Parkinson's Disease Detection from Speech Using Convolutional Neural Networks

Evaldas Vaiciukynas[1,2](✉), Adas Gelzinis[1], Antanas Verikas[1,3], and Marija Bacauskiene[1]

[1] Department of Electrical Power Systems, Kaunas University of Technology, Studentu 50, 51368 Kaunas, Lithuania
{evaldas.vaiciukynas,adas.gelzinis,marija.bacauskiene}@ktu.lt

[2] Department of Information Systems, Kaunas University of Technology, Studentu 50, 51368 Kaunas, Lithuania

[3] Centre for Applied Intelligent Systems Research, Halmstad University, Kristian IV:s väg 3, PO Box 823, 30118 Halmstad, Sweden
antanas.verikas@hh.se

Abstract. Application of deep learning tends to outperform hand-crafted features in many domains. This study uses convolutional neural networks to explore effectiveness of various segments of a speech signal, – text-dependent pronunciation of a short sentence, – in Parkinson's disease detection task. Besides the common Mel-frequency spectrogram and its first and second derivatives, inclusion of various other input feature maps is also considered. Image interpolation is investigated as a solution to obtain a spectrogram of fixed length. The equal error rate (EER) for sentence segments varied from 20.3% to 29.5%. Fusion of decisions from sentence segments achieved EER of 14.1%, whereas the best result when using the full sentence exhibited EER of 16.8%. Therefore, splitting speech into segments could be recommended for Parkinson's disease detection.

Keywords: Parkinson's disease · Audio signal processing
Convolutional neural network · Information fusion

1 Introduction

Parkinson's disease (PD) is the second most common neurodegenerative disease after Alzheimer's [1] and it is expected that the prevalence of PD is going to increase due to population ageing. Medical intervention could be considered to slow down the progression of PD if it is detected early, resulting in increased life span and life quality for PD patients. Acoustic analysis of voice or speech signal is considered as an important non-invasive tool in screening for PD. Related work is summarized in [2], where splitting of speech into voiced and unvoiced parts is recommended.

B. Guidi et al. (Eds.): GOODTECHS 2017, LNICST 233, pp. 206–215, 2018.
https://doi.org/10.1007/978-3-319-76111-4_21

Recent advances in deep learning helped reaching the state-of-the-art performance in various domains – not only for images, but for audio data as well. For example, the combination of deep neural networks with hidden Markov models (DNN-HMM) in [3] outperformed traditional the Gaussian mixture model-based (GMM-HMM) solution. Convolutional neural networks (CNN) were successfully applied to automatic speech recognition [4,5], speech activity detection [6,7] or acoustic scene classification [8]. When applying CNN to audio data it is common to characterize an underlying signal using spectrograms, obtained by calculating the Mel-frequency spectral coefficients (MFSCs) from extracted fragments of a signal, as in [5,6,9,10]. Besides spectrograms, their first and second temporal derivatives (delta and double delta) can also be considered as additional input feature maps [4,7,8,11].

This study explores speech recordings of a four-words sentence in Lithuanian language processed by the spectrogram-based CNN model for the task of PD detection. Splitting a sentence into various segments, corresponding to separate words and combinations of words or syllables, is considered. Besides MFSCs and their first and second derivatives, usage of various other input feature maps is also proposed. Due to different lengths of speech segments, corresponding to the same part of the sentence, a solution to obtain a fixed length spectrogram by image interpolation is compared to the commonly used sampling of a fixed size window at random locations. Decision-level fusion is applied to improve PD detection.

2 Data

Pronunciation of a phonetically balanced sentence in a native Lithuanian language "turėjo senelė žilą oželį" (which translates into "granny had a little greyish goat") was recorded in a sound-proof booth. Recordings were done using an acoustic cardioid (AKG Perception 220, frequency range 20–20000 Hz) microphone. Microphone was located at ∼10 cm distance from the mouth. The audio

Table 1. Sentence segments, containing separate words (# 1–4), transitions between words when split on syllables (# 5–6), pairs of words (# 7–8), or full sentence (# 9).

#	Sentence fragments				Notation
1	tu rė jo				TUREJO
2		se ne lė			SENELE
3			ži lą		ZILA
4				o že lį	OZELI
5		**lė**	ži lą	**o**	LE_O
6	**rė** jo	**se**			RE_SE
7	**tu** rė jo	se ne **lė**			TU_LE
8			**ži** lą	o že **lį**	ZI_LI
9	tu rė jo	se ne lė	ži lą	o že lį	SENTENCE

format was mono PCM wav (16 bits at 44.1 kHz sampling rate). A mixed gender database collected contains 268 subjects (194 healthy controls and 74 PD cases) ranging from 22 to 85 years in age.

Each speech recording was manually annotated and split into sentence segments, containing: separate words, transitions between words and pairs of consecutive words. A full sentence without any splitting (*SENTENCE* segment) was also considered. Details on sentence segments used are in Table 1.

3 Methodology

3.1 Input Feature Maps

An important step in acoustic analysis is characterization of an audio signal by various features. Mel-frequency spectral coefficients (MFSC) is a commonly applied transformation to audio signal spectrum, resulting in the Mel-warped spectrogram. The MFSC spectrogram contains values of amplitude for each frequency coefficient (on the vertical axis) and a time moment (on the horizontal axis). Logarithmic energy values computed after converting an audio signal into Mel-frequency without application of the direct cosine transform, as proposed in [12,13], are used here for MFSCs. Several feature maps, e.g. after considering temporal derivatives of MFSCs, can be stacked on top of each other to form a 3-dimensional array, similarly to the RGB channels in image data.

This work considers several variants of short-term audio features, resulting in nine input feature maps in total:

1. Mel-frequency spectral coefficients ($MFSC$).
2. First temporal derivative of MFSC ($MFSC_{\Delta}$).
3. Second temporal derivative of MFSC ($MFSC_{\Delta\Delta}$).
4. Levinson-Durbin reflection coefficients (LRC) [14].
5. Vocal tract area coefficients ($VTAC$) [15].
6. Ratio of the area of the two successive vocal tract tubes calculated along the frames ($VTAC_R$) [15].
7. Classical spectrogram - frequency spectral coefficients (FSC).
8. Linear predictive coding coefficients ($LPCC$).
9. Mel-frequency cepstral coefficients ($MFCC$).

Figure 1 illustrates an example speech signal and input feature maps extracted from it. The number of coefficients (ticks on the vertical axis) considered was 80.

3.2 Convolutional Neural Network

The convolutional neural network (CNN) is a variant of standard neural network. Instead of fully connected layers CNN has architecture composed of consecutive pairs of *convolution* and *pooling* layers. CNN enables learning of local features and promotes weight sharing, where internal representations are a result of convolving the input with a filter mask. Each convolutional layer has a number

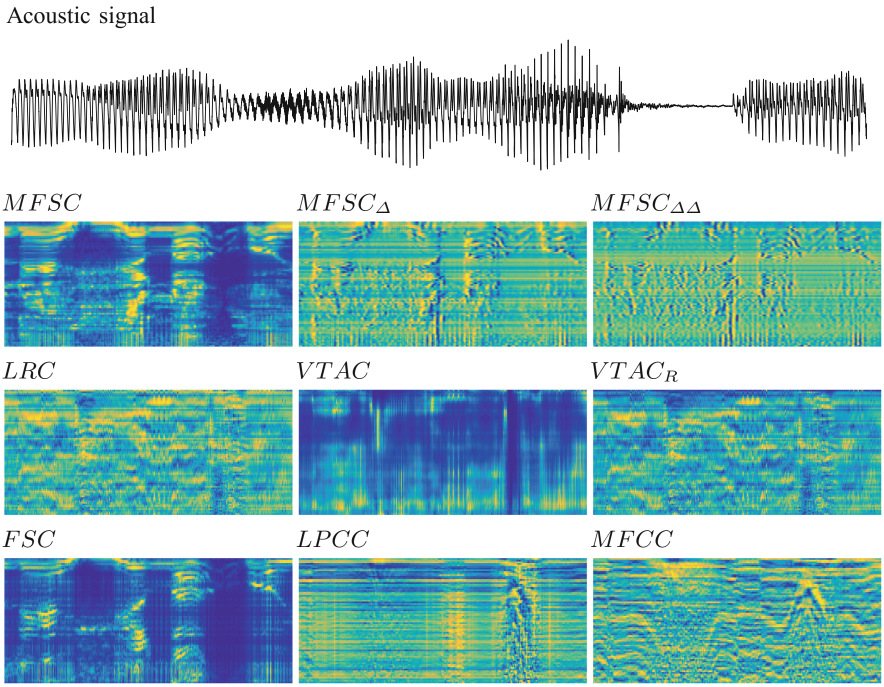

Fig. 1. An example of input feature maps extracted from speech signal.

of filter masks, which are learned during training, and application of convolution results in a new feature map, which is downsampled to a smaller size by using a pooling layer. Input feature maps in CNN are usually organized as a 3-dimensional array, where flat 2D planes in such an array are stacked input feature maps. For example, when CNN is applied on images, an array of 2D input feature maps corresponds to RGB channels. As shown in Fig. 2, every input feature map $\mathbf{O}_i (i = 1, ..., I)$, where I is the number of channels, is connected to further feature maps $\mathbf{Q}_j (j = 1, ..., J)$ through application of convolution. The i-th input feature map is connected to j-th feature map through convolution with a local weight matrix $\mathbf{w}_{ij}$, known as a filter mask. The filter mask is defined by a size $(m \times n)$, where (m) corresponds to a few spectrogram frequencies and (n) to a small temporal window. A non-input feature map $\mathbf{Q}_j$ is obtained using the convolution operation $*$:

$$\mathbf{Q}_j = \sigma\left(\sum_{i=1}^{I} \mathbf{O}_i * \mathbf{w}_{ij}\right) \qquad (j = 1, ..., J) \tag{1}$$

where $\mathbf{O}_i$ is the i-th input feature map, $\mathbf{w}_{ij}$ corresponds to a filter mask, σ is an activation function of the neural network. Therefore, CNN could be considered as a transformation of an input image through a chain of convolutions by the filter masks w learned from the data.

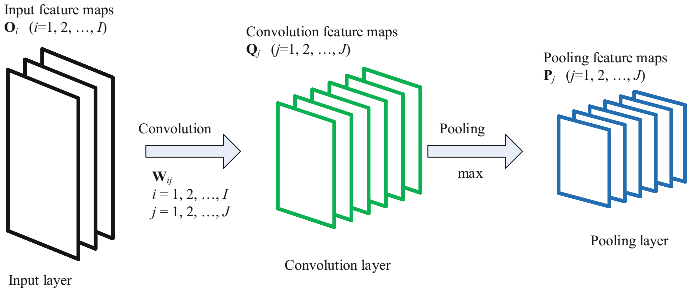

Fig. 2. A pair of *convolution* and *pooling* layers in the CNN architecture.

As shown in Fig. 2, the *max-pooling* operation is performed on each feature map, resulting from the convolution. The goal of *pooling* is to downsample the feature maps to smaller resolution. The max-pooling operation can be written as:

$$p_{ijk} = \max_{n=1,m=1}^{G} q_{i,(j-1)\times s+n,\,(k-1)\times s+m} \tag{2}$$

where G is the pooling size, s denotes the *shift size* (by how many pixels the pooling window is shifted), q_{ijk} is the jk-th element of i-th convolutional feature map Q.

After a pair (or several pairs) of *convolution-pooling* layers, the CNN is completed using a fully-connected dense layer, which uses feature maps from the last pair of *convolution-pooling* layers flattened into a one-dimensional vector. The output layer, connected to the dense layer, realizes the detection task by the *soft-max* of 2 neurons. An example of the CNN architecture, containing two pairs of *convolution-pooling* layers, is illustrated in Fig. 3.

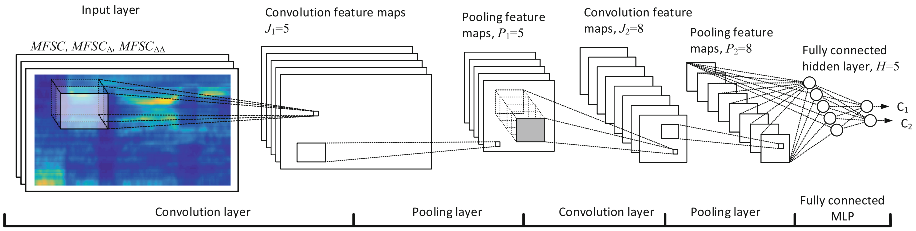
Input layer
MFSC, MFSC$_\Delta$, MFSC$_{\Delta\Delta}$
Convolution feature maps
J_1=5
Pooling feature maps, P_1=5
Convolution feature maps, J_2=8
Pooling feature maps, P_2=8
Fully connected hidden layer, H=5
C$_1$
C$_2$
Convolution layer
Pooling layer
Convolution layer
Pooling layer
Fully connected MLP

Fig. 3. An example of the CNN architecture, having *convolution-pooling* layers and a fully-connected multilayer perceptron (MLP) with 2 output neurons (predicted classes).

4 Experimental Investigations

A CNN with 4 pairs of *convolution-pooling* layers and a fully-connected dense layer with 2 *soft-max* neurons in the output (as shown in Fig. 3) was used for the experiments. The number of neurons in the dense layer (hidden layer of MLP before the *soft-max*) was 256, whereas other parameters of the CNN architecture are listed in Table 2. Filter masks of the first *convolution* layer had rectangular-shaped dimensions of 7 × 5, as recommended by [16] for spectrogram data, whereas filters in subsequent convolutions had traditional square-shaped dimensions of 3 × 3.

Table 2. Parameters for 4 pairs of the *convolution-pooling* layers.

Pair of layers	Convolution layer		Pooling layer
	# of maps	Filter size	Window size
I	96	7 × 5	3 × 3
II	256	3 × 3	3 × 3
III	384	3 × 3	3 × 3
IV	256	3 × 3	3 × 3

Length of speech recordings varies, therefore the size of the input feature maps is not constant with respect to the temporal axis. Input feature maps of fixed size were obtained using the following techniques: (a) cutting out signal fragments of 80 pixels in width at n random locations (F_{80}); (b) cutting out signal fragments of 120 pixels in width at n random locations (F_{120}); (c) re-sizing (squeezing or extending) a spectrogram image using the bi-cubic interpolation into a fixed width, corresponding to the mean width of all images (F_{mean}); (d) squeezing spectrogram image to the minimum width of all images (F_{min}). The number of random locations to sample from in the F_{80} and F_{120} cases was set to $n = 41$, which corresponds to the factor of data augmentation. Therefore, each subject, represented by a speech recording, has a single data example in the case of F_{mean} and F_{min}, but n examples in the case of F_{80} and F_{120}.

Detection performance was evaluated using the stratified 20-fold cross-validation. When performing data splits for the cross-validation, subject-level disjointedness was respected and all data examples for a subject (in F_{mean} and F_{min} cases) were either in a training or in a test fold. The equal error rate (EER), measured at the operating point of detector where sensitivity becomes equal specificity (or false alarm becomes equal miss rate) [17], was obtained for test data and is reported as a goodness-of-detection metric.

Results of initial experiments, not reported here, indicated improved detection performance when using all nine available input feature maps compared to a smaller combination or a single feature map. Detection results when using all nine feature maps are reported in Table 3. Interpolation by squeezing input maps to the smallest width (F_{min}) had slightly better performance than interpolation to the average width (F_{mean}). Nonetheless, the type of fragmentation did not affect the EER considerably. Meanwhile, the choice of sentence segment had a stronger influence on the EER varying from 20.3% for the ZILA segment to 29.5% for the TUREJO segment.

Table 3. Detection performance by EER (in %) for each type of fragmentation and sentence segment when using all input feature maps (shown in Fig. 1). Mean EER through all types of fragmentation is in the last line.

Type	TUREJO	SENELE	ZILA	OZELI	LE_O	RE_SE	TU_LE	ZI_LI	SENTENCE
F_{mean}	30.5	***23.1***	25.1	26.4	27.0	25.3	24.6	25.9	23.4
F_{min}	29.5	23.4	21.9	25.6	24.6	26.7	23.1	24.7	***21.4***
F_{120}	30.6	23.2	***20.3***	24.0	24.6	28.7	26.1	24.7	24.6
F_{80}	34.0	23.8	***22.0***	24.0	24.7	28.3	27.6	25.6	24.9
Mean	31.2	23.4	22.4	25.0	25.2	27.2	25.4	25.2	23.6

Due to the fact that the dataset was expanded in the case of F_{80} and F_{120} types, the example-wise EER shown in Table 3 is not particularly informative subject-wise. To obtain the subject-wise EER, the output class probabilities of n examples from a single recording were fused by averaging. Results in the subject-wise EER form after such fusion are given in Table 4. We can notice lower EER values, especially for the longer segments. It is also worth mentioning that data augmentation resulting from the random sampling (F_{80} and F_{120}), after fusion tends to outperform the interpolation approaches (F_{mean} and F_{min}) irrespective of sentence segment used.

Table 4. Detection performance by EER (in %) after fusing subject-wise decisions from n examples through averaging of output class probabilities.

Type	TUREJO	SENELE	ZILA	OZELI	LE_O	RE_SE	TU_LE	ZI_LI	SENTENCE
F_{120}	27.3	18.7	18.1	22.2	20.4	25.8	20.9	20.5	***16.8***
F_{80}	35.0	19.3	20.2	21.3	18.8	22.2	21.2	20.3	***17.1***

Aiming to improve over the best detection result (EER of 16.8% when using the SENTECE segment and the F_{120} fragmentation type), decision-level fusion of all nine segments was considered. Fusion of decisions arising for each kind of sentence segment (and n examples in the F_{80} and F_{120} cases) was done by: (a) voting (*vote*), where the majority class is considered; (b) averaging output probabilities (*prob*); (c) weighted average of output probabilities (*prob_w*), where weights are set based on the accuracy obtained using that segment; (d) the random forest [18] classifier (*RF*). The results of the decision-level fusion of sentence segments are given in Table 5. We can again notice that interpolation deteriorates detection, but now re-sizing to the average width outperforms squeezing. Meanwhile, the lowest EER of 14.1% was achieved for the (b) fusion case when random sampling of short fragments (F_{80}) was used.

Table 5. Detection performance by EER (in %) for decision-level fusion of all sentence segments. Results are reported by fragmentation type and fusion variant.

Type	*vote*	*prob*	*prob_w*	*RF*
F_{mean}	18.4	18.2	***17.8***	21.0
F_{min}	***19.2***	20.0	19.8	20.1
F_{120}	15.8	14.9	***14.8***	16.8
F_{80}	15.0	***14.1***	14.3	14.4

5 Conclusions

This work investigated PD detection from a speech signal using convolutional neural networks. Spectrograms and several other types of short-term features were considered as stacked 2D input maps to the CNN. A speech recording was split into various sentence segments and influence of each segment to the PD detection performance was evaluated and compared to the decision-level fusion of all segments case. The detection performance measured in EER varied from 29.5% for the TUREJO segment to 20.3% for the ZILA segment. This indicates that some parts of speech recording are more effective for the PD detection task than others.

Interpolation of spectrogram to the fixed length could not outperform the case of using fragments of fixed length taken at random locations and resulted in worse performance when fusion was considered. Therefore, data augmentation, arising from sampling of fixed length fragments, can be considered as more beneficial for CNN than using interpolation and less data.

The best detection result, EER of 14.1%, was achieved when using the decision-level fusion of sentence segments, whereas the best result without splitting a sentence into segments showed EER of 16.8%. Therefore, splitting a speech signal into several potentially overlapping segments and later combining decisions, obtained on each segment as well as a full-length signal, helps to improve PD detection. In this work splitting was done manually, but an automatic way of segmentation for text-dependent recordings should be devised in the future.

Acknowledgements. Funding for this work was provided by a grant (No. MIP-075/2015) from the Research Council of Lithuania. The dataset was collected by the Department of Otorhinolaryngology at Lithuanian University of Health Sciences.

References

1. de Rijk, M., Launer, L., Berger, K., Breteler, M., Dartigues, J., Baldereschi, M., Fratiglioni, L., Lobo, A., Martinez-Lage, J., Trenkwalder, C., Hofman, A.: Prevalence of Parkinson's disease in Europe: a collaborative study of population-based cohorts. Neurologic diseases in the elderly research group. Neurology **54**(11 Suppl 5), S21–S23 (2016)

2. Orozco-Arroyave, J.R., Hönig, F., Arias-Londoño, J.D., Vargas-Bonilla, J.F., Daqrouq, K., Skodda, S., Rusz, J., Nöth, E.: Automatic detection of Parkinson's disease in running speech spoken in three different languages. J. Acoust. Soc. Am. **139**(1), 481–500 (2016)
3. Hinton, G., Deng, L., Yu, D., Dahl, G.E., Mohamed, A.R., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T.N., Kingsbury, B.: Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. IEEE Signal Process. Mag. **29**(6), 82–97 (2012)
4. Abdel-Hamid, O., Mohamed, A.R., Jiang, H., Deng, L., Penn, G., Yu, D.: Convolutional neural networks for speech recognition. IEEE/ACM Trans. Audio Speech Lang. Process. **22**(10), 1533–1545 (2014)
5. Sainath, T.N., Kingsbury, B., Saon, G., Soltau, H., Mohamed, A.R., Dahl, G., Ramabhadran, B.: Deep convolutional neural networks for large-scale speech tasks. Neural Netw. **64**, 39–48 (2015). Special Issue on "Deep Learning of Representations"
6. Zhang, H., McLoughlin, I., Song, Y.: Robust sound event recognition using convolutional neural networks. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 559–563, April 2015
7. Thomas, S., Ganapathy, S., Saon, G., Soltau, H.: Analyzing convolutional neural networks for speech activity detection in mismatched acoustic conditions. In: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2519–2523, May 2014
8. Han, Y., Lee, K.: Acoustic scene classification using convolutional neural network and multiple-width frequency-delta data augmentation. Computing Research Repository (CoRR) arXiv:1607.02383 (2016)
9. Dennis, J., Tran, H.D., Li, H.: Spectrogram image feature for sound event classification in mismatched conditions. IEEE Signal Process. Lett. **18**(2), 130–133 (2011)
10. Deng, L., Abdel-Hamid, O., Yu, D.: A deep convolutional neural network using heterogeneous pooling for trading acoustic invariance with phonetic confusion. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6669–6673, May 2013
11. Adi, Y., Keshet, J., Goldrick, M.: Vowel duration measurement using deep neural networks. In: 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP), pp. 1–6, September 2015
12. Godino-Llorente, J.I., Gomez-Vilda, P.: Automatic detection of voice impairments by means of short-term cepstral parameters and neural network based detectors. IEEE Trans. Biomed. Eng. **51**(2), 380–384 (2004)
13. Dibazar, A.A., Narayanan, S., Berger, T.W.: Feature analysis for automatic detection of pathological speech. In: Proceedings of the 2th Joint EMBS/BMES Conference, Houston, USA, pp. 182–183 (2002)
14. Verikas, A., Gelzinis, A., Vaiciukynas, E., Bacauskiene, M., Minelga, J., Hållander, M., Uloza, V., Padervinskis, E.: Data dependent random forest applied to screening for laryngeal disorders through analysis of sustained phonation: acoustic versus contact microphone. Med. Eng. Phys. **37**(2), 210–218 (2015)
15. Muhammad, G.: Voice pathology detection using vocal tract area. In: 2013 European Modelling Symposium, pp. 164–168, November 2013

16. Hrúz, M., Kunešová, M.: Convolutional neural network in the task of speaker change detection. In: Ronzhin, A., Potapova, R., Németh, G. (eds.) SPECOM 2016. LNCS (LNAI), vol. 9811, pp. 191–198. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-43958-7_22
17. Faundez-Zanuy, M., Monte-Moreno, E.: State-of-the-art in speaker recognition. IEEE Aerosp. Electron. Syst. Mag. **20**(5), 7–12 (2005)
18. Breiman, L.: Random forests. Mach. Learn. **45**, 5–32 (2001)