Computer program for visual diagnosis of the speech disorders

Marek Wiśniewski[1*], Wiesława Kuniszyk-Jóźkowiak[2],
Elżbieta Smołka[2], Waldemar Suszyński[2]

*[1]Institute of Physics, Maria Curie-Skłodowska University,
Pl. Marii Curie-Skłodowskiej 1, 20-031 Lublin, Poland
[2]Institute of Computer Science, Maria Curie-Skłodowska University,
Pl. Marii Curie-Skłodowskiej 1, 20-031 Lublin, Poland*

Abstract

The diagnosis of speech disorders requires not only audio but also visual analysis. The speaker often performs many articulation moves, e.g.: a movement of mouth without emitting sound. Therefore a computer program was developed for simultaneous analysis of several elements in conjunction with the digital recording of a person's speaking face. These elements are: sound, a spectrogram picture and an oscillogram. The program can read audiovisual recordings and allows for multiple playing with a selected pace. What follows from the previous work with the program, it gives much important information about the kind and the essentiality of the speech disorders.

1. Introduction

The therapeutic process of people suffering from disturbances of speech fluency should be adapted to the type of the disturbance and to the individual psychological features of a patient. Therefore it is important that the type and the source of the disturbance are properly diagnosed earlier. A commonly used method for diagnosis is the analysis of sound recordings by the auditory method. It gives a lot of valuable information but is very time consuming and, in many cases, not effective. It particularly concerns disturbances related to the articulation of single phonemes and blockades [1].

In that aspect computer methods have appeared very helpful, especially the frequency-time analysis one. The methods gave a great potentiality and considerably increased effectiveness of diagnosis. Generally computer aided examination depends on the analysis of the following elements [1]:
- spectrograms of non-fluent utterances and their comparison them to spectrograms of fluent utterances,
- distribution of signal amplitudes in a frequency scale within words,

---

*Corresponding author: *e-mail address*: marek.wisniewski@umcs.lublin.pl

- time-frequency parameters characteristic of the utterance envelope of stuttering people,
- evolution of frequencies and formant levels of utterance in time,
- utterances shown as stratum graphs (data are earlier filtered in a suitable way).

The next step related to the use of computers is to develop programs for automatic detection and classification of non-fluent speech [2].

The diagnostic methods described above depend on the analysis of sound recordings and, considering the specifics of human speech, are not totally effective. Therefore new sources of additional information are looked for. Such an additional source seems to be the facial expression of the stuttering person.

Below, a computer program, named "Spectrum" is presented. It was written by the author and allows for examinations of audiovisual recordings.

2. The "Spectrum" program

The program must be run under Microsoft Windows operating system (Windows XP and Windows 98 were tested) with the DirectX 9.x library installed. The capabilities of the program are following:

- playback of audiovisual samples saved in the "avi" file format, where both video and sound signals are not compressed,
- rehearing of "wav "sound files,
- fragment selection from bigger recordings, subjecting them to frequency analysis and showing the results in the form of a colorful spectrogram,
- playback of audiovisual samples in slower frame rates,
- saving selected fragments of audiovisual recordings in the form of "avi" files.

Recorded sound tracks should have sampling frequency of at least 20050 Hz and 16 bit resolution.

After a file is opened one needs to find a proper and interesting fragment of the recording. Next the fragment (max. 10 seconds) can be subjected to the Fourier analysis and is displayed as a spectrogram. Simultaneously, there is an oscillogram of the selected fragment displayed. On the oscillogram interesting elements of the expression can be selected and then played (Fig. 1). During the play a cursor over both the oscillogram and the spectrogram is moving and in the top right corner the patient's face picture is changing. It has proved to be very helpful to play with slower pace when observing the patient's face.

For frequency analysis two methods were implemented: DFT and FFT Radix 2. Analysis parameters can be accessed in the options window and concern the time window change as well as the weight function choice (Fig. 2). Additionally, the dynamics range of the spectrogram can be narrowed. From the

maximum dynamics value found in the investigated signal the selected range is cut off and then displayed. Thanks to that noise, usually appearing at low ranges of frequency, can be removed.
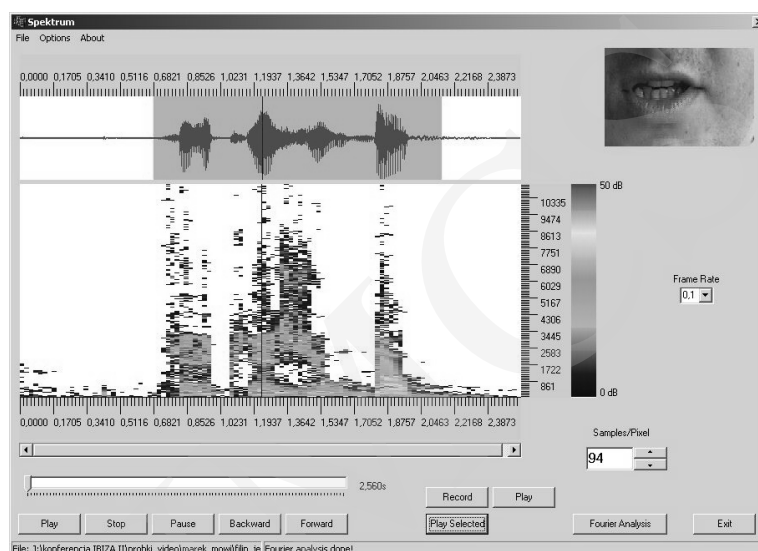

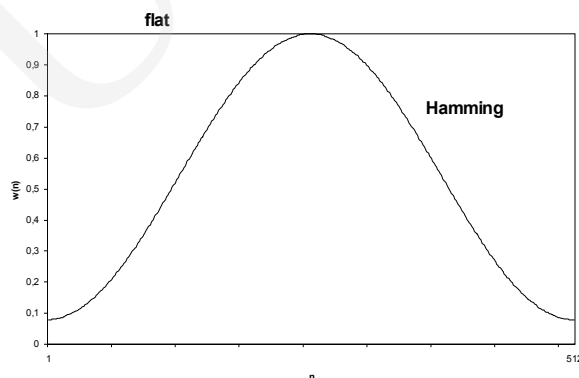Fig. 1. The main window of the „Spektrum" program

Fig. 2. The Hamming's weight function

3. The „Spektrum" program construction

The program was written under Borland C++ Builder 6.0 environment. There were elements of DirectShow [5,6] library (part of DirectX library for video processing) utilized.

The heart of the application developed with the use of the library is the concept of a Filter Graph. The filter graph consists of objects called filters, which perform specialized operations on a data stream. Before the moment of

the proper operation of the application, the filters must be properly connected to create the filter graph. The application usually controls a filter graph by calling the methods of helper objects (Filter Graph Managers) and by processing of the messages generated by the filters in the graph.

The "Spektrum" program uses several graphs: basic graph for handling "avi" files, a graph for saving selected audio-video fragments to files and a graph for handling "wav" files.

The basic filter graph consists of the following objects (Fig. 3): AsyncReader, AviSplitter, MySound, VideoRenderer and DSoundRender.
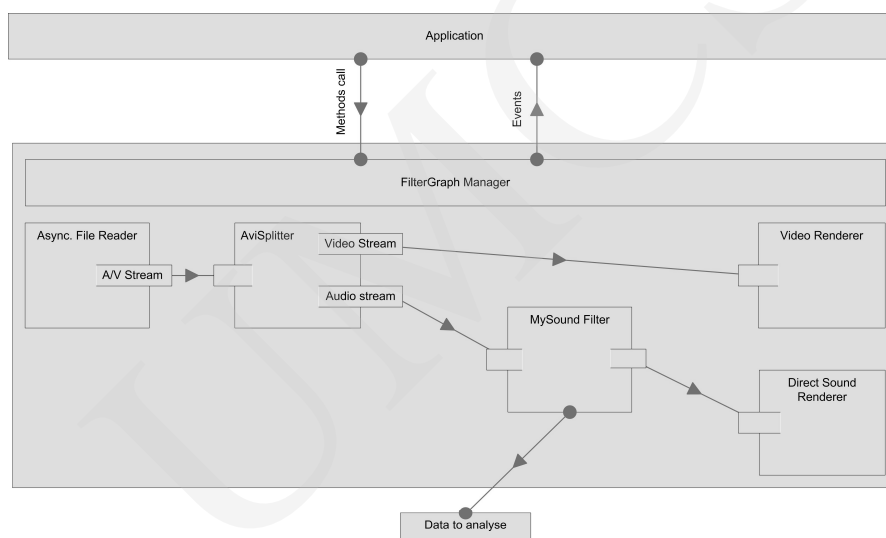

Fig. 3. The basic filter graph of the "Spektrum" program

The AsyncReader object allows for reading the content of a video file and passing it down to the AviSplitter object. That object separates sound data from the video stream and passes it to the MyFilter object. The basic function of the element is copying earlier the settled portion of data from the sound stream and placing them in the buffer for further processing. At the same time data are transferred to the DSoundRender object and sound is generated by the sound card.

The data of the video stream extracted by the AviSplitter object are directly passed to the VideoRenderer object and then rendered on a monitor screen.

The graph for saving selected audio-video fragments is an extension of the basic graph with the following filters: InfTee, AviDest, FileWriter. The InfTee object is used for quantity duplication of the audio and video stream. These extra streams are then passed to the AviDest object where they are prepared to be saved to the "avi" file format. Such prepared data are written to a hard disk with the FileWriter object.

In the case of handling "wav" files, the following objects were used: AsyncReader, WAVEParser, MySound and DSoundRender. The "wav" type file is read by the AsyncReader object and the content is passed to the WAVEParser object. Next, the data, in the form of audio stream, are transferred to MySound object and further rendered by the DSoundRender object. For filter graphs creation, helper objects CaptureGraphBuilder2 and FilterGraph were used.

Connections between filters are made with the help of the RenderStream() method from the CaptureGraphBuilder2 object.

The following objects: AsyncReader, AviSplitter, DSoundRender, VideoRenderer, InfTee, AviDest, FileWriter and WAVEParser are included in the DirectX library. The MySound filter has been implemented by the author.

Extracted sound data are immediately shown on the screen in the form of an oscillogram and further can be put through the Fourier analysis procedure. The process of Fourier analysis is as follows. The consecutive portion of data, whose size is defined by program configuration (the window size), is taken from the buffer. Next the data can be, in the case of need multiplied by the Hamming weight function (Fig. 2) [5]. Such prepared data are processed by DFT or FFT algorithm and the results are put into the computer memory as a table. The process is repeated until the data in the buffer have run out.

After the Fourier analysis is done, an earlier defined dynamics range of data is selected from the results and displayed in the form of a colorful spectrogram (Fig. 1).

4. Examples of applications

For the program test recording of people suffering from disturbances of speech fluency was made. The digital camcorder was used for recording. Then the obtained recording (in DV format, PAL standard) was processed with the help of the VirtualDub application [6]. Video samples were cropped so that only the mouth of a speaking person was visible and then saved to a file in the form accepted by the "Spectrum" program. Such prepared recordings were subject to analysis by the author's program.

Figure 4 shows an analysis of a recording where a patient says the words "między przesmyk". One can easily notice that the patient has problems with the beginning of the articulation of the word "przesmyk" (marked on the oscillogram). Between the words "między" and "przesmyk" there is an unnatural gap which can be classified as a blockade. Additionally, during the gap an intrusion is noticeable (laryngeal "y") which is an attempt at staring of the articulation of the next word.
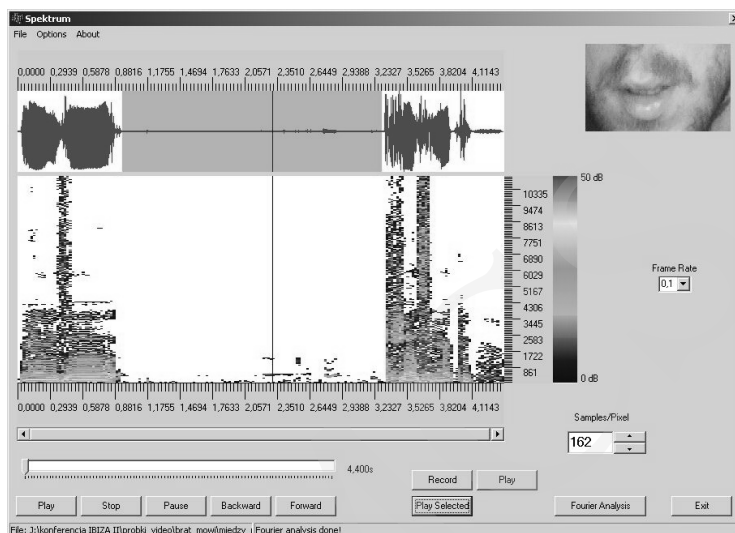
Fig. 4. The non-fluent articulation of the words: "między przesmyk"

Mouth movement is a very interesting source of information in that case. During the gap between the words it is clearly visible that the patient tries to prepare himself to express the next word.

Picture 5 presents an analysis of the fluent articulation of the above mentioned expression. The pictures of the mouth match the cursor position on the oscillograms and spectrograms. Comparing both pictures one can see that during non-fluent speech (Fig. 4) the mouth is open. Such a situation does not happen during proper speech (Fig. 5.).
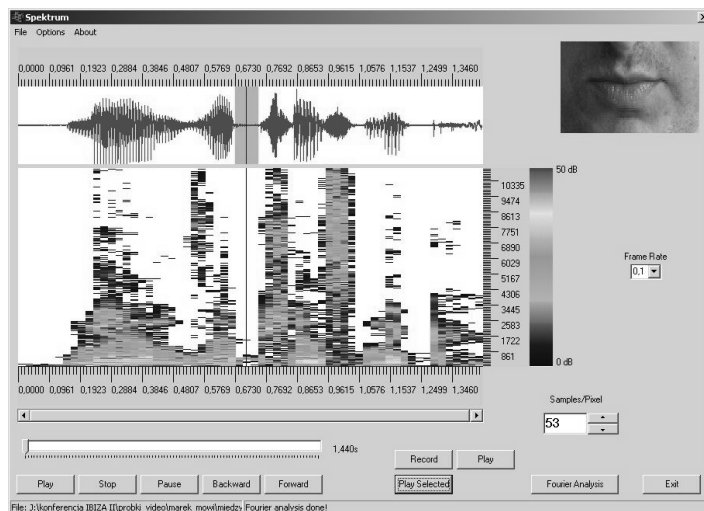

Fig. 5. The fluent articulation of the words: "między przesmyk"

The next non-fluent speech example is shown in figure 6. A young boy says the phrase "myślę, że był głodny" and commits two errors. The first one is the lengthening of the "ż" phoneme and the second is the repetition of the "gło" syllable. During the "ż" phoneme articulation, the boy arranges his mouth in a specific way – they are strongly extended horizontally (shown in the picture). During the articulation of the word "głodny", the boy also makes characteristic moves. It can be a valuable hint in the diagnostic process.
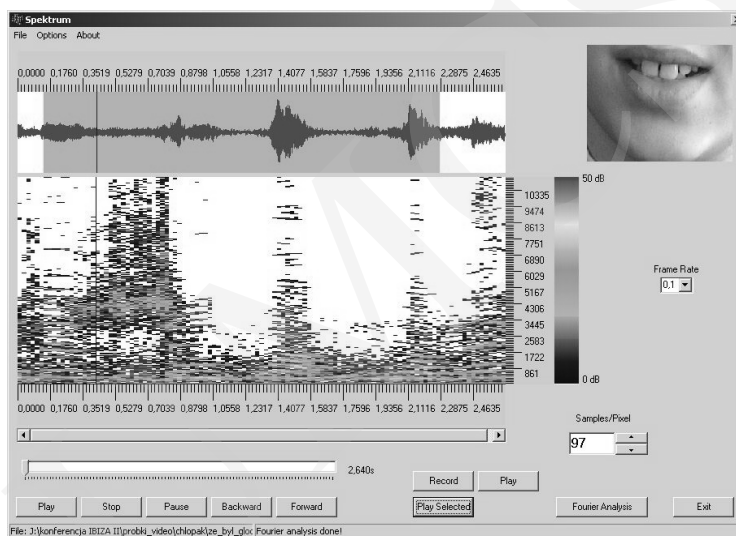

Fig. 6. The non-fluent articulation of the phrase: "myślę, że był głodny"

Conclusions

Diagnostic methods used for speech disturbance examination need to be continually improved. It is obvious that computer methods are utilized for that. The main aim is to free a therapist from arduous examinations as well as to increase the objectivism of diagnosis. The most recent studies led up to the development of a method for automatic detection and classification of speech disturbances [2]. They are based on the analysis of acoustical recordings and for obvious reasons, not always satisfactory results are achieved.

The described above the "Spektrum" program can be very helpful in the diagnostic process and can facilitate the search for characteristic visual features that appear in non-fluent speech. They can be used as additional parameters for the improvement of automatic detection methods.

Acknowledgements

References

[1] Kuniszyk-Jóźkowiak W., Smołka E., Suszyński W., *Computer audio-visual diagnosis of speech disorders*, XII BIB Conference, Warszawa, (2001).
[2] Suszyński W., *Computer analysis and detection of speech disorders*, Doctoral Dissertation, Gliwice, (2005).
[3] Microsoft DirectX 9.0 SDK
[4] http://clootie.narod.ru/cbuilder/
[5] Tadeusiewicz R., *The speech signal*, Warszawa, (1988).
[6] http://www.virtualdub.org/