Advances in Electrical and Computer Engineering **Volume 11, Number 3, 2011**

Self-Organizing Maps For Identifying Impaired Speech

Ovidiu GRIGORE , Valentin VELICAN
Politehnica University of Bucharest, 061071, Romania
Department of Applied Electronics and Information Technology
Electronics and Telecommunications Faculty
ovidiu.grigore@upb.ro

***Abstract*—This paper presents a method of identifying heavily impaired pronunciations of 'r' consonant in Romanian language using Kohonen neural networks. The study focused on words that contain 'r' as the first letter and used signals recorded mainly from children, as mispronunciations occur most of the time at young age persons. Parameters of the alternating component of each speech sample's envelope are used as feature vectors in the classification stage..**

***Index Terms*—speech processing, impaired speech evaluation, self-organizing maps.**

I. INTRODUCTION

The increasing number of children with impaired speech problems is reflected in the need of developing new and more efficient therapy methods that can supplement the speech specialist.

Due to the fact that speech therapy requires many visits to the specialist, an efficient automated method for "home training" would be more convenient in many cases. Unfortunately, automated methods developed around automatic-speech-recognition (ASR) algorithms are sparse and still in the research stage, mostly due to the characteristics of the sounds analyzed and also due to the large amount of pronunciations defects existing [1], [2]. These generate medium performance in the recognition task and the goal to reach human level performance has not been yet achieved. Also, even though there are several projects trying to build automatic recognition systems customized for specific languages, e.g. [3] for Spanish, [4] for English, [5] for American sign language (!), to our knowledge there is only one other project except ours [6], [7], [8] that tries to build a good ASR based system for Romanian language.

Therefore we concentrated our efforts on developing the very first methods that can be applied in identifying mispronunciations in Romanian. Due to the large number of defects, as stated before, for the beginning we only focused in analyzing mispronunciations of 'r'. This paper presents the architecture and the performances of an already proposed feature extraction technique applied to the mentioned phoneme but using a different classifier.

In [9] we have proposed the extraction of the alternating component of the signals' envelope as a possible method of making difference between correct and incorrect pronunciations of 'r'. We also tested a k-NN classifier [10], [11], [12] and revealed that a reasonable level of performance can be achieved using our method. In this paper we use the Kohonen neural network as a classifier and we will compare the level of performance achieved by the SOM with those obtained by using the k-NN algorithm, in [9].

Moreover, using the topological property of the Kohonen Map, we introduce a new method of the impaired speech analysis, putting in evidence that the different kind of defects existing in speech pronunciation can be correct estimated by the location of the Kohonen output layer winner neuron.

II. PROBLEM FORMULATION

There are many issues to address in our study, each one with its own importance and impact on the overall result. But what has to be stressed, from the very beginning, is that the vast complexity of the subject – impaired speech assessment – can only be overcame by fragmenting the problem and studying it step by step. By doing so, one should be able to construct a more robust 'release version' of the algorithm that can address all the different mispronunciation types in existence. Therefore, this article only presents a method of identifying mispronunciations of 'r' consonant, maybe the most common verbal defect in the Romanian language.

Secondly, the most speech processing algorithms used nowadays in different applications (e.g. speech recognition, speaker identification) are based on vowels as feature extraction source, for the very simple reason of packing more energy and, consequently, more information than consonants [13], [14] , [15], [16].

Unfortunately from the speech processing procedure point of view, the most known classified pronunciation defects are generally related with problems in consonants uttering (e.g. not knowing how to do it, or the inability to do it), meaning that identifying specific features for the mispronunciation evaluation task, is not only original, but also considerably more difficult to deal with, being necessary to develop new feature extraction algorithms for consonants analysis.

III. PROBLEM SOLUTION

The 'r' phoneme is in Romanian language a hard rhotic consonant, being also one of the most commonly mispronounced sounds. The defect is generally known as "rhotacism" and it can be observed from young age children to adults. In the worst possible utterances, the 'r' is replaced by other sounds, like /l/, /î/ or can completely miss from

This work was supported by CNCSIS–UEFISCSU, project number 846/2009.

Digital Object Identifier 10.4316/AECE.2011.03007

words [17], [18] In the mild, linguistic acceptable mispronunciations, the phoneme is replaced with a guttural ‘r’, resembling the pronunciation of ‘r’ in French.

The problem of identifying particular features for correct / wrong pronunciations of this consonant proved to be difficult and after several failed attempts of frequency and cepstral analysis we decided to search more into the time domain representation of the signal. We then noticed the difference in shape of the envelope of correct and wrong pronunciations of ‘r’. What has been worked in order to represent this characteristic is presented below, structured on dedicated paragraphs for each important step.

In the end of the study we concentrate on implementing a Self Organizing Map [19] capable of presenting similarities between features extracted from voice recordings, according to our method.

A. Database of Recordings

The database used in the study was recorded from 15 children and 5 adults pronouncing words starting with ‘r’ like *rac, rană, ramă, raţă,* etc. (representing the Romanian words for the English: crab, wound, frame, duck). It was taken care to collect sounds that have the same “*starting consonant – vowel*” combination as transitions from the consonant to the vowel influence the wave shape of the recorded signal [17].

From the entire batch of 65 input signals, 25 different recordings (15 correct and 10 incorrect pronunciations) were selected for the ‘learning stage’ of the classification algorithm. This learning set is important as the Kohonen network uses feature vectors extracted from it in order to establish the criteria by which other unknown samples are differentiated in the testing session. Care was taken when selecting the correct samples in order to construct a collection with phonemes as closely as possible to the correct pronunciation rules established by specialists. It is also important to mention that all of the adults had a correct pronunciation while only three children correctly pronounced words containing ‘r’.

The test database was comprised of 20 correct and 20 incorrect samples of ‘r’, different from the samples in the upper mentioned ‘learning set’

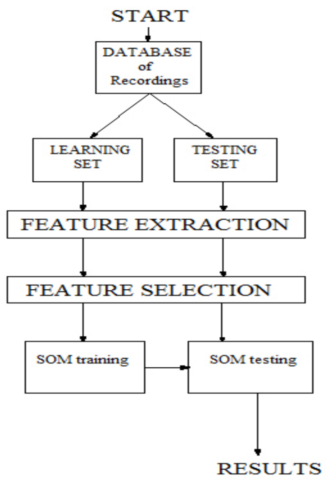


Figure 1. The processing block used in studying the feature extraction methods for identifying correct / incorrect ‘r’ pronunciations.

B. Feature Extraction

The core of this study is the feature extraction method. To stress the importance of this stage of every speech processing application in general and in our case in particular, we present a theoretical sequence of steps defined by the blocks in Fig. 1 that have been followed by this paper.

Studying the shape of the signal in time domain it was observed that correctly pronounced ‘r’, independent from the subject’s age, sex or voice timbre, resembled very well amplitude modulated (AM) signals with an envelope frequency of about 25 – 30Hz. Unlike this, all other phonemes and, of course, the incorrectly pronounced ‘r’ lacks the AM shape of the signal, resembling in most cases with vowels, but with less amplitude – Fig. 2.

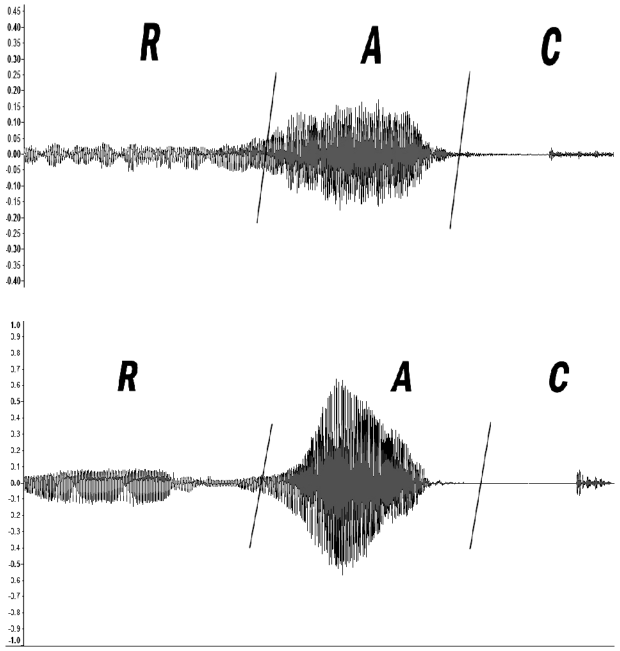


Figure 2. Correctly (above) and incorrectly (below) pronounced ‘r’ in ‘rac’. (The samples belong to kindergarten children)

It becomes obvious that a good feature by which correctly pronounced ‘r’ phonemes differ from their incorrect counterparts is the shape of the signal in the time domain. More exactly, the clearest difference between these two classes is the signal’s envelope.

Extracting the signal’s envelope in a software manner can be done in several ways: by using the Hilbert Transform or by using the square of the signal and then filtering the result, etc. For this study we have chosen another approach, extracting the maximum signal value for every N sample intervals of length L, where:

$$N = \frac{length\_(Signal)}{L} \tag{1}$$

and then linearly interpolating the resulting points, knowing that a straight line that contains $P_1(x_1,y_1)$ and $P_2(x_2,y_2)$ satisfies the equation:

$$(y - y_1) = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) \tag{2}$$

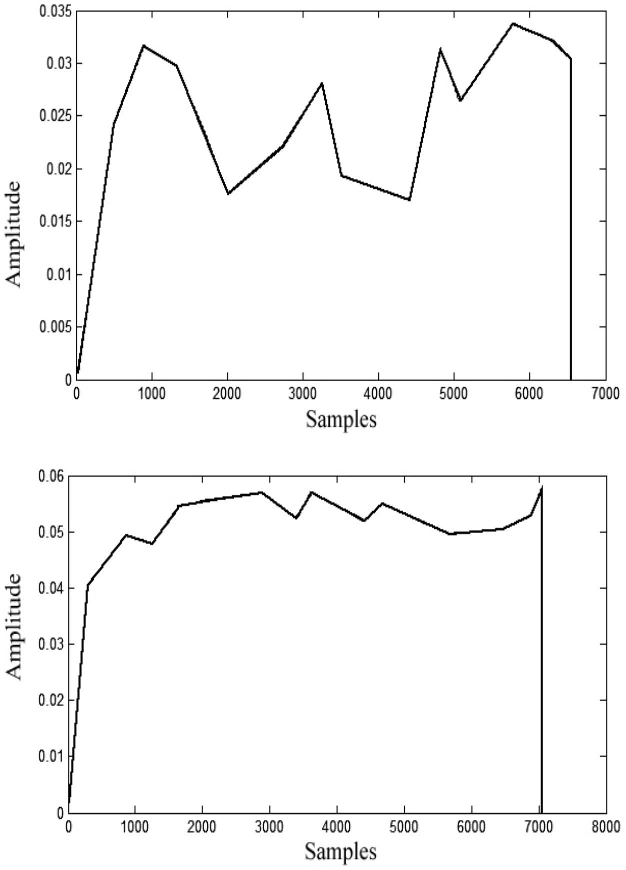


Figure 3. Envelope extraction of a correctly (above) andan incorrectly (below) pronounced 'r' in 'rac'. The graph is plotted in samples versus amplitude.

The result of applying the above "envelope detector" to both a correct and an incorrect signal can be seen in – Fig.3. It is important to notice that the processed signals have the same number of samples as the original ones due to the interpolation process. In order to eliminate a potential error that can appear in the classification process due to the signals' difference in amplitude, the extracted envelopes are being normalized to one by dividing each sample in the current analyzed signal window to the maximum contained sample value.

At this point, and as expected, the shape of the envelope for the correctly pronounced signals contains large variations (described more or less as sinusoidal) whereas in the other case, the envelope's amplitude is more constant (less variations) or it modifies slowly in time.

To use this observed characteristic, we first decided to extract from the signal's envelop the short-time continuous component by using a constant sized moving window. This operation is described by the following expressions:

$$vectAvg(k) = \frac{1}{M} \cdot \sum_{u=k}^{M-1+k} Envelope(u) \qquad (3)$$

where: k = 1, 2, 3…

To be noticed that when:

$$k = length\_Envelope - M + 2 \qquad (4)$$

the averaging window of length *M* overshoots the end of the envelope. In this case, and not from scientific considerations, but only because of potential implementing errors, we decided to reduce the length of the window (*M*) by one unit for every increase of *k* such that the last *M – 1* values of *vectAvg* will be computed using *M –1, M –2,…, 1* values from the envelope.

Then the feature value is extracted in the form of:

$$vectFeat(k) = \left(vectAvg(k) - Envelope(k)\right)^2 \qquad (5)$$

where: *k = 1, 2, 3…*

Going a bit more into detail, (3) defines the local continuous component of the signal or the local mean value. Fig. 4 presents this next step in processing the signals of Fig. 3. Now, the amplitude variation of the correct pronunciations becomes even more evident compared to the incorrect case. This is due to the fact that (3) overpasses the imperfections generated by the "envelope detector" and filters most of the spikes in the signal. Equation (5) defines the energy of the alternating component of the signal's envelope or, better said, a measure of how much does the envelope varies in amplitude with time – Fig. 5. Naturally, correctly pronounced samples of 'r' will have higher values in *vectFeat* than wrong pronunciations. The idea to use this method of extracting the continuous and alternating component of the signal, came from [20], [21] where *SCR – skin conductance response* and *SCL – skin conductance level* are none other (from a mathematical point of view) than the alternating / continuous component for a physiological signal analyzed.

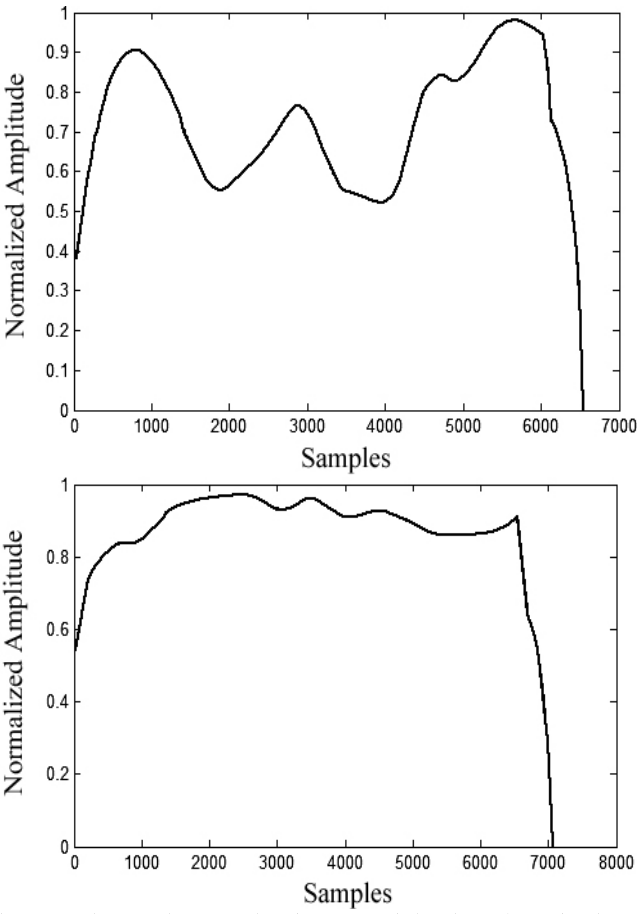


Figure 4. First step in processing the extracted signal envelope: the short-time continuous component. Above: a correct pronunciation of 'r' in 'rac'. Below: an incorrect pronunciation of 'r' in 'rac'.

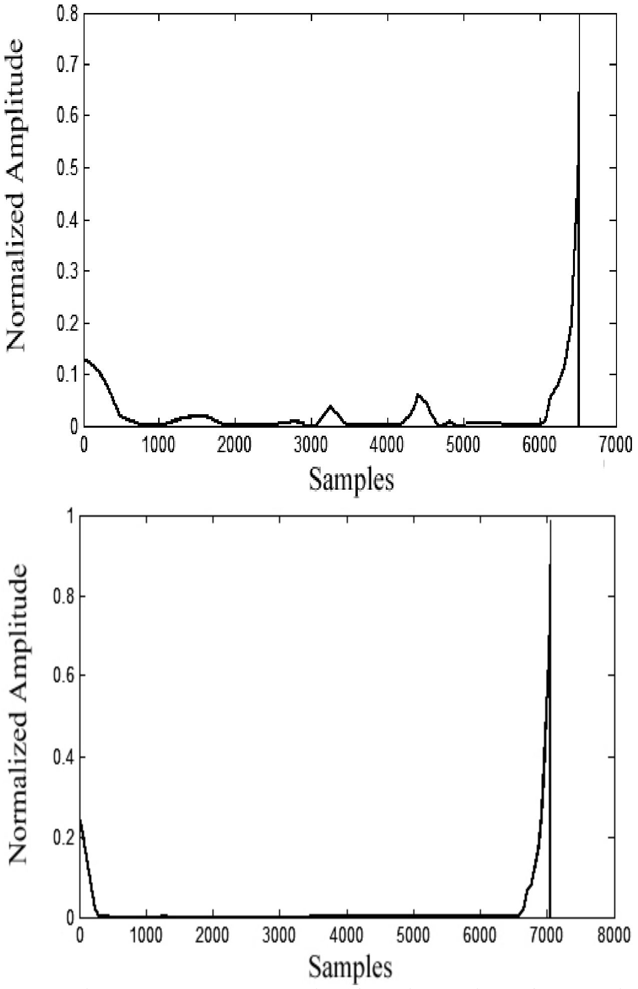


Figure 5. Alternate component extraction, from the envelope of a correctly (above) and an incorrectly (below) pronounced ‘r’ in ‘rac’.

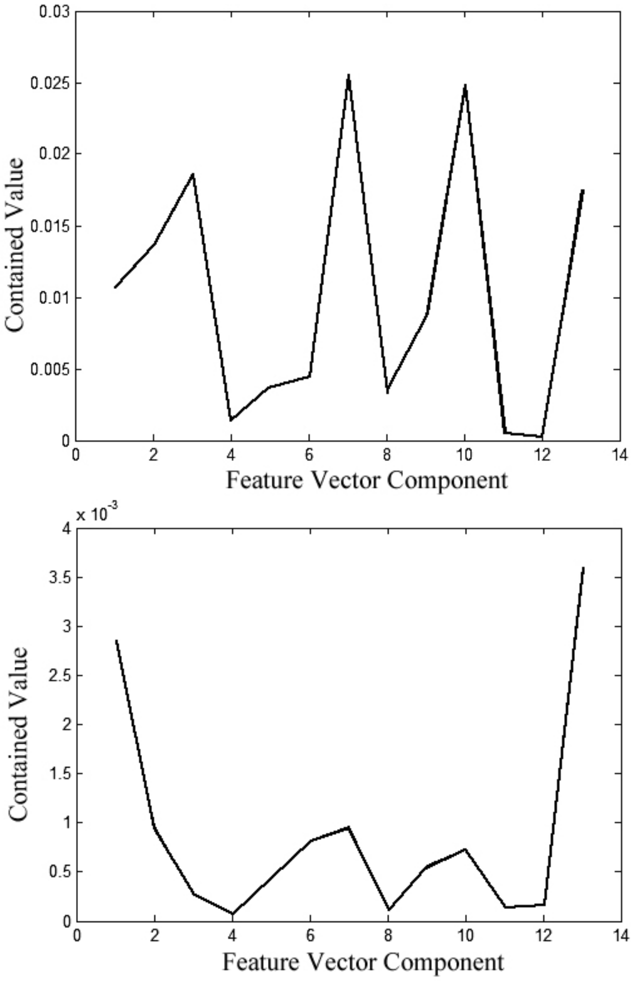


Figure 6. Feature vectors extracted using Eq.6 for a correct (above) and an incorrect (below) occurrence of ‘r’ in ‘rac’ (Romanian for ‘crab’).

Yet, due to the fact that *vectFeat* is still a very high dimension vector (most of the ‘r’ phoneme recorded are in excess of 5000 samples), running the classifier at this stage is not computationally efficient. Moreover, input signals have different lengths generating feature vectors of different lengths, all ending in a situation that is not desirable: the classifier has to test feature vectors of different lengths. Therefore a *feature selection* stage is implemented by dividing each *vectFeat* in the database into *INT* intervals of equal *sizeINT* length and calculating the mean on every such interval – this being equivalent with computing the power of alternating component of the signal’s envelope, i.e. the normalization of the above estimated energy in Eq.(5):

$$vectMean(i) = \frac{1}{sizeINT} \sum_{k=(i-1)sizeINT+1}^{i-sizeINT} vectFeat(k) \qquad (6)$$

where $i = 1, 2, \dots, INT$

What can be noticed is that *vectMean* is now a feature vector of equal length for any input signal, that contains the information of *vectFeat* condensed in much less coefficients, being more computational efficient to process them in the classification stage. (Fig.6)

C. Data Analysis / Classification

The aim of this stage was to observe relationship between input data (features of the speech samples). By clustering the pronunciations we intended to discover how closely related are the different input samples and how the speech samples group together relative to theirs correctness.

We decided to use a Kohonen Neural Network (also known as a Self Organizing Map - SOM), one of the best known data clustering algorithms and in addition we used the network as a classifier and compared the results to the results obtained in [9].

A SOM is an unsupervised neural network able to produce a representation of the input features in a lower dimensional space, called map. What is important is that the map consists of neurons that are continuously tweaked in the learning stage, to resemble as much as possible the topology of the input data. Also neighboring nodes resemble each other defining regions of resemblance and making the map a good data classifier for subsequent queries.

Briefly, the SOM consists of a collection of nodes organized in two layers. The input layer is a one dimensional array of length *INT* corresponding to the length of each input feature vectors x. The output layer is an NxN array of neurons. Each node in the map is linked to all the input layer nodes by a weight vector w_{ij} – Fig.7.

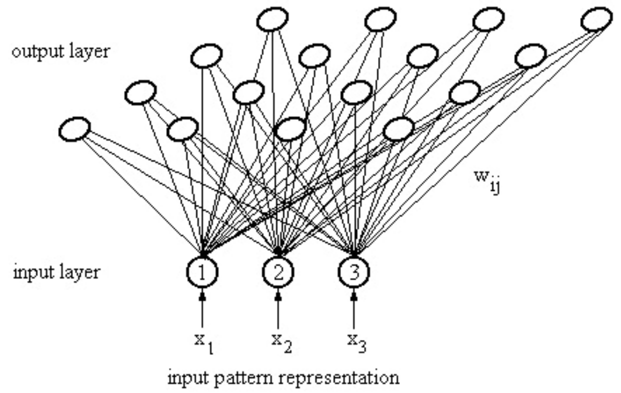


Figure 7. Representation of a Self Organizing Map (a Kohonen Neural Network)

All the weights arriving at the same output neuron *j*, forms the prototype vector associated to that neuron, noted as w_j

The learning process, responsible for the computation of the organized map follows the algorithm:

1. Initialize all weights w_{ij} with random values similar to those in the input space.
2. for each learning epoch $t = 1$ to *S*:
   - randomly extract an input $\boldsymbol{x}(t)$, from the learning data set
   - compute a measure of resemblance (in our case Euclidian distance) to each node (neuron) in the SOM
   - find the winner neuron (the closest match to the current input) – the winner is defined by Eq. (7)

$$\|\boldsymbol{x}(t)-\boldsymbol{w}_{winner}(t)\| \le \|\boldsymbol{x}(t)-\boldsymbol{w}_i(t)\| \qquad \forall i \neq winner \quad (7)$$

   - tweak the associated w_{ij} using Eq. (8) for both the winner neuron and the neighboring nodes that are in the boundaries defined by Eq.(9)

$$w_{ij}(t+1) = w_{ij}(t) + f_{c(x),i}(x_i(t) - w_{ij}(t)) \qquad (8)$$

$$f_{c(x),i} = \alpha(t)\exp\left(\frac{\|r_i - r_c\|^2}{2\sigma^2(t)}\right) \qquad (9)$$

   - adjust the function that defines the neighboring region for the winner node (fc(x),i); the function, usually a Gaussian, has the width defined by σ(t); the latter is monotonically decreased as the iterations progress.
   - adjust the learning rate α(t); it too, monotonically decreases as t increases.

D. Experimental Conditions

The experiment included a *learning stage* in which a SOM is trained and a *testing stage* in which it is observed how samples of voice are mapped. Both the learning and the training data are part of the database of recordings described in section III.A.

The network output layer was dimensioned as a 3 x 3 array of neurons, the input layer having the theoretical length *INT* – Eq.6. Going back now to Fig.5: it is important to say that due to the existence of high values at the beginning and especially at the end of *vectMean*, for the classification stage, we only used the mean values computed for intervals 2 to *INT*-2, as was discussed in section… These high values appear as a result of applying Eq.(4).

Several tests were done for every set of parameters in order to observe how samples distribute themselves on different SOMs generated.

More details on the input parameters and the debate on the results are presented in the following section.

E. Results

The most important input parameters used during the test session are described below:

- ‘*Ages*’ represent the number of iterations that passes in order to train the SOM. *Ages = S.*

- ‘*Map Size*’ is the total number of neurons in the SOM output layer.

- ‘*L*’ (samples) is the dimension of the processing window used in the phonemes envelope extraction stage

- ‘*M*’ (samples) is the number of samples in the moving average window; this is necessary when computing the alternate component of the signal

- ‘*INT*’ is the number of intervals used in the feature selection stage – Eq.(6)

Some representative output maps are presents in Fig.8, but more detailed results are presented in Table I. What can be noticed is that the bulk of the incorrect phonemes tend to concentrate on a single neuron of the map whereas correct pronunciations find matches in more neurons.

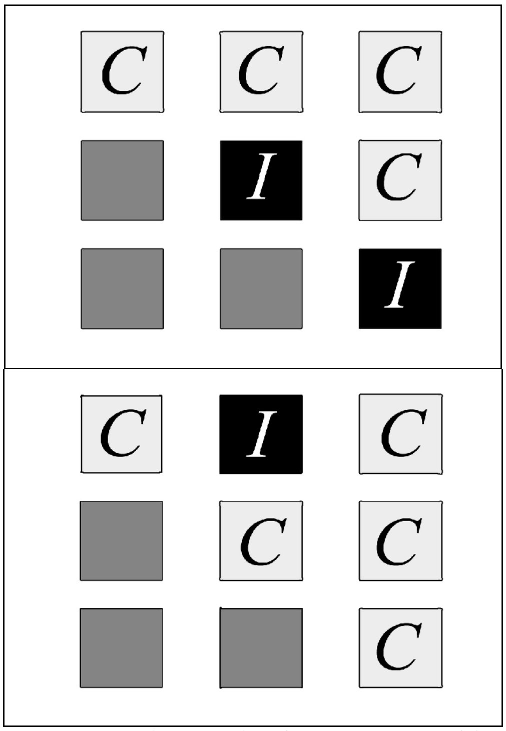
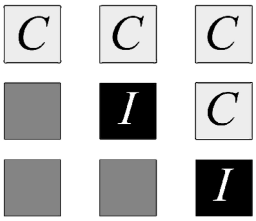
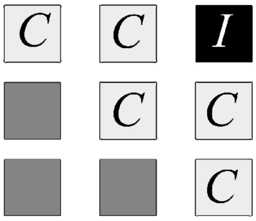
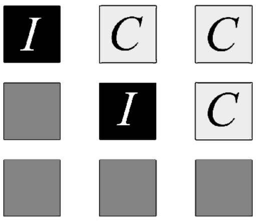
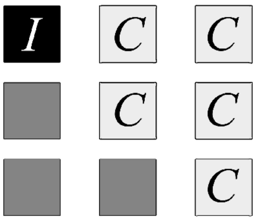
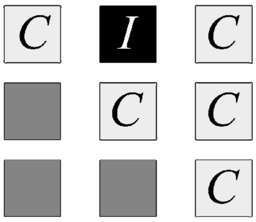


Figure 8. Two representative examples of 3x3 output maps of the Kohonen neural network.
Dark Gray = nodes unassigned
Light Gray = nodes assigned as ‘correct’
Black = nodes assigned as ‘incorrect’

TABLE I. TEST RESULTS

Note: *Black* – node assigned as incorrect, after the training process
Light Gray – node assigned as correct, after the training process
Dark Gray – node unassigned after the training process.

Test No.	Processing and NN parameters	Output neurons class labels assigned after the learning session	Correct Class. Rate	Number of output neurons activation when ***correct pronounced*** samples from the test set are applied at the input	Number of output neurons activation when ***incorrect pronounced*** samples from the test set are applied at the input
1	S = 1000 M = 400 L = 600 INT = 12		80%	5 2 7 0 1 2 0 0 3	0 4 0 0 2 0 0 0 14
2	S = 1000 M = 400 L = 600 INT = 12		67.5%	1 2 4 0 7 2 0 0 4	1 0 11 0 1 2 0 0 5
3	S = 1000 M = 400 L = 600 INT = 12		75%	5 5 5 0 0 5 0 0 0	2 0 0 0 13 3 0 0 2
4	S = 1000 M = 400 L = 600 INT = 12		65%	3 0 3 0 3 7 0 0 4	9 4 3 0 3 0 0 0 1
5	S = 1000 M = **600** L = 600 INT = 12		82.5%	3 3 2 0 3 4 0 0 5	0 16 0 0 0 2 0 0 2

6	S = 1000 M = 600 L = 600 INT = 12	C C C ▨ I C ▨ ▨ I	70%	$\begin{matrix} 3 & 4 & 1 \\ 0 & 3 & 3 \\ 0 & 0 & 6 \end{matrix}$	$\begin{matrix} 0 & 1 & 1 \\ 0 & 6 & 1 \\ 0 & 0 & 11 \end{matrix}$
7	S = **1500** M = 600 L = 600 INT = 12	C C I ▨ C C ▨ ▨ C	77.5%	$\begin{matrix} 3 & 1 & 6 \\ 0 & 0 & 5 \\ 0 & 0 & 5 \end{matrix}$	$\begin{matrix} 0 & 0 & 17 \\ 0 & 0 & 1 \\ 0 & 0 & 2 \end{matrix}$
8	S = 1500 M = 600 L = 600 INT = 12	C I C ▨ C C ▨ ▨ C	72.5%	$\begin{matrix} 2 & 9 & 1 \\ 0 & 4 & 4 \\ 0 & 0 & 0 \end{matrix}$	$\begin{matrix} 0 & 18 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{matrix}$
9	S = 2000 M = 600 L = 600 INT = 12	C I C ▨ C ▨ ▨ ▨ C	72.5%	$\begin{matrix} 3 & 7 & 4 \\ 0 & 2 & 2 \\ 0 & 0 & 2 \end{matrix}$	$\begin{matrix} 0 & 18 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}$
10	S = 2000 M = 600 L = 600 INT = 12	C C I ▨ C ▨ ▨ ▨ C	70%	$\begin{matrix} 4 & 3 & 6 \\ 0 & 3 & 2 \\ 0 & 0 & 2 \end{matrix}$	$\begin{matrix} 4 & 0 & 16 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{matrix}$

Let us remember that the paper aimed at running a classification test, also. This was done by running the learning set on a just computed map and labelling the nodes as *correct* or *incorrect* by counting for each neuron the inputs it classified. It can be seen that the results are quite promising: up to 82.5% correct classification rate compared to 82% at top in [9]. Furthermore, a best classification rate is obtained when the number of training ages is around 1000 to 1500. Above or below that, the classification rate decreases slightly.

IV. CONCLUSION

The paper presented a method of extracting features for voice samples containing ‘r’ phoneme and it compared different pronunciation, both correct and incorrect, using a Kohonen neural network.

Due to the large amount of defects existing and dissimilarities in the voice timbre, accents or language characteristics the method only applies to severely wrong pronounced ‘r’ phoneme in Romanian, where the consonant is heavy altered or replaced with a different sound.

The method is concentrated on extracting the alternate component of a recorded speech sample.

It has been observed that the alternate component is much higher in correctly pronounced ‘r’ compared to an incorrect pronunciation.

The database on which test have been conducted contained speech samples from fifteen children and five adults pronouncing words that start with ‘ra-’.

Results of the test showed that wrongly pronounced phoneme tend to organize on a single neuron whereas the correct phoneme are spread on more nodes of the map.

Using the SOM as a classifier, we obtained a correct classification rate of up to 82.5%, better than the results obtained in [1], [2] or [9] – other studies of the authors.

REFERENCES

[1] O. Grigore, C. Grigore, V. Velican, “Intelligent System for Impaired Speech Evaluation”, in *Proceedings of the International Conference on Circuits, Systems, Signals,* 10/2010, ISSN: 1792-4324, pp. 365-368. Available at: http://www.wseas.us/e-library/conferences/2010/Malta/CSS/ CSS-60.pdf

[2] O. Grigore, C. Grigore, V. Velican, “Impaired Speech Evaluation using Mel-Cepstrum Analysis”, in *International Journal Of Circuits, Systems And Signal Processing*, ISSN: 1998-4464, pp. 70-77. Available at: http://www.naun.org/journals/circuitssystemssignal/19-537.pdf

[3] C. Vaquero, O. Saz, W. Rodriguez, E. Lleida, “Human Language Technologies for Speech Therapy in Spanish Language”, Available at: http://dihana.cps.unizar.es/~alborada/docu/2008cvaquero2.pdf

[4] G. Potamianos, C. Neti, “Automatic Speechreading of Impaired Speech”, in *Proceedings of the Audio-Visual Speech Processing Workshop*, Scheelsminde, Denmark, 2001. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.23.2169&rep=rep1&type=pdf

[5] T. Starner, J. Weaver, A. Pentland, “Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video”, in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, no. 12, december 1998, pp. 1371 - 1375. Available at: http://luthuli.cs.uiuc.edu/~daf/courses/Signals%20AI/Papers/HMMs/00735811.pdf

[6] S.-G. Pentiuc, I. Tobolcea, O. A. Schipor, M. Danubianu, M.D. Schipor, “ Translation of the Speech Therapy Programs in the Logomon Assisted Speech Therapy System”, in *Advances in Electrical and Computer Engineering*, Vol. 10, no.2, 2010. Available at: aece.ro/displaypdf.php?year=2010&number=2&article=8

[7] O. A. Schipor, S.-G. Pentiuc, M.D. Schipor, “Improving Computer Based Speech Therapy Using a Fuzzy Expert System”, in *Computing and Informatics*, Vol. 22, 2003. Available at: http://www.eed.usv.ro/~schipor/publications/10_1.pdf

[8] M. Danubianu, S.-G. Pentiuc, O.A. Schipor, M. Nestor, I. Ungureanu, “Distributed Intelligent System for Personalized Therapy of Speech Disorders”, in *The Third International Multi-Conference on Computing in the Global Information Technology*, 2008, Athens, Greece. Available at: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4591364

[9] O. Grigore, V. Velican, “Simple Method of Identifying Impaired Speech”, in *Proceedings of the International Conference on Multimedia and Signal Processing* 2011, to be published

[10] L. Baoli, Y. Shiwen, L. Qin, “An Improved *k*-Nearest Neighbour Algorithm for Text Categorization”, in *Proceedings of the 20th International Conference on Computer Processing of Oriental Languages,* Shenyang, China, 2003. Available at: http://arxiv.org/ftp/cs/papers/ 0306/0306099.pdf

[11] X. Wu, V. Kumar, J.R. Quinlan et al. “Top 10 algorithms in data mining” – survey paper. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.144.5575&rep=rep1&type=pdf

[12] L. Haifeng, L. Shousheng, S. Zhan, “An Improved kNN Text Categorization on Skew Sort Condition”, in *2010 International Conference on Computer Application and System Modeling (ICCASM)*, Available at: http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5620491

[13] L. Rabiner, R. Schafer *Digital Processing of Speech Signals*, Prentice Hall, 1978

[14] M. Grimm, K. Kroschel, *Robust Speech Recognition and Understanding*, I-Tech Education and Publishing, 2007

[15] D. Jurafsky, J.H. Martin, *Speech and Language Processing,* Prentice Hall, 2000

[16] F. Mihelic, J. Zibert, *Speech Recognition*, IN-TECH, 2008

[17] D.V. Popovici, C. Buica-Belciu, A. Iordan, “Particularitatile Fonetice ale Pronuntiei Copiilor Dislalici”, in *O Scoala Deschisa,* 2/2009, Ed.SS6SN, pp. 116-124.

[18] I. Moldovan, *Date privind raportul dintre capacitatea de pronuntare si cea de diferentiere la palatolalici. Elemente de psihopedagogia handicapatilor.* Bucuresti, Tipografia Universitatii din Bucuresti, 1990.

[19] T. Kohonen, *Self-Organizing Maps*, Berlin, Edition Springer, 2001.

[20] O. Grigore, I. Gavat, C. Grigore, M. Cotescu, “An Adaptive Lighting System Using the Simulated Annealing Algorithm” in *Proceedings of the International Conference on Simulation, Modeling and Optimization,* 09/2008, ISSN: 1790-2769, pp. 143-147. Available at: http://www.wseas.us/e-library/conferences/2008/spain/smo/smo22.pdf

[21] O. Grigore, I. Gavat, C. Grigore, M. Cotescu, “Psycho-Physiological Signal Processing and Algorithm for Adaptive Lighting Control”, ISEEE – Galati, 2008