Politechnika Śląska
Wydział Automatyki, Elektroniki
i Informatyki
Instytut Informatyki

Autoreferat rozprawy doktorskiej

Komputerowe rozpoznawanie niepłynności mowy z zastosowaniem transformaty falkowej i sztucznych sieci neuronowych

Ireneusz Codello

Promotor: dr hab. Wiesława Kuniszyk-Jóźkowiak, prof. nadzw.

Praca wykonana w Zakładzie Biocybernetyki Instytutu Informatyki
Uniwersytetu Marii Curie-Skłodowskiej,
w ramach studiów doktoranckich Politechniki Śląskiej na kierunku Informatyka,
przedstawiona Radzie Wydziału Automatyki, Elektroniki i Informatyki
Politechniki Śląskiej, jako rozprawa doktorska.

GLIWICE 2014

Wstęp

Analiza mowy jest dziś bardzo ważną gałęzią informatyki - ustna komunikacja z komputerem może być pomocna przy pisaniu dokumentów, tłumaczeniu języków lub po prostu w codziennej pracy przy komputerze. Bardzo praktycznym zastosowaniem jest również rozpoznawanie mówców, płci czy analiza schorzeń narządów traktu głosowego [4] [5] [18] [19] [20] [21] [22] [23] [24] [31] [68] [71]. Z tego powodu to zagadnienie (oraz inne zagadnienia przetwarzania sygnałów np. sonarowych czy sejsmicznych) jest analizowane od wielu lat, co doprowadziło do stworzenia lub zaadoptowania na potrzeby cyfrowego przetwarzania sygnałów (ang. digital signal processing – DSP) wielu algorytmów służących do ekstrakcji danych, takich jak: transformata Fouriera [29] [41] [53] [62], analiza cepstralna [10] [26] [41], liniowa predykcja [7] [32] [33] [35] [39] [41], analiza falkowa [9] [15] [16] [17], filtry [51] i inne... jak i algorytmów rozpoznawania mowy, mówców lub innych cech w sygnale: ukryte łańcuchy Markowa [69] [70] [72], sieci neuronowe [30] [45] [46] [47] [50] [49] [48] [56] [55], logika rozmyta [51], metody analityczne jak analiza obwiedni [36], współczynników korelacji [52] i inne.

Odsłuchowe oznaczanie niepłynności przez terapeutę jest obarczone dużym błędem. Trzeba wielokrotnie odsłuchać nagranie, żeby wychwycić wszystkie zaburzenia. Ponieważ percepcja terapeuty znacząco zależy od jego wypoczęcia, poziomu koncentracji w danej chwili oraz od innych subiektywnych czynników – terapeuta ten sam fragment raz może uznać jako płynny, a po ponownym odsłuchaniu jako niepłynny. Automatyczne rozpoznawanie niepłynności, które działa zawsze na podstawie takich samych reguł – jest bardziej obiektywne. Ponadto automatycznie generowane statystyki mowy patologicznej byłoby dużym ułatwieniem dla terapeutów w ich próbach oszacowania postępów terapii. Terapeuta mógłby nagrywać wypowiedzi pacjenta podczas terapii i uruchamiać program zliczający niepłynności w tych nagraniach, ponieważ robienie tego manualnie jest procesem żmudnym i czasochłonnym. Oprócz statystyk liczby i czasów trwania niepłynności, terapeuta otrzymywałby listę wszystkich niepłynności, dzięki której łatwo mógłby wyszukiwać i odsłuchiwać wyselekcjonowane fragmenty. Narzędzie automatycznie rozpoznające niepłynności:

- wyręczyłoby terapeutę w żmudnym zliczaniu i klasyfikowaniu niepłynności w nagraniach pacjenta,

- zwiększyłoby obiektywizm oceny – w terapii, z sesji na sesję, pokazując statystyki oparte na dokładnie takich samych regułach wyszukiwania (brak subiektywnych „ludzkich" czynników), co mogłoby mieć duże znaczenie dla diagnozy i wyboru ćwiczeń dla pacjenta,
- zaoszczędziłoby czas terapeuty, który mógłby go przeznaczyć chociażby na dłuższe sesje terapeutyczne.

Cel i tezy pracy

Celem pracy jest opracowanie systemu do automatycznego rozpoznawania niepłynności w mowie ciągłej z precyzyjną lokalizacją ich w czasie. Przyjęte zostały następujące tezy:

1) **Parametryzacja sygnału mowy przy zastosowaniu ciągłej transformaty falkowej pozwala na optymalne przybliżenie własności percepcyjnych słuchu ludzkiego, przy odpowiedniej konstrukcji falki macierzystej. Taki sposób parametryzacji ma bardzo istotne znaczenie przy rozpoznawaniu niepłynności, gdyż tworzony system powinien naśladować odbiór zaburzeń przez słuchaczy i samego mówiącego.**
2) **Optymalnym sposobem redukcji nadmiarowości wektorów wejściowych jest zastosowanie sieci Kohonena, pozwala ona bowiem na nieeliminowanie czasu, który w przypadku tych zaburzeń odgrywa istotną rolę. Ponadto odpowiednia modyfikacja algorytmu uczenia tych sieci daje możliwość bardzo dobrego odwzorowania przedłużeń oraz powtórzeń głosek.**
3) **Sieci neuronowe są bardzo dobrymi klasyfikatorami powtórzeń głosek.**
4) **Klasyfikacja powtórzeń sylab jest możliwa przy zastosowaniu odpowiednio dostosowanych algorytmów korelacyjnych.**
5) **Możliwe jest skutecznie rozpoznawanie niepłynności w mowie ciągłej z precyzyjną lokalizacją w skali czasu przy zastosowaniu ciągłej transformaty falkowej z konstrukcją skal barkowych.**

Przegląd dotychczasowych badań rozpoznawania niepłynności mowy przedstawiony został w rozdziale 3 rozprawy doktorskiej. Z analizy tej literatury wynika:

- badacze skupiają się na detekcji przedłużeń, powtórzeń głosek oraz powtórzeń sylab (rzadko pojawiają się też blokady oraz wtrącenia),

- do tej pory nikt nie próbował używać algorytmu CWT do parametryzacji sygnału mowy w celu detekcji niepłynności,
- prawie wszyscy korzystają z odsłuchowo segmentowanych danych wejściowych – jedynie Suszyński, Kuniszyk-Jóźkowiak [53] oraz Wiśniewski, Kuniszyk-Jóźkowiak [69] opracowali algorytmy rozpoznawania przedłużeń w mowie ciągłej.

Część badań opisanych w tej pracy została opublikowana w 12 pracach [6] [7] [8] [9] [10] [11] [12] [13] [14] [15] [16] [17]. Najnowsza z tych prac [13] uzyskała wyróżnienie na międzynarodowej konferencji CORES 2013 oraz uzyskała zaproszenie do publikacji pokonferencyjnej w czasopiśmie "Pattern Analysis and Applications" i w wersji rozszerzonej została wysłana do druku.

Do pracy dołączono 4 pliki multimedialne prezentujące:

- ogólny opis funkcjonalności programu WaveBlaster,

- procedurę detekcji przedłużeń przy użyciu programu WaveBlaster,

- procedurę detekcji powtórzeń głosek przy użyciu programu WaveBlaster,

- procedurę detekcji powtórzeń sylab przy użyciu programu WaveBlaster.

Parametryzacja sygnału mowy

Do parametryzacji zastosowano algorytm ciągłej transformaty falkowej [1] [3] [26] [39] [40] [43] [66]. Aby ją opisać, najpierw należy przedstawić definicję falki, czyli funkcji $\psi(n)$, która spełnia następujące warunki:

- jej wartość średnia wynosi 0,
- ma wartości niezerowe tylko na skończonym przedziale $<u,v>$.

Ponadto falka może zaczynać i kończyć się próbkami o wartości 0, czyli $\psi(u)$ nie musi być pierwszą niezerową próbką, a $\psi(v)$ nie musi być ostatnią niezerową próbką.

Z każdej z falek bazowych tworzy się rodzinę falek względem parametrów a i b:

$$\psi_{a,b}(n) = \frac{1}{\sqrt{a}}\psi\left(\frac{n-b}{a}\right)$$

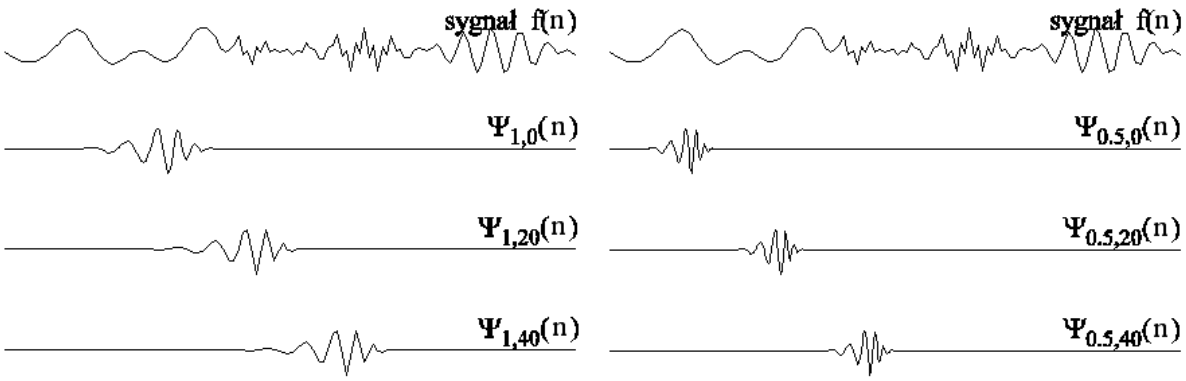
gdzie: *a* – skala, *b* – przesunięcie, *n* – numer próbki (czas), *ψ(n)* – falka bazowa.

Współczynnik *a* odpowiada za skalowanie poziome falki (rozciąganie i zwężanie), natomiast współczynnik *b* odpowiada za przesunięcie falki.

Ciągłą transformatę falkową dla sygnałów dyskretnych można przedstawić wzorem:

$$CWT_{a,b} = \sum_{n} f(n) \cdot \psi_{a,b}(n)$$

gdzie: n – numer próbki (czas), *f(n)* – sygnał wejściowy, $\psi_{a,b}(n)$ – falka bazowa w skali *a* i przesunięciu *b*.



rys. 1 Schemat obliczania CWT

W pracy przyjęto, że przeciwne wartości $CWT_{a,b}$ odzwierciedlają ten sam stopień podobieństwa (czyli wartość 2 i -2 oznaczają takie samo podobieństwo). W związku z tym do wszystkich obliczeń brane są wartości $|CWT_{a,b}|$. Ponadto wszędzie użyto relatywnej skali decybelowej, gdzie największej wartości skalogramu CWT_{MAX} przyporządkowano wartość 0dB, a pozostałe wartości decybelowe są ujemne:

$$CWT^0_{a,b} = 20\ log_{10}\left(\frac{|CWT_{a,b}|}{|CWT_{MAX}|}\right)$$

gdzie: CWT_{MAX} – największa wartość *CWT* dla wszystkich skal *a* i przesunięć *b*.

Na falkę macierzystą wybrano falkę powstałą na bazie rzeczywistej falki Morleta [27], ze względu na jej kształt (rys.2), który jest podobny do przebiegu sygnału mowy:

$$\psi^*(t) = e^{-t^2/2}\cos(2\pi F_C t)$$

gdzie: F_C – częstotliwość środkowa falki, t – czas.

W literaturze można znaleźć również znormalizowaną postać tej falki [37] [42] – jest ona przemnożona przez stałe $\frac{1}{\sqrt[4]{\pi}}$ lub $\frac{1}{\sqrt{b\pi}}$ (b – współczynnik szerokości pasma). Można je jednak pominąć - ponieważ autor pracy stosuje w skalogramach CWT relatywną skalę decybelową, z której wynika, że dowolna stała przemnażająca falkę zostanie skrócona w wyniku dzielenia $\frac{|CWT_{a,b}|}{|CWT_{MAX}|}$.

Falka ta ma jeszcze jedną bardzo istotną zaletę – może mieć różną częstotliwość środkową F_C, ponieważ F_C jest parametrem funkcji cosinus. Jako, że badacze podają różne dolne zakresy częstotliwości słyszalnych przez człowieka (od 15 do 20 *Hz*), za F_C przyjęto wartość *20Hz*. Na potrzeby tej rozprawy nazwiemy tę falkę Morlet20.

rys.2 Falka Morleta20 (po lewej) oraz jej widmo Fouriera (po prawej)

CWT, z racji tego, że jest ciągłą transformatą, może być wyliczona dla dowolnych skal *a*. Jednak wyliczanie ich wszystkich wydaje się nadmiarowe i na pewno wymagające obliczeniowo – ze względu na samo CWT jak i na późniejsze jego przetwarzanie (skalogram składałby się z ogromnej liczby danych). Uznano, iż rozpoznawanie niepłynności na bazie sygnału sparametryzowanego przy zastosowaniu skali percepcyjnej, czyli bazującej na charakterystyce słuchu człowieka, powinno przynieść zadowalające rezultaty. Z tego powodu zdecydowano się na wybór skali barkowej, jednej z kilku popularnych skal percepcyjnych (takich jak skala melowa czy ERB [44]).

Wybrano reprezentację Hartmuta [67]:

$$B = \frac{26.81}{1+1960/f} - 0.53$$

gdzie: f – częstotliwość w Hz.

Surowy wynik CWT^0 obliczony dla skal barkowych, jest punktem wyjścia dla wszystkich użytych metod detekcji niepłynności mowy prezentowanych w tej pracy. Oznacza to, że wszystkie algorytmy post-procesingu oczekują skalogramu, jako danych wejściowych. W przypadku przedłużeń i powtórzeń głosek zastosowano sieci Kohonena do grupowania wektorów, a w przypadku powtórzeń głosek dodatkowo użyto perceptronu 3-warstwowego przy klasyfikacji fragmentów do zbiorów płynny/niepłynny.

Grupowanie wektorów

W procedurach detekcji niepłynności zastosowano sieci Kohonena [25] [28] [34] [56] [57] [58] [59], ze względu na bardzo dobrą właściwość grupowania, świetnie nadają się do redukcji wymiaru danych [57] [58] [59]. Można zamienić wynik analizy CWT (czyli skalogram), który reprezentuje przestrzeń 3D na dane dwuwymiarowe w postaci sekwencji indeksów wygrywających neuronów. W tym celu należy podzielić skalogram na odcinki (okna) wzdłuż osi czasu, a następnie, używając ich jako wektorów wejściowych (jedno okno – to jeden wektor), wytrenować sieć Kohonena. Na końcu, za pomocą już wytrenowanej sieci, należy dla każdego wektora wejściowego (czyli okna skalogramu) wyznaczyć jedną wartość – numer wygrywającego neuronu. Tak zredukowany wynik jest łatwiejszy do analizy z powodu mniejszej ilości danych.

Przed treningiem sieć jest inicjowana danymi losowymi, dlatego też ten sam ciąg wektorów wejściowych może dać inną sekwencję indeksów wygrywających neuronów. Dzieje się tak, ponieważ ze względu na tę losowość, tworzy się taka sama liczba skupisk (map), ale w innych rejonach sieci. Autor rozprawy wprowadził następującą modyfikację algorytmu trenowania – **‘zerowanie pierwszego neuronu’**. Po zainicjowaniu sieci, neuron o numerze 0 (lewy górny róg w topologii prostokątnej) jest zerowany i oznaczany jako ‘tylko do odczytu’. Bierze on udział we wszystkich obliczeniach, ale jego wagi nie są zmieniane, więc zawsze są zerowe. Z tego powodu neuron ten zawsze przyciąga ciszę i inne ‘słabe’ sygnały do lewego górnego rogu sieci. W naturalny sposób, ze względu na modyfikację sąsiadów, wektory o średnich wartościach są grupowane w środkowych rejonach sieci, a wektory wejściowe o największych wartościach umieszczane są w prawym dolnym rogu.

Neurony sieci Kohonena grupują podobne wektory wejściowe, dlatego oczekiwano, iż na odcinkach z przedłużeniami będzie wygrywał tylko jeden neuron. Efekt ten uzyskiwano tylko wtedy, jeżeli liczba neuronów sieci (czyli jej rozmiar) była zbliżona do liczby fonemów we fragmencie. Jeżeli neuronów było za mało w stosunku do liczby fonemów w sygnale wejściowym – odmienne fonemy były grupowane przez ten sam neuron (czyli uzyskiwaliśmy wrażenie przedłużenia w płynnych fragmentach), jeżeli neuronów było za dużo – kilka neuronów grupowało ten sam fonem (czyli uzyskiwaliśmy efekt płynności we fragmentach z zaburzeniami). Fragmenty z mową płynną posiadają dużo większe zróżnicowanie fonemów od fragmentów tej samej długości zawierających przedłużenie, ponieważ przedłużenie wypełnia jego znaczną część. Ponieważ rozmiar sieci musi być stały dla całej wypowiedzi (tj. dla wszystkich fragmentów) dobranie rozmiaru sieci, która odwzorowałaby tylko jeden wygrywający neuron tylko dla odcinka z przedłużeniem okazało się niemożliwe.

Z tego powodu, po wytrenowaniu sieci Kohonena wektorami CWT^0, ale przed wygenerowaniem sekwencji indeksów neuronów wygrywających, zastosowano dodatkową, autorską modyfikację nazwaną **'redukcją sieci Kohonena'**. Ideą 'redukcji' jest wygładzenie sekwencji indeksów wygrywających neuronów sieci Kohonena na odcinkach, na których znajduje się ten sam fonem. Procedura redukcji jest następująca:

- Znajdź dwa najbliższe neurony k_A, k_B (odległość mierzona metryką Euklidesową pomiędzy wagami neuronów).
- Jeżeli odległość jest mniejsza niż zadany dystans ε wypełnij wagi 'słabszego' neuronu zerami (tj. neuronu z niższą ilością przypisanych wektorów wejściowych). Dzięki temu wektory wejściowe 'słabszego' neuronu będą najprawdopodobniej przypisywane 'silniejszemu' neuronowi:

$$\begin{cases} |\overrightarrow{k_A} - \overrightarrow{k_B}| < \varepsilon, \overrightarrow{k_A} \vee \overrightarrow{k_B} = \vec{0} \\ w\ p.p., \quad nic\ nie\ rób \end{cases}$$

gdzie:

$\overrightarrow{k_A}, \overrightarrow{k_B}$ – wektory reprezentowane przez wagi neuronów k_A, k_B,

ε – dystans, czyli wartość graniczna odległości pomiędzy wektorami $\overrightarrow{k_A}$ *i* $\overrightarrow{k_B}$.

- Powtarzaj dwa poprzednie kroki dopóki istnieje para neuronów, których odległość jest mniejsza od w/w progu.

Klasyfikacja

Na podstawie badań Szczurowskiej [54] [55] [57] [58] [59], do klasyfikacji powtórzeń głosek zastosowano perceptron 3-warstowyw [60] [61] [62] [63] [64] [65] uczony algorytmem wstecznej propagacji i gradientów sprzężonych. Wykazała Ona ich wysoką przydatność w wykrywaniu wzorców mowy niepłynnej. Dla pozostałych niepłynności, do klasyfikacji użyto metod analitycznych.

Algorytmy do automatycznego rozpoznawania niepłynności w mowie ciągłej

Rozpoznawanie niepłynności realizowano według następującego planu:

1. parametryzacja plików dźwiękowych algorytmem CWT,
2. uzyskanie sekwencji indeksów wygrywających neuronów na podstawie skalogramów CWT,
3. analiza tak uzyskanych sekwencji i dobranie odpowiednich algorytmów rozpoznawania dla każdej grupy niepłynności,
4. zastosowanie miar oceny wyników rozpoznawania (tj. parametrów czułości i przewidywalności) [2]:

$$czul = \frac{P}{A}, przew = \frac{P}{P + B}$$

gdzie: *czul* – czułość (sensability), *przew* – przewidywalność (predictability),

P – liczba poprawnie rozpoznanych niepłynności,

B – liczba płynnych fragmentów błędnie oznaczonych jako niepłynne,

A – liczba niepłynności.

Rodzaje niepłynności analizowane w tej rozprawie istotnie różnią się od siebie – z tego powodu autor pracy postanowił każdy z nich analizować niezależnie, aby jak najlepiej dopasować algorytmy rozpoznawania do rodzaju wykrywanych cech. Proces ten można podzielić na cztery etapy:

1. Analiza materiału dźwiękowego – tj. czasów trwania niepłynności w badanych sygnałach, jak i czasów przerw między nimi, czasów ciszy przed lub po niepłynności na podstawie obszernego zbioru niepłynnych wypowiedzi osób jąkających się i porównaniu ich z płynnymi odpowiednikami. Na tym etapie utworzono różne statystyki czasowe, na bazie których wyznaczano warunki brzegowe dla analizowanych fragmentów mowy.
2. Analiza wyników parametryzacji fragmentów niepłynnych oraz ich płynnych odpowiedników. Na tym etapie szczegółowo przeanalizowano skalogramy CWT dla fragmentów płynnych i niepłynnych jak również ich sekwencje indeksów neuronów wygrywających sieci Kohonena. Na tej podstawie zaproponowano algorytmy rozpoznawania danej niepłynności.
3. Utworzenie procedur detekcji niepłynności, wyselekcjonowanie przestrzeni parametrów, które będą w nich zmieniane.
4. Przeprowadzenie rozpoznawania niepłynności w/w procedurami – wybranie wartości początkowych dla parametrów oraz modyfikację tych wartości w kolejnych etapach testowania na podstawie analiz otrzymywanych wyników. Ostateczne wyselekcjonowanie najlepszych wyników rozpoznawania.

Procedura detekcji przedłużeń

Przyjęto następującą procedurę detekcji przedłużeń (podkreślone elementy są parametrami algorytmu):

1)	wczytaj plik dźwiękowy
2)	oblicz współczynniki CWT^0 dla skal barkowych
3)	wykryj fragmenty mowy dla odcięcia szumów na poziomie -55dB dla szerokości okna 23,2ms i 50% przesunięcia okna (50% dla zwiększenia rozdzielczości wykrywania fonacji) *następnie każdy fragment podziel na okna o szerokości 23,2ms i 100% przesunięcia okna i ponownie oblicz wektory*
4)	dla fragmentów mowy dłuższych niż 200ms:

5) wytnij fragment CWT^0 z zadanym otoczeniem (*otocz_wycinania)*

6) zredukuj współczynniki CWT^0 do jednego indeksu neuronu zwycięskiego w każdym oknie dla sieci Kohonena o zadanym rozmiarze (*rozm_Koh*) i zadanym sąsiedztwie (*sas_Koh*). Sieć uczona była przez 100 epok z liniowo malejącym współczynnikiem uczenia 0.20-0.10

7) dla tak wytrenowanej sieci Kohonena zastosuj 'redukcję' neuronów z zadanym dystansem (*dyst_redukcji*) i ponownie wygeneruj sekwencję indeksów neuronów zwycięskich

8) jeżeli w w/w sekwencji istnieje odcinek dłuższy od zadanej długości (*dl_sekwencji*), w którym wygrywa tylko jeden neuron, oznacz go jako przedłużenie

9) na podstawie odsłuchowych i automatycznych zaznaczeń wygeneruj współczynniki wykrywalności *czul* i *przew*

Procedura detekcji powtórzeń głosek

Przyjęto następującą procedurę detekcji powtórzeń głosek (podkreślone elementy są parametrami algorytmu):

1) wczytaj plik dźwiękowy

2) oblicz współczynniki CWT^0 dla skal barkowych

3) wykryj fragmenty mowy (fonacje) dla zadanego odcięcia szumów (*poziom_szumów*) i zadanej minimalnej przerwy pomiędzy wyrazami (*min_przerwa*) – dla szerokości okna 23,2ms i 50% przesunięcia okna
następnie każdy fragment podziel na okna o szerokości 23,2ms i 100% przesunięcia okna i ponownie oblicz wektory

4) dla fonacji z plików plik2 i plik3 (czyli około 75% danych) krótszych niż 200ms:

5) wytnij fragment CWT^0 z zadanym otoczeniem (*otocz_wycinania*) – każdy fragment z zaburzeniem składał się z 500 milisekundowego prefixu, następnie niepłynności oraz postfixu odpowiedniej długości tak, aby łączna długość

	była równa zadanemu otoczeniu
6)	zredukuj CWT^0 do sekwencji indeksów wygrywających neuronów sieci Kohonena – na podstawie wyników wstępnych testów użyto tylko 16 skal barkowych: 6,7,..21 oraz tylko sieci 5x5 uczoną dla 100 epok przy wsp. uczenia 0.20-0.10 i wsp. sąsiedztwa 2.5-0.5.
7)	manualnie oznacz fragment jako płynny/niepłynny
8)	używając narzędzia „Intelligent Problem Solver" pakietu STATISTICA znajdź najlepszy perceptron 3-warstwowy dla danych z kroku 7) i wyeksportuj jego wagi do programu WaveBlaster
9)	dla 100% wyrazów krótszych niż 200ms:
10)	wytnij fragment CWT^0 i wygeneruj sekwencję indeksów wygrywających neuronów sieci Kohonena
11)	używając nauczonego perceptronu oznacz fragment jako płynny/niepłynny
12)	na podstawie odsłuchowych i automatycznych zaznaczeń wygeneruj statystyki wykrywalności

Procedura detekcji powtórzeń sylab

Powtarzane sylaby generują podobne wartości współczynników CWT, dlatego oczekiwano, że odpowiadające im sekwencje indeksów wygrywających neuronów sieci Kohonena będą również zbliżone. Dzięki temu, traktując w/w sekwencje indeksów jako wektory n-wymiarowe, będzie można wyliczać odległość między nimi (powinna być mała) lub wyliczać dla nich wartość korelacji (powinna być wysoka). Niestety, mimo podobieństw na poziomie współczynników CWT, nie udało się uzyskać satysfakcjonujących wyników na podstawie w/w wektorów. W tej sytuacji zdecydowano się na korelowanie wartości skalogramu CWT (bez udziału sieci Kohonena).

Przyjęto następującą procedurę detekcji powtórzeń sylab (podkreślone elementy są parametrami algorytmu):

1)	wczytaj plik dźwiękowy

2) oblicz współczynniki CWT^0 dla zadanych skal barkowych (*skale_barkowe*) i falki Morleta o zadanej częstotliwości środkowej (*Fc*)

3) wykryj fragmenty mowy (fonacje) dla szerokości okna 23,2ms i 50% przesunięcia okna

   następnie każdy fragment podziel na okna o szerokości 23,2ms i 100% przesunięcia okna i ponownie oblicz CWT^0

4) dla każdej pary sąsiadujących fragmentów z których:

   - pierwszy jest dłuższy niż 70ms i krótszy niż 500ms
   - drugi jest krótszy od pierwszego o co najwyżej 100ms

5) wyodrębnij odpowiadający fragment CWT^0 dla tych wyrazów dla zadanego odcięcia szumów (*odc_szumów*)

6) oblicz ich współczynnik korelacji

7) wyznacz optymalną wartość graniczną G oraz linię graniczną $L=cx+d$ korelacji , oddzielającą płynne i niepłynne powtórzenia sylab

8) dla tak wyznaczonej granicy korelacji, oznacz automatycznie pary sylab jako płynne/niepłynne

9) na podstawie odsłuchowych i automatycznych zaznaczeń wygeneruj statystyki wykrywalności

Program WaveBlaster

Autor rozprawy stworzył program „WaveBlaster” – w ogromnej mierze ułatwiający proces badawczy.

Funkcjonalność prezentowanego programu obejmuje wiele algorytmów, paneli konfigurujących i wizualizujących wyniki tych algorytmów, paneli umożliwiających wczytywanie oraz zapis wszelakich danych wejściowych, wyjściowych i konfiguracyjnych. Duża część tak rozbudowanej aplikacji była tworzona na potrzeby badań prowadzonych pod kątem tej rozprawy – mimo, że ostatecznie wykorzystano tylko niewielką jej część do napisania

tej pracy (obrazuje to jak wiele analiz i opcji było sprawdzanych, zanim autor uzyskał satysfakcjonujące wyniki rozpoznawania niepłynności w mowie ciągłej). Dzięki temu, że WaveBlaster zawiera różnorakie algorytmy z bogatą liczbą opcji do ich konfiguracji (DWT/CWT z dużą ilością falek, analiza Fouriera, analiza liniowej predykcji, generowanie modelu traktu głosowego, detekcja formantów, wyliczanie filtrów tercjowych, mapy Kohonena, algorytmy pre-empfazy, korelacji, k-Means, wyliczanie energii i obwiedni sygnału) – program ten stał się ogólnym narzędziem do analizy sygnałów (na przykład sygnału EMG [38]). Mimo, iż stworzenie narzędzia WaveBlaster pochłonęło bardzo dużo czasu, przyspieszyło to (lub wręcz umożliwiło) uzyskanie satysfakcjonujących wyników badań. Dzięki łatwej i szybkiej obsłudze, przeprowadzono nim niezliczoną ilość eksperymentów (analizy i wyniki przedstawione w tabelach tej rozprawy stanowią niewielką ich część).

Wszystkie komponenty programu, mimo iż dość liczne, są ze sobą powiązane. Dzięki temu szukając zależności, regularności w badanym sygnale, możemy oglądać dane z wielu perspektyw (powiększanie, pomniejszanie, przesuwanie, zmiana kolorów i skal). Dokładny system skal i miar jest bardzo przydatny w poszukiwaniu/oglądaniu choćby najmniejszych szczegółów. Wszystkie wykresy są sprzężone ze sobą (powiększanie, pomniejszanie, przesuwanie, zaznaczanie),
w związku z tym, dla wybranego fragmentu możemy jednocześnie oglądać i porównywać wyniki wielu analiz (oscylogram, spektrogramy, widma, obrys sieci ohonena) – co jest bardzo przydatną funkcjonalnością.

Prawie wszystkie badania przedstawione w niniejszej rozprawie zostały przeprowadzone z wykorzystaniem tego programu. Wyjątek stanowi wyszukiwanie najlepszego perceptronu przy detekcji powtórzeń głosek, gdzie dodatkowo użyto pakietu STATISTICA.

Podsumowanie

Przegląd użytych metod

Autor pracy opracował algorytmy rozpoznawania trzech rodzajów niepłynności, najczęściej badanych przez naukowców (wnioski z rozdziału opisującego obecny stan badań), tj. przedłużeń, powtórzeń głosek, powtórzeń sylab. Dla każdego rodzaju opracowano odrębną procedurę rozpoznawania. We wszystkich procedurach:

- sygnał dźwiękowy najpierw parametryzowano ciągłą transformatą falkową przy wykorzystaniu falki Morlet20 (Morlet39,5 dla powtórzeń sylab) oraz skal barkowych,
- następnie dzielono go na okna i uśredniano wartości skal tworząc szesnasto lub osiemnasto elementowe wektory (w zależności od procedury),
- wektory o wartościach poniżej ‘odcięcia szumów’ oznaczano jako ciszę, wyznaczając na tej bazie fragmenty fonacji,
- dzięki dodatkowym kryteriom charakterystyk czasowych niepłynności (wyznaczonych na podstawie analizy materiału dźwiękowego), redukowano fonacje nie spełniające czasowych warunków brzegowych.

W przypadku detekcji przedłużeń:

- zmniejszano wymiar wektorów za pomocą sieci Kohonena z nowatorskim, zmodyfikowanym algorytmem trenującym (‘redukcja neuronów sieci Kohonena’),
- wyszukiwano fragmentów, w których indeks wygrywającego neuronów utrzymywał się zadany parametrem czas trwania i oznaczano je, jako przedłużenia.

W przypadku detekcji powtórzeń głosek:

- zmniejszano wymiar wektorów za pomocą sieci Kohonena z nowatorskim, zmodyfikowanym algorytmem trenującym (‘zerowanie pierwszego neuronu’),
- klasyfikowano wektory do zbiorów powtórzenie głoski/płynność przy użyciu wielowarstwowego perceptronu.

W przypadku detekcji powtórzeń sylab:

- obliczano wartości korelacji sąsiadujących fonacji,
- na podstawie wartości granicznej korelacji (lub linii granicznej), klasyfikowano fonacje do zbiorów powtórzenie sylaby/płynność.

Mimo iż w/w procedury różnią się od siebie, dzięki temu, że bazującą na wspólnych algorytmach parametryzacji i podobnej metodologii, są do siebie zbliżone – stanowiąc niejako rodzinę procedur.

Podsumowanie wyników rozpoznawania

Jako, że w mowie płynnej zbiór wektorów zawierających niepłynność jest dużo mniejszy od liczby wszystkich analizowanych wektorów, autor pracy uznał, iż pojedynczy parametr oznaczający liczbę wykrytych niepłynności byłby wysoce niewystarczający. Nie odzwierciedlałby on liczby pomyłek w mowie płynnej, który przy takiej dysproporcji wektorów płynnych do niepłynnych jest równie ważny. Z tego powodu, do oceny rozpoznawania zaburzeń zastosowano współczynniki czułości (wartość proporcjonalna do liczby wykrytych niepłynności) i przewidywalności (wartość odwrotnie proporcjonalna do liczby popełnionych błędów w mowie płynnej).

Do badań użyto nagrania dwudziestu osób jąkających się oraz czterech osób zdrowych. Charakterystykę użytego materiału dźwiękowego przedstawia poniższa tabelka:

tab. 1 Analiza badanego materiału dźwiękowego dla poszczególnych rodzajów niepłynności

rodzaj niepłynności	**łączny czas trwania nagrań**	**liczba niepłynności**
przedłużenia	18 min. 32 s.	373
powtórzenia głosek	9 min. 43s.	294
powtórzenia sylab	5 min. 26 s.	106

Widać, że analizowany materiał był dość obszerny i zróżnicowany, zatem wydaje się, że opracowane metody powinny być uniwersalne, tj. powinny być tak samo skuteczne dla dowolnych wypowiedzi – oczywiście wypowiedzi w języku polskim. Trudno powiedzieć, jakie wyniki moglibyśmy uzyskać dla innego języka – z racji pojawiania się w nim innych fonemów i innych ich sekwencji, odpowiedź nie jest oczywista.

Uzyskano następujące wyniki rozpoznawania:

tab. 2 Najlepsze wyniki rozpoznawania poszczególnych rodzajów niepłynności

rodzaj niepłynności	**czułość**	**przewidywalność**	**konfiguracja parametrów**
przedłużenia	92%	82%	falka Morleta20 okno 23,2 ms, 100% przesunięcia

			18 skal barkowych: 4-21 poziom odcięcia szumów: 55dB otoczenie wycinania: 2,5s. rozmiar sieci Kohonena: 4x4 sąsiedztwo uczenia sieci: 2,5-0,5 współczynniki uczenia: 0,2-0,1 długość uczenia: 100 epok dystans redukcji sieci: 0,55 minimalna długość sekwencji 250ms otoczenie wycinania 2000ms *wykrywanie fonacji:* falka Morleta20 okno 23,2 ms, 50% przesunięcia 18 skal barkowych: 4-21 poziom odcięcia szumów: 55dB
powtórzenia głosek	86%	95%	falka Morleta20 okno 23,2 ms, 100% przesunięcia 16 skal barkowych: 6-21 poziom odcięcia szumów 54dB otoczenie wycinania 1,5s rozmiar sieci Kohonena: 5x5 sąsiedztwo uczenia sieci: 2,5-0,5 współczynniki uczenia: 0,2-0,1 długość uczenia: 100 epok perceptron: MLP 65-83-1 uczenie perceptronu: BP100,CG28b minimalna przerwa między fonacjami 50ms *wykrywanie fonacji:* falka Morleta20 okno 23,2 ms, 50% przesunięcia

			16 skal barkowych: 6-21 poziom odcięcia szumów 54dB
powtórzenia sylab	81%	83%	falka Morleta39,5 okno 23,2 ms, 100% przesunięcia 16 skal barkowych: 6-21 poziom odcięcia szumów 60dB linia graniczna korelacji zbliżona do prostej y= 0,635650ms *wykrywanie fonacji:* falka Morleta20 okno 23,2 ms, 50% przesunięcia 16 skal barkowych: 6-21 poziom odcięcia szumów 55dB

Zaproponowane procedury rozpoznawania są w pełni automatyczne – fragmenty na żadnym etapie nie są manualnie (odsłuchowo) segmentowane ani oznaczane. Podejście polegające na automatycznym i sekwencyjnym wycinaniu dużej liczby wektorów mowy płynnej powoduje, że są one bardzo zróżnicowane i często są bardzo podobne do wektorów mowy zaburzonej – co stanowi dużą trudność. Z tego powodu, w początkowych fazach tworzenia każdej z procedur, wyniki rozpoznawania były bardzo niskie, mimo iż wzorowano się na sprawdzonych przez innych badaczy rozwiązaniach. Z tego względu powstała potrzeba opracowania i weryfikacji nowych algorytmów, dzięki którym uzyskano wysoką skuteczność identyfikacji niepłynności w mowie ciągłej.

Wnioski końcowe

Realizując cel pracy, udało się stworzyć program automatycznie rozpoznający niepłynności w mowie ciągłej z jednoczesnym, precyzyjnym wyznaczaniem początku i końca każdego zaburzenia.

Parametryzowanie sygnału mowy algorytmem ciągłej transformaty falkowej jest dobrym rozwiązaniem, dzięki któremu możemy elastycznie wybierać pasma częstotliwości, w których

chcemy wyznaczyć współczynniki spektrogramu. Parametryzacja taka wraz z zastosowaniem modelu percepcyjnego, polegającym na wyliczaniu spektrogramu dla skal barkowych, które odzwierciedlają charakterystykę słuchową człowieka, przyniosła zamierzone rezultaty – wyniki rozpoznawania niepłynności są wysokie.

Sieci Kohonena bardzo dobrze nadają się do redukcji danych wejściowych. Uzyskiwane sekwencje wygrywających neuronów bardzo upraszczają postać danych, czyniąc je łatwiejszymi do dalszej analizy. Przy dodatkowej modyfikacji algorytmu uczenia otrzymywano bardzo dobre (tj. łatwe do sklasyfikowania) odwzorowania przedłużeń i powtórzeń głosek. Powtórzenia sylab nie są już tak dobrze odwzorowywane (przynajmniej przy testowanych przez autora konfiguracjach parametrów sieci Kohonena) – to znaczy przebiegi wygrywających neuronów sieci Kohonena dla takich samych sylab są istotnie różne. Natomiast algorytm korelacyjny zastosowany bezpośrednio dla parametrów CWT (bez redukcji sieciami Kohonena) – generuje bardzo wysokie współczynniki korelacji.

Program WaveBlaster ma charakter badawczy. Zawiera bardzo dużo opcji oraz wylicza różnorodne (często nadmiarowe) współczynniki, ponieważ jest to niezbędne w procesie wyszukiwania zależności między danymi, tym samym optymalizując proces wyszukiwania niepłynności. W kolejnym kroku należałoby znacznie uprościć interfejs użytkownika, który na razie jest ukierunkowany na zastosowania naukowe – dzięki czemu, mógłby zacząć być stosowany przez logopedów. Należałoby również zoptymalizować program WaveBlaster, zarówno pod kątem szybkości działania jak również pod kątem potrzebnej pamięci RAM tak, aby mógł zacząć być efektywnie stosowany na mniej wymagających maszynach, dzięki czemu mógłby być powszechniej stosowany.

Bibliografia

A.N. Akansu and R.A. Haddad, *Multiresolution signal decomposition.*: Academic 1] Press, 2001.

S. Barro and R. Marin, *Fuzzy Logic in Medicine*. New York: Physica-Verlag 2] Heidenberg, 2002.

J. T. Białasiewicz, *Falki i Aproksymacje*. Warszawa: Wydawnictwo Naukowo-

3] Techniczne, 2000.

J.P. Campbell Jr., "Speaker recognition: a tutorial," *Proceedings of the IEEE*, vol. 85, 4] no. 9, pp. 1437 - 1462, 1997.

K. Chanwoo, S. Kwang-deok, and S. Wonyong, "A robust formant Extraction 5] algorithm combining spectral peak picking and root polishing," *EURASIP Journal on Applied Signal Processing*, pp. 33-33, 2006.

I. Codello and W. Kuniszyk-Jóźkowiak, "„Wave Blaster" – a comprehensive tool for 6] speech analysis and its application for vowel recognition using wavelet continuous transform with bark scales," *56 Otwarte Seminarium z Akustyki OSA*, pp. 63-68, 2009.

I. Codello and W. Kuniszyk-Jóźkowiak, "Digital signals analysis with LPC method," 7] *Annales UMCS Informatica AI 5*, pp. 315-321, 2006.

I. Codello and W. Kuniszyk-Jóźkowiak, "Formant paths tracking using Linear 8] Prediction based methods," *Annales UMCS Informatica AI X(2)*, pp. 7-12, 2010.

I. Codello and W. Kuniszyk-Jóźkowiak, "Wavelet analysis of speech signal," *Annales* 9] *UMCS Informatica AI 6*, 2007.

I. Codello, W. Kuniszyk-Jóźkowiak, T. Gryglewicz, and W. Suszyński, "Utterance 10] intonation imaging using the cepstral analysis," *Annales UMCS Informatica AI 8(1)*, pp. 157-163, 2008.

I. Codello, W. Kuniszyk-Jóźkowiak, and A. Kobus, "Kohonen networks application in 11] speech analysis algorithms," *Annales UMCS Informatica AI X(2)*, pp. 13-19, 2010.

I. Codello, W. Kuniszyk-Jóźkowiak, E. Smołka, and A. Kobus, "Automatic disordered 12] sound repetition recognition in continuous speech using CWT and Kohonen network," *Annales UMCS Informatica*, 2012.

I. Codello, W. Kuniszyk-Jóźkowiak, E. Smołka, and A. Kobus, "Automatic disordered 13] syllables repetition recognition in continuous speech using CWT and correlation," *Advances in Intelligent and Soft Computing*, 2013.

I. Codello, W. Kuniszyk-Jóźkowiak, E. Smołka, and A. Kobus, "Automatic
14] prolongation recognition in disordered speech using CWT and Kohonen network," *Journal Of Medical Informatics & Technologies*, 2012.

I. Codello, W. Kuniszyk-Jóźkowiak, E. Smołka, and A. Kobus, "Disordered sound
15] repetition recognition in continuous speech using CWT and Kohonen network," *Journal Of Medical Informatics & Technologies, Vol. 17*, pp. 123-130, 2011.

I. Codello, W. Kuniszyk-Jóźkowiak, E. Smołka, and A. Kobus, "Prolongation
16] Recognition in Disordered Speech," *Proceedings of International Conference on Fuzzy Computation*, pp. 392-398, 2010.

I. Codello, W. Kuniszyk-Jóźkowiak, E Smołka, and W. Suszyński, "Speaker
17] Recognition using Continuous Wavelet Transform with Bark Scales," *Polish J. of Environ. Stud. Vol. 18*, pp. 78-82, 2009.

A. Czyżewski, B. Kostek, and H. Skarżynski, *Technika komputerowa w audiologii,*
18] *foniatrii i logopedii*. Warszawa: Exit, 2002.

M. Dzieńkowski, "Komputerowe słuchowo-wizualne diagnozowanie i terapia
19] niepłynności mowy," *praca doktorska, Instytut Biocybernetyki i Inżynierii Biomedycznej PAN*, 2007.

M. Dzieńkowski, W. Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński, "Computer
20] programme for speech impediment diagnosis and therapy," *Annales Informatica Universitatis Mariae Curie-Skłodowska*, pp. 21-29, 2003.

M. Dzieńkowski, W. Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński, "Computer
21] speech echo-corrector," *Annales Informatica Universitatis Mariae Curie- Skłodowska*, pp. 315-322, 2004.

M. Dzieńkowski, W. Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński, "Cyfrowa
22] analiza plików dzwiekowych," *Lubelskie Akademickie Forum Informatyczne*, pp. 71- 78, 2002.

M. Dzieńkowski, W. Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński,

23] "Komputerowa zindywidualizowana terapia niepłynnej mowy," *XIII Konferencja Naukowa Biocybernetyka i Inżynieria Biomedyczna*, vol. I, pp. 546-551, 2003.

M. Dzieńkowski, W. Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński,
24] "Komputerowe słuchowo-wizualne echo dla celów terapii niepłynnosci mowy," *Obliczenia naukowe - wybrane problemy, PTI*, pp. 57-63, 2003.

S. Garfield, M. Elshaw, and S. Wermter, "Self-organizing networks for classification
25] learning from normal and aphasic speech," *In The 23rd Conference of the Cognitive Science Society*, 2001.

B. Gold and N. Morgan, *Speech and audio signal processing*. New York: John Wiley
26] & Sons Inc., 2000.

P. Goupillaud, A. Grossmann, and J. Morlet, "Cycle-octave and related transforms in
27] seismic signal analysis," *Geoexploration 23*, pp. 85-102, 1984-1985.

A. Horzyk and R. Tadeusiewicz, "Self-optimizing neural networks, Advances in
28] neural networks," *Lecture notes in computer science*, pp. 150-155, 2004.

X. Huang and A. Acero, *Spoken Language Processing: A Guide to Theory, Algorithm
29] and System Development*.: Prentice-Hall Inc., 2001.

A. Izworski and W. Wszołek, "Wykorzystanie metod sztucznej inteligencji w
30] diagnostyce i przetwarzaniu patologicznych sygnałów akustycznych," *Speech and Language Technology 3*, pp. 299-319, 1999.

A. Izworski, W. Wszołek, R. Tadeusiewicz, and T. Wszołek, "Understanding of
31] deformed speech signals using vocal tract simulation," *Advances of Medicine and Health Care through Technology - the Challenge to Biomedical Engineering in Europe*, pp. 532-533, 2002.

A. Kobus, W. Kuniszyk-Jóźkowiak, E. Smołka, and I. Codello, "Speech Nonfluency
32] Detection and Classification Based on Linear Prediction Coefficients and Neural Networks," *Journal of Medical Informatics & Technologies, Vol. 15*, pp. 135-144, 2010.

A. Kobus, W. Kuniszyk-Jóźkowiak, E. Smołka, W. Suszyński, and I. Codello, "A new
33] elliptical model of the vocal tract," *Journal Of Medical Informatics & Technologies, Vol. 17*, pp. 131-139, 2011.

T. Kohonen, "Self-Organizing Maps," pp. 2173-2179, 2001.
34]

A. Komaee and A. Sepehri, "Linear Prediction and Synthesis of Speech Signals,"
35] *Department of Electrical and Computer Engineering, University of Maryland*. [Online]. http://www.enee.umd.edu/~afshin/adsp2/proj2.pdf

W. Kuniszyk-Jóźkowiak, "A comparison of speech envelopes of stutterers and
36] nonstutterers," *J. Acoust. Soc. Am.*, pp. 1105-1110, 1996.

W. Kuniszyk-Jóźkowiak, *Przetwarzanie sygnałów biomedycznych*. Lublin:
37] Uniwersytet Marii Curie-Skłodowskiej w Lublinie, 2011.

W. Kuniszyk-Jóźkowiak, J. Jaszczuk, T. Sacewicz, and I. Codello, "Time–frequency
38] Analysis of the EMG Digital Signals," *Annales UMCS Informatica AI XII*, pp. 19-25, 2012.

R.D. Lyons, *Wprowadzenie do cyfrowego przetwarzania sygnałów*. Warszawa,
39] Wydawnictwa Komunikacji i Łączności, 2003.

R. Polikar. The wavelet tutorial. [Online].
40] http://users.rowan.edu/~polikar/WAVELETS/WTpart1.html

L.R. Rabiner and R.W. Schafer, *Digital Processing of Speech Signals*. New Jarsey:
41] Prentice-Hall, Inc., 1978.

I. Simonovski and M. Boltezar, "The norms and variances of the Gabor, Morlet and
42] general harmonic wavelet functions," *Journal of Sound and Vibration* , vol. 264, no. 3, pp. 545-557, 2003.

S. W. Smith, *Digital signal processing*. San Diego, California, 1994.
43]

J. Smith and J. Abel, *Bark and ERB Bilinear Transforms*.: IEEE Transactions on

44] Speech and Audio Processing, 1999.

E. Smołka, W. Kuniszyk-Jóźkowiak, M. Dzieńkowski, W. Suszyński, and M.
45] Swietlicki, "Rozpoznawanie samogłosek w izolacji i mowie ciagłej z wykorzystaniem perceptronu wielowarstwowego," *Structures Waves - Human Health*, pp. 143-148, 2005.

E. Smołka, W. Kuniszyk-Jóźkowiak, and W. Suszyński, "Odwzorowanie płynnych i
46] niepłynnych słów w sieci Kohonena," *XLIX Otwarte Seminarium z Akustyki*, 2002.

E. Smołka, W. Kuniszyk-Jóźkowiak, and W. Suszyński, "Reflection of fluent and
47] nonfluent words in Kohonen network," *XLIX Open Seminar on Acoustics*, pp. 371-376, 2002.

E. Smołka, W. Kuniszyk–Jóźkowiak, W. Suszyński, and M. Dzieńkowski, "Speech
48] syllabic structure extraction with application of Kohonen network," *Annales Informatica UMCS*, pp. 125-131, 2003.

E. Smołka, W. Kuniszyk-Jóźkowiak, W. Suszyński, M. Dzieńkowski, and I.
49] Szczurowska, "Speech nonfluency recognition in two stages of Kohonen networks," *Structures - Waves - Human Health*, vol. XIII, no. 2, pp. 139-142, 2004.

E. Smołka, W. Kuniszyk-Jóźkowiak, E. Suszyński, and M. Wiśniewski, "Vowel
50] recognition in continuous speech with application of MLP neural network," *Annales UMCS, Sectio AI Informatica vol. V*, pp. 139-144, 2006.

W. Suszyński, *Automatyczne rozpoznawanie niepłynności mowy*. Lublin-Gliwice:
51] Praca doktorska, 2005.

W. Suszyński and M. Dzieńkowski, "Detekcja niepłynnosci mowy przy
52] wykorzystaniu funkcji korelacji," *51 Otwarte Seminarium z Akustyki*, pp. 386-389, 2004.

W. Suszyński, W. Kuniszyk-Jóźkowiak, E. Smołka, and M. Dzieńkowski, "Automatic
53] recognition of nasals prolongations in the speech of persons who stutter," *Structures - Waves - Human Health*, pp. 175-184, 2003.

I. Swietlicka, W. Kuniszyk-Jóźkowiak, and E. Smołka, "Artificial neural networks in

54] the disabled speech analysis," *Computer Recognition Systems (Advances in Soft Computing), Verlag Berlin Heidelberg, Springer*, pp. 347-354, 2009.

I. Swietlicka, W. Kuniszyk-Jóźkowiak, and E. Smołka, "Detection of syllable
55] repetition using two-stage artificial neural networks," *Polish Journal of Environmental Studies*, vol. 17, pp. 462- 466, 2008.

I. Szczurowska, W. Kuniszyk-Jóźkowiak, and E. Smołka, "Application of Artificial
56] Neural Networks In Speech Nonfluency Recognition," *Polish Jurnal of Environmental Studies, 2007 16(4A)*, pp. 335-338, 2007.

I. Szczurowska, W. Kuniszyk-Jóźkowiak, and E. Smołka, "Speech nonfluency
57] detection using Kohonen networks," *Neural Computing and Application*, vol. 18, no. 7, pp. 677-687, 2009.

I. Szczurowska, W. Kuniszyk-Jóźkowiak, and E. Smołka, "The application of
58] Kohonen and Multilayer Perceptron network in the speech nonfluency analysis," *Archives of Acoustics 2006. 31 (4 (Supplement))*, pp. 205-210, 2006.

I. Szczurowska, E. Smołka, W. Kuniszyk-Jóźkowiak, W. Suszyński, and M.
59] Dzieńkowski, "The application of neural networks in the speech nonfluency analysis," *Structures Waves - Human Health*, vol. XIV, no. 1, pp. 173-176, 2005.

R. Tadeusiewicz, *Elementarne wprowadzenie do sieci neuronowych z przykładowymi*
60] *programami*. Warszawa: Akademicka Oficyna Wydawnicza, 1998.

R. Tadeusiewicz, *Sieci neuronowe*. Warszawa: EXIT, 1993.
61]

R. Tadeusiewicz, *Sygnał mowy*. Warszawa: Wydawnictwa Komunikacji i Łączności,
62] 1988.

R. Tadeusiewicz, *Wstęp do sieci neuronowych*. Warszawa: Akademicka Oficyna
63] Wydawnicza Exit, 2000.

R. Tadeusiewicz, "Zastosowanie sieci neuronowych do rozpoznawania mowy,"

64] *Analiza, synteza i rozpoznawanie sygnału mowy dla celów automatyki, informatyki, lingwistyki i medycyny*, pp. 137-150, 1994.

R. Tadeusiewicz, W. Wszołek, and A. Izworski, "Sieci neuronowe jako narzedzie do
65] symulacji przetwarzania informacji akustycznej systemu słuchowego," *X Krajowa Konferencja Naukowa Biocybernetyka i Inżynieria Biomedyczna*, pp. 801-807, 1997.

The MathWorks, “Matlab 7 Help – Wavelets: A New Tool for Signal Analysis”.
66]

H. Traunmüller, "Analytical expressions for the tonotopic sensory scale," *J. Acoust.*
67] *Soc. Am. 88*, pp. 97-100, 1990.

H. Wakita, "Direct Estimation of the Vocal Tract Shape by Inverse Filtering of
68] Acoustic Speech Waveforms," *IEE Transactions on Audio and Electroacoustics*, vol. AU-21, no. 5, 1973.

M. Wiśniewski, W. Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński, "Automatic
69] detection of disorders in a continuous speech with the Hidden Markov Models approach," *Advances in Soft Computing 45, Computer Recognition Systems 2, Springer-Verlag, Berlin Heildelberg*, 2007.

M. Wiśniewski, W. Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński, "Automatic
70] detection of prolonged fricative phonemes with the Hidden Markov Models approach," *Journal of Medical Informatics and Technogies vol. 11/2007, Computer System Dept. University of Silesia*, 2007.

M. Wiśniewski, W. Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński, "Vision
71] echo," *nnales UMCS, Sectio AI Informatica*, vol. III, pp. 139-144, 2005.

M. Wiśniewski, W, Kuniszyk-Jóźkowiak, E. Smołka, and W. Suszyński, "Automatic
72] detection of speech disorders with the use of Hidden Markov Models," *Annales UMCS, Sectio AI Informatica vol. VI-VII*, 2007.