Modern Applied Science; Vol. 9, No. 8; 2015
ISSN 1913-1844 E-ISSN 1913-1852
Published by Canadian Center of Science and Education

# Robust Voice Activity Detection with Deep Maxout Neural Networks

Valentin Sergeyevich Mendelev[1], Tatiana Nikolaevna Prisyach[2] & Alexey Alexandrovich Prudnikov[3]

[1] St. Petersburg National Research University of Information Technologies, Mechanics & Optics, Speech Technology Center, Russia

[2] STC-innovations Limited, Russia

[3] St. Petersburg National Research University of Information Technologies, Mechanics & Optics, Russia

Correspondence: Valentin Sergeyevich Mendelev, St. Petersburg National Research University of Information Technologies, Mechanics & Optics, Speech Technology Center, 197101, St.Petersburg, Kronverkskiy pr, 49, Russia. E-mail: mendelev@speechpro.com/prisyach@speechpro.com/prudnikov@speechpro.com

Received: January 9, 2015 Accepted: March 20, 2015 Online Published: June 25, 2015

doi:10.5539/mas.v9n8p153 URL: http://dx.doi.org/10.5539/mas.v9n3p153

**Abstract**

Voice activity detection (VAD) under non-stationary noises is a very important task to solve when using a real-life system of automatic speech recognition, especially if a remote microphone is used. Many existing methods do not work well with noise that changes over time or with very low signal-to-noise ratio (SNR). This paper proposes a method based on deep maxout neural networks with dropout regularization. The method is effective even for very low SNR (up to -5dB). The robustness of the method is demonstrated by low FR/FA error rates on a test dataset that was recorded under conditions different from the training dataset.

**Keywords:** voice activity detection, maxout networks, non-stationary noise, deep learning, dropout

## 1. Introduction

Noise robustness of a speaker activity detector is a very important requirement for real-life use. That is especially evident when using a remote micriphone. Noise introduces substantial distortions into the speech signal. If the system is trained on clean data and used for noisy data, that leads to a significant accuracy reduction in detecting the boundaries of speaker activity. Most existing approaches require the information about statistical characteristics of the noise to be known beforehand. These methods can be divided into three categories: approaches based on a deterministic rule, statistical approaches and neural networks-based approaches.

Approaches based on a deterministic rule use a number of characteristics, such as zero crossing rate, short-time energy, autocorrelation coefficients, in order to compare acoustic features with a certain preset threshold to make a decision. In (Rabiner & Sambur, 1975) two acoustic features (log energy and zero crossing rate) are used to detect the boundaries of isolated words. This algorithm is very simple but it does not work in noisy conditions. The authors of (Savoji, 1989) first calculate the probability density function for the spectrum of each frame and then the entropy. They obtain speech and pause labels by using certain preset thresholds. This method does work for a noisy signal, but only for slowly changing noise levels, and it is not stable under low SNR. In (Krubsack & Niederjohn, 1991), a deciding rule is used based on pitch detection. A speech confidence measure is determined using a euristic procedure based on three features extracted from the autocorrelation function. In (Junqua et al., 1994) the method for detecting word boundaries is based on a time-frequency parameter which is formed from the energy in the frequency band and the log of short-time energy. The noise threshold is first calculated based on several initial frames of the input signal and then compared to the time-frequency parameter in order to determine the initial boundaries. Then the threshold rule is used to determine initial and final boundaries of words. The main drawback of approaches based on deterministic rules is that they use thresholds extracted empirically from a segment of non-speech signal. Consequently, such methods do not work for cases when noise levels change over time. They are also not effective for low SNRs.

Statistical approaches (employing hidden Markov models (HMM), Gaussian mixture models (GMM)) use maximum aposteriori probability (MAP) or maximum likelihood (ML) criteria for speech detection. It is assumed that a feature vector belongs to a certain class. Different clustering methods are used for solving the task. However, a great amount of training data for different types of background noises is needed for estimating the probability distribution. The quality of these approaches depends on the choice of probability distribution and the possibility of estimating the parameters of noise distribution. (Atal & Rabiner, 1976) solved the problem of speech detection for clean speech using an approach based on image **recognition**. Five acoustic features were used: zero crossing rate, short-time energy, the first coefficient of the autocorrelation function, the first-order linear prediction coefficient and the residual energy of linear prediction. The model for each class was a multidimensional Gaussian distribution. The MAP criterion was used for making the decision. In (Acero et al., 1993) HMM was used for modeling speech and pause classes, and the Viterbi algorithm was used for searching. (Bhiksha & Rita, 2003) describes using non-linear likelihood obtained from a Bayesian classifier. The main drawback of statistical methods is that the distribution of acoustic features for each class must be known beforehand. (Wu and Zhang, 2011) proposed using a linear weighted combination of different statistical models as the input of the unsupervised SVM.

Neural networks-based approaches use neural networks as template classifiers. There are two advantages in using such an approach. The first is that a neural network classifier is built directly on the training data without a strict assumption about the distribution of its classes. The second is the high discrimination capability of neural networks. (Qi & Hunt, 1993) proposes a multilayer neural network for detecting voiced and non-voiced fragments and pauses. Several features are used: cepstral coefficients, zero crossing rate and mean square energy. However, this approach completely ignores context information. In (Hong & Lee, 2013) RNN is used for classifying speech and non-speech fragments under noisy conditions. The authors demonstrate the advantage of using a RNN classifier compared to a GMM classifier. They describe the efficiency of the method under changing noise levels, however they only deal with different automobile noises. The paper (Zhang & Wu, 2013) proposes a deep belief network (DBN)-based VAD. DBN is a powerful hierarchical generative model for feature extraction. Unlike traditional methods of training deep models, DBN can prevent overfitting by using a special unsupervised pretraining procedure. A DBN-based VAD first connects acoustic features in a long feature vector, which is used as a visible layer or input DBN. Then a new feature is extracted as a result of the transition of the long feature vector through multiple nonlinear hidden layers. As a result, each class of observation is predicted by the linear classifier, so the output is the softmax layer of the DBN with a new feature at the entrance.

Deep neural networks have a long history. They may describe a highly variant function using several parameters. If the training is completed successfully, they can achieve good generalization capability even with a small volume of training data.

We propose using a deep neural network with a maxout activation function and dropout regularization. Dropout technology has shown its efficiency on small training data. Using maxout improves the accuracy of model averaging with dropout. The trained neural networks are highly effective for noisy data even under low SNR and in case of training and test data mismatch.

## 2. The Proposed Technique

### *2.1 The Structure of VAD*

Figure 1 shows the structure of the speaker activity detector.

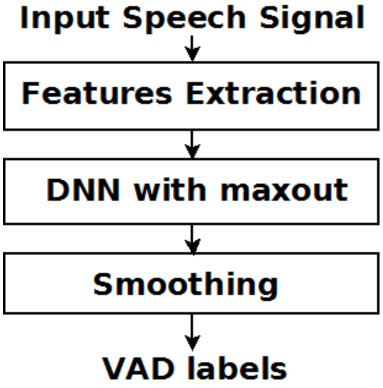

Figure 1. VAD structure.

The first stage is feature extraction. We use Fbank with context length 15.

Then the features are fed into a trained deep maxout neural network. The output of the network are the aposteriori probabilities of each frame belonging to one of the classes (speech, pause). For the correct interpretation of the speech segments and pause segments, a threshold is used for aposteriori probabilities. The threshold value may be selected automatically depending on the SNR. The threshold increases for larger values of SNR, so that VAD can separate speech from pause with greater certainty. Thus, the values of aposteriori probabilities will be close to 1 at high SNR. At low SNR, the aposteriori probabilities of speech segments may decline to 0.6. Anyway, the threshold does not fall below 0.55 in our case, since it is important to identify all the speech.

The last stage is smoothing the frame labeling. By default, fragments shorter than 1 second are smoothed.

### *2.2 Training Features*

The choice of training features is critical for any classification task. For a speaker activity detector, good features must satisfy two conditions: 1) the distribution of speech and non-speech fragments must be different, that is, good features must not overlap for speech and noise classes; 2) the features must be robust to noise.

The following feature types are used in the literature: energy features (Rabiner & Sambur, 1975), spectrum features (Boll, 1979), cepstral features (Kinnunen et al., 2007), harmonic features (Kingsbury et al., 2002) and long-term features (Fukuda et al., 2008).

We examined and compared the following features: mel-frequency cepstral coefficients (MFCC) (Kinnunen et al., 2007) with context, filter banks (Fbank) with context and with normalization of the cepstral average, gammatone frequency cepstral coefficients (GFCC) (Shao wt al., 2009) with context. The advantage of GFCC features compared to the others is that they are more robust to noise, so they work better for speech detection.

### *2.3 The Training Network*

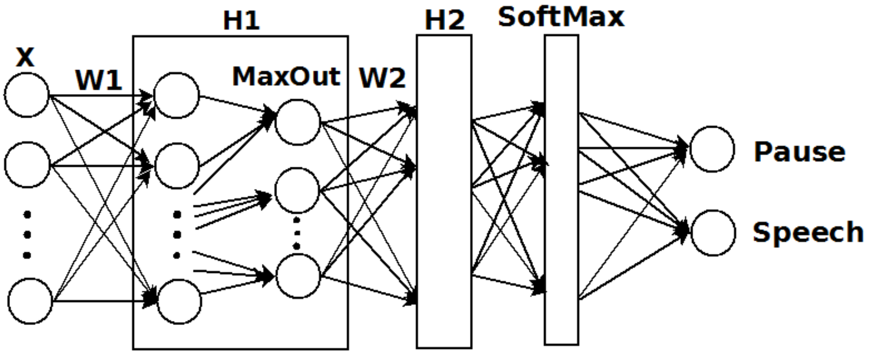

Figure 2. The neural network for VAD with two hidden layers $H_1$ and $H_2$.

#### 2.3.1 Network Structure

Figure 2 shows the structure of the neural network used for training two classes: "speech" and "pause". It contains two hidden layers $H_1$ and $H_2$. The structure of $H_2$ is analogous to $H_1$.

Let us examine the fully connected layer of the neural network with the input feature vector $X = [x_1, x_2, ..., x_n]^T$ and the weight matrix of the first hidden layer $W1$ with dimenstion $d \times n$, $d$ in the experiments eqals 1200, $n$ is the dimension of the minibatch. It is assumed that in order to increase the accuracy of network training the input features are normalized for the mean and variance. The initial weights are selected randomly in the interval $[-0.01, 0.01]$ according to uniform distribution. The output is the vector $Y1 = [y1_1, y1_2, ..., y1_d]^T$, which is calculated as matrix multiplication between the input vector of the layer and the weight matrix using the activation function $A$. As a result,

$$Y1_{ij} = A\left(W1_{ij} * X + b1_{ij}\right) \ , \ z1_{ij} = W1_{ij} * X + b1_{ij} \,,$$

where $W1_{ij}$ are the elements of the weight matrix of the first hidden layer, $b1_{ij}$ are the corresponding offsets, $i = 1, ..., n, j = 1, ..., d$.

After that, the neurons in the network are united into groups, each of which consititutes a maxout node. The number of groups in the experiment is 5. As the activation function $A$ we use the maximum selected from several candidates of the maxout node. (Goodfellow et al., 2013) shows the advantage of the maxout network compared to differentiated activation functions, such as tanh (hyperbolic tangent), which consists of better approximation of model averaging. The maxout activation function is represented as

$$Y1_r = \max_r (z1_r),$$

where $z1_r$ are the values of the neurons in the $r$th maxout node, $r = 1,...,R$, $R$ is the number of neurons in the maxout node.

Dropout is used at the output of the hidden layer. Most of the literature on deep training focuses mainly on regularizing the network so as to avoid overfitting. Different regularizing methods exist (L1, L2 (Bengio, 2012), L2-prior regularizing (Liao, 2013)). Dropout (Hinton, 2012), (Wang & JaJa, 2014) is a widely used and effective regularizing method for DNNs. It makes it possible to avoid complicated coadaptations on training data. On the other hand, the dropout procedure is an efficient way of averaging models with neural networks. A good way to reduce error on the test set is to average the predictions obtained from a very large number of different networks.

The standard solution is to train many separate networks and then apply each of them to the test data, but this process is very labor-intensive both for training and for testing. Random assignment of a zero value to a neuron makes it possible to train a large number of different neural networks in reasonable time. Networks for each training vector are trained in this way, but all networks have a common weight matrix. At the output, each neuron of the layer is assigned a zero value with the probability $1-p$. Experiments show that dropout increases generalization capability of the neural network and improves results on test data. Dropout is also efficient for small training datasets. Combined with maxout, dropout makes it possible to achieve exact rather than approximate model averaging and to fully utilize its potential. The use of dropout is illustrated in Figure 3.

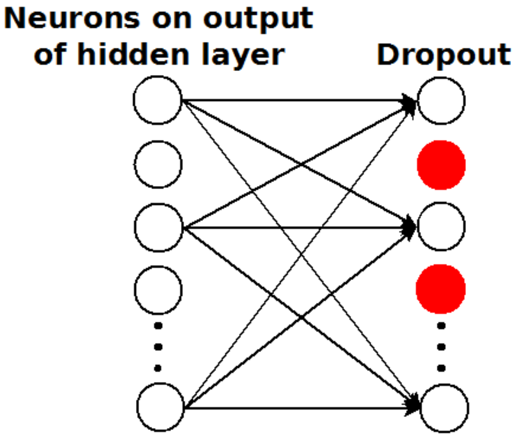

Figure 3. Regularizing using dropout.

Using dropout at the output of the layer we get

$$Y1 = M * A(W1 * X),$$

where $M$ is the vector binary mask with the dimension $d$, $M_j \sim Bernoulli(p), j = 1,...,d$. So the fully connected layer with DropOut becomes a sparse layer in which the values of neurons are updated randomly during training. Each element of the mask is independent for each training feature vector and in fact establishes different connections for each new feature vector from the training dataset. In addition, the mask is also applied to offsets during training.

As with $Y1$, at the output of the second hidden layer we get the vector $Y_2 = [y2_1, y2_2,..., y2_d]^T$ .

At the output of the classifier network we use a softmax layer which normalizes the sum of output values to equal 1 and makes it possible to interpret the outputs of the neural network as aposteriori probabilities: $O = S(Y2; W_S)$, $W_S$ is the vector of dimension parameters $k$, $k$ is the number of classes (in our case, 2: speech and pause).

### 2.3.2 Training Conditions

According to the target function, we calculate the value of the training error $E$. We use the cross-entropy criterion as the target function (Golik, 2013)

$$E(D,O) = -\sum_{i=1}^{m} D_i \log(O_i), m = 1,...,k ,$$

$D$ - $k$ - dimensional desired response of neural network.

At the final training stage we calculate the increments $\Delta W$ for the weights of each neuron for their subsequent updating. We use the standard backward propagation for that (Rojas, 1996). Weight increments are calculated starting from the softmax layer and ending with the first layer. For softmax layer

$$W_S = W_S - \eta E'_{W_S} ,$$

$\eta$ is the training speed.

We introduce two differences from the standard backward propagation procedure. Firstly, increments for the dropout layer are calculated: $Y2 = Y2 - \eta(M * A'_{Y2})$. Here the weights that were active during the direct pass are updated according to the mask $M$. Secondly, such a mask is also used for the maxout layer, that is, only the weights corresponding to maximum values are updated. The increments for the weights of first hidden layer $H_1$ are determined in the same way.

## 3. Experiments and Results

### *3.1 Training and Testing Datasets*

The training and testing data for the neural networks were taken from the speaker database recorded at Speech Technology Center (STC). The database contains recordings of phonetically rich sentences using a remote all-direction microphone under various acoustic conditions (office, home, car, street). The microphone was located at the distance of 2 to 3m from the speaker, with a 0.5m error. The experimental dataset is described in Table 1.

Table 1. Experimental dataset

| Dataset | Sampling rate, Hz | Duration, hours | Men | Women | SNR, dB |
|---|---|---|---|---|---|
| **STC dataset** | 16000 | 131 | 188 | 220 | [-5; 20] |

### 3.2 Experimental Results

For testing the robustness of VAD with maxout DNN we trained several DNNs with different features. All the networks had two hidden 1000-dimensional layers.

The fbankCMN_2HLx1000_L2.net DNN was trained using Fbank with context length 15 using cepstral mean normalization. L2 regularization was used during training, network configuration was fully connected.

The mfccCMN_2HLx1000_L2.net DNN was trained using MFCC, in other respects it was similar to the previous network.

The gammatoneNet_2HLx1000_L2.net DNN was trained using gammatone features with context length 15. L2 regularization was used during training, network configuration was also fully connected.

The final network, maxoutCMN_2HLx1000.net, was trained using Fbank, context length 15, using cepstral mean normalization. We used dropout regularization for training, the network configuration was described above in Section 4, maxout activation function was used.

The test data were remote microphone recordings with the total duration of 3 hours, containing different types of office and home noises, as well as street, automobile, construction noises. The test data did not match the training data: a different type of remote microphone was used, the distance was not the same (the microphone could be further away from the speaker than 3m).

Table 2 shows the results of FR/FA, where FR is the error “speech as pause” and FA is “pause as speech”.

Table 2. Comparison of FR/FA for different neural networks

| Network configuration | Features | FR | FA |
|---|---|---|---|
| fbankCMN_2HLx1000_L2.net | 31xFbank+CMN | 8 | 13 |
| mfccCMN_2HLx1000_L2.net | 31xMFCC+CMN | 10 | 12 |

| | | | |
|---|---|---|---|
| gammatoneNet_2HLx1000_L2.net | 31xGF | 11 | 13 |
| **maxoutCMN_2HLx1000.net** | 31xFbank+CMN | **2.6** | **2.5** |

## 4. Discussion

The table shows that using the maxout-activation function in combination with the dropout regularization reduces "speech as pause" and "pause as speech" errors four times. The reasons why this result is achieved are as follows. First, maxout does not use a fixed activation function, instead the function is created during training. Second, maxout is a universal approximator. Any continuous function can be approximated arbitrarily well on a compact domain by a maxout network with two maxout hidden units. Third, dropout performs model averaging, so maxout in conjunction with dropout enhances the accuracy of dropout model averaging technique and improves optimization. The maxout model benefits more from dropout than other activation functions. Through the use of such technology we can achieve greater robustness in noisy conditions.

This paper presents the results of the first experiments with DNN with maxout activation function and dropout regularization on noisy features. In the future, we are particularly interested in the following topics. We plan to perform experiments on the selection of dropout-regularization parameters and the size of maxout groups. Perhaps an increased number of hidden layers can improve the final layout. We would also like to conduct more comprehensive experiments on different ranges of SNR. The described DNN with maxout activation function and dropout regularization was trained on noise fbank. We plan to use other features with different context lengths. Using gammatone features may give the best results under very noisy data.

## 5. Conclusion

The paper presents a robust speaker activity detector based on DNNs with maxout activation function and dropout regularization. It is well-known that the main problem for DNN training is often the insufficient amount of training data. Our experiments show a high efficiency of speech/non-speech detection for the proposed method even in case of mismatched training and test data. The effectiveness of the method is demonstrated by lower error rates compared to standard DNNs under low SNR (up to -5 dB).

Further research will focus on the use of more robust features with different context lengths and on different SNR ranges. Experiments are planned to select optimal parameters of DNN training (selection of maxout group size, dropout-regularization parameters, number of hidden layers).

## Acknowledgements

The work was financially supported by the Ministry of Education and Science of Russian Federation (Contract 14.579.21.0057, ID RFMEFI57914X0057) and by the Government of the Russian Federation, Grant 074-U01.

## References

Acero, A., Crespo, C., Torre, D. L., & Torrecilla, J. C. (1993). Robust HMM-based endpoint detector. Proc. Eurospeech (pp. 1151-1154).

Atal, B. S., & Rabiner, L. R. (1976). A pattern recognition approach to Voiced-Unvoiced-Silence classification with application to speech recognition. IEEE Trans. Acoust. Speech Sig. Process, 24. (pp. 201-212). http://dx.doi.org/10.1109/TASSP.1976.1162800.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. arXiv report:1206.5533, Lecture Notes in Computer Science Volume 7700, Neural Networks: Tricks of the Trade Second Edition, Editors: Gregoire Montavon, Genevieve B. Orr, Klaus-Robert Muller (pp. 437-478). http://dx.doi.org/10.1007/978-3-642-35289-8_26.

Bhiksha, R., & Rita, S. (2003). Classifier-based non-linear projection for adaptive endpointing of continuous speech. Computer Speech and Language, 17 (pp. 5-26). http://dx.doi.org/10.1016/S0885-2308(02)00028-1.

Boll, S. F. (1979). Suppression of acoustic noise in speech using spectral subtraction. Acoustics, Speech and Signal Processing. IEEE Transactions on, vol. 27, no. 2, (pp. 113–120). http://dx.doi.org/10.1109/TASSP.1979.1163209

Fukuda, T., Ichikawa, O., & Nishimura, M. (2008). Phone-duration-dependent long-term dynamic features for a stochastic model-based voice activity detection. In Ninth Annual Conference of the International Speech Communication Association.

Golik, P., Doetsch, P., & Ney, H. (2013). Cross-entropy vs. squared error training: a theoretical and experimental comparison. INTERSPEECH, ISCA. (pp. 1756-1760).

Goodfellow, I. J., Warde, F. D., Mirza, M., Courville, A., & Bengio, Y. (2013). Maxout networks. arXiv preprint arXiv:1302.4389.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. Presented at CoRR. arXiv:1207.0580v1.

Hong, W. T., & Lee, C. C. (2013). Voice Activity Detection based on Noise-Immunity Recurrent Neural Networks. *IJACT, 5*(5), 338-345. http://dx.doi.org/10.4156/ijact.vol5.issue5.41.

Junqua, J. S., Mak, B., & Reaves, B. (1994). A robust algorithm for word boundary detection in the presence of noise. IEEE Trans. *Speech and Audio Processing, 2*, 406-412. http://dx.doi.org/10.1109/89.294354.

Kingsbury, B., Saon, G., Mangu, L., Padmanabhan, M., & Sarikaya, R. (2002). Robust speech recognition in noisy environments: The 2001 IBM spine evaluation system, in Acoustics, Speech, and Signal Processing (ICASSP). *2002 IEEE International Conference on, 1*, I–53–I–56.

Kinnunen, T., Chernenko, E., Tuononen, M., Frnti, P., & Li, H. (2007). Voice activity detection using MFCC features and support vector machine. *Int. Conf. on Speech and Computer, 2*, 556–561.

Krubsack, D. A., & Niederjohn, R. J. (1991). An autocorrelation pitch detector and voicing decision with confidence measures developed for noise-corrupted speech. *IEEE Trans. Sig. Process, 39*, 319-329. http://dx.doi.org/ 10.1109/78.80814.

Liao, H. (2013). Speaker adaptation of context dependent deep neural networks, in Proc. ICASSP, 2013. (pp. 7947-7951). http://dx.doi.org/10.1109/ICASSP.2013.6639212.

Qi, Y., & Hunt, B. R. (1993). Voiced-unvoiced-silence classifications of speech using hybrid features and a network classifier. IEEE Trans. *Speech and Audio Process, 1*, 250-255. http://dx.doi.org/10.1109/89.222883.

Rabiner, L. R., & Sambur, M. R. (1975). An algorithm for determining the endpoints of isolated utterances. *Bell System Technical Journal, 54*(2), 297-315. http://dx.doi.org/10.1002/j.1538-7305.1975.tb02840.x

Rojas, R. (1996). Neural Networks: A Systematic Introduction. Springer-Verlag, New York, USA. 502.

Savoji, M. H. (1989). Robust algorithm for accurate endpointing of speech signal. *Speech Communication, 8,* 45-60. http://dx.doi.org/10.1016/0167-6393(89)90067-8.

Shao, Y., Jin, Z., Wang, D., & Srinivasan, S. (2009). An auditory based feature for robust speech recognition. In Proc. ICASSP'09, 2009. (pp. 4625–4628). http://dx.doi.org/10.1109/ICASSP.2009.4960661.

Wang, Q., & JaJa, J. (2014). From Maxout to Channel-Out: Encoding Information on Sparse Pathways. ICANN 2014. (pp. 273-280). http://dx.doi.org/10.1007/978-3-319-11179-7_35.

Wu, J., & Zhang, X. L. (2011) Maximum margin clustering based statistical VAD with multiple observation compound feature. *IEEE Signal Process. Lett., 18*(5), 283–286. http://dx.doi.org/10.1109/LSP.2011.2119482.

Zhang, X. L., & Wu, J. (2013). Deep belief networks based voice activity detection. IEEE Trans. Audio, Speech, *Lang. Process., 21*(4), 697–710. http://dx.doi.org/10.1109/TASL.2012.2229986

## Copyrights