

УДК 534.4: 621.391

О.І. Лучинкіна, О.Н. Карпов, А.Г. Матвєєнко

*Дніпропетровський національний університет ім. Олеся Гончара*

## **РОЗРОБКА АЛГОРИТМІВ І ПРОГРАМ РОЗУМІННЯ ТА СИНТЕЗУ ШЕПІТНОЇ МОВИ**

Розглядається шепітна мова, як одна з різновидів мови, для якої з'являється проблема складного і неоднозначного сприйняття. Пропонується алгоритм її обробки з метою підвищення розбірливості і подальшого синтезу.

**Ключові слова:** *апроксимація, дзвіноподібні функції, лінкер, локон Аньєзі, неоднозначність, розпізнавання мови, основний тон, спектр, тональні характеристики сигналу, функція Гауса, частота, шепітна мова, matlab*

Рассматривается шепотная речь, как одна из разновидностей речи, для которой появляется проблема сложного и неоднозначного восприятия. Предлагается алгоритм ее обработки с целью повышения разборчивости и дальнейшего синтеза.

**Ключевые слова:** *апроксимация, колоколообразные функции, линкер, локон Аньези, неоднозначность, распознавание речи, основной тон, спектр, тональные характеристики сигнала, функция Гауса, частота, шепотная речь, matlab*

The whispered speech is considered as a form of speech, for which a problem of complex and ambiguous perception take place. An algorithm for processing it in order to improve legibility and further synthesis is proposed.

**Key words:** *approximation, bell-shaped functions, linker, witch of Agnesi Agnese, ambiguity, speech recognition, the fundamental tone, spectrum, tonal characteristics of the signal, the the function Gauss, frequency, whisper, matlab*

**Огляд проблеми.** Шепітна мова – один з дефектів функціонування мовотворчої системи.

Причини розладів голосу [1; 2] вельми різноманітні: захворювання і травматичні ушкодження гортані та голосових зв'язок; порушення резонаторних систем; хвороби органів дихання; захворювання серця і серцево-судинної системи; ендокринні розлади, зокрема, захворювання щитовидної залози; порушення слуху, що утрудняють загальне «настроювання» голосотворного апарату зважаючи на відсутність або недостатність слухового контролю; тривале куріння; систематичне вживання алкоголю; вплив отрутохімікатів; часте

перебування у запорошених приміщеннях; систематичне перенапруження голосу, особливо при неправильному користуванні ним; різкі температурні коливання, зокрема, пиття холодної води і особливо холодного молока і соків; психічні травми.

Зазначені етіологічні фактори призводять до органічних і функціональних порушень голосу.

Дефект може бути викликаний різними захворюваннями і виражається в тому, що голосової м'язи не напружується, внаслідок чого в мові не з'являється тональна компонента.

З точки зору утворення мовного сигналу на вхід мовотворчого тракту надходять сигнали від двох генераторів:

- а) тонального – коливання голосової м'язи;
- б) шумового – турбулентний потік повітря.

Такий дефект може з'являтися через різні хвороби та виражається у тому, що голосові зв'язки не напружуються, в результаті чого з мови зникає тональна компонента.

При відсутності коливань голосових м'язів мова формується як звичай, але не звучить. Огинають спектрів тональних звуків такі ж, як і при наявності коливань голосових м'язів. Істотна відмінність – у внутрішній структурі спектрів. Спектри тональних звуків при наявності голосового джерела є лінійчатыми з частотами ліній кратними частоті основного тону (ОТ), де  $\omega_k = k\omega_0$ , де  $\omega_0$  - частота ОТ. У спектрах шепітної мови немає ліній на частотах. Завдання озвучування тональних звуків є завданням розпізнавання слів, визначення меж тональних звуків і їх озвучення відповідно до їх спектрів.

Задача розпізнавання за методом та алгоритмом добре відома [3], хоча надійність розпізнавання – це величина що залежить від багатьох причин. Для ізольованих слів можна отримати гарну надійність розпізнавання – понад 90 % при налаштуванні системи на конкретного диктора. Для невеликого словника – це близько до 99 %. Реальна задача озвучування шепітної мови пов'язана з безперервною мовою, а надійність розпізнавання безперервної мови в загальному випадку низька – нижче 80 %.

Завдання озвучування – це завдання синтезу за відомою спектрально-часовою функцією тональної частини шепітної мови. Істотно важливим є завдання сегментації, тому що треба максимально точно визначати межі озвучування. Оскільки чергування гучних і тональних звуків носить випадковий характер, то при вирішенні задачі синтезу необхідно вести підрахунок ділянок мови для того, щоб

визначати які з них відповідно до транскрипції необхідно озвучувати. При вирішенні задачі розпізнавання на фонемному рівні завдання визначення цих ділянок вирішується автоматично.

Тому постає проблема боротьби з таким дефектом. Одна з можливостей вирішення такої проблеми полягає у спробі озвучення сегментів тональних звуків шепітної мови. Отримання звичайної вимови за шепітним сигналом покращить можливості спілкування з людьми, з яких пошкоджені голосові зв'язки.

Сутність проблеми, полягає в тому, щоб розробити таку систему, яка б по вхідному шепітному сигналу могла б автоматично розпізнати та синтезувати сигнал з тональною компонентою. Також проблема ускладнюється тим, що більшість існуючих методів розпізнавання і сегментації розроблені для сигналів зі звичайною вимовою. Тому система повинна використовувати алгоритми, які не спираються на тональні характеристики сигналу. Завдяки використанню таких алгоритмів якість розпізнавання та сегментації шепітної мови має бути не гіршою, ніж для звичайної мови.

Проблема розпізнавання шепітної мови – це не лише проблема налаштування системи. Це проблема, яка, насамперед, пов'язана з розумінням мови. Навіть людині іноді важко розпізнати шепітну мову.

У статті будуть розглянуті методи та алгоритми, які допоможуть вирішити не лише задачу розпізнавання. Вони направлені на те, щоб, насамперед, вирішити задачу розуміння мови та перетворення її таким чином, щоб вона була зрозуміла як людині, так і машині.

**Постановка задачі.** Вхідними даними для подальшого аналізу є звуковий сигнал шепітної мови,  $\{S_i, i = \overline{1, k}\}$ .

Уведення сигналу відбувається через завантаження з wave-файлу.

При розпізнаванні шепітної мови особлива увага приділяється якості сигналу, тому що додавання шумів або зниження якості сигналу відповідно знизить надійність розпізнавання. Таким чином повинна забезпечуватися належна якість аудіосигналу.

Необхідно розробити алгоритм, за допомогою якого можливо:

- 1) отримати мовленевий сигнал шляхом завантаження звукового файлу;
- 2) обробити мовленевий сигнал таким чином, щоб шепітна мова стала більш розбірливою та придатною для розпізнавання;
- 3) озвучити отриманий за допомогою алгоритму сигнал.

**Методи та алгоритми розв'язання задачі розпізнавання шепітної мови.** Розглянемо алгоритм розпізнавання шепітної мови,

який засновано на апроксимації мовленевого сигналу за допомогою дзвіноподібних функцій [4].

**1. Первинна обробка сигналу.** Над сигналом проводиться базова фільтрація, яка здійснюється за принципом відокремлення шумів на початку та в кінці звукового потоку залежно від маскувального рівня сигналу.

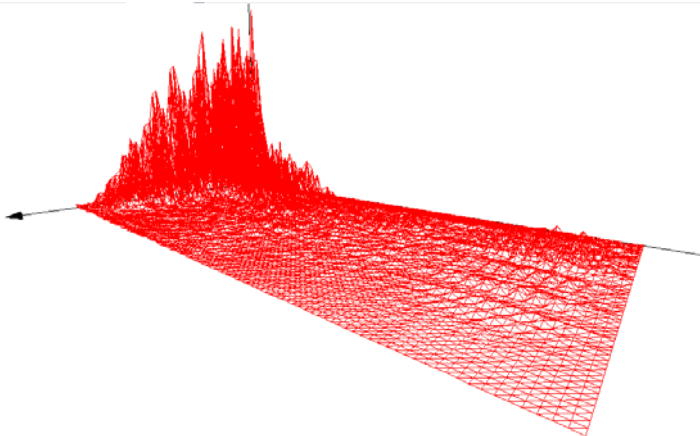
Нормалізація даних в задачах моделювання процесів аналізу мовних сигналів необхідна через те, що перешкоджає різним носіям мови впливати на кінцевий результат аналізу, зводить параметри мовного сигналу до канонічного вигляду.

У процесі подальшої обробки ми отримаємо спектрально-часове представлення вхідного звукового сигналу.

Спектрально-часове представлення може бути отримане у різних ортогональних базисах функцій:

- Фур'є;
- Мат'є;
- Лежандра;
- Чебишева;
- Уолша;
- Лаггера;
- Ерміта та інших.

На рис.1 зображене спектрально-часове представлення слова «ноль».



**Рис. 1.** Спектрально-часове представлення слова «ноль»

**2. Апроксимація спектру класу дзвіноподібних функцій.** Для подальшого аналізу сигналу спектрально-часове представлення, що отримано на попередньому кроці, необхідно очистити від шумів.

Можливі варіанти апроксимуючих дзвіноподібних функцій:

1. У класі комплексних функцій:

$$W_i(w_k)^2 = \frac{c_i^2(a_i^2 + w_k^2)}{(b_i^2 + a_i^2 - w_k^2)^2 + (2w_k a_i)^2}, h_i(t_l) = d_i e^{-(t_l - T_i)^2 f_i}, \quad (1)$$

де  $t_l = \ln \lambda_l$ ,  $\lambda_l$  – час чи номер інтервалу.

2. Функції другого порядку у вигляді полінома:

$$W_i(w_k) = a_{i2} w_k^2 + a_{i1} w_k + a_{i0}, h_i(t_l) = b_{i2} t_l^2 + b_{i1} t_l + b_{i0}. \quad (2)$$

На основі парабол можна побудувати дзвіноподібну функцію з властивостями асимптотичності як сплайн з двома вузлами.

3. Модифікований локон Аньєзі:

$$W_i(w_k) = \frac{a_i^3}{c_i^2 + (w_k - \beta_i)^2}, h_i(t_l) = \frac{b_i^2}{d_i^2 + (t_l - T_i)^2}. \quad (3)$$

4. Функція Гауса:

$$W_i(w_k) = a_i e^{-(w_k - \beta_i)^2 v_i}, h_i(t_l) = b_i e^{-(t_l - T_i)^2 \tau_i}. \quad (4)$$

5. Експоненціальна функція:

$$W_i(w_k) = a_i e^{b_i w_k + v_i w_k^2}, h_i(t_l) = r_i e^{d_i t_l + f_i t_l^2}. \quad (5)$$

**3. Вибір і формування типів мовних одиниць.** В якості мовних одиниць можуть виступати:

- а) сегменти;
- б) склади;
- в) фонемі;
- г) слова.

**4. Сегментація.** Сегментація може проводитися одним з наступних методів:

- а) фільтрація огинають спектра;
- б) застосування порогових методів за груповими ознаками;
- в) диференціювання функцій параметрів;
- г) верифікація за сукупністю параметрів.

**5. Навчання системи.** Алгоритм розпізнавання вхідного сигналу, оснований на пошуку за базовим словником. Тому рішення задачі розпізнавання мови складається з двох частин: навчання системи (накопичення еталонів) і власне розпізнавання.

Етап навчання полягає в тому, що системі послідовно подаються сигнали навчальної вибірки, які будуть проаналізовані та додані до

словника. Наприклад, це можуть бути цифри від одного до дев'яти. Для кожного еталона виконуються всі етапи обробки сигналу:

- первинна обробка;
- частотно-часове перетворення;
- згладжування (апроксимація колоколоподібними функціями).

**6. Розпізнавання мовленевого сигналу.** Етап розпізнавання полягає у зіставленні поточного сигналу для розпізнавання з еталонами із словника. Поточний сигнал також повинен бути оброблений, щоб його можна було порівнювати з іншими еталонами. Для обробки, сигнал проходить всі описані вище етапи.

Цифрове представлення сигналів являє собою вектори частотних коефіцієнтів для кожного часового інтервалу (розбиття на часові інтервали описане вище). Для порівняння двох сигналів, попарно порівнюються вектори коефіцієнтів часових інтервалів цих сигналів. Для порівняння частотних коефіцієнтів використовується евклідова відстань між векторами

$$dist = \|V_1 - V_2\|, \quad (6)$$

де  $V_1, V_2$  – вектори коефіцієнтів,  $\|A\|$  – евклідова норма вектору.

Таким чином визначається відстань між двома сигналами  $d_i$  – ця відстань знаходиться за допомогою методу динамічного програмування. Метод динамічного програмування дозволяє вирішити завдання ослаблення часової нестационарності шляхом нелінійного розтягування короткої реалізації сигналу щодо довгої. Розпізнаним еталоном є еталон з найменшим значенням відстані

$$index = \arg \min_i (d_i). \quad (7)$$

Потім за номером еталона  $index$  знаходиться відповідна транскрипція для шепітного сигналу.

**Апроксимація мовленевого сигналу.** Сформулюємо задачу апроксимації мовленевого сигналу [5,6]. Нехай задана прямокутна область  $R = [w_a, w_b] \times [T_c, T_d]$ , а в області  $R$  задана дискретна спектрально-часова функція  $S(w_k, t_l)$ , де  $w_k$  – дискретно задана частота, а  $t_l$  – дискретно заданий час. Область задана граничними значеннями  $w_a, w_b$  – частоти,  $T_c, T_d$  – часу.

Також нехай визначені класи функцій  $\{W_i(w_k)\}$  та  $\{h_i(t_l)\}$ , які мають наступні властивості:

- функція повинна бути дзвонообразною (мати максимум на заданій частоті) для представлення резонансу;
- форма функції повинна залежати від деяких параметрів;

- функція повинна асимптотично наближатись до площини області R в будь-якому напрямку від максимуму;
- гілки функції повинні описувати інерційні та диференційні властивості при моделюванні частин системи чи монотонно спадати (зростати), якщо максимум знаходиться поза границями області.

Функція спектру сигналу  $S(w_k, t_l)$  має будь-яку кількість сплесків спектральної енергії, розташованих у даній області. Необхідно описати функцію  $S(w_k, t_l)$  в класі функцій  $\{W_i(w_k)\}$  та  $\{h_i(t_l)\}$ , і визначити параметри сплесків функції спектру як параметри функцій  $\{W_i(w_k)\}$  та  $\{h_i(t_l)\}$ .

Згідно з теоремою Колмогорова, будь-яка неперервна функція п змінних може бути отримана за допомогою композиції неперервних функцій однієї змінної та єдиної функції двох змінних  $g(x, y) = x + y$ . Він довів, що будь-яка функція f, неперервна на n-вимірному кубі, може бути представлена у вигляді

$$f(x_1, \dots, x_n) = \sum_{i=1}^{2n+1} h_i \left( \sum_{j=1}^n \varphi_{ij}(x_j) \right), \quad (8)$$

де функції  $h_i$ ,  $\varphi_{ij}$  – неперервні, а функції  $\varphi_{ij}$  також стандартні, тобто не залежать від вибору функції f.

Визначимо модель опису функції  $S(w_k, t_l)$  у вигляді

$$Z(w_k, t_l) = \sum_{i=1}^n |S_{gi}(w_k)| |W_i(w_k)| h_i(t_l) \quad , \quad (9)$$

де  $S_{gi}(w_k)$  – спектральна функція генератора сигналу, n – кількість інформаційних сплесків.

Для визначення параметрів усіх функцій застосуємо схему послідовних вилучень інформаційних складових мовного сигналу (СЕТ). Параметри всіх функцій визначають послідовно для кожного доданка суми у функції  $Z(w_k, t_l)$ . Спектрально-часова функція може бути представлена у вигляді

$$S(w_k, t_l) = Z_{(1)}(\langle param_1 \rangle, w_k, t_l) + S_{(1)}(w_k, t_l). \quad (10)$$

Тоді після визначення параметрів  $params_1$  для функції  $Z_{(1)}$ , можна визначити залишок спектрально-часової функції у області

$$S(w_k, t_l) = Z_{(1)}(\langle param_1 \rangle, w_k, t_l) - S_{(1)}(w_k, t_l). \quad (11)$$

Дана процедура виділення залишку і знаходження параметрів повторюється далі для всіх доданків:

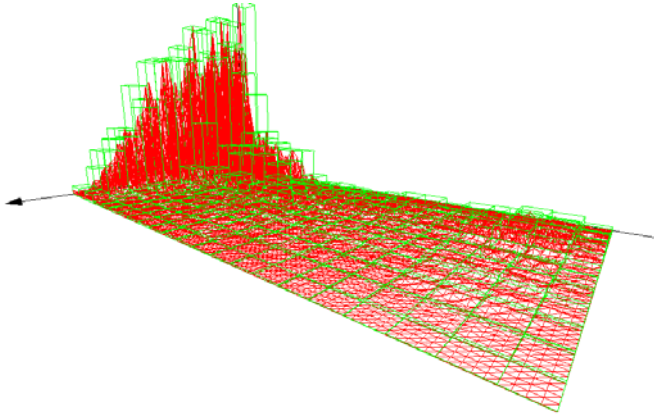
$$\begin{aligned} S(w_k, t_l) &= Z_{(i)}(\langle param_1 \rangle, w_k, t_l) + S_{(i)}(w_k, t_l) \\ S(w_k, t_l) &= Z_{(i)}(\langle param_1 \rangle, w_k, t_l) - S_{(i)}(w_k, t_l) \end{aligned} \quad (12)$$

Функція  $S$  буде апроксимована повністю, коли будуть знайдені всі параметри кожної з  $n$  функцій доданків у  $Z$ .

Розглянемо алгоритм апроксимації.

1. Нехай маємо дискретну спектрально-часову функцію  $S(w_k, t_l)$ .

Для її апроксимації за схемою послідовних вилучень цю функцію розбивають на прямокутні регіони за часом та частоті. Розбиття за часом генерує інтервали однакової довжини на всьому часі сигналу – наприклад по 40 мс чи 4 шаги. Розбиття по частоті може бути рівномірним та нерівномірним. Низькі частоти несуть більше інформації, тому там потрібне детальніше розбиття. Вигляд розбиття зображено на рис. 2, регіони відділяються обмежувачами паралелепіпедами.



**Рис. 2. Рівномірне розбиття на регіони для апроксимації для слова «ноль»**

2. За допомогою методу покоординатного спуску проводимо апроксимацію частини спектру в кожному регіоні.

Суть методу полягає у послідовній мінімізації функції для кожного параметру окремо до того моменту поки не буде виконуватись умова зупинки алгоритму.

$$Z(w_k, t_l) = Z(p_1, \dots, p_m, w_k, t_l), \quad (13)$$

У даному випадку для функції вибирається наступний параметр для мінімізації  $p_j$ , а інші параметри фіксуються. Далі відбувається мінімізація одновимірної функції від обраного параметра



$$Z(w_k, t_l) = Z(p_j, w_k, t_l), \quad (13)$$

Так як градієнт даної функції невідомий, застосовується адаптивний алгоритм для знаходження значення параметра, який мінімізує функцію.

- 1) обирається початковий шаг step параметра;
- 2) функція тестується, щоб зробити крок у напрямках  $-\text{step}$  та  $+\text{step}$ ;
- 3) якщо функція зменшується у деякому напрямку, то точка мінімуму зміщується і параметр змінюється відповідно;
- 4) якщо в обох напрямках функція більша за поточне значення, шаг параметра зменшується вдвічі;
- 5) умова зупинки алгоритму – шаг став менший деякого критичного значення.

Процес мінімізації багатовимірної функції повторюється для кожного параметра послідовно. Після останнього параметра знову обирається перший, доки не виконається умова зупинки алгоритму по координатного спуску. Ця умова – зміна значення функції на поточному та попередньому кроці менше деякого значення.

**Інтеграція MATLAB в .NET.** Для промови було використано програмне забезпечення синтезу реалізоване у середовищі MATLAB, тому постало питання інтеграції MATLAB у .NET [6].

Дуже часто перед програмістом постає завдання обчислення складної математики. MATLAB у свою чергу є відмінним засобом для його вирішення, але слабким у створенні повноцінного користувацького інтерфейсу.

Інструменти:

- Microsoft Visual Studio 2010 SP1
- MATLAB 2010a
- MATLAB Component Runtime

**Крок 1.** Налаштування лінкера.

Щоб зібрати dll-бібліотеку MATLAB'а для інтеграції у C#.NET, потрібно налаштувати лінкер, тобто яким середовищем ми будемо збирати проект. Для початку потрібно встановити середовище виконання MCR (MATLAB Component Runtime). Це набір dll-бібліотек для повної підтримки мови MATLAB. Установчий файл можна знайти: ... \ MATLAB \ R2011b \ toolbox \ compiler \ deploy \ win32 \ MCRInstaller. Для налаштування лінкера в командному вікні MATLAB'а набираємо mbuild -setup. З усім погоджуємося і вибираємо потрібну нам середу, в нашому випадку це MVS 2010 SP1. Результат представлено на рис.3.

**Крок 2.** Пишемо m-функцію.

Компілятор MATLAB'a розуміє тільки функції, тобто, кожен сценарій повинен починатися з function (бажано закінчуватися end) і бути окремим m-файлом.

```
Please choose your compiler for building standalone MATLAB applications:
```

```
Would you like mbuild to locate installed compilers [y]/n? y
```

```
Select a compiler:
```

```
[1] Lcc-win32 C 2.4.1 in C:\PROGRA-1\MATLAB\R2010a\sys\lcc
```

```
[2] Microsoft Visual C++ 2008 SP1 in C:\Program Files\Microsoft Visual Studio 9.0
```

```
[0] None
```

```
Compiler: 2
```

```
Please verify your choices:
```

```
Compiler: Microsoft Visual C++ 2008 SP1
```

```
Location: C:\Program Files\Microsoft Visual Studio 9.0
```

```
Are these correct [y]/n? y
```

### Рис. 3. Налаштування лінкера

**Крок 3.** Отримуємо динамічну бібліотеку.

Набираємо в командному вікні MATLAB'a `deploytool`. Створюємо новий .NET Assembly проект MATLABplane, вказуємо розміщення.

Далі створюємо клас, додаємо в нього \*.m файл і натискаємо кнопку build.

Після успішної компіляції створюється бібліотека \*.dll, її шлях: ... \ distrib \ \*.dll.

**Крок 4.** Створюємо додаток C#.NET.

У MVS 2010 SP1 створюємо додаток Windows Forms на C#.

**Крок 5.** Додаємо посилання на бібліотеки.

Перед використанням методів проекту необхідно додати посилання на скомпільовану бібліотеку \*.dll і на бібліотеку MWArray.dll, знайти її можна за адресою ... \ MATLAB \ R2010a \ toolbox \ dotnetbuilder \ bin \ win32 \ v2.0.

Для використання бібліотек у проекті необхідно додати опис простору імен:

```
using MathWorks.MATLAB.NET.Utility;  
using MathWorks.MATLAB.NET.Arrays;
```

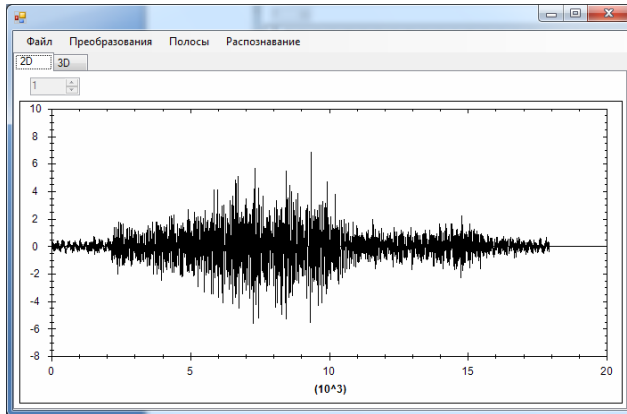
using MATLABplane; // Простір імен з отриманої бібліотеки

Для отримання будь-яких значень з MWAгау потрібно використовувати приведення типів. Тепер можна вільно використовувати написані на MATLAB'ї функції.

**Результати.** На основі запропонованого алгоритму було створено програмний продукт розпізнавання шепітної мови.

Розглянемо результат роботи програми.

На рис. 4 представлена шепітна промова слова «ноль».



**Рис. 4.** Вигляд сигналу для шепітного слова «ноль»

Спектрально-часове (рис. 5, 6) та спектрально-смугове (рис.7) представлення звукового сигналу свідчить про його сильну зачумленність.

Апроксимація сигналу за допомогою дзвіноподібних функцій поліпшує картину. Як ми бачимо з рис. 8 за допомогою апроксимації спектрально-часове представлення (рис. 5, 6) було фідфільтровано та позбавлено зайвих високочастотних відхилень.

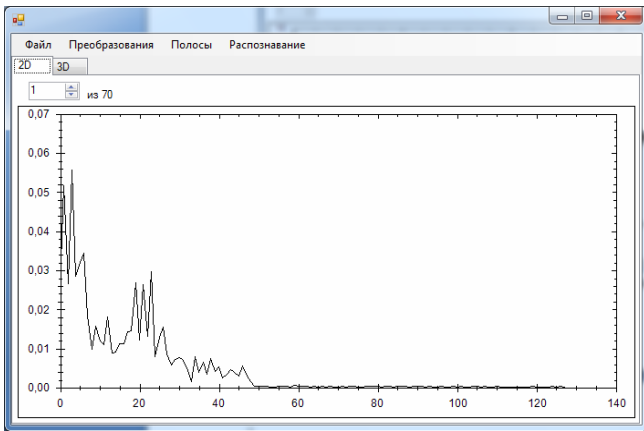


Рис. 5. Дискретне перетворення Фур'є для шепітного слова «ноль»

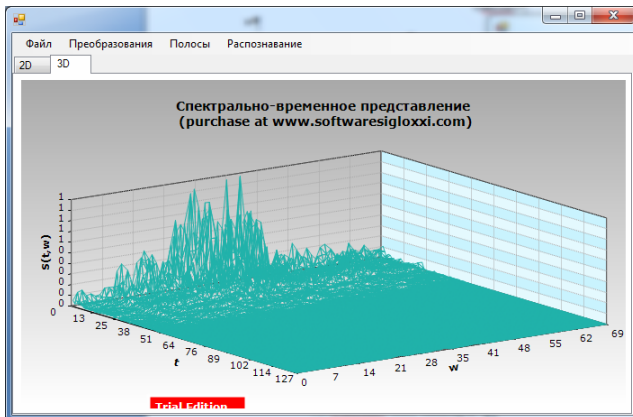
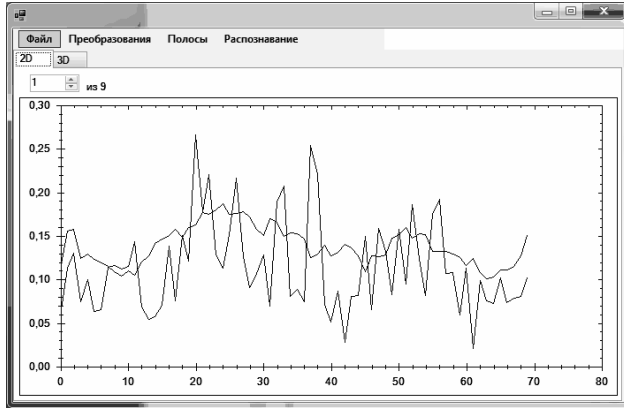
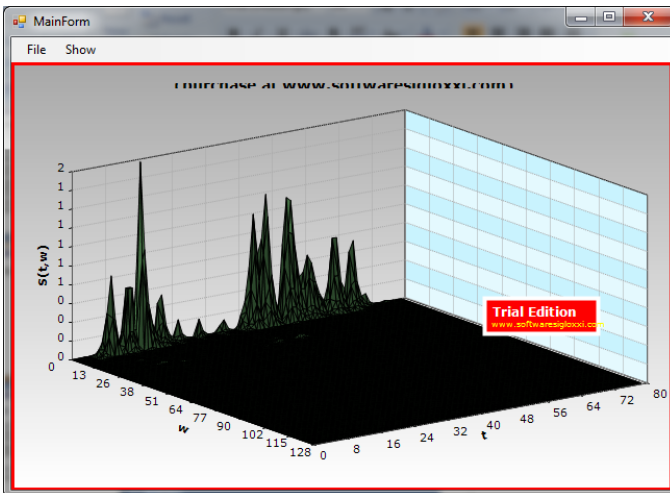


Рис. 6. Спектрально-смугове представлення шепітного слова «ноль»



**Рис. 7. Спектрально-смуговое представление та фільтрація слова «ноль»**

Порівняння якості розпізнавання стандартного підходу з використанням спектрально-смугового представлення та запропонований підхід з використанням розпізнавання апроксимованого за допомогою дзвоноподібних функцій спектрально-часового представлення показало, що запропонований підхід дає кращий результат для розпізнавання шепітної мови.



**Рис. 8. Апроксимоване спектрально-часове представлення шепітного слова «ноль»**

Недоліком даного підходу є складність програмної реалізації та потреба у більших системних ресурсах.

Отримані за допомогою побудованої системи розпізнавання мови результати було передано до системи MathLab для подальшого синтезу.

**Висновки.** Задача розпізнавання шепітної мови – це задача, по-перше, її розуміння та адаптації для сприйняття, а вже потім самого процесу розпізнавання.

Мовний сигнал формується під дією двох генераторів: тонального та шумового. При шепітній вимові звук утворюється тільки шумовим генератором, а голосові зв'язки не коливаються. Через зашумленість та специфіку вхідного сигналу задача розпізнавання шепітної мови не є тривіальною і не може бути вирішена за допомогою простих систем розпізнавання мови. У ході дослідження шепітної мови по запропонованому вище алгоритму було розроблено програмний продукт, за допомогою якого було поліпшено розбірливість мови, що сприяло підвищенню якості розпізнавання. Крім того, оброблений мовленевий сигнал синтезовано за допомогою системи Mathlb.

Отриманий сигнал є більш сприятним для розпізнавання та зрозумілим.

### Бібліографічні посилання

1. **Алмазова Е.С.** Логопедическая работа по восстановлению голоса у детей / Е.С. Алмазова – М., 2005. – 192 с.
2. **Парамонова Л.Г.** Логопедия для всех / Л.Г. Парамонова. – СПб, 1997. – 464 с.
3. **Карпов О.Н.** Компьютерные технологии распознавания речевых сигналов: моногр./ О.Н. Карпов, А.Г. Габович, Б.Г. Марченко. – К., 2005. – 138 с.
4. **Карпов О.Н.** Технология построения устройств распознавания речи: Моногр. / О.Н. Карпов – Д., 2001. – 182 с.
5. **Карпов О.Н.** Вычислительные схемы представления функций многих переменных в классе функций меньшего числа переменных. Методы анализа речевых сигналов: Моногр. / О.Н. Карпов. – Д., 2003. – 180 с.
6. **Карманов В.Г.** Математическое программирование / В.Г. Карманов. – М., 1980 – 256 с.
7. <http://habrahabr.ru/post/132487/>

*Надійшла до редколегії 15.06.2012*