Emotion Recognition from Speech: Stress Experiment

Stefan Scherer*, Hansjörg Hofmann†, Malte Lampmann†, Martin Pfeil†, Steffen Rhinow†, Friedhelm Schwenker*, Günther Palm*

* Institute of Neural Information Processing,
† Institute of Information Technology,
University of Ulm, 89069 Ulm, Germany
{firstname.lastname}@uni-ulm.de

Abstract

The goal of this work is to introduce an architecture to automatically detect the amount of stress in the speech signal close to real time. For this an experimental setup to record speech rich in vocabulary and containing different stress levels is presented. Additionally, an experiment explaining the labeling process with a thorough analysis of the labeled data is presented. Fifteen subjects were asked to play an air controller simulation that gradually induced more stress by becoming more difficult to control. During this game the subjects were asked to answer questions, which were then labeled by a different set of subjects in order to receive a subjective target value for each of the answers. A recurrent neural network was used to measure the amount of stress contained in the utterances after training. The neural network estimated the amount of stress at a frequency of 25 Hz and outperformed the human baseline.

1. Introduction

Affective computing aims at providing more effective and natural human computer interfaces. The goal of most of the efforts in affective computing is recognizing emotions, such as anger or boredom. However, stress is paid hardly any attention, even though everybody experiences stress at work or in everyday situations. These situations may even be dangerous, for example while driving the car during rush hour. An understanding car management system could be of great importance in these situations, by calming down the driver by playing different music or warning him about his current emotional status. This example illustrates the usefulness of an automatic stress recognizer. However, currently it is difficult to implement such applications since data is difficult to obtain and the available datasets are mostly recordings made by the military. Furthermore, the vocabulary of the datasets is limited (Vlasenko et al., 2007; Hansen and Bou-Ghazale, 1997).

The goal of this work is to obtain a large dataset comprising a large vocabulary at different stress levels. Therefore, an approach to record audio signals containing actual stress induced by an air controller simulation, described in detail in Sect. 2., is introduced in this paper. Subjects are asked to play this air controller simulation and answer different questions taken randomly from a pool of questions comprising among others personal, political, mathematical, or geographical topics. Additionally, a *Jeopardy* category of questions was included in order to provoke the forming of complete questions, while under stress.

Methods to evaluate the recorded data and an automatic stress recognition architecture are proposed in Sects. 3. and 4.. For the evaluation of the recorded data a second experiment had to be conducted with different subjects, who had to listen to the recordings and label them with a numerical value between 0 (no stress) and 100 (very stressed). The results of this evaluation experiment will be presented in 3.1.. An additional goal is the implementation of a stress recognizer that performs close to real-time. The recognition and evaluation parts of this proposed work are based on recently published work (Maganti et al., 2007b; Scherer et al., 2007; Scherer et al., 2008). For the recognition features extracted from the audio signal resembling the rate of change of frequency in time windows of 100 ms, called modulation spectrum features, will be used as input to a novel type of recurrent neural networks, namely echo state networks (ESNs), that can be trained very efficiently using the direct pseudo inverse calculation to adapt their output weights. Furthermore, they are capable of taking previous feature vectors sequentially into account. In Sect. 4.3. the experiments and recognition results of the automatic system will be presented. Finally, Sect. 5. concludes this work.

2. Experimental Setup

The subject enters a normal office. A comfortable seat awaits the subject in front of a computer monitor. The game is already running, only the login for the subject is missing. The countdown is on. Three, two, one. The subject sees a black screen with green lines resembling a radar screen, as shown in Fig. 1. On the screen little objects in the shape of planes appear. In the beginning of the experiment there are only a few of them moving slowly. Later as the experiment continues, there will be more and the subject has to coordinate their flight paths in order to prevent collisions. The exact number of planes for each level, the duration in seconds, the speed of the planes, and the size of the exit areas, respectively the directions, in percent are listed in Table 1. The task of the game is to send as many planes as possible to their desired target. The plane that should receive directions is selected by a mouse click. After a plane is selected a small label indicating the desired destination is shown, as well as a green circle around the plane. The direction it should fly towards is indicated using the number pad of the keyboard. If a plane is directed the correct exit the player

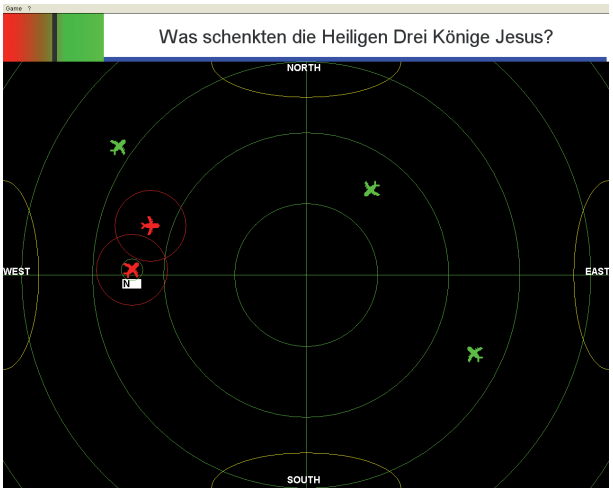

Figure 1: Example screen of medium difficulty.

Level	Planes	Speed	Target Size	Duration
1	2	2	100%	60s
2	3	2	100%	60s
3	3	3	90%	30s
4	4	3	80%	30s
5	4	4	70%	30s
6	4	5	60%	30s
7	5	6	50%	120s
8	5	7	50%	120s
9	5	7	40%	60s
10	5	7	30%	60s

Table 1: Difficulty levels of the air controller simulation.

is rewarded 100 points, otherwise 50 points are subtracted if a plane exits on a wrong path or 100 points if two planes crash. Before planes crash they change their color to red in order to warn the player of possible collisions. The current performance of the player is shown color coded, in order not to distract him too much from the actual tasks. Additionally, on top of the screen questions appear. The time window for an answer is open for eight seconds. Now the subject should answer the question correctly within the continuously shrinking time window. The spoken answers are recorded along with the corresponding time window, the question itself, and the difficulty degree of the air controller simulation. Possible questions comprise small enumeration problems, political, general knowledge, and personal questions, and *Jeopardy* style answers to enlarge the possible output vocabulary. Such a *Jeopardy* task is, according to the famous quiz game, solved correctly by formulating a corresponding question to a given answer. Example questions could look like the following:

- What is the name of the highest mountain in Europe?
- Which kind of music do you like?
- He was the first man to put his feet on the moon. (*Jeopardy*)
- Name three different car brands.

The experiment continues for around 10 minutes and becomes increasingly difficult over time inducing more and more stress. At the beginning of three phases the subjects are asked how stressed they are on a scale from one to ten in order to assess their personal feeling about the experiment. The question is asked for the first time at the beginning of level two, the second time at the beginning of level five, and the last time at the beginning of level ten. Furthermore, the performance of the players is recorded for each trial. These two values could also indicate the sensitivity of the subject towards increasing stress or difficulty in a computer game. To summarize, the subjects' assignment is to use the mouse to select and change the flight paths of multiple planes flying at different speeds, according to the difficulty level, and to say the correct answers for the questions shown on top. The subjects will be given the same instructions at their first visit and are encouraged to come back in order to improve their performance. The evaluation of the performance however, will be "black boxed". The game only gives reviews, such as "You can do better than that! Would you like to try again?", even if the subject performed well. These comments intend to increase the stress level further during successive runs of the game. This setup clearly intends to induce real stress in a playful manner and not to induce boredom over time. Repeated trials will show if the program succeeds in doing that. However, this is only one part of the experiment, since evaluation of the recorded content as well as training an artificial recurrent neural network for automatic stress recognition, are as important as recording it. Section 3. introduces the intended procedure of data evaluation and Sect. 4. describes the utilized neural network architecture.

3. Evaluation and Labeling

For the evaluation of the recorded audio material another experiment with different subjects was necessary. Since everybody reacts differently towards stress, as well as some people have more experience playing computer games and may be able to plan flight paths faster than others, it is not possible to draw any direct relationship between the difficulty level of the game and the amount of stress in the speech signal. Therefore, an evaluation experiment with different subjects had to be conducted. The subjects hear randomly presented, and segmented recordings. For the segmentation a novel and robust algorithm utilizing similar features as in the recognition process presented in Sect. 4.1. was applied to purge the recordings of pauses and noise (Maganti et al., 2007a). The subjects are then asked to assign a number between 0 (no stress) and 100 (extremely stressed) to each of the audio segments. To simplify the labeling process, a tool was implemented that helps the subjects to determine the stress level in a fast manner. In order not to bore the subjects, they have the opportunity to stop the labeling process at any time and continue where they left off. Finally, the mean value of stress as indicated by several labeling subjects is used as a target signal and reference in an automatic stress recognition experiment, as described in Sect. 4.. The variance of the numeric labels over the different trials will be used for the evaluation of the automatic stress recognition experiment. The relation-

ship between stress level and difficulty is determined by the human evaluation subjects only. The results of the evaluation experiment are presented in the following section.

3.1. Evaluation Results

In Table 2 a few statistics of the evaluation experiment are summarized. The first column indicates the number of the speaker, assigned according to the recording order. The second and third columns indicate statistics extracted from the evaluation experiment. The mean value of all the evaluations for one speaker are shown along with the standard deviation. These two values indicate the amount of stress the subjects perceived by listening to the recordings. A higher mean value indicates a higher stress level and a larger standard deviation gives a hint towards the range of stress the subjects experienced during the experiment. The 25th and 75th percentile are shown in the last two columns in order to give some information about the range of stress experienced. It is seen that the mean values range from 31 for the lowest stressed subject to 49.6 for the subject experiencing the highest amount of stress.

Speaker	Mean	Std.	P_{25}	P_{75}
1	35.8	16.1	24	47
2	41.9	17.8	25	59
3	45.2	19.2	29.5	61
4	**31.0**	**17.1**	**20**	**40**
5	43.2	20.5	25	61
6	43.0	21.4	23	60
7	31.2	17.6	21	37
8	33.2	16.1	21	41
9	38.0	17.8	23	51
10	35.7	16.9	22	49
11	**49.6**	**20.0**	**31.75**	**65**
12	49.1	19.3	32	65
13	43.4	20.2	26	62
14	32.1	15.8	22	41
15	41.6	18.8	26	56

Table 2: The mean values and standard deviation of the labels for each speaker along with the self-assessment of stress of the speakers while testing, and the number of crashed and misled planes for each of the three phases (beginning/middle/end).

It is also interesting that the self-assessment values of those two subjects are also the minimum and maximum respectively, as it is seen in Table 3. Furthermore, it is seen that players who are performing better in sending the planes towards the correct directions are less stressed than others performing worse. The amount of crashed and misled planes seems also connected to the amount of stress experienced by the subjects, indicating that the game is actually inducing stress in some of the subjects.

A spearman correlation test was conducted between the three factors mean labeled stress value per speaker, mean of self-assessment, and the mean of crashes. The results of the three tests are shown in Table 4, indicating a strong correlation between the labeled stress and the number of crashes and the labeled stress and the self-assessed stress. The correlation between those is significant with p-values below 0.05. The correlation between the self-assessment and the number of crashed planes however, is not significant.

Speaker	Self-Assessment	Crashes
1	1/2/4	0/4/13
2	2/4/?	0/4/30
3	7/6/8	1/10/37
4	**1/1/2**	**0/2/16**
5	7/8/9	0/3/28
6	4/4/6	0/3/26
7	1/3/7	0/1/23
8	1/1/3-4	0/0/8
9	1/1-2/5	0/6/31
10	1/2/5	0/3/11
11	**7/9/10**	**5/9/17**
12	4/4/?	0/5/27
13	1/3/4	9/22/38
14	2/5/8	1/1/26
15	2/3/7	0/2/19

Table 3: The self-assessment of stress of the speakers while testing, and the number of crashed and misled planes for each of the three phases (beginning/middle/end). The question marks indicate that there was no recording of the answer available, since some of the subjects did not find the time to answer all the questions while playing.

	ρ	p-value
M vs SA	0.61	0.01
M vs C	0.68	0.005
C vs SA	0.40	0.13

Table 4: Results of the spearman correlation test. M indicates mean labeled stress value, SA the average self-assessment, and C the mean of the crashes.

Over all five different subjects listened to the recordings and labeled each sentence using the previously described tool. In order to compare the evaluation results with previously published work the stress values were mapped to a scale of $[+1, -1]$ (Grimm et al., 2007). The average standard deviations in the human evaluator's ratings of all the utterances is 0.32. After removing the unusable recordings, where for example no answer was given, 619 recordings remained. The mean values of the labels are used as target signal for the automatic stress recognizer and the average standard deviation is used to assess the accuracy of the recognizer. The results of the automatic stress recognition experiment are described in Sect. 4.3..

4. Automatic Stress Recognition

The setup of the automatic stress recognition system will be very similar to previously published systems, and work to be published (Maganti et al., 2007b; Scherer et al., 2007; Scherer et al., 2008). Biologically motivated modulation

spectrum features will be used as input for the artificial neural network. The utilized network architecture is a so called echo state network (ESN). ESNs utilizing the sequential characteristics of modulation spectrum features are easy to train, since the weights of the neurons are trained using the direct pseudo inverse calculation instead of gradient descent training. Furthermore, the network is robust towards noisy real world conditions as well as the features. The goal of this work is to recognize stress from the speech signal close to real-time, as in recently published work (Scherer et al., 2008), which is of great advantage in time sensitive applications. Additionally, it would be interesting to see the performance of the ESN under noisy conditions, such as in a car or office, by adding noise to the recorded speech. As mentioned before, the labels obtained through experiments with human subjects and the corresponding variances of the labels, will be used for evaluation of the ESN performance.

4.1. Modulation Spectrum Features

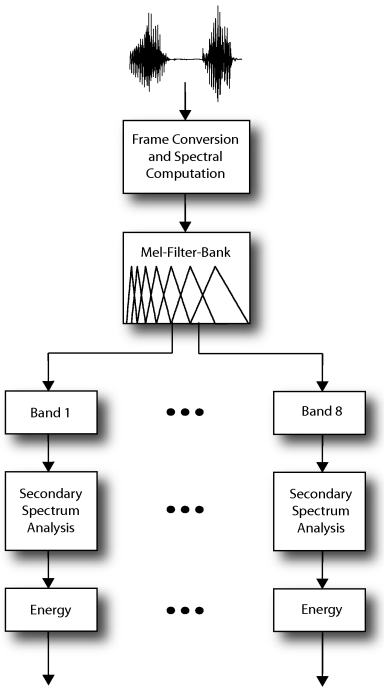

Figure 2: Schematic description for feature extraction.

Short term analysis of the speech signal, such as extracting spectral features from frames not more than several milliseconds, dominates speech processing for many years. However, these features are strongly influenced by environmental noise and renders them therefore unstable. In (Hermansky, 1997), it is suggested to use the so called modulation spectrum of speech to obtain information about the temporal dynamics of the speech signal to extract reliable cues for the linguistic context. Since emotion in speech is often communicated by varying temporal dynamics in the signal the same features are used to classify emotional speech in the following experiments (Scherer et al., 2003). The proposed features are based on long-term modulation spectrum. In this work, the features based on slow temporal evolution of the speech are used to represent the emotional status of the speaker. These slow temporal modulations of

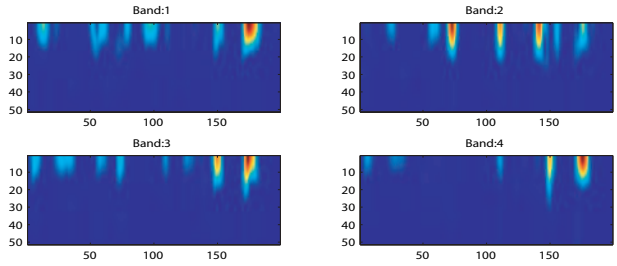

Figure 3: Modulation spectrum for the first four bands of a single angry utterance. The x-axis represents the time scale, in frames and the y-axis, the frequency in Hz.

speech emulate the perception ability of the human auditory system. Earlier studies reported that the modulation frequency components from the range between 2 and 16 Hz, with dominant component at around 4 Hz , contain important linguistic information (Hermansky, 1996; Drullman et al., 1994; Kanedera et al., 1999). Dominant components represent strong rate of change of the vocal tract shape. This particular property, along with the other features has been used to discriminate speech and music (Scheirer and Slaney, 1997). In this work, the proposed features are based on this specific characteristic of speech, to recognize the emotional state of the speaker.

The block diagram for the feature extraction for a system to recognize emotions is shown in Fig. 2. The fast Fourier transform (FFT) for the input signal $x(t)$ is computed over N points with a shift of n samples, which results in a $\frac{N}{2}$ dimensional FFT vector. Then, the Mel-scale transformation, motivated by the human auditory system, is applied to these vectors. The Mel-filter bank with eight triangular filters $H_i[k]$, is defined by:

$$H_i[k] = \begin{cases} \frac{2(k-b_i)}{(d_i-b_i)(c_i-b_i)} & b_i \leq k \leq c_i \\ \\ \frac{2(d_i-k)}{(d_i-b_i)(d_i-c_i)} & c_i \leq k \leq d_i \end{cases}, \qquad (1)$$

where $i = 1, ..., 8$ indicates the index of the i-th filter. b_i and d_i indicate the frequency range of filter H_i and the center frequency c_i is defined as $c_i = (b_i + d_i)/2$. These ranges are equally distributed in the Mel-scale, and the corresponding frequencies b_i and d_i are listed in Table 5. For $k < b_i$ and $k > d_i$ $H_i[k] = 0$.

For each of the bands, the modulations of the signal are computed by taking FFT over the P points, shifted by p samples, resulting in a sequence of $\frac{P}{2}$ dimensional modulation vectors. Most of the prominent energies can be observed within the frequencies between 2 - 16 Hz. Figure 3 illustrates the modulation spectrum based energies for a single angry utterance, for the values $N = 512$, $n = 160$, $P = 100$ and $p = 1$ for the first four bands. For the classification task following values were used: $N = 1600$, $n = 640$, $P = 10$, $p = 1$. Since the signal is sampled with 16 kHz, N corresponds to 100 ms and n to 40 ms resulting in a feature extraction frequency of 25 Hz. According to the window size P a lead time of 400 ms is necessary. Therefore, one feature vector in the modulation spectrum takes 400 ms into account with an overlap of 360 ms, due to p.

Band	Start Freq. (Hz)	End Freq. (Hz)
1	32	578
2	257	964
3	578	1501
4	966	2217
5	1501	3180
6	2217	4433
7	3180	6972
8	4433	8256

Table 5: Start and end frequencies of the triangular Mel-filters.

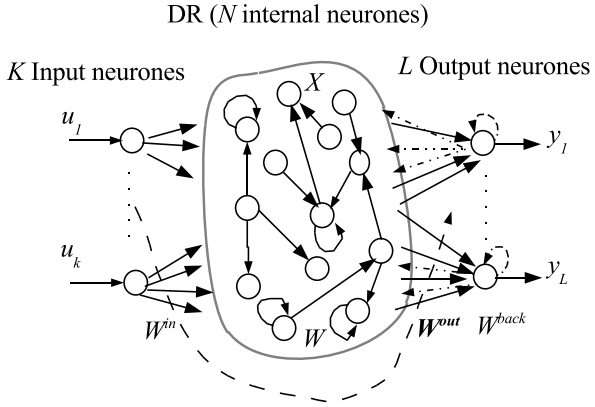

Figure 4: Basic architecture of ESN. Dotted arrows indicate connections that are possible but not required.

4.2. Echo State Network

Feed forward neural networks have been successfully used to solve problems that require the computation of a static function, i.e. a function whose output depends only upon the current input. In the real world however, many problems cannot be solved by learning a static function because the function being computed may produce different outputs for the same input if it is in different states. Since expressing emotions is a constantly changing signal, emotion recognition falls into this category of problems. Thus, to solve such problems, the network must have some notion of how the past inputs affect the processing of the present input. In other words, the network must have a memory of the past input and a way to use that memory to process the current input. This limitation can be rectified by the introduction of feedback connections in the network. The class of Neural Networks which contain feedback connections are called RNNs. In principle RNNs can implement almost arbitrary sequential behavior, which makes them promising for adaptive dynamical systems. However, they are often regarded as difficult to train. Using ESNs only two steps are necessary for training: First, one forms a dynamic reservoir (DR), with input neurons and input connections, which has the echo state property. The echo state property says: "if the network has been run for a very long time, the current network state is uniquely determined by the history of the input and the (teacher-forced) output." (Jaeger, 2002). According to experience, it is better to ensure that the internal weight matrix has maximum eingenvalue $|\lambda_{max}| < 1$. Second, one attaches output neurons to the network and trains suitable output weights.

As presented in (Fig. 4), we consider a network with K inputs, N internal neurons and L output neurons. Activations of input neurons at time step n are $U(n) = (u_1(n), \ldots, u_k(n))$, of internal units are $X(n) = (x_1(n), \ldots, x_N(n))$, and of output neurons are $Y(n) = (y_1(n), \ldots, y_L(n))$. Weights for the input connection in a $(N\text{x}K)$ matrix are $W^{in} = (w_{ij}^{in})$, for the internal connection in a $(N\text{x}N)$ matrix are $W = (w_{ij})$, and for the connection to the output neurons in an L x $(K+N+L)$ matrix are $W^{out} = (w_{ij}^{out})$, and in a $(N\text{x}L)$ matrix $W^{back} = (w_{ij}^{back})$ for the connection from the output to the internal units.

The activation of internal and output units is updated according to:

$$X(n+1) = f(W^{in}U(n+1) + WX(n) + W^{back}Y(n)) \quad (2)$$

where $f = (f_1, \ldots, f_N)$ are the internal neurons output sigmoid functions. The outputs are computed according to:

$$Y(n+1) = f^{out}(W^{out}(U(n+1), X(n+1), Y(n))) \quad (3)$$

where $f^{out} = (f_1^{out}, \ldots, f_L^{out})$ are the output neurons output sigmoid functions. The term $(U(n+1), X(n+1), Y(n))$ is the concatenation of the input, internal, and previous output activation vectors. The idea of this network is that only the feed forward weights for connections from the internal neurons to the output, the connections from the input layer to the output, and the connections from the output towards itself are to be adjusted.
Here we present briefly an off-line algorithm for the learning procedure:

1. Given I/O training sequence $(U(n), D(n))$
2. Generate randomly the matrices (W^{in}, W, W^{back}), scaling the weight matrix W such that it's maximum eigenvalue $|\lambda_{max}| \leq 1$.
3. Drive the network using the training I/O training data, by computing

$$X(n+1) = f(W^{in}U(n+1) + WX(n) + W^{back}D(n)) \quad (4)$$

4. Collect at each time the state $X(n)$ as a new row into a state collecting matrix M, and collect similarly at each time the sigmoid-inverted teacher output $tanh^{-1}D(n)$ into a teacher collection matrix T.
5. Compute the pseudo inverse M^+ of M and the output weights

$$W^{out} = (M^+T)^t \quad (5)$$

t: indicates transpose operation.

For exploitation, the trained network can be driven by new input sequences and using the equations (2) and (3).

4.3. Experiments and Results

For the automatic stress recognition experiments an ESN with 100 neurons in the DR was trained. The network was randomly initialized and the connectivity within the network was 0.25 which indicates that 25% of the connections were set within the network. Additionally, the spectral width $\lambda_{\max}$ was set to 0.2.

	MSE	ME
Labeler 1	0.284	0.421
Labeler 2	**0.151**	**0.281**
Labeler 3	0.291	0.422
Labeler 4	0.241	0.384
Labeler 5	0.211	0.365
ESN	**0.084**	**0.235**

Table 6: Results for the classification experiments. MSE denotes the mean square error, ME the mean absolute error, and ESN the echo state network.

A standard 10-fold cross validation was conducted, where a randomly chosen tenth of the data was used for testing and the rest for training in each fold, in order to test the stress recognition capabilities of the ESN. Since there is no way to assess the "true" amount of stress for each file, the targets for each of the utterances were the mean values over all the five labelers. In order to compare the performance of the automatic recognizer the mean square errors (MSE) and mean absolute errors (ME) for each of the labeler and the ESN towards the target are listed in Table 6. It is shown, that the ESN outperforms each of the labelers even though the human labelers are in favor since their decision is directly included in the targeted mean value. The MSE for the ESN is with 0.084 below the best human labeler reaching a MSE of 0.151. Furthermore, the ESN outputs decisions on a frame wise basis at a frequency of 25 Hz, as mentioned in Sect. 4.1.. The results show that a relatively simple artificial neural network can recognize stress better than the human baseline.

5. Conclusions

This paper presented an experimental setup to record speech data containing different levels of stress. It targeted the lack of freely available datasets comprising stress in speech. An air controller simulation was used to induce stress in fifteen subjects. Furthermore, a tool was presented that was used to label the audio signal by a different set of subjects, since not all the subjects reacted in the same way with respect to the experienced stress. A thorough statistical analysis of the labeled and recorded data was given, indicating that the labels correlate with the self assessed stress perception and the playing skills. Additionally, a recurrent neural network namely an echo state network (ESN) was trained with the labeled audio data using the computationally inexpensive direct pseudo inverse method, and used to recognize the amount of stress frame wise at a frequency of 25 Hz. The results were promising and the automatic recognizer outperformed the human recognition rates. For future work the dataset could still be expanded and the number of labelers should be increased, as well as different architectures may perform even better than the proposed ESN in automatically detecting stress in speech.

6. References

R. Drullman, J. Festen, and R. Plomp. 1994. Effect of reducing slow temporal modulations on speech reception. *Journal of the Acousic Society*, 95:2670–2680.

M. Grimm, K. Kroschel, H. Harris, C. Nass, B. Schuller, G. Rigoll, and T. Mossmayr. 2007. On the necessity and feasibility of detecting driver's emotional state while driving. In *Proceedings of Affective Computing and Intelligent Interaction*, pages 126–138.

J. H. L. Hansen and S. Bou-Ghazale. 1997. Getting started with susas: a speech under simulated and actual stress database. In *Proceedings of Eurospeech 1997*.

H. Hermansky. 1996. Auditory modeling in automatic recognition of speech. In *Proceedings of Keele Workshop*.

H. Hermansky. 1997. The modulation spectrum in automatic recognition of speech. In *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*.

H. Jaeger. 2002. Tutorial on training recurrent neural networks, covering bppt, rtrl, ekf and the echo state network approach. Technical Report 159, Fraunhofer-Gesellschaft, St. Augustin Germany.

N. Kanedera, T. Araib, H. Hermansky, and M. Pavele. 1999. On the relative importance of various components of the modulation spectrum for automatic speech recognition". *Speech Communications*, 28:43–55.

H. K. Maganti, P. Motlicek, and D. Gatica-Perez. 2007a. Unsupervised speech/non-speech detection for automatic speech recognition. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*.

H. K. Maganti, S. Scherer, and G. Palm. 2007b. A novel feature for emotion recognition in voice based applications. In *Proceedings of ACII*, pages 710–711.

E. Scheirer and M. Slaney. 1997. Construction and evaluation of a robust multifeature speech/music discriminator. In *Proceedings of ICASSP*, volume 1, pages 1331–1334.

K. R. Scherer, T. Johnstone, and G. Klasmeyer, 2003. *Handbook of Affective Sciences - Vocal expression of emotion*, chapter 23, pages 433–456. Affective Science. Oxford University Press.

S. Scherer, F. Schwenker, and G. Palm. 2007. Classifier fusion for emotion recognition from speech. In *Proceedings of Intelligent Environments 07*.

S. Scherer, M. Oubbati, F. Schwenker, and G. Palm. 2008. Real-time emotion recognition from speech using echo state networks. In *to be published in Proceedings of ANNPR 2008*.

B. Vlasenko, B. Schuller, A. Wendemuth, and G. Rigoll. 2007. Frame vs. turn-level: emotion recognition from speech considering static and dynamic processing. In *Proceedings of Affective Computing and Intelligent Interaction 2007*.