

Содержание

Введение	2
1 Теория распознавания речи	3
1.1 Способы распознавания речи	3
1.2 Ограничения систем распознавания	5
1.3 Способы описания речевых сигналов	7
1.4 Психофизические единицы измерения	10
1.5 Кепстральный анализ	12
2 Реализованные алгоритмы	13
2.1 Алгоритм нахождения мел-частотных кепстральных ко- эффициентов	13
2.2 Алгоритм сравнения речевых сигналов с применением динамического программирования (DTW)	15
2.3 Листинг программы на языке MATLAB	22
Заключение	30

Введение

В настоящее время, с развитием компьютерных технологий, использование систем автоматического распознавания речи в качестве интерфейса приобретает все большую популярность. Однако, создание таких систем является нетривиальной задачей. Это связано прежде всего с тем, что каждая реализация произнесенного слова может отличаться от любой другой по целому ряду признаков. Например, по частоте или громкости. Также анализируемый речевой сигнал может быть растянут или сжат по времени относительно образца, с которым производится сравнение. Именно для минимизации временных различий служит рассмотренный в данной работе алгоритм.

1 Теория распознавания речи

1.1 Способы распознавания речи

Выделяют несколько основных способов распознавания речи:

Распознавание отдельных команд.

- Суть технологии: раздельное произнесение и последующее распознавание слова или словосочетания из небольшого заранее заданного словаря.
- Техническая реализация: точность распознавания ограничена объемом заданного словаря. При соблюдении этого условия данная технология позволяет достичь самой высокой достоверности распознавания.
- Применение: в настоящее время наиболее ярким примером использования технологии распознавания отдельных команд в коммерческих приложениях является голосовая навигация по сайтам.

Распознавание по грамматике

- Суть технологии: распознавание фраз, соответствующих определенным заданным правилам (грамматике).

- Техническая реализация: для задания грамматик используются стандартные XML-языки (VoiceXML), обмен данными между системой распознавания и приложением, как правило, осуществляется по протоколу MRCP.
- Применение: технология распознавания по грамматике широко применяется в системах голосового самообслуживания (СГС).

Поиск ключевых слов в потоке слитной речи.

- Суть технологии: распознавание отдельных участков речи.
- Техническая реализация: в этом случае речь может быть как спонтанной, так и соответствующей определённым правилам. Произнесенная речь не полностью преобразуется в текст - в ней автоматически находятся лишь те участки, которые содержат заданные слова или словосочетания.

Распознавание слитной речи на большом словаре (LVCSR — large vocabulary continuous speech recognition).

- Суть технологии: эта технология наиболее близка к мечте человека о взаимодействии человека и машины — все, что сказано, дословно преобразуется в текст. Поэтому иногда эта технология так и называется STT — speech to text.

- Техническая реализация: задача полноценного распознавания слитной речи не решена нигде в мире, однако, достоверность распознавания уже достаточно высока для использования технологии на практике.
- Применение: потенциальная сфера применения технологии в коммерческих целях довольно широка.

В данной работе реализован способ распознавания отдельных речевых команд.

1.2 Ограничения систем распознавания

Задача распознавания речи состоит в восстановлении по звуковому сигналу слова естественного языка (из ограниченного словаря), произнесением которого является звуковой сигнал. Она обычно решается путем задания эталонов слов словаря и последующего сравнения звуковых сигналов с этими эталонами. Звуковой сигнал представляет из себя целочисленный вектор значений звукового давления, измеренного в равноотстоящие друг от друга моменты времени. Мощность пространства звуковых сигналов огромна. Для решения задачи распознавания обычно сначала равномерно разбивают сигнал на окна одинаковой длины. Окна преобразуют из временной области

в частотную (например, с помощью преобразования Фурье), чтобы близость окон относительно простых метрик (типа Евклидовой) соответствовала близости участков сигнала "на слух". Затем решается задача нахождения соответствия между окнами звукового сигнала и окнами эталонов словаря. Сложность последней задачи заключается в том, что различные участки звукового сигнала в различных произнесениях одного и того же слова отличаются разной степенью сжатия или растяжения (вовсе не пропорционального). Для решения задачи нахождения соответствия между окнами сигналов традиционно используются методы динамического программирования. Создание компьютерных систем распознавания речи связано со множеством объективных трудностей, накладывающих на подобные системы искусственного интеллекта ряд ограничений. Предельные возможности компьютера по распознаванию речи связаны прежде всего с тем, что человек, которого можно взять за эталон распознающей системы, распознает осмысленную речь, а компьютеру в полной мере это не дано. Компьютер принципиально не может с требуемой надежностью исправлять ошибки и неоднозначности распознавания, используя синтаксическую и семантическую связь слов предложения. Вместо этого в современных системах используется статистическая модель, задающая связь последовательных троек слов предложения.

Кроме того человек использует зачастую дополнительную, незвуковую информацию. Самым ярким примером здесь может служить так называемое «чтение по губам», которому могут обучиться глухие люди. Известно, что в шумной обстановке человеку легче распознать речь, если он следит за губами говорящего. Человек воспринимает речь объемно, что позволяет ему производить шумоочистку и пространственное выделение сигнала более качественно, чем ЭВМ. Дополнительно картина осложняется тем, что все известные алгоритмы распознавания речи являются дикторозависимыми. После настройки на голос одного диктораа распознающие системы дают удовлетворительные результаты распознавания для этого типа голоса, но хуже работают на других голосах. Надежность распознавания речи человеком, напротив, не зависит от типа голоса диктора.[4]

1.3 Способы описания речевых сигналов

При распознавании речевых сигналов, как правило, оперируют не с исходным речевым сигналом, получаемым на выходе микрофона, а с так называемым описанием речевого сигнала, экономно представляющим речевой сигнал и содержащим информацию о том, что говорится. Обычно принято описывать (задавать) речевой сигнал последовательностью $X_I = \{x_1, x_2, \dots, x_i, \dots, x_I\}$ из элементов x_i , ко-

торые являются отсчетами векторной функции $x(t)$ в дискретные равноотстоящие моменты времени $t_i = i\Delta T$ с шагом ΔT , принимаемым равным, например, 15 мс. Тогда I это длина речевого сигнала в дискретном равномерном времени с шагом ΔT . Вообще говоря, может быть использовано и неравномерное время с изменяющимся шагом ΔT , выбираемым, например, из диапазона 8-25 мс. В любом случае, однако, речевой сигнал будет представляться последовательностью X_I , из элементов x_i . Последовательности X_I получают в результате предварительной обработки речевого сигнала на выходе микрофона, чем существенно сокращается объем информации. Так, исходный речевой сигнал, который характеризуется объемом 200000 бит/с, как правило, описывается существенно меньшим объемом информации — от 9600 до 600 и менее бит/с, однако все еще сохраняющим существенную информацию о том, что говорится, чтобы по ней отвечать на вопросы о распознаваемом классе. Вопросам предварительной обработки речевого сигнала посвящено огромное количество работ. Несмотря на многочисленность предложений все они сводятся к тому, что элементы речи описываются величинами, представляющими в той или иной форме мгновенные передаточную характеристику речевого тракта и параметры источников его возбуждения. Поскольку эти величины изменяют свои значения срав-

нительно медленно в процессе произнесения речи, то для подробного описания речевых сигналов вполне достаточно ограничиться временной дискретизацией элементов с шагом $\Delta T = 15$ мс. Чаще всего элементами речи x_i выступают мгновенный амплитудный спектр речи или мгновенная автокорреляционная функция, мгновенный продольный профиль акустической трубы речевого тракта, мгновенные значения параметров линейной системы, представляющей речевой тракт, мгновенные значения системы двоичных признаков, характеризующих звуки по месту и способу образования и т. п. Для многих описаний речевого сигнала могут быть указаны взаимнооднозначные преобразования, позволяющие переходить от одного описания к другому. Элементы x_i могут содержать компоненты, описываемые разнородными физическими величинами. Например, наряду с компонентами, представляющими форму амплитудного спектра речи или передаточную характеристику речевого тракта, могут быть компоненты, характеризующие интенсивность элемента, способ его образования (с участием голоса или только шума), относительную частоту основного тона и т. п. Последовательности X_I , элементов x_i получают, анализируя речевой сигнал на интервале (окне) анализа продолжительностью $\Delta T' \geq \Delta T$ и перемещая это окно вдоль оси времени с шагом ΔT . Таким образом, интервалы анализа либо

соприкасаются, либо перекрываются. [1]

1.4 Психофизические единицы измерения

Психоакустика - наука о природе восприятия звука. Органы слуха выполняют двойное кодирование звука, как спектральное, так и временное. В отдельном звуке восприятие выделяет пять основных свойств. Это громкость, тембр, высота, продолжительность и пространственная локализация. При этом громкость можно соотнести с амплитудой колебаний, тембр - с формой волны, высоту - с частотой колебаний.

Использование психофизических единиц для описания речевых сигналов позволяет приблизиться к механизмам человеческого восприятия речи, которые, во многом, являются образцом для систем автоматического распознавания. В данной работе для описания характеристик речевых сигналов используется единица высоты звука мел.

Мел — психофизическая единица высоты звука, применяется главным образом в музыкальной акустике. Количественная оценка звука по высоте основана на статистической обработке большого числа данных о субъективном восприятии высоты звуковых тонов. Результаты исследований показывают, что высота звука связана главным

образом с частотой колебаний, но зависит также от уровня громкости звука и его тембра. Звуковые колебания частотой 1000 Гц при эффективном звуковом давлении $2 \cdot 10^{-3}$ Па (то есть при уровне громкости 40 фон), воздействующие спереди на наблюдателя с нормальным слухом, вызывают у него восприятие высоты звука, оцениваемое по определению в 1000 мел. Звук частоты 20 Гц при уровне громкости 40 фон обладает по определению нулевой высотой (0 мел). Зависимость нелинейна, особенно при низких частотах (для «низких» звуков). Преобразовать значение частоты звука (Гц) в значение высоты (мел) можно по формуле: $m = 1125 \ln(1 + f/700)$

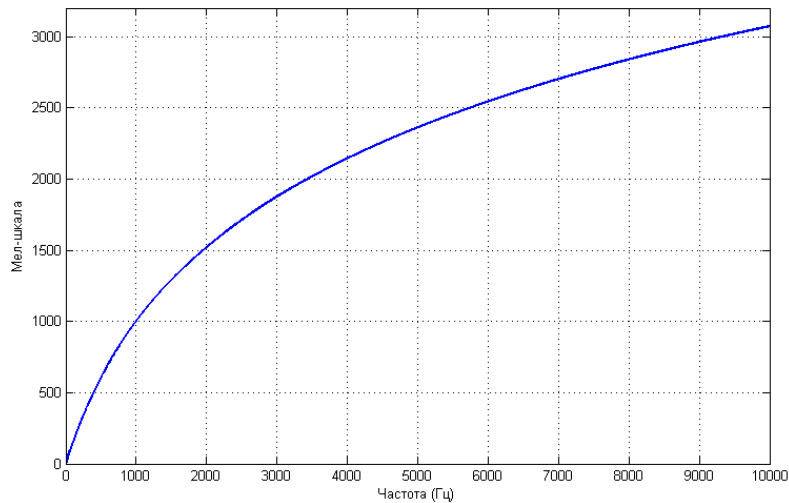


Рис. 1: Зависимость воспринимаемой высоты звука от его частоты

1.5 Кепстральный анализ

Кепстральный анализ применяют для сигналов, представляющих собой свертку двух временных функций, причем таких, что после преобразования их в спектр они образуют неперекрывающиеся на оси q импульсы.

Пример. Задан сигнал $s(t)$ на выходе линейного тракта. Требуется определить некоторую информацию о входном сигнале $s(t)$ и самом тракте, например, о его импульсной характеристике $h(t)$.

Выходной сигнал определяется сверткой $s(t) = s(t) \otimes h(t)$.

Т.к. $S(\omega) = S(\omega)H(\omega)$ в линейном тракте, то после логарифмирования получаются функции $\ln S^2(\omega)$ и $\ln H^2(\omega)$, а на выходе сумма кепстров $C_s(q)$ и $C_h(q)$.

Таким образом, кепстральный анализ позволяет развернуть речевой сигнал, и получить информацию о состоянии артикуляционного аппарата, которая недоступна в частотной или временной области.

2 Реализованные алгоритмы

2.1 Алгоритм нахождения мел-частотных кепстральных коэффициентов

Запишем исходный речевой сигнал как

$$x[n], \quad 0 \leq n < N$$

Получим спектр сигнала, используя ДПФ

$$X_a[k] = \sum_{n=0}^{N-1} x[n] e^{\frac{-2\pi i}{N} kn}, \quad 0 \leq k < N$$

Определим оконные функции. В данном случае использованы треугольные окна, равномерно расположенные относительно мел-шкалы

$$H_m = \begin{cases} 0, & k < f[m-1] \\ \frac{(k-f[m-1])}{(f[m]-f[m-1])}, & f[m-1] \leq k < f[m] \\ \frac{(f[m+1]-k)}{(f[m+1]-f[m])}, & f[m] \leq k \leq f[m+1] \\ 0, & k > f[m+1] \end{cases}$$

Граничные частоты $f[m]$ получены из равенства

$$f[m] = \left(\frac{N}{F_s}\right) B^{-1} \left(B(f_1) + m \frac{B(f_h) - B(f_1)}{M+1} \right)$$

где $B(f)$ — операция перевода значений частоты в мел-шкалу

$$B(f) = 1125 \ln(1 + f/700)$$

Соответственно, обратная операция

$$B^{-1}(b) = 700(\exp(b/1125) - 1)$$

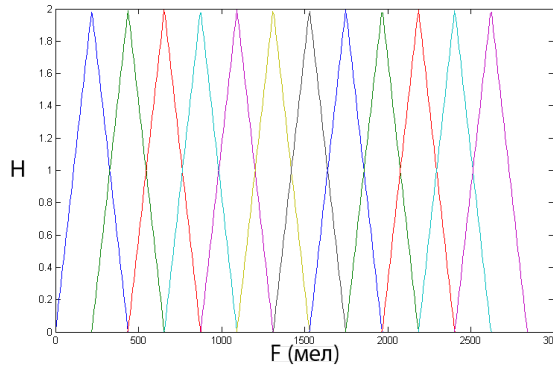


Рис. 2: Окна (мел-шкала)

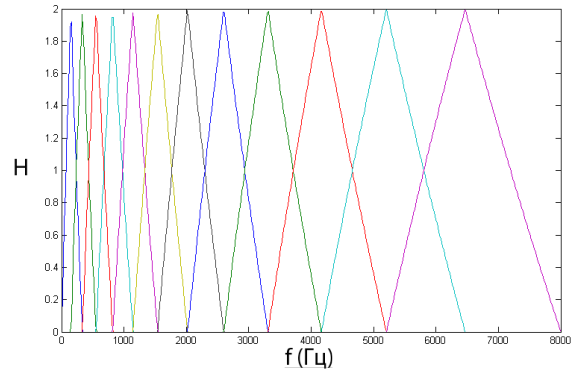


Рис. 3: Окна (частотная шкала)

Вычислим значения энергии сигнала для каждого окна

$$S[m] = \ln\left(\sum_{k=0}^{N-1} |X_a[k]|^2 H_m[k]\right), \quad 0 \leq m < M$$

К полученным значениям применим ДКП

$$c[n] = \sum_{m=0}^{M-1} S[m] \cos(\pi n(m + 1/2)/M), \quad 0 \leq n < M$$

Полученные в результате значения, а также их изменения во времени, используются в дальнейшем как описание речевого сигнала.

2.2 Алгоритм сравнения речевых сигналов с применением динамического программирования (DTW)

Алгоритм динамического трансформирования времени (DTW) вычисляет оптимальную последовательность трансформации (деформации) времени между двумя временными рядами. Алгоритм вычисляет оба значения деформации между двумя рядами и расстоянием между ними.

Предположим, что есть две числовые последовательности $A = a_1, a_2, \dots, a_I$ и $B = b_1, b_2, \dots, b_J$. Длина двух последовательностей может быть различной.

Временные различия между A и B могут быть описаны с помощью некоторой последовательности $c = (i, j)$:

$$F = c(1), c(2) \dots, c(k), \dots, c(K)$$

где $c(k) = (i(k), j(k))$. Данная последовательность представляет собой функцию, которая позволяет отобразить временную ось A на временной оси B . Назовем ее функцией деформации. [3]

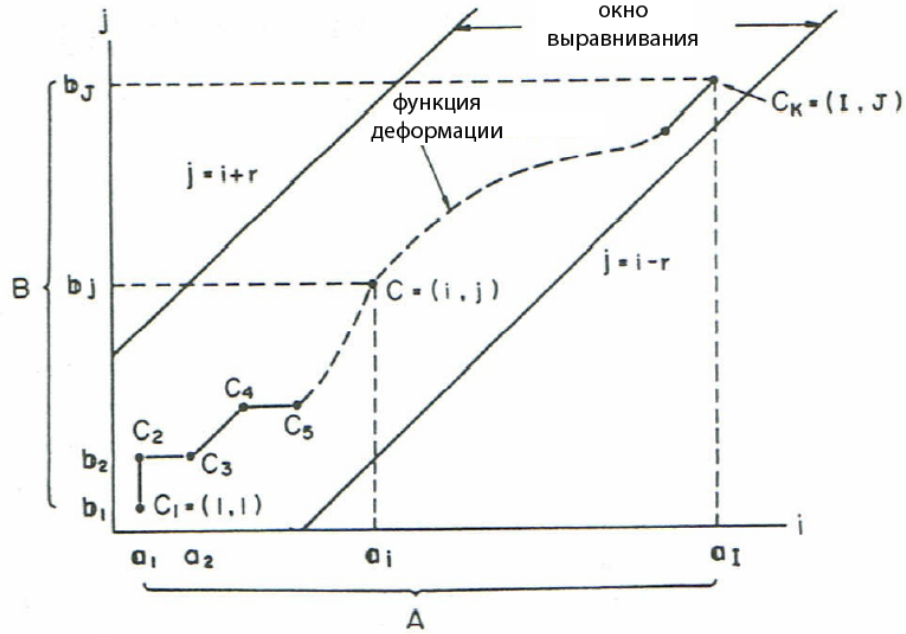


Рис. 4: Функция деформации и окно выравнивания

Алгоритм начинается с расчета локальных расстояний между элементами двух последовательностей. Самый распространенный способ для вычисления расстояний является метод, рассчитывающий модуль разности между значениями двух элементов (Евклидова метрика). В результате получаем матрицу расстояний, имеющую n строк и m столбцов общих членов:

$$d(c) = d(i, j) = |a_i - b_j|, i = 1..I, j = 1..J$$

Взвешенная сумма значений метрик в точках, принадлежащих

функции деформации F

$$E(F) = \sum_{k=1}^K d(c(k)) \cdot w(k)$$

(где $w(k)$ - неотрицательный весовой коэффициент) является мерой доброкачественности функции F . Она принимает минимальное значение, когда функция F оптимально выравнивает временные различия между A и B . Минимальное остаточное расстояние между A и B , которое остается после устранения временных различий, может служить мерой различия речевых последовательностей A и B

$$D(A, B) = \min_F \left[\frac{\sum_{k=1}^K d(c(k)) \cdot w(k)}{\sum_{k=1}^K w(k)} \right]$$

Существует три условия, налагаемых на DTW алгоритм для обеспечения быстрой конвергенции:

1. Монотонность – путь никогда не возвращается, то есть: оба индекса, i и j , которые используются в последовательности, никогда не уменьшаются.
2. Непрерывность – последовательность продвигается постепенно: за один шаг индексы i и j , увеличиваются не более чем на 1.

3. Предельность – последовательность начинается в $(1,1)$ и заканчивается в (I,J) .

Практическая реализация данного алгоритма представляет собой нахождение значения нормированного расстояния

$$D(A, B) = \frac{1}{N} g_K(c(K))$$

где $g_k(c(k))$ можно найти из уравнения:

$$g_k(c(k)) = \min_{c(k-1)} [g_{k-1}(c(k-1)) + d(c(k)) \cdot w(k)]$$

Начальное условие:

$$g_1(c(1)) = d(c(1)) \cdot w(1)$$

Из ограничений следует, что $c(1) = (1, 1)$,

$$g(i, j) = \min \begin{bmatrix} g(i, j-1) + d(i, j) \\ g(i-1, j-1) + 2d(i, j) \\ g(i-1, j) + d(i, j) \end{bmatrix}.$$

В данном случае нормированное расстояние можно записать как

$$D(A, B) = \frac{1}{N} g(I, J), \text{ где } N = I + J.$$

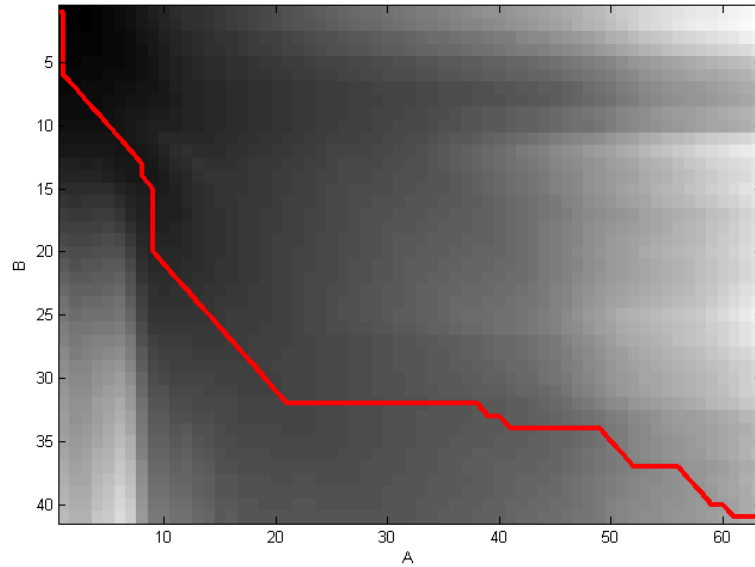


Рис. 5: Пример нахождения функции деформации

Эксперименты проводились для словаря из 10 слов (цифры от 0 до 9). Одним диктором было записано 100 повторений для сравнения (по 10 для каждого в словаре).

С помощью данного алгоритма производится сравнение анализируемого сигнала с сохраненными в памяти компьютера эталонами. В результате выбирается пара с минимальной дистанцией и делается вывод о соответствии сигнала слову из словаря. Пример результата сравнения можно видеть на рис. 7 (Меньшее значение дистанции означает большее сходство).

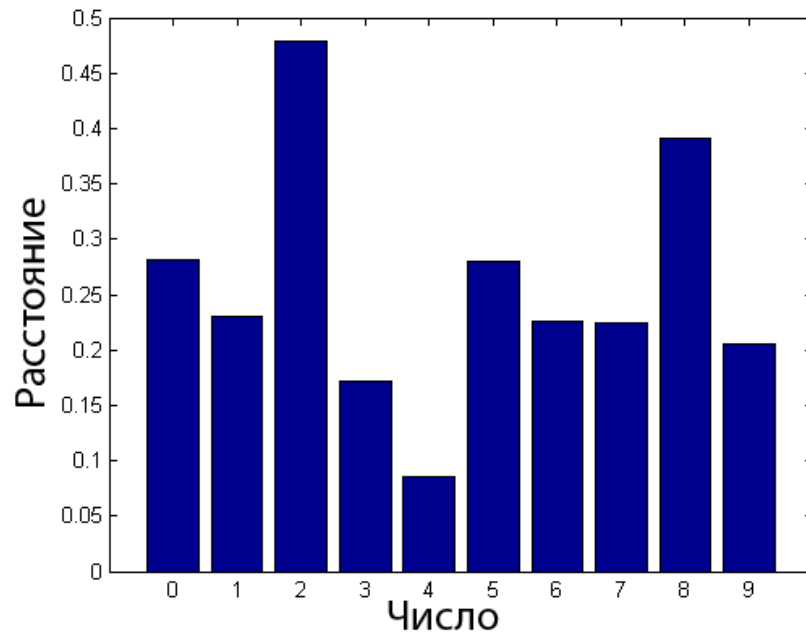


Рис. 6: Диаграмма расстояний для слова «четыре»

В результате, на 100 повторений было обнаружено 2 ошибки распознавания, что позволяет говорить о достаточной точности выбранного алгоритма. На рис. ?? показана диаграмма расстояний для ошибочно распознанного слова «восемь»

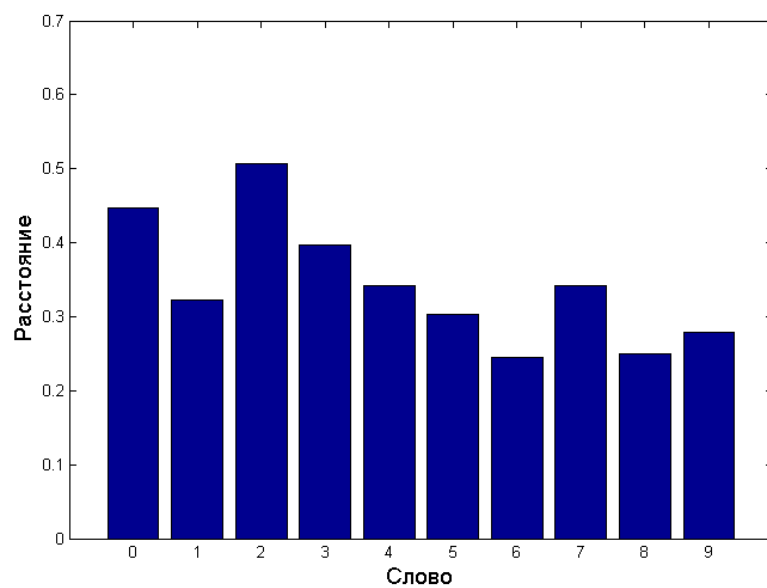


Рис. 7: Диаграмма расстояний для ошибочно распознанного слова «восемь»

2.3 Листинг программы на языке MATLAB

```
1 function out = mfcc(sig, nfilt, wlen, woverlap) % MFCC calculation
2 % sig = input signal
3 % nfilt = number of coefficients
4 % wlen = window width
5 % woverlap = window increment
6 %
7 %
8
9 if nargin < 4
10     woverlap = 128;
11     if nargin < 3
12         nfilt = 24;
13         if nargin < 2
14             wlen = 256;
15         end
16     end
17 end
18
19 bank = melbankm(nfilt, wlen, 16000, 0, 0.5, 'm');
20 bank = full(bank);
21 bank = bank / max(bank(:));
22
23 dctcoef = zeros(nfilt / 2, nfilt);
24
25 for k = 1:(nfilt / 2)
26     n = 0: nfilt - 1;
```

```

27     dctcoef(k,:)=cos((2*n+1)*k*pi/(2*nfilt));
28 end
29
30
31 w = 1 + 6 * sin(pi * (1:nfilt/2) ./ (nfilt/2));
32 w = w/max(w);
33
34
35 xx=double(sig);
36 xx=filter([1 -0.9375],1,xx);
37
38
39 xx=enframe(xx,wlen,woverlap);
40
41 m=zeros(size(xx,1),size(dctcoef,1));
42
43 for i=1:size(xx,1)
44     y = xx(i,:);
45     s = y' .* hamming(wlen);
46     t = abs(fft(s));
47     t = t.^2;
48     c1=dctcoef * (log(bank * t(1:(wlen/2+1))));
49     c2 = c1.*w';
50     m(i,:)=c2';
51 end
52
53
54 dtm = zeros(size(m));
55 for i=3:size(m,1)-2
56     dtm(i,:) = -2*m(i-2,:) - m(i-1,:) + m(i+1,:) + 2*m(i+2,:);

```

```

57 end
58 dtm = dtm / 3;
59
60
61 out = [m dtm];
62
63 out = out(3:size(m,1)-2,:);

```

```

1 function [Dist, p, q] = dtw( siga, sigb )
2 %DIW Dynamic time warp
3
4 S1=siga';
5 S2=sigb';
6 ES1 = sqrt(sum(S1.^2))+1e-12;
7 ES2 = sqrt(sum(S2.^2))+1e-12;
8 M = (S1'*S2)./(ES1'*ES2);
9 M=1-M;
10
11 [r,c] = size(M);
12
13
14 D = zeros(r+1, c+1);
15 D(1,:) = NaN;
16 D(:,1) = NaN;
17 D(1,1) = 0;
18 D(2:(r+1), 2:(c+1)) = M;
19
20

```



```

21     phi = zeros(r,c);
22
23
24     for i = 1:r;
25         for j = 1:c;
26             if (abs(i-j*r/c)<r/4)
27                 [dmax, tb] = min([2*D(i, j), D(i, j+1), D(i+1, j)]);
28             elseif (i>j)
29                 dmax=D(i, j+1);
30                 tb=2;
31             else
32                 dmax=D(i+1,j);
33                 tb=3;
34             end
35             D(i+1,j+1) = D(i+1,j+1)+dmax;
36             phi(i, j) = tb;
37         end
38     end
39
40
41     i = r;
42     j = c;
43     p = i;
44     q = j;
45     while i > 0 && j > 0
46         tb = phi(i, j);
47         if (i==1)
48             j=j-1;
49         elseif (j==1)
50             i=i-1;

```

```

51     else
52         if (tb == 1)
53             i = i - 1;
54             j = j - 1;
55         elseif (tb == 2)
56             i = i - 1;
57         elseif (tb == 3)
58             j = j - 1;
59         else
60             error;
61         end
62     end
63     p = [i, p];
64     q = [j, q];
65 end
66
67
68 D = D(2:(r+1), 2:(c+1));
69 Dist=D(end,end)/length(p);
70
71 end

```

```

1 function err = compare10(suffix, Dict)
2 err=0;
3 for count=0:9
4     [s, sr] = wavread(strcat(num2str(count), '_ ', suffix, '.wav'));
5
6

```

```

7      s=truncword(s);
8      %sound(s, sr);
9      %s=awgn(s,20, 'measured ');
10
11
12      hashKeys = Dict.keys;
13      i=1;
14      while hashKeys.hasMoreElements
15
16          key=hashKeys.nextElement;
17          wordslist(i)=key;
18          mfcc_s=mfcc(s);
19          mfcc_e=Dict.get(key);
20          distancelist(i)=dtw(mfcc_s(:,1:end),mfcc_e(:,1:end));
21          i=i+1;
22      end
23
24
25      [m, k] = min(distancelist);
26
27      if (10-k)~=count
28          err=err+1;
29
30      end
31
32  end

```

```

1  clc

```

```

2 clear
3 close all
4 %Creating dictionary hashtable
5 Dict = java.util.Hashtable;
6
7 [s_e1_3, sr] = wavread('1_3.wav');
8 s_e1_3=mfcc(truncword(s_e1_3));
9 Dict.put('1',s_e1_3);
10
11 [s_e2_3, sr] = wavread('2_2.wav');
12 s_e2_3=mfcc(truncword(s_e2_3));
13 Dict.put('2',s_e2_3);
14
15 [s_e3_3, sr] = wavread('3_3.wav');
16 s_e3_3=mfcc(truncword(s_e3_3));
17 Dict.put('3',s_e3_3);
18
19 [s_e4_3, sr] = wavread('4_3.wav');
20 s_e4_3=mfcc(truncword(s_e4_3));
21 Dict.put('4',s_e4_3);
22
23 [s_e5_3, sr] = wavread('5_2.wav');
24 s_e5_3=mfcc(truncword(s_e5_3));
25 Dict.put('5',s_e5_3);
26
27 [s_e6_3, sr] = wavread('6_3.wav');
28 s_e6_3=mfcc(truncword(s_e6_3));
29 Dict.put('6',s_e6_3);
30
31 [s_e7_3, sr] = wavread('7_3.wav');

```

```

32 s_e7_3=mfcc(truncword(s_e7_3));
33 Dict.put('7',s_e7_3);
34
35 [s_e8_3, sr] = wavread('8_3.wav');
36 s_e8_3=mfcc(truncword(s_e8_3));
37 Dict.put('8',s_e8_3);
38
39 [s_e9_3, sr] = wavread('9_3.wav');
40 s_e9_3=mfcc(truncword(s_e9_3));
41 Dict.put('9',s_e9_3);
42
43 [s_e0_3, sr] = wavread('0_2.wav');
44 s_e0_3=mfcc(truncword(s_e0_3));
45 Dict.put('0',s_e0_3);
46
47 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
48
49 err=0;
50 for i=2:8
51     err=err+ compare10(num2qstr(i),Dict);
52 end

```

Заключение

Рассмотренный в работе алгоритм показал достаточно высокую точность (98%). При этом минимизировано влияние на точность распознавания громкости (за счет нормирования) и скорости произнесения (за счет методов динамического программирования). Вопросы устойчивости к влиянию шума и дикторозависимости, которые не исследовались в данной работе, могут быть предметом дальнейших исследований.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Винцюк Т.К. Анализ, распространение и интерпретация речевых сигналов. Киев: Наукова думка, 1987.
- [2] Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, Spoken Language Processing: A Guide to Theory, Algorithm, and System Development, Prentice Hall, 2001, ISBN:0130226165
- [3] H. Sakoe and S. Chiba, «Dynamic programming optimization for spoken word recognition», IEEE Trans. Acoust. Speech Signal Process., Vol. ASSP-26, No. 1, Feb. 1978
- [4] Мазуренко И.Л. Компьютерные системы распознавания речи. Интеллектуальные системы, Москва, 1998 г.
- [5] П. Линдсей, Д. Норман Переработка информации у человека — М: Мир, 1974