced. The are described by Table1, Table2 and Table3.

Table1

Webcam multi-speaker database	
Microphone	High sensibility
Duration	230 minutes (161 men+69 women)
Speakers	5 men, 5 women

Table 2

Webcam mono-speaker database	
Microphone	High sensibility
Duration	124 minutes
Speakers	1 man

Table3

Headset mono-speaker database	
Microphone	Low sensibility
Duration	20 minutes
Speakers	1 man

5.2 Mono-speaker speech recognition

This paragraph is concerned with mono-speaker speech recognition experiments. Several acoustic models are developed and evaluated in order to find the best. The system's size should be less than 5 mega bytes and the decoding time should be less than real time (RT).

5.2.1 Primary experiments

In these primary experiments we used webcam and headset microphones. Twenty minutes of each mono-speaker database and context independent models are employed to develop and test two ASR systems. Results are reported in Figure11.

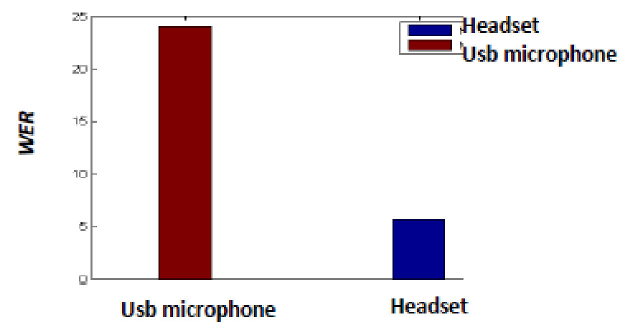

Figure11. WER for the headset and the webcam ASRs

We notice that the headset based system results are better than the webcam based system ones (WER < 5%). Despite these results, we choose to continue the experiments with the webcam because it present many advantages such as the possibility of moving (the voice is detected in about 16m²) and that it can easily be used by handicapped and aged people.
Using the webcam, the sampling frequency decreases from 16 KHz to 8 KHz so the hardware interrupt (when receiving the signal samples) is reduced and the other programs implemented in the box turn with less disturbance.

5.2.2 Continuous Context dependant models

We developed 1200 continuous context dependant models (triphones). We varied the number of Gaussians per state. Results are reported in Figure12.

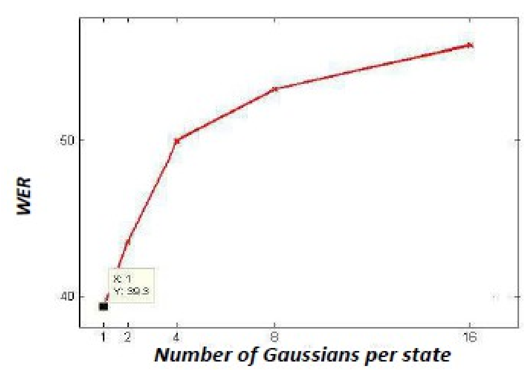

Figure 12. WER for Continuous Context Dependant models

We notice that the WER value is too high and increase with the number of Gaussians. We conclude that the amount of training data is insufficient for models training. Therefore these models are useless for this embedded system.

5.2.3 Continuous Context Independent Models

To reduce the number of models, we developed 35 context independent models and test them. Results are reported in the following figure.

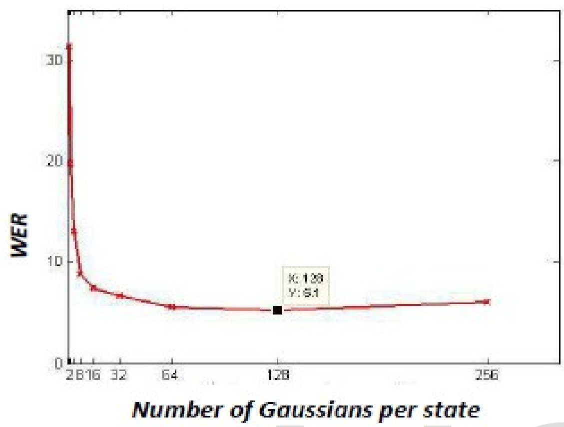

Figure13. WER for Continuous Context Independent models

The best results correspond to WER = 5,1 %. The variation of the memory size in desribed in Figure14.

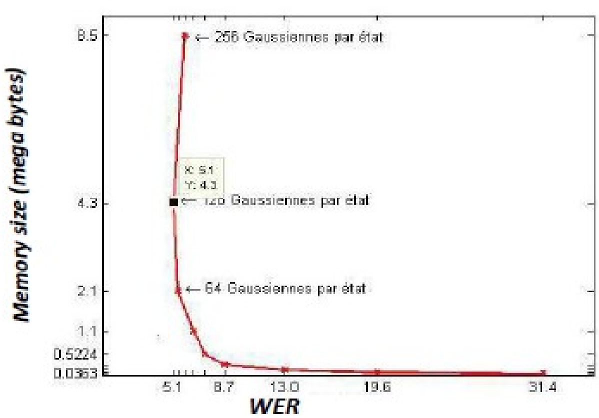

Figure14. Memory size for the Continuous Context Independent system

The memory used by the ASR system is limited enough (4.3 mega bytes < 5 mega bytes) to be implemented in the inner Box.
The main problem with these models is the computation time which is 2,9* RT for 128 Gaussians/state and 1,7 * RT for 64 Gaussians per state. So these models are also useless for the smart home command.

5.2.4 Semi-Continuous Context Independent Models

To reduce the computation time of the previous experiments we developed semi-continuous context independent models and varied the number of shared Gaussians. The results are reported below.

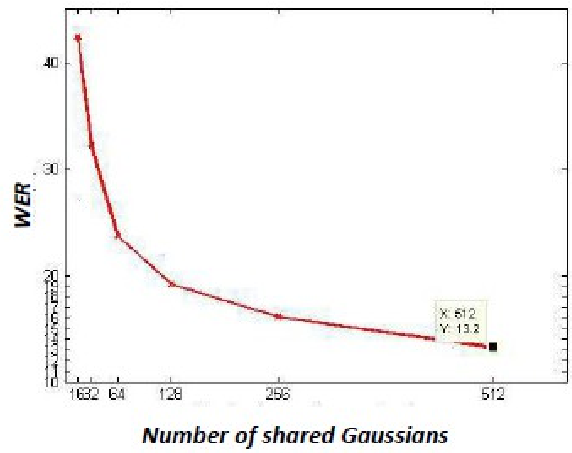

Figure 15. WER for Semi - Continuous Context Independant models

The memory size is described in Figure16.

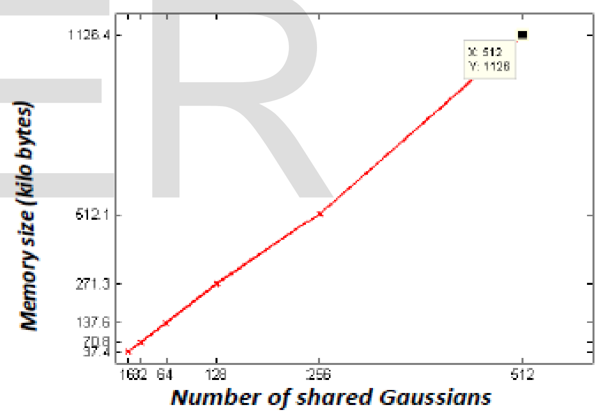

Figure 16. Memory size for the Continuous Context Independent system

The memory size is convenient but WER is about 10% which is useless for this application.

5.2.5 Semi-Continuous Context Dependent Models

To reduce the WER of the previous experiments we developed 1200 semi continuous triphones and varied the number of shared Gaussians. The results are below.

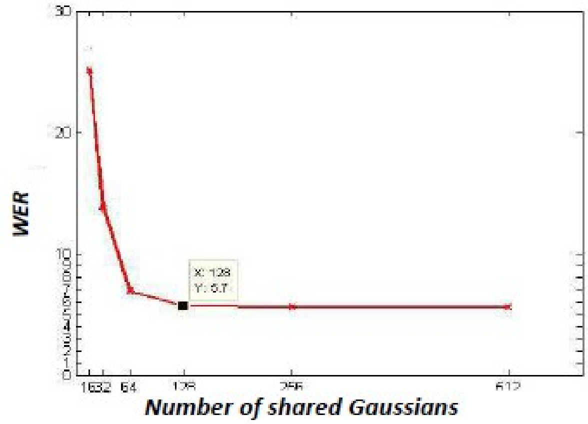

Figure 17. WER for Semi - Continuous Context Dependant models

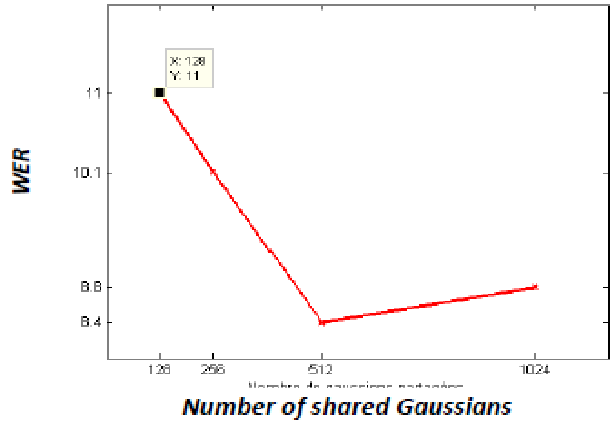

Figure19. WER for Multi-speaker speech recognition

Then we report the memory size in Figure18.

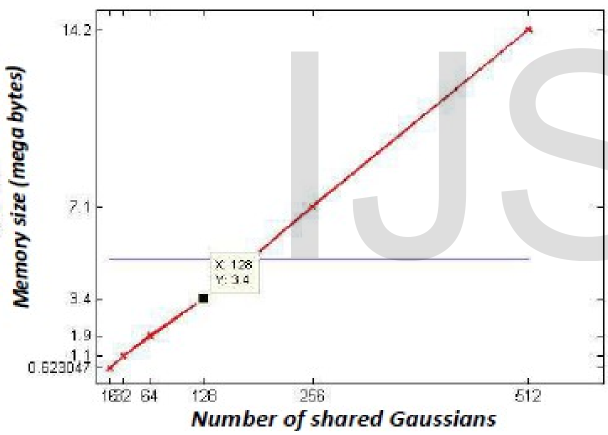

Figure 18. Memory size for the Semi - Continuous Context Dependent system

We notice that for 128 shared Gaussians, WER is low (5,1%), the memory size (3,4 Mega bytes) which is less than 5 mega bytes and the system turns in 0,2 * RT which is convenient for our application.

5.3 Multi-speaker speech recognition

We extend the previous experiments to multi-speaker database (5 men and 5 women). We develop 1200 semi-continuous context dependant models and vary the number of shared Gaussians. Results are reported in Figure19.

The best WER = 8,4 is obtained with 512 shared Gaussians but the memory size is more than 5 mega bytes. To improve these results, we developed models for women and models for men and apply gender detection.

One hour is used for women train and test models with cross validation and the same thing for men. The mean value of the men and women results is computed and plotted in the same scheme in red color.

Results are reported below:

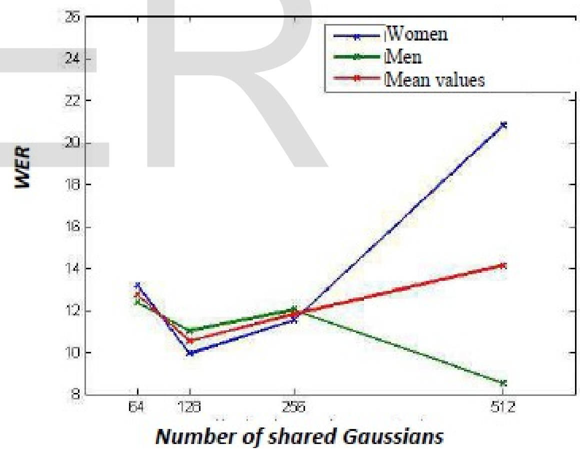

Figure 20. WER for gender based Multi-speaker speech recognition

Then we report the curve of Figure19 and the mean values of men and women (Figure20) in the same the figure (Figure21) to compare them.

We notice that using gender separate models is interesting only with 64 Gaussians per state. Otherwise, we must increase the database lengh to improve the speaker independent ASR system performance.

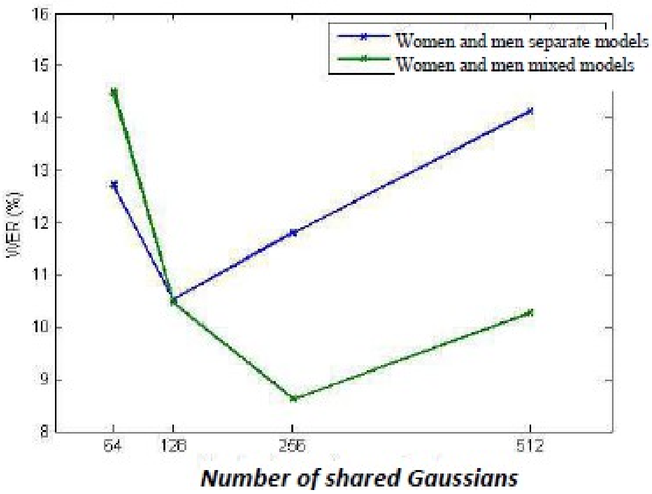

Figure 21. WER for Multi-speaker speech recognition (separate and mixed models)

6 Conclusion

This work concerns the development and the evaluation of an ASR and its integration in a smart home local box. The embedded system should detect continuously the speech signal, recognize the sentence and execute the corresponding command in less than real time.

After a market survey, three microphone devices were chosen: the headset, the webcam and the smartphone. For the smartphone, a vocal activity detection is algorithm have been implemented.

Further several realistic scenarios have been prepared, recorded and manually transcribed. Therefore speaker dependant and independent databases have been developed. The primary experiments shows that the WER headset based ASR is good enough even by using context independant models. For the webcam microphone, speaker dependant and speaker independent ASRs have been tested and evaluated by their WER, their speed and their memory size.

As reducing the number of Gaussians without decreasing the system performance is of major interest, after applying continuous context dependent and independent models we noticed that the semi continuous models improve the system speed without decreasing the performance. The best results for speaker dependant system with semi continuous context dependent models are : wer = 5,7%, it turns in 0,2 * Real time and its memory size is 3,4 Mega bytes.

Future work concern the increasing of speaker independent database to improve the speaker independent system.

Acknowledgment

This work has been conducted in the framework of “Maison intelligente” project with the collaboration of Chifco Company and Eniso Engineering School. The author thanks Amir Louati for his contribution of this work.

References

[1] Adda G., Chollet G, Essid S., Fillon T., Garnier-Rizet M., Hory C., and Beltaifa-Zouari L. Traitement des modalités « audio» et « parole » In Campedel, M. - Hoogstoel, P. - , Sémantique et multi modalité en analyse de l'information, Hermès, 2011, 143-188p. ISBN : 978-2-7462-3139-9.

[2] Aiyer, A., Gales, MJF., & Picheny, MA., Rapid Likelihood Calculation of Subspace Clustered Gaussian Components, International Conference on Acoustics, Speech, and Signal Processing, 2000, 1519-1522

[3] Chan, A., Ravishankar, M., & Rudnicky, A., On Improvements to CI based GMM Selection, European Conference on Speech Communication and Technology, Lisbon, September 2005.

[4] Chan, A., Sherwani, J., Mosur, R., & Rudnicky, A., Four Layer Categorization Scheme of Fast GMM Computation Techniques in Large Vocabulary Continuous Speech Recognition Systems, International Conference on Spoken Language Processing, Korea, 2004.

[5] Filali, K., Li, X., & Bilmes, J., Data-driven Vector Clustering for Low Memory Footprint ASR, International Conference on Spoken Language Processing, 2002

[6] Huggins-Daines David, Mohit Kumar, Arthur Chan, Alan W Black, Mosur Ravishankar, and Alex I. Rudnicky. POCKETSPHINX: A Free, Real-time continuous speech Recognition System for Hand-Held devices. ICASSP 2006

[7] Lee, A., Kawahara, T., Takeda, K., & Shikano, K., A New Phonetic Tied Mixture Model for Efficient Decoding, International Conference on Acoustics Speech and Signal Processing, 2001, 1269-1272

[8] Leppanen, J., and Kiss, I., Gaussian Selection with Non-Overlapping Clusters for ASR in Embedded Devices, International Conference on Acoustics Speech and Signal Processing, 2006

[9] Li, X., Malkin, J., & Bilmes, J., A High-speed, Low-Resource ASR Back-end based on Custom Arithmetic, IEEE Transactions on Audio, Speech and Language Processing, 2006

[10] Lingdell Mattias, “Embedded, Speaker Independent Speech Recognition of Connected Digits on Java ME Enabled Cell Phones.

[11] Milhoret Pierrick, Dan Istate, Jerôme Boudy and Gérard Chollet. Intéractions sonores et vocales dans l'habitat. JEP-TALN-RECITAL 2012

[12] Pellom, B., Sarikaya, R., & Hansen, J.H.L., Fast Likelihood Computation Techniques for Nearest-Neighbor Based Search for Continuous Speech Recognition, IEEE Signal Processing Letters, 2001, vol. 8, no. 8, 221-224

[13] Transcriber a tool for segmenting, labeling and transcribing speech Manuel du transcripteur. https://www.u-picardie.fr/LESCLaP/rey/ Tutoriel_transcriber.pdf

[14] Thilo w. Kohler, Christian Fuger, Sebastien Stuker and Alex Waibel. Rapid porting of ASR systems to mobile devices. Interspeech 2005.

[15] Vacher Michel, François Portet, Frédéric Aman, Benjamin Lecouteux, Solange Rossato, et al. Reconnaissance automatique de la parole dans les habitats intelligents : Application à l’assistance à domicile. 4è Journées Annuelles de la société Française des Technologies pour l’autonomie et de Géron technologie JASFTAG 2014. Paris France.

[16] Vashney Nikila, Sukhwinter Singh Embedded speech Recognition System. International Journal of advanced Research in Electrical, Electronics and Instrumentation Engineering IJAREEIE. Vol3, Issue4, April 2014.

[17] Zaykovskiy Dmitry, “Survey of the Speech Recognition Techniques for Mobile Devices,” SPECOM'2006, St. Petersburg, 25-29 June 2006

[18] Zouari Leila and Chollet Gérard. Efficient Codebooks for Fast and Accurate Low Resource ASR Systems. Speech Communication. March 2009, pages 732-743