A Vocal User Interface Plug-in for jMRUI

Rui Guimaraes, Theologos Athanaselis, Stelios Bakamidis, Ioannis Dologlou, Stavroula-Evita Fotinea
Institute for Language and Speech Processing (ILSP) / R.C. "Athena"
Artemidos 6 & Epidavrou, Maroussi GR 151 25
Athens, Greece
{rui; tathana; bakam; ydol; evita}@ilsp.gr

Abstract— **This paper presents a review of the necessary technology in order to develop a Vocal User interface to be integrated into the jMRUI [1]. jMRUI allows magnetic resonance (MR) spectroscopists to easily perform time-domain analysis of *in vivo* MR Data and might in the future be used during intraoperative MRI scanning. An operation room with an MRI scanner is a highly noisy environment which degrades speech recognition. These specific circumstances must be taken in consideration and are described in this paper along with the structure of the vocal interface.**

Keywords- VUI, Speech, Interface, jMRUI

I. INTRODUCTION

jMRUI is a software package developed and managed by FAST (Advanced Signal-Processing for Ultra-Fast Magnetic Resonance) Research and Training Network [1]. It is a graphical user interface that allows MR spectroscopists to easily perform time-domain analysis of *in vivo* MR Data [2]. More than 1400 research groups worldwide in 53 countries benefit from the MRUI software and the number keeps growing. The software is modular allowing several independent analyses. Besides the preprocessing and quantitation algorithms (Figure 1) users can add their own custom plug-in. The goal of the project described in this paper is to provide a Vocal User Interface (VUI) for several analysis allowed by this software. The vocal interface will allow the users to be free to operate other devices or other software while instructing jMRUI to perform the needed tasks.

The possibility that the software will be used for intra-operative MRI scanning allowing hands free analysis is also considered. These circumstances imply extra challenges to an Automatic Speech Recognizer (ASR). MRI acoustic noise can go as up as 120dB for a 3T system [3], therefore, robust denoising software will have significant influence on the ASR performance. This fact is exhibited later in this paper, detailed in the preprocessing section of the system structure. Also, the way the software is organized and how it is normally used may also influence the vocal command structure and definition. The challenge faced here is then to build a list of voice commands that can be used efficiently in compliance with the typical way a user may interact with the software. Therefore, the system presents challenges in each of its modules: sound capture, preprocessing, automatic speech recogniser (ASR) and commands (linguistic framework). Each of these modules is described in the sections that follow. The overall performance of the system depends on how each module is adjusted according to the results from the other modules. The full system will be implemented as a custom plug-in for jMRUI (Figure 1). This paper presents the work done so far to achieve the goals of this project and makes considerations on future work.

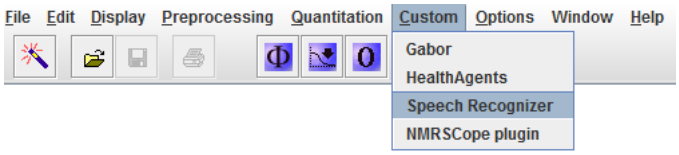

Figure 1- jMRUI showing custom plug- in for ASR

II. SYSTEM STRUCTURE

The overall performance of the system will depend on several parts that can be improved independently. The changes in one module of the system will require adjustment on the other module. For example the preprocessing software will depend on the quality of the noise cancelling hardware during the sound capture. The modules of the system are sketched below (figure 2).

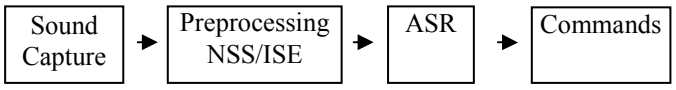

Figure 2- Schematic representation of system architecture

a. Sound Capture

The sound capture might need to take into consideration the presence of the MRI magnet. A fiber optic microphone would be ideal as it normally includes powerful noise cancelling. However, these microphones are expensive and some other user devices might be used as the MRI safe sound collector seen in [4].

b. Preprocessing

Given that the output of the sound capture may include immense acoustic noise depending on the device used,

978-1-4244-6494-4/10/$26.00 ©2010 IEEE

preprocessing is required to attenuate this noise. As reported in [5], non linear spectral subtraction (NSS) seems to perform better than Iterative Signal Enhancement (ISE) based on Singular Value Decomposition (SVD). The test consisted of 100 sentences uttered by male and female speakers. The speakers are native English (UK), with no speaking or hearing disability whatsoever. They read the test sentence in a natural and relaxed manner. In order to evaluate the speech recognition system performance regarding the word based confidence score, MRI noise is used. The difference becomes more significant at a Signal to Noise Ratio of 10 or 15 (dB) where improvement in confidence scores after denoising goes up 50%.

The results of the above reported experimentation are shown in Figure 3. The symbol (◊) of diamond represents the confidence score (%) achieved for input signals with different SNR values and no enhancement whatsoever. The symbol (Δ) of triangle corresponds to the confidence score (%) achieved for input signals where the non-linear spectral subtraction method is applied. The symbol (■) of square corresponds to the confidence score (%) achieved for input signals where the ISE method is applied. The confidence score seems to be unaffected by the noise reduction method when the SNR value is greater than 25dB. This can be explained knowing that NSS/ISE improvement is rather small for high SNR.

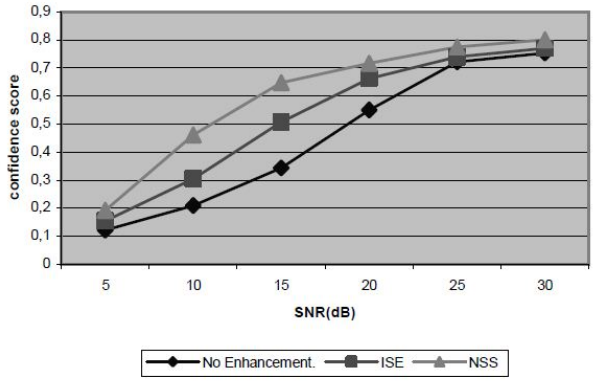

Figure 3- ISE vs NSS - confidence score comparison

1. *Speech enhancement based on non-linear spectral subtraction (NSS)*

In additive noise, the degraded speech can be described by

$$y[k] = x[k] + n[k] \tag{1}$$

where $x[k]$ and $n[k]$ represent the noise-free speech and corrupting noise sequences respectively. Assuming the speech and noise processes are uncorrelated, the relationship can be described in the short time power spectral domain by,

$$\left|Y_i(\omega_k)\right|^2 = \left|X_i(\omega_k)\right|^2 + \left|N_i(\omega_k)\right|^2 \tag{2}$$

Note that the index, i is used to represent the i - th windowed frame of speech.

Non-linear spectral subtraction (NSS) takes into account the frequency-dependent signal to noise ratio (SNR) of coloured noise. Here, the algorithm reduces subtraction for spectral components of high SNR and increases subtraction for spectral components of low SNR. In addition, the noise model includes an averaged noise spectrum. NSS enhancement can be expressed in terms of a filtering operation,

$$\left|\hat{X}_i(\omega_k)\right| = H_i(\omega_k) \bullet \left|Y_i(\omega_k)\right| \tag{3}$$

where $H_i(\omega_k)$ depends on a smoothed estimate of the noise-corrupted speech magnitude spectrum $\left|\ddot{Y}_i(\omega_k)\right|$, and non-linear subtraction term, $\Phi_i(\omega_k)$,

$$H_i(\omega_k) = \frac{\left|\ddot{Y}_i(\omega_k)\right| - \left|\Phi_i(\omega_k)\right|}{\left|\ddot{Y}_i(\omega_k)\right|} \tag{4}$$

The subtraction term, $\Phi_i(\omega_k)$, is given by

$$\Phi_i(\omega_k) = \frac{\max_{i-40\le\tau\le i}\left|\hat{N}_T(\omega_k)\right|}{1+\gamma\rho_t(\omega_\kappa)} \tag{5}$$

with $\rho_t(\omega_\kappa) = \dfrac{\left|\ddot{Y}_i(\omega_k)\right|}{\left|\ddot{N}_i(\omega_k)\right|}$, where γ is a constant dependent on the range of $\rho_t(\omega_\kappa)$. For practical purposes, the dynamic range varies between 1 to 3 times the smoothed noise magnitude estimate (i.e., $\left|\ddot{N}_i(\omega_k)\right| \le \Phi_i(\omega_k) \le 3\left|\ddot{N}_i(\omega_k)\right|$) and a noise-floor is established during subtraction [8].

2. *Outline Of The Truncated SVD Procedure*

Consider the clean speech signal $x[k]$ and the noise signal $n[k]$ (both unknown). If we assume the noise to be additive, we can write $y[k] = x[k] + n[k]$, where $y[k]$ corresponds to the recorded noisy signal. From the vector $\mathbf{y} = [y[0], y[1], \cdots, y[N-1]]^{\mathrm{T}}$ we can construct the Hankel matrix $\mathbf{Y} \in \Re^{L\times M}$,

$$\mathbf{Y}=\begin{bmatrix} y[0] & y[1] & \dots & y[M-1] \\ y[1] & y[2] & \dots & y[M] \\ \vdots & \vdots & & \vdots \\ y[L-1] & y[L] & \dots & y[N-1] \end{bmatrix} \quad (6)$$

with $L \geq M$ and $L = N+1-M$. Assuming that the clean signal $x[k]$ consists of a sum of p complex exponentials, then the Hankel matrix containing the clean signal is rank-deficient and has rank $p \leq M$. If $n[k]$ consists of broadband noise, the matrix **Y** will in general not be rank-deficient and will have rank M.

From the SVD of **Y** it is possible to construct a least-squares estimate of the Hankel matrix containing the clean signal. When we set the $M-p$ smallest singular values, corresponding to the noise, to zero and we only retain the p largest singular values, corresponding to the signal, we are able to construct the matrix $\mathbf{Y}_p$,

$$\mathbf{Yp} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} = U_1 \Sigma_1 V_1^T \quad (7)$$

which is the best rank-p approximation of the original matrix **Y**.

In general, the matrix $\mathbf{Y}_p$ does not have a Hankel structure. A simple procedure for restoring the Hankel structure is to average along the anti-diagonals of the matrix and to construct a Hankel matrix $\hat{X}$,

$$\hat{X} = \begin{bmatrix} \hat{x}[0] & \hat{x}[1] & \dots & \hat{x}[M-1] \\ \hat{x}[1] & \hat{x}[2] & \dots & \hat{x}[M] \\ \vdots & \vdots & & \vdots \\ \hat{x}[L-1] & \hat{x}[L] & \dots & \hat{x}[N-1] \end{bmatrix} \quad (8)$$

$$\hat{x}[k] = \frac{1}{\beta-\alpha+1} \sum_{i=\alpha}^{\beta} \mathrm{Y}_p(k-i+2, i) \quad (9)$$

$$\alpha = \max(1, k-L+2), \beta = \min(M, k+1) \quad (10)$$

Because of the averaging, the matrix $\hat{\mathbf{X}}$ in general does not have rank p any more. Still, because $\hat{\mathbf{X}}$ is closer to $\mathbf{Y}_p$ than the original matrix **Y**, the signal $\hat{\mathbf{x}} = [\hat{x}[0], \hat{x}[1], \dots, \hat{x}[N-1]]^T$ will be more compatible with the p–th order model than the signal **y**. It has been shown that for speech applications this simple procedure is indeed able to reduce additive noise [9].

a) Iterative Signal Enhancement Algorithm (ISE)

The relatively new algorithm that is known in the literature as ISE is based on the combination of the SVD of the input signal and an iterative process that each time extracts from the signal its most energetic spectral properties. The signal is processed frame by frame and for a given frame an *M*-th order SVD is computed.

The algorithm begins with the calculation of the rank-1 signal decomposition $s[k]$ from the corrupted signal $y[k]$. This calculation is realised by truncating the Hankel matrix to rank one and averaging the anti-diagonals. Then the residual signal $r[k]$ is computed by subtracting the signal decomposition $s[k]$ from the input signal $y[k]$. The procedure is repeated using the residual signal as input for the next iteration. The algorithm terminates when the energy of the residual signal is equal to the energy of the noise. The enhanced signal $\hat{x}[k]$ is obtained by accumulating the signal decompositions $s[k]$ over all iterations [9].

c. Automatic Speech Recognition - ASR

Confidence scores provide an estimate of the likelihood of the recognised words being correct. Although in the experiment sketched above which concerned denoising, this was used to derive conclusions, it is more reliable to use the word error rate by testing the full system with different sound capture devices. The ASR is also intended to be robust with respect to word order problem. For example, the voice command „rotate the voxel 10 degrees towards the frontal lobe“ and „rotate the voxel towards the frontal lobe 10 degrees“ should be translated into the same action.

Another point that merits attention is that, being jMRUI so popular across the globe, many of its users will not be native English speakers. This means that it should have a training feature to lower the word error rate. Optionally a mechanism can be set up where the user defines its own voice commands and their corresponding actions.

1. System overview

The ASR system is based on Hidden Markov Models (HMM) [6]. The unknown speech input is converted into a sequence of acoustic vectors $\mathrm{Y} = y_1, y_2, \dots, y_n$, by means of a parameter extraction module. The goal of the ASR system is to determine the most probable word sequence $\hat{W}$ given the observed acoustic signal Y, based on the Bayes' rule for decomposition of the required probability $\mathrm{P}(W \mid \mathrm{Y})$ into two components, that is,

$$\hat{W} = \arg\max_{W} P(W/Y) = \arg\max_{W} \frac{P(W)P(Y/W)}{P(Y)} \quad (11)$$

The prior probability $P(W)$ is determined directly from the language model. The likelihood of the acoustic data $P(Y \mid W)$ is computed using a composite HMM representing W constructed from simple HMM phoneme models joined in sequence according to word pronunciations stored in a dictionary. Figure 4 illustrates the main components of the speech recognition module.

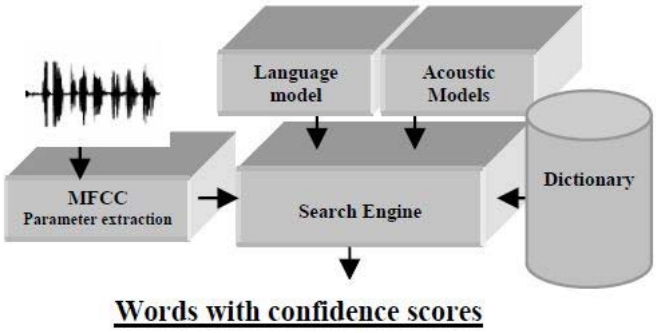

Figure 4- The architecture of the Speech Recognition engine

2. Parameter extraction

The prime function of the parameter extraction module is to divide the input speech into blocks and for each block to derive a smoothed spectral estimate. The spacing between blocks is typically 10 msecs and blocks are normally overlapped to give a longer analysis window of typically 25 msecs. A Hamming window weighting is applied to each block and Mel-Frequency Cepstral Coefficients (MFCCs) are used to model its spectral characteristics.

3. Acoustic models

The purpose of the acoustic models is to provide a method of calculating the likelihood of any vector sequence Y given a word *w*. For a small vocabulary system, and digit recognition systems, we can have whole word models and achieve good performance.

However for ASR systems this is impractical. In this case word sequences are decomposed into basic sounds called phonemes. Each individual phoneme is represented by an HMM. HMM phoneme models typically have three emitting states and left-to-right topology. For the English language, 45 phonemes are used to describe the pronunciation of all words. The corresponding HMMs were trained using the well known WSJCAM0 British English speech database comprising 8000 utterances (92 speakers, 90 utterances per speaker) (Robinson et al., 1995).

4. Language model

The language model used by the ASR system is the standard statistical N-grams [6]. The N-grams provide an estimate of $P(W)$, the probability of observed word sequence W.

Assuming that the probability of a given word in an utterance depends on the finite number of preceding words, the probability of N-word string can be written as:

$$P(W) = \prod_{i=1}^{N} P(w_i \mid w_{i-1}, w_{i-2}, ..., w_{i-(n-1)}) \quad (12)$$

N-grams simultaneously encode syntax, semantics and pragmatics and they concentrate on local dependencies. This makes them very effective for languages where word order is important and the strongest contextual effects tend to come from near neighbours. A statistical language model describes probabilistically the constraints on word order found in language: typical word sequences are assigned high probabilities, while atypical ones are assigned low probabilities. N-grams have also been chosen, because the N-gram probability distributions can be computed directly from text data, yielding hence no requirement to have explicit linguistic rules (e.g. formal grammars). The statistical language model of the ASR consists of bigrams (N=2) and trigrams (N=3).

5. Search engine

The basic recognition problem is to find the sequence of words that maximizes equation (11). The search engine used here applies beam search and Viterbi decoding. The branching tree of HMM-state nodes are connected by state transitions and word-end nodes are connected by word transitions. Any path from the start node to an arbitrary point in the tree is denoted by a movable token placed in the node at the end of the path. The score of the token is the total log probability up to that point, and the history of the token records the sequence of word-end nodes that the token has passed through. For every time frame, the best score in any token is noted and any token that lies more than a beam-width below this best score is destroyed.

d. jMRUI vocal operation command set

To improve the chances of success the voice command should be defined always as sets of 2 or more words. For example, the action „Open“ could be given as „Open File“. As the ASR exploits the n-gram probabilities the use of group of words can significantly reduce the word error rate using sets of words.

Given the complex interface of jMRUI it is important that the user has a simple set of commands that have a very low word error rate. Otherwise, the user might feel unmotivated to use the voice interface at all as it has happened in other systems. The base commands can then be used for other, more complex, according to a pyramid scheme as in Figure 5.

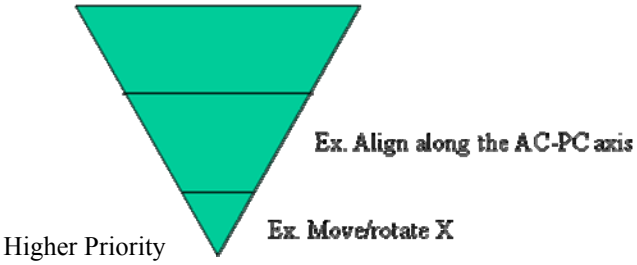

Figure 5 - Pyramid of commands

For example, the command "Align along the AC-PC axis" can converted in a series of commands "Move X" and "Rotate X" which can also be given by the users themselves.

Defining commands in this fashion improves the usability of the system. There will be a subset of commands that a user has a high guarantee that they will respond correctly. Given the complexity of the system, the commands with less priority will only be used by advanced and trained users.

III. Conclusion

A Vocal User Interface can provide a quicker access to all functionalities of a software package like jMRUI or even allow the user to do simultaneous work improving his or her experience with the software.

For example, in case of intra-operative MRI a doctor might want to make spectroscopy analysis while his or her hands are busy with medical instruments. Also, a voice command can be easier to remember than the key shortcuts. This voice command could be defined by the user.

The specific circumstances under which the proposed interface is used imposes extra challenges that can be dealt by using denoising algorithms and thoroughly thought set of commands. Correctly defining these commands is no easy task and must take in consideration how the choices of words affect the performance of the overall system. For example, it is important to use more than one word for each command. The final performance of the system will depend how each module is adjusted when it receives information from the predecessor. We have the freedom to do this adjustment as the modules for preprocessing and ASR are implemented in-house. Nevertheless, in the future some commercial packages for ASR can be tested for comparison.

The work reported in this paper can be replicated to other projects with similar challenges as, for example, helicopter piloting or healthcare equipment interfaces. The success of the system depends on adjusting its parts to a point it reaches high reliability (normally superior to 95% of correct recognised words).

Acknowledgments

This work has been partially supported by the European Project FAST-Advanced Signal Processing for Ultra Fast Magnetic Resonance Spectroscopic Imaging, and Training, Marie Curie Research Training Network, MRTN-CT-2006-035801.

The authors also wish to thank the Co-ordinator Danielle Graveron (Laboratoire Creatis-RMN, Universite Claude Bernard LYON I - UCBL, France) and CERMEP as well as Christian Labadie (Max Planck Institute for Human Cognitive and Brain Sciences, Germany) for providing the audio data.

References

[1] http://www.fast-mariecurie-rtn-project.eu/

[2] http://sermn02.uab.cat/mrui/mrui_Overview.shtml

[3] Price DL, De Wilde JP, Papadaki AM, Curran JS, Kitney RI, "Investigation of acoustic noise on 15 MRI scanners from 0.2 T to 3 T" J Magn Reson Imaging 2001;13:288–293.

[4] Lukkari, T., Malinen, J., and Palo, P. (2007). "Recording speech during magnetic resonance imaging," in "MAVEBA 2007," Florence, Italy, 163 – 166.

[5] T., Athanaselis, S., Bakamidis, I., Dologlou, E., Fotinea, (2009), "Impact of MRI scanner noise on Speech Recognition confidence score", in Proceedings of the 4rd Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, November 6-8, Poznań, Poland.

[6] Young, S.J. (1996) "Large Vocabulary Continuous Speech Recognition", IEEE Signal Processing Magazine 13, (5) (pp45-57).

[7] Athanaselis, T., Fotinea, S-E., Bakamidis, S., Dologlou, I. (2003), "Impact of speech enhancement on ASR confidence score", Proceedings of the 3rd European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems, EUNITE-2003 (pp548-553).

[8] M. Dendrinos, S. Bakamidis, G. Carayannis, "Speech enhancement from noise: A regenerative approach", Speech Communication, Vol. 10, no.2, February, 1991, pp. 45-57.

[9] S. Doclo, I., Dologlou, M. Moonen, "A novel iterative signal enhancement algorithm for noise reduction in speech", Proceedings of ICSLP-98, Sydney, Australia, 1998, pp. 1435-1439.

[10] C. Kyriakou, S. Bakamidis, I. Dologlou, G. Carayannis, "Robust Continuous Speech Recognition in the Presence of Coloured Noise", Imaging Systems and Techniques, 2008. IST 2008. IEEE International Workshop on , vol., no., pp.349-352, 10-12 Sept. 2008