





Оглавление

Введение	5
1 Обзор существующих методов.....	9
1.1 Формирование системы признаков	10
1.1.1 Спектрально-временные признаки	10
1.1.2 Кепстральные признаки	10
1.2 Классификаторы	11
1.2.1 Скрытые Марковские модели	11
1.2.2 Метод опорных векторов	13
1.2.3 Модель гауссовых смесей.....	14
1.2.4 Векторное квантование	15
1.3 Обзор существующих методов идентификации звукового сигнала в базах данных .	
1.3.1 Алгоритм поиска музыкальных композиций <i>Shazam</i>	16
1.3.2 Алгоритм поиска музыкальных композиций <i>Яндекс</i>	18
1.4 Выводы по главе 1	20
2 Информационная модель поиска звукового сигнала в базе данных	22
2.1 Теория активного восприятия	22
2.2 Система поиска звукового сигнала в базе данных с использованием теории активного восприятия	24
2.2.1 Предварительная обработка сигнала	25
2.2.2 Формирование системы признаков.....	25
2.2.3 Размещение сегментов сигнала в бинарном дереве	27
2.2.4 Поиск звукового сигнала в базе данных.....	28
2.3 Выводы по главе 2	29
3 Вычислительный эксперимент	31
3.1 Описание эксперимента.....	31
3.2 Описание программного продукта	32
3.3 Анализ результатов	35
3.4 Выводы по главе 3	41
Заключение.....	43
Список литературы.....	44
ПРИЛОЖЕНИЕ А	46

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)						
Изм.	Лист	№ докум.	Подпись	Дата	Информационная модель и алгоритмы идентификации звукового сигнала			Лит.	Лист	Листов	
Разраб.		Шарутин К. Н.		02.07.2020						4	45
Провер.		Гай В. Е.		02.07.2020							
Н. Контр.		Гай В. Е.		02.07.2020							
Утверд.		Жевнерчук Д.В.		02.07.2020				НГТУ НГТУ кафедра ВСТ			

Введение

В настоящее время, в связи с развитием и внедрением в повседневную жизнь информационных технологий, одной из самых динамично развивающихся областей является область, связанная с задачей распознавания звуковых сигналов.

Перспективность исследований в данной области подтверждается тем, что на текущий момент не существует систем, которые решали бы проблему распознавания на уровне аналогичном способностям человеческого мозга распознавать звуковые сигналы.

Технология распознавания звуковых сигналов имеет широкое применение в таких задачах как, например, поиск произведения по его фрагменту в базах данных, а также в задачах выявления плагиата.

Рассмотрим более подробно области возможного применения, приведя следующую классификацию задач, в которых могут быть востребованы технологии распознавания звуковых сигналов:

1) Поиск композиции по ее фрагменту – в качестве фрагмента используется некоторая часть музыкальной произведения или же самостоятельно воспроизведенный фрагмент, на основе которого производится поиск в базе данных. База данных чаще всего представляет собой набор заранее собранных композиций, которые могут быть преобразованы в сжатое символьное описание для уменьшения вычислительных затрат.

2) Выявление плагиата – используется в сфере охраны интеллектуальной собственности. Позволяет автоматически определять заимствования певцами или композиторами звуковых фрагментов из чужих песен.

3) Криминалистика – поиск и сопоставление двух записей голосов, выявление сходства.

4) Голосовая биометрия - анализ уникальных характеристик речи диктора, обусловленных манерами разговора и строением речевого аппарата. Преимуществом данной технологии является возможность идентификации пользователя на расстоянии.

На основании перечисленных сфер применения, а также отсутствия общепринятых методов решения задачи распознавания можно сделать вывод, что существует необходимость разработки программных систем, реализующих распознавание звуковых сигналов.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						5
Изм	Лист	№ докум.	Подп.	Дата		

Цель работы и задачи исследования

Целью исследования данной работы является разработка информационной модели системы идентификации звуковых сигналов с использованием положений теории активного восприятия. Кроме того, необходимо разработать алгоритм поиска звукового сигнала в базе данных, а также выполнить исследования эффективности разработанного алгоритма и провести сравнительного анализа полученных результатов с другими результатами и методами.

Задачи, которые необходимо решить для достижения поставленной цели:

- 1) Анализ существующих методов в данной проблемной области.
- 2) Построение информационной модели системы, осуществляющей распознавание музыкальных произведений.
- 3) Разработка алгоритма поиска звукового сигнала в базе.
- 4) Проведение вычислительного эксперимента.
- 5) Сравнительный анализ полученных результатов.

Объект исследования

Объектом исследования данной научной работы выступает сигнал, содержащий запись музыкального произведения для поиска схожих сигнатур в звуковой базе данных.

Предмет исследования

Предметом исследования в данной выпускной квалификационной работе являются модели и алгоритмы, реализующие поиска звукового сигнала в базе данных.

Методы исследования

В качестве методов исследования в ходе выполнения использовались методы формирования признакового описания звукового сигнала, основанные на теории активного восприятия, аналитический метод и вычислительный эксперимент.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						6
Изм	Лист	№ докум.	Подп.	Дата		

Положения, выносимые на защиту

- 1) Модель системы идентификации звукового сигнала в базе данных с использованием теории активного восприятия;
- 2) Алгоритм поиска звукового сигнала в базе данных с использованием бинарных деревьев, содержащих признаковые описания сигналов.
- 3) Результаты вычислительного эксперимента.

Научная новизна

Научная новизна работы заключается в разработке информационной модели системы распознавания музыкальных произведений на основе теории активного восприятия, построении базы данных на основе бинарного дерева и создании алгоритма поиска звуковой композиции.

Теоретическая и практическая ценность работы

Теоретическая ценность работы заключается в разработанной модели системы идентификации звукового сигнала в базе данных с использованием признакового описания, разработанного на основе теории активного восприятия.

Практическая ценность работы заключается в разработке программного комплекса для решения поставленной задачи, а также его тестирования и сравнения с другими методами идентификации звукового сигнала в базе данных.

Апробация работы

Апробация выпускной квалификационной работы была проведена в ходе защиты публикации по теме данной работы в рамках XXVI международной научно-технической конференции «Информационные системы и технологии – 2020».

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						7
Изм	Лист	№ докум.	Подп.	Дата		

Обоснованность и достоверность

Обоснованность и достоверность проведенного исследования, полученных результатов и выводов доказывается проведением вычислительного эксперимента, сравнением полученных результатов с известными и апробацией основных положений выпускной квалификационной работы в докладе на международной научно-технической конференции.

Публикации

1) К. Н. Шарутин, В.Е. Гай – Информационная модель и алгоритмы идентификации звукового сигнала // Материалы XXVI международной научно-технической конференции «Информационные системы и технологии - 2020», ИСТ-2020, Россия, Н. Новгород, 2020г.

Объем и структура работы

Выпускная квалификационная работа состоит из введения, трёх глав основной части, заключения и списка литературы. Общий объём работы – 46 страниц. Диссертация содержит 10 рисунков, 12 таблиц, 14 формул. Список литературы включает в себя 19 наименований.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						8
Изм	Лист	№ докум.	Подп.	Дата		

1 Обзор существующих методов

Задача распознавания звукового сигнала включает в себя три основных этапа (рисунок 1):

- предварительная обработка сигнала;
- формирование системы признаков;
- принятие решения (классификация);

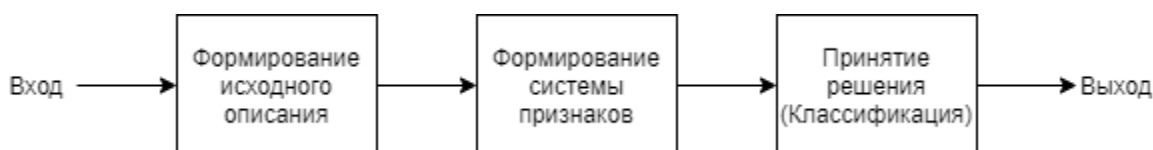


Рисунок 1 Схема системы распознавания звукового сигнала

Этап предварительной обработки сигнала включает в себя нормализацию входного сигнала, возможное удаление шумов, а также представление сигнала в удобной для анализа форме.

Этап формирования системы признаков позволяет получить информацию об исследуемом элементе, которая однозначно характеризовала бы его.

На этапе принятия решения используются признаковые описания, полученные на предыдущем шаге, для того идентифицировать принадлежность сигнала к той или иной группе. Существует большое число различных алгоритмов классификации, которые используются при принятии решения.

Основными этапами при поиске звукового сигнала в базе данных являются формирование признакового описания входного сигнала, а также сам поиск сигнала в базе данных.

Для формирования системы признаков часто используется подход основанный на быстром преобразовании Фурье, а также кепстральные признаки. В данной работе предполагается использование теории активного восприятия для формирования системы признаков.

На этапе поиска сигнала в звуковой базе данных используются алгоритмы с применением скрытых марковских моделей, метода опорных векторов, а также хэш таблиц с ключами. В данной работе система признаков будет храниться в узлах сформированного бинарного дерева, а поиск музыкальных композиций заключается в

сравнении признакового описания фрагмента музыкальной композиции с фрагментами из звуковой базе данных.

1.1 Формирование системы признаков

1.1.1 Спектрально-временные признаки

Спектрально-временные признаки характеризуют сигнал речевого аппарата человека исходя из его физико-математической сущности [1]. Данные признаки отражают формы временного ряда, а также спектр голосовых импульсов совместно с особенностями фильтрации речевого аппарата у разных людей. Речевой сигнал представляет собой последовательность значений энергетических спектров, которые были измерены в различные моменты времени. После этого речевой сигнал подвергается спектральному анализу с использованием быстрого преобразования Фурье, результатом которого является набор амплитудного спектра i -ой частоты. Данный набор представляет собой систему признаков для конкретного звукового сигнала.

Признаки, полученные с помощью быстрого преобразования Фурье, позволяют создавать ощущение звуков, но могут терять важную динамическую составляющую информации. Они уменьшают точность распознавания, ассоциируясь с другими признаками, добавляя сигналу лишние шумы.

Данный подход обеспечивает лучшую точность распознавания в фиксированных условиях. Является неустойчивым к влиянию внешней среды и к голосу самого говорящего.

1.1.2 Кепстральные признаки

Использование кепстральных признаков является очень распространенным методом для определения признаков звукового сигнала.

Данный метод основывается на восприятии звуковой информации по подобию человеческих органов слуха. Кепстральные признаки используют мэл-шкалу, которая связана с критическими полосами слуха. Мэл – единица высоты звука, основана на восприятии этого звука органами слуха [2].

Имеется входная последовательность, на которую накладывается весовая функция. Распространенный вариант весовой функции – окно Хэмминга, представленное

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						10
Изм	Лист	№ докум.	Подп.	Дата		

следующей формулой:

$$w_n = 0,54 - 0,46 \cdot \cos\left(\frac{2\pi n}{N-1}\right), n = 0, \dots, N-1 \quad (1)$$

На следующем шаге происходит применение дискретного преобразования Фурье
Дискретное преобразование Фурье:

$$X_k = \sum_{n=0}^{N-1} x_n \cdot w_n \cdot e^{\frac{-2\pi i}{N} k \cdot n}, k = 0, \dots, N-1 \quad (2)$$

После того, как была произведено преобразование Фурье, полученный спектр сигнала подается на вход фильтров разбиения сигнала в частотной области на диапазоны. Граница каждого из фильтров вычисляется с помощью шкалы мэл.

Перевод в мэл-частотную область можно описать формулой:

$$B(f) = 1127 \cdot \ln\left(1 + \frac{f}{700}\right) \quad (3)$$

Существует некоторое количество фильтров и исследуемый диапазон частот. Разбивая исследуемый диапазон на 24 равномерно-распределенных перекрывающихся диапазона, образуется мэл шкала. Кроме того, необходимо вычислить соответствующие границы в области линейных частот.

Данные фильтры применяются к квадратам модулей коэффициентов преобразования Фурье. Результаты, полученные на данном этапе, логарифмируются.

Для получения коэффициентов необходимо произвести дискретное косинусное преобразование, чтобы завершения вычисления мел-кепстральных коэффициентов.

Преимуществами использования мэл-кепстральных признаков является использование мэл-шкалы, выделяющая наиболее близкие человеческому уху признаки, а также ограниченное число коэффициентов для сжатия обрабатываемой информации.

К недостаткам данного метода можно отнести низкую устойчивость в внешнем шума, что значительно ухудшает показатели распознавания.

1.2 Классификаторы

1.2.1 Скрытые Марковские модели

Конечный автомат – основная единица, которая является основой скрытых марковских моделей [3]. В алгоритме скрытых марковских моделей переход между состояниями определяется текущим состоянием системы. Кроме того, для перехода между состояниями скрытой марковской модели используется вероятностный закон. На рисунке 2 представлена структурная схема переходов между состояниями скрытой

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						11
Изм	Лист	№ докум.	Подп.	Дата		

марковской модели (рисунок 2):

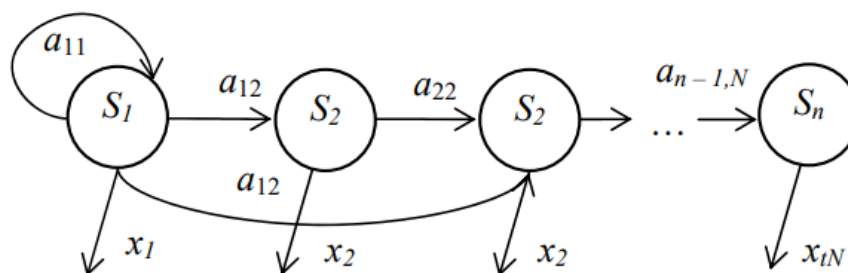


Рисунок 2 Диаграммы переходов между состояниями СММ.

Выходной вектор при использовании данного алгоритма представлен в виде последовательности векторов (наблюдение) $\{x_1, x_2, \dots, x_T\}$ длиной T .

Важным преимуществом скрытых марковских моделей перед нейронными сетями является способность обрабатывать входные последовательности и сигналы разных длин.

Использование скрытых марковских моделей в задаче распознавания осуществляется в два этапа:

1) Обучение – обучение позволяет определить такие параметры, как матрица вероятностей переходов между состояниями скрытых марковских моделей, параметры гауссовых смесей, среди них математическое ожидание, матрица ковариации, а также веса для каждого из состояний. На данном этапе распространенным решением является использование алгоритма Баума-Велча.

2) Декодирование – процесс декодирования позволяет определить вероятность получения последовательности векторов конкретной модели. Для определения данной вероятности используется алгоритм максимума правдоподобия Витерби.

Эффективное обучение скрытой марковской модели напрямую зависит от числа входных образцов.

Также важно учесть условие линейной независимости данных образцов. Отсутствие данного условия приводит к вырождению матрицы ковариации в следствии чего модель становится полностью неработоспособной.

Практический подход к использованию метода скрытых марковских моделей позволяет решать следующие задачи [4]:

- 1) Задача выбора параметрических векторов, среди которых используются кепстральные коэффициенты или коэффициенты линейного предсказания.

- 2) Нормализация параметрического вектора.
- 3) Выбор количества состояний модели N , а также числа компонент гауссовой смеси M .
- 4) Необходимость сегментация обучающих векторов, что позволяет найти значений математических ожиданий гауссовых смесей на первоначальном шаге обучения.

1.2.2 Метод опорных векторов

Метод опорных векторов – метод, который используется в задачах бинарной классификации. Данный метод строит разделяющую функцию вида:

$$f(x) = w \cdot x + b \quad (4)$$

Пусть задана последовательность точек пространства признаков $X = \{x_1, x_2, \dots, x_n\}$ с метками $Y = \{y_1, y_2, \dots, y_n\}$, $y_i \in \{-1, 1\}$, $1 \leq i \leq n$, соответствующими двум классам. Рассмотрим случай линейной разделимости данных для нахождения функции $f(x)$:

$$\begin{cases} w \cdot x_i + b \geq 1, y_i = 1 \\ w \cdot x_i + b \leq -1, y_i = -1 \end{cases} \Leftrightarrow y_i \cdot (w \cdot x_i + b) - 1 \geq 0, 1 \leq i < n \quad (5)$$

Увеличение расстояния между плоскостями, разделяющими признаки, позволяет добиться большей надежности при разделении классов.

В случае неразделимости гиперплоскостей необходимо использовать функцию ядра [5].

Функция ядра может быть представлена в виде скалярного произведения в новом пространстве. В качестве нового пространства выступает текущее пространство, но отраженное с более глубокой размерностью, что позволяет произвести линейное разделение.

Функция ядра в виде скалярного произведения в новом пространстве:

$$K(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j) \quad (6)$$

Метод опорных векторов использует разделяющие параметры функции $f(x)$ и параметры функции ядра в качестве сохраняемой модели диктора.

После вычисления решающей функции $f(x)$ определяется принадлежность вектора x' соответствующему классу на основании знака выражения $f(x')$.

Среди преимуществ метода опорных векторов можно выделить высокую точность классификации, которой можно добиться, правильно подобрав функции ядра.

К недостаткам данного метода можно отнести проблему, связанную с выбором

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						13
Изм	Лист	№ докум.	Подп.	Дата		

функции ядра. Кроме того, данный алгоритм очень медленно обучается при использовании его в задачах классового распознавания.

1.2.3 Модель гауссовых смесей

В современных системах распознавания речи широко используются модели гауссовых смесей. Данный алгоритм использует метод максимального ожидания, чтобы оценить степень правдоподобия используемых моделей. Кроме того, метод максимального ожидания позволяет веса компонентов [6].

Метод максимального ожидания состоит из двух шагов:

- 1) Вычисляются вероятности принадлежности каждой точки данных к кластеру на основании параметров модели.
- 2) Происходит обновление модели в соответствии с кластерным распределением, проведенным на первом шаге.

В качестве входных данных подается обучающая последовательность векторов $X = \{x, \dots, x_n\}$.

Данная модель представляет собой взвешенную сумму Гауссиан

$$p\left(\frac{x}{\alpha}\right) = \sum_{i=1}^M \omega_i \cdot p_i(x) \quad (7)$$

где α – модель диктора, M – количество компонентов модели, ω – веса компонентов,

такие что $\sum_{i=1}^M \omega_i = 1$,

Существует целый ряд акустических моделей, которые связаны между собой. Результаты выполнения каждой модели, которая предшествовала текущей модели, могут быть использованы как входные параметры для работы с более сложной моделью. Таким образом получается, что более точные данные у моделей, которые основываются на предыдущих (рисунок 3).

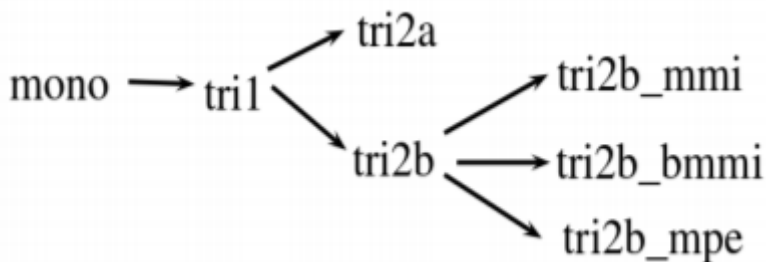


Рисунок 3 Иерархия акустических моделей.

Первоначально обучается моно-фононная модель “*mono*”, которая использует мел-частотные кепстральные коэффициенты.

Следующим шагом обучается модель три-фононная “*tri2a*”, вместе с ней запускается обучение модели “*tri2b*”, на вход которой подаются линейные преобразования, которые уменьшают размерность входных данных, и оценка параметров линейного преобразования для того, чтобы максимально повысить вероятность.

С использованием *tri2b* можно обучить еще три акустические модели по следующим инструкциям:

- *Maximum mutual information* – увеличение количества обучения, максимизируют вероятность правильного выбора слова. Является чувствительным к отсутствию транскрипции или при противоречии высказываний.
- *Boosted maximum mutual information* – модель обучается несколько раз с увеличенным параметром $beam=0.05$, что дает более высокую скорость обучения и более точные результаты.
- *Minimum phone error* – модель также обучается несколько раз, использует сглаженное приближение к ошибке в фонемах.

1.2.4 Векторное квантование

Метод векторного квантования использует разбиение множества обучающих векторов в множество кодовых векторов фиксированного размера [7].

В основе данного метода лежит создание кодовых книг. Под кодовой книгой подразумевается кодовых векторов.

Чтобы получить первоначальную кодовую книгу можно воспользоваться алгоритмом *K*-средних.

На вход алгоритма подается кодовый вектор, который сравнивается с каждым существующим кодовым словом используя для этого евклидово расстояние. Происходит распределение текущего множества по группам. Каждая новая итерация заключается в распределении векторов в ближайшие группы и вычисления расстояния.

Кодовое слово с наименьшим расстоянием является эталонным.

К преимуществам данного метода можно отнести простоту реализации,

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						15
Изм	Лист	№ докум.	Подп.	Дата		

применимость к задачам текстонезависимой идентификации диктора, а также низкую вычислительную нагрузку.

Среди недостатков можно отметить низкую точность распознавания при временном изменении сигнала, так как происходит смешивание векторов.

1.3 Обзор существующих методов идентификации звукового сигнала в базах данных

Наиболее популярными на данный момент алгоритмами распознавания звуковых фрагментов являются алгоритмы, которые используют предварительно сформированную базу данных.

Крупные компании чаще всего используют максимальные колебания амплитуды в песне для уникальной идентификации звукового сигнала. Набор подобных колебаний позволяет описать музыкальные композиции.

В качестве признаков описаний сигнала чаще всего используются частотные характеристики, полученные с помощью преобразования Фурье, или мел-кепстральные коэффициенты.

Важным фактор при построении подобных алгоритмов является скорость поиска схожих компонентов с входным сигналом из базы данных. Подобные задачи решаются с помощью нейронных сетей, алгоритмов фильтрации, а также простого перебора элементов в базе данных.

Принцип работы систем поиска в базах данных заключается в отправке фрагмента аудиосигнала на сервер, дальнейшей его обработке и поиске среди существующих музыкальных композиций.

Далее будет представлен обзор двух наиболее популярных алгоритмов распознавания музыкальных фрагментов в базах данных.

1.3.1 Алгоритм поиска музыкальных композиций *Shazam*

Приложение *Shazam* предназначено для поиска песни в звуковой базе данных на основании некоторого звукового фрагмента, который подвергается анализу для поиска соответствий.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						16
Изм	Лист	№ докум.	Подп.	Дата		

Для получения признакового описания звуковых сигналов данное приложение использует быстрое преобразование Фурье [8].

Данный алгоритм использует метод скользящего окна для анализа участков фиксированной длины, после чего выдает признаковое описание для текущего участка.

Для получения цифровой сигнатуры сигнала выбираются наиболее значимые частоты, то есть те частоты, которые обладают наибольшей амплитудой. Выбор частот с максимальной амплитудой обусловлен тем, что данные частоты имеют больший шанс сохранить свою эффективность в условиях, когда звуковой сигнал может быть подвергнут зашумленности.

Эффективным способом хранения и поиска музыкальных композиций является использование их признаковых описаний в качестве ключей хэш-таблицы. Ключ каждой композиции представлен некоторым набором метаданных. Среди этих метаданных присутствуют название песни, время появления определенных частот, а также исполнитель. В таблице 1 представлен пример, как подобные записи могут храниться в базе.

Таблица 1 Соотношение хэш-значения с метаданными музыкальной композиции

Хэш-значение	Время появления частот, с	Метаданные
43 55 77 171 394	52,01	Песня А Исполнитель А
53 56 92 151 899	39,08	Песня В Исполнитель В
23 57 95 111 127	22,10	Песня Г Исполнитель Г
60 55 79 199 255	6,75	Песня С Исполнитель С

Для поиска музыкальной композиции происходит запись звука, выполняется вышеописанный процесс определения сигнатур, после чего начинается непосредственно сам поиск композиций в музыкальной базе по вычисленному хэш-значению.

При идентификации композиции иногда случается, что одно хэш-значение полученного сигнала может соответствовать сразу двум или более различным песням. Чтобы избежать данной ситуации, необходимо произвести дробление сигнала и проверить на совпадении хэш-значения нескольких сегментов.

1.3.2 Алгоритм поиска музыкальных композиций *Яндекс*

Приложение *Яндекс.Музыка* является одним из самых популярных приложений для прослушивания потоковой музыки. Кроме того, данное приложение обладает функционалом для распознавания музыкальных треков. *Яндекс.Музыка* обладает большой базой данных музыкальных треков, среди них присутствуют композиции, приобретенные у правообладателей, а также подборка самых популярных треков из интернета.

На этапе разработки своего алгоритма компания отвергла ряд подходов к поиску звукового фрагмента в базе данных:

- побитовое сравнение. Помехи, шумы, громкоговоритель источника звука, акустика помещения, неравномерная амплитудно-частотная характеристика микрофона - все эти факторы создают неточности в перекодировании записанного фрагмента и делают неприемлемым побитовое сравнение.
- использование водяных знаков. Данный подход является эффективным, если компания встраивает водяные знаки в музыкальные композиции еще на этапе звукозаписи. Данные водяные знаки легко распознаются алгоритмами, но недоступны человеческому уху. *Яндекс.Музыка* является потоковым сервисом воспроизведения музыки, а не написания ее, поэтому данный подход не является актуальным в этом случае.
- нестрогое сравнение спектрограмм. Спектрограммы можно рассматривать как изображения и использовать для поиска интересных композиций. Но данный метод обладает рядом проблем, таких как, например, большой объем изображений треков, так как внутренняя база музыкальных композиций компании насчитывает более 8 миллионов треков. Данный процесс является сильно ресурсозатратной, а также очень длительной.

Пики спектрограмм, локального максимума амплитуды, являются устойчивыми к помехам и шумам. Положение данных пиков в системе “частота-время” почти не меняется. В среднем один музыкальный трек представлен в виде 250 тысяч пиков, что является гораздо меньшим числом, чем при использовании изображений треков. Учитывая большую музыкальную базу данных компании, данный подход является наиболее эффективным [9].

В данном алгоритме используется двухэтапное сравнение из-за того, что при записи фрагменты добавляют новые пики и перекрывают уже имеющиеся:

- поиск смещения для каждого трека, при котором с запросом совпадает

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						18
Изм	Лист	№ докум.	Подп.	Дата		

максимальное количество пиков;

- поиск из выбранных треков трека с наибольшим числом совпадений;

Чтобы выполнить данное сравнение необходимо построить гистограмму, в которой для каждой частоты пика из запроса и трека откладывается +1 по оси Y в том смещении, где нашлось совпадение (рисунок 4):

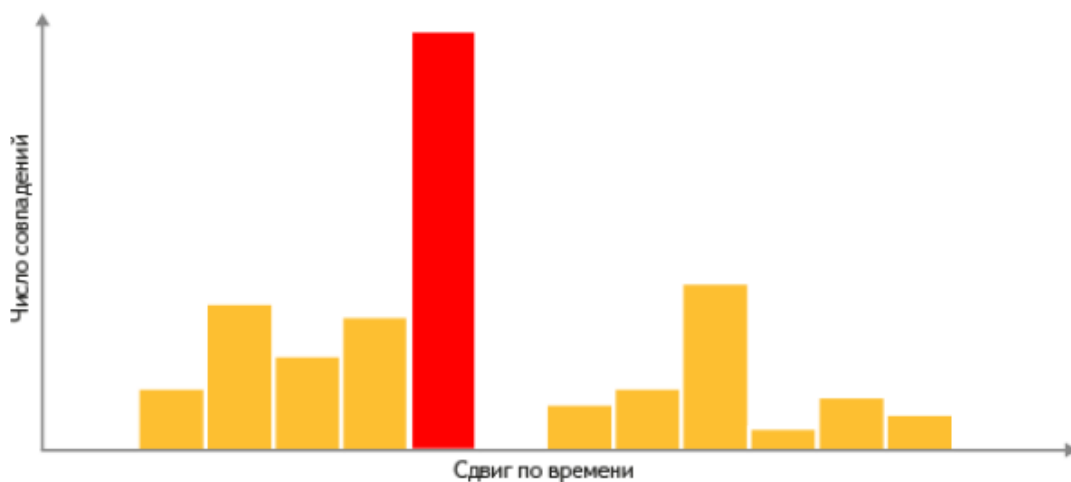


Рисунок 4 Гистограмма отношения числа совпадений к сдвигу по времени

Среди всех композиций выбирается композиция с наибольшим числом совпадений, которые являются мерой близости данного трека с запросом.

Стоит заметить, что среди всех пиков выбираются лишь те, которые являются устойчивыми к искажениям. Алгоритм выбора устойчивых к искажениям пиков:

- в рамках одной частоты вводится порог, который проходит по оси времени. Данный порог при обнаружении пика выше текущего положения срезает разницу между значением порога и верхней точкой. Потом порог поднимается на первоначальную высоту этого пика.
- среди всех частот выбираются пики, обладающие наибольшими локальными максимумами среди вершушек пиков, которые были срезаны на предыдущем шаге.

При выборе пиков учитываются скорость опускания порога, число выбираемых пиков в каждом временном интервале, также важным критерием является окрестность влияния пиков на соседей.

Выбранные критерии позволяют добиться снижения числа пиков до минимального значения.

Большей эффективности, а также скорости работы алгоритма можно добиться с использованием пары пиков. Использование данного подхода обусловлено тем, что при увеличении числа пиков, возрастает также вероятность встретить в исходной песне два оригинальных пика.

Поиск музыкальных композиций в базе данных заключается в получении выборки музыкальных треков, имеющих набор ярко выраженных пиков. После чего происходит определение гарантированно наилучшего ответа.

1.4 Выводы по главе 1

Данная глава включает в себя краткий обзор существующих методов распознавания звуковых сигналов.

Обзоры включают в себя описание основных подходов к формированию признакового описания сигнала, методов классификации, а также описание уже существующих программных продуктов, которые реализуют поиск музыкальных композиций в базах данных. После того, как было описано множество методов, предназначенных для решения поставленной задачи, можно сделать вывод о том, что данная тематика является актуальной.

В процессе обзора существующих методов распознавания сигналов был выявлен ряд проблем, таких как:

- проблема формирования системы признаков, которая обеспечивала бы однозначность классификации и последующего анализа на схожесть отдельных фрагментов сигнала;
- проблема величины входного сигнала. Многие алгоритмы используют для анализа звуковой фрагмент достаточно большой длительности, что не всегда хорошо сказывается на производительности;
- проблема поиска звукового фрагмента в базах данных. Данная проблема заключается в подборе наиболее удобного и быстрого способа поиска схожих с интересующим звуковой фрагментов для получения информации о нем.

Данная дипломная работа предлагает по-новому решить вышеописанные проблемы. Использование Теории активного восприятия для построения системы признаков позволит эффективно использовать признаковое описание сигнала, путем сокращения размерности. Было решено разработать новый алгоритм, который

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						20
Изм	Лист	№ докум.	Подп.	Дата		

позволял бы эффективно хранить звуковые записи в базе данных, содержащий в себе бинарное дерево. С помощью бинарного дерева возможно упорядочить звуковые записи, и позволяет быстро и эффективно найти схожие записи в базе данных, предоставить информацию о них.

Далее в главе 2 присутствует подробное теоретическое описание системы идентификации звукового сигнала в базе данных

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	<i>Лист</i>
						21
<i>Изм</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		

2 Информационная модель поиска звукового сигнала в базе данных

2.1 Теория активного восприятия

Теория активного восприятия была предложена профессором В.А. Утробиним. Ее применение хорошо себя зарекомендовало в сфере распознавания изображений. Кроме того, последнее время данная теория активно используется для анализа звуковых сигналов [10], [11].

Теория активного восприятия предназначена для реализации двух первых этапов распознавания – предварительной обработки сигнала и формирования системы признаков.

Этап предварительной обработки сигнала подразумевает процесс интегрирования. Происходит разбиение сигнала на части, после чего амплитуды отсчетов для каждой части складываются (рисунок 5). Данный этап называется *Q*-преобразованием. Результатом интегрирования является массив *S* - множество сумм отсчетов сегмента звукового сигнала.

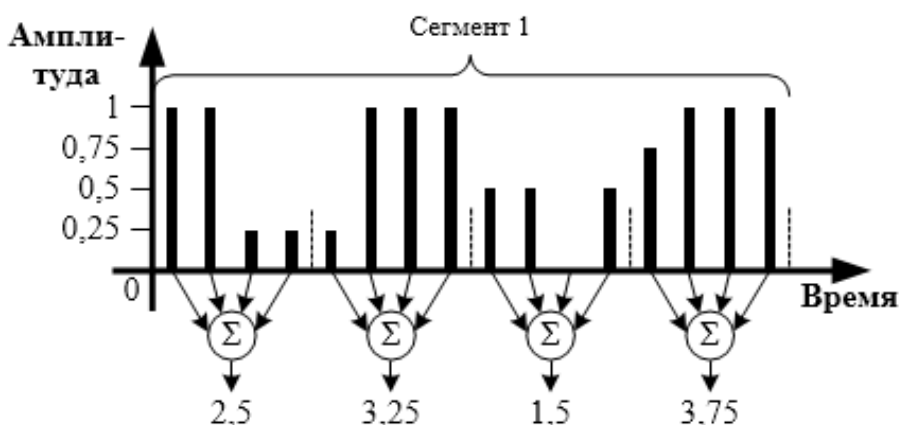


Рисунок 5 *Q*-преобразование

Этап формирования признакового описания подразумевает процесс дифференцирования. На полученный ранее массив сумм *S* накладываются фильтры (рисунок 6), в результате чего происходит вычисление значений спектральных коэффициентов. Для этого необходимо поэлементно умножить каждый фильтр на массив сумм *S*.

Произведенные преобразования определяют *U*-преобразование, которое является главным в теории активного восприятия.

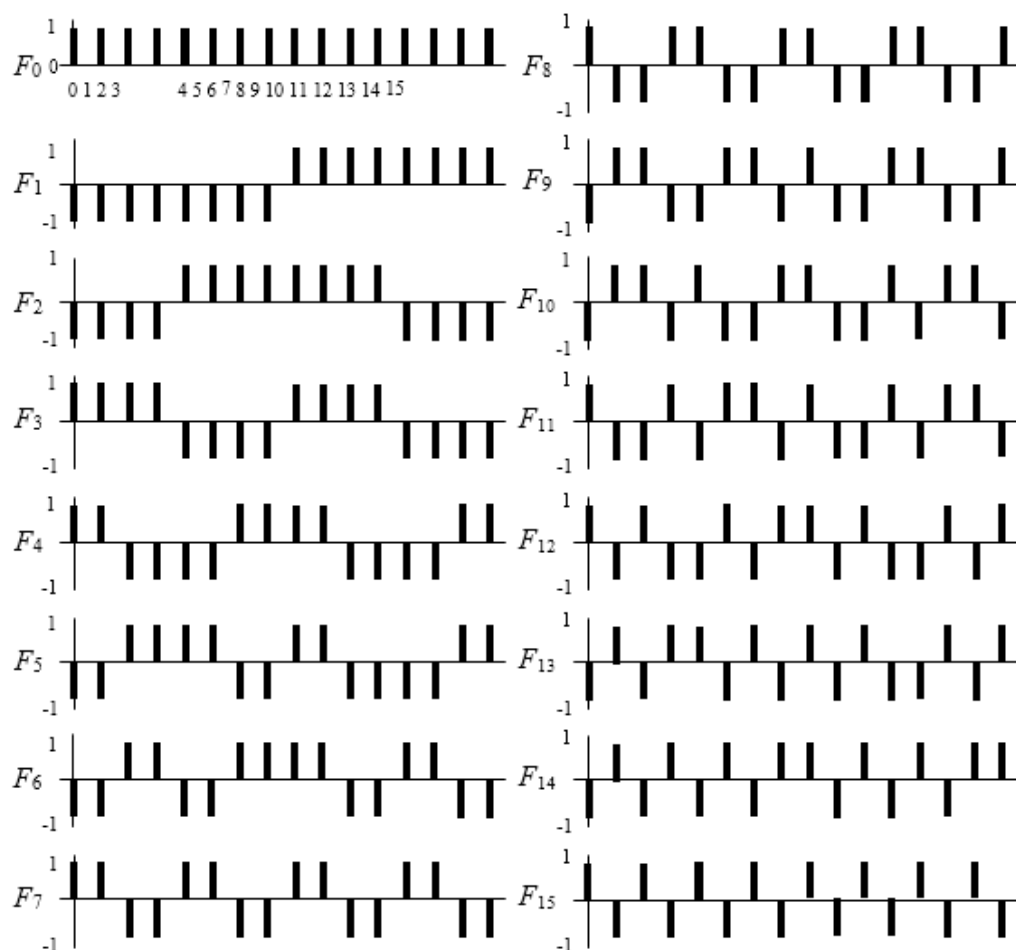


Рисунок 6 Фильтры

Одной из составляющих теории активного восприятия является раздел Алгебра групп. Алгебра групп используется на этапе формирования признакового описания. Позволяет определять зависимости между спектральными коэффициентами, полученными в ходе U -преобразования. Алгебра групп включает в себя полные группы, образованные на тройках операторов, а также замкнутые группы, образованные на четверках операторов.

Различают полные и замкнутые группы как на операцию сложения, так и на операцию умножения. Кроме того, использование инверсных операторов позволяет расширить полные группы до 140, а замкнутые до 840 групп.

Полученный набор признаковых описаний сигнала позволяет добиться большей эффективности при распознавании.

2.2 Система поиска звукового сигнала в базе данных с использованием теории активного восприятия

Система поиска звукового сигнала в базе данных позволяет найти решение двух основных поставленных задач:

- формирование базы данных звуковых сигналов;
- поиск музыкального фрагмента в существующей базе данных.

База данных звуковых сигналов представляет собой бинарное дерево, хранящее в своих узлах признаки описания сегментов сигнала [12].

Задача поиска объекта в базе данных делится на три основных этапа:

- Этап предварительной обработки сигнала;
- Этап формирования системы признаков сигнала;
- Этап принятия решения.

Полученная база данных звуковых сигналов будет использоваться для поиска в ней фрагмента некоторой песни.

Фрагмент звукового сигнала, подающийся на вход алгоритма, подвергается нормализации, а также к нему применяются перечисленные выше этапы обработки и формирования признакового описания. Входной сигнал будет также представлен в виде бинарного дерева. На следующем шаге происходит сравнение признакового описания входного фрагмента и базы данных, анализ результатов и вывод найденной композиции.

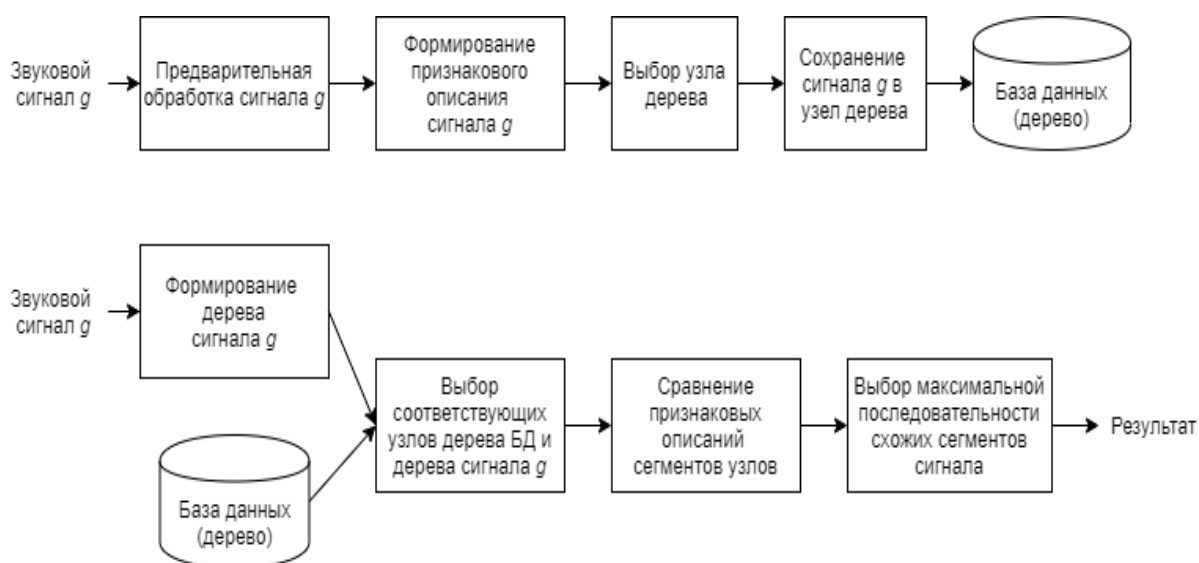


Рисунок 7 Информационная модель системы

2.2.1 Предварительная обработка сигнала

Этап предварительной обработки сигнала включает в себя нормирование звукового сигнала к диапазону $[0, 1]$ и выполнение Q -преобразования. Данное преобразование подразумевает применение к сегментам входного сигнала операции сложения. Нормирование сигнала происходит по следующей формуле:

$$S_{dsc} = S_{dsc} - \min(S_{dsc}), S_{dsc} = \frac{S_{dsc}}{\max(S_{dsc})}, \quad (8)$$

где S_{dsc} – дискретный звуковой сигнал.

Нормированный сигнал представляется в виде множества сегментов s . Каждый сегмент – это последовательность значений из исходного сигнала S :

$$s = \{S_k\}, k = 1: K, \quad (9)$$

где K – число сегментов, длина сегмента составляет M отсчётов, сегменты выбираются из сигнала с равномерным шагом в N отсчётов.

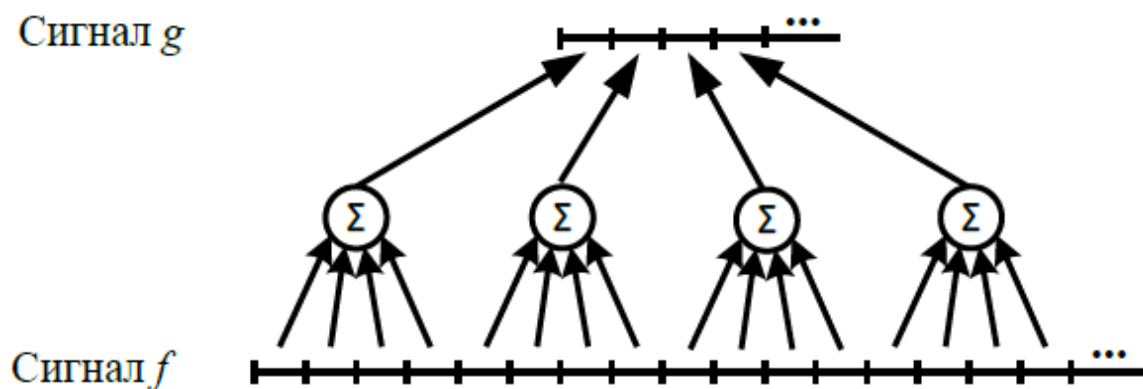


Рисунок 8 Схема предварительной обработки сигнала

2.2.2 Формирование системы признаков

На этапе формирования системы признаков вычисляются признаки сигнала, предварительно обработанного на предыдущем этапе, которые будут использоваться в момент принятия решения при поиске звукового фрагмента в базе данных.

Алгоритм формирования системы признаков состоит из следующих шагов:

- 1) U -преобразование – каждый сигнал подвергается делению на M равных частей, после чего применяется операция суммирования к каждой из частей (результат

Q -преобразования). В результате вычисления U -преобразования формируется спектральное разложение сигнала:

$$q_k[i] = \sum_{l \in T_i} S_k[l], \quad (10)$$

где T_i – индексы отсчётов сегмента S_k , принадлежащие i -ой части сегмента.

2) Применение к полученному вектору множества q_k фильтров Уолша системы Хармута, в результате чего формируется вектор спектральных коэффициентов μ_k :

$$\mu_k[i] = \sum_{j=0}^{M-1} q_k[j] F_i[j], \quad i = 1, M, k = 1, K, \quad (11)$$

где M – число используемых фильтров Уолша системы Хармута.

Одной из важных частей теории активного восприятия является алгебра групп. В данном разделе описаны зависимости между спектральными коэффициентами. Использование подобных зависимостей позволяет добиться большей эффективности распознавания на этапе поиска звукового сигнала в базе данных.

В соответствие каждому из фильтров можно поставить бинарный оператор V_i . Операторы распределяются таким образом, что в случае, когда спектральный коэффициент является положительным, либо отрицательным, то ему в соответствие ставится прямой или инверсный оператор. В случае нулевого значения спектрального коэффициента бинарный оператор не ставится.

Совокупность полученных операторов может быть представлена в виде полных или же замкнутых групп, которые образуются по тройке и четверке бинарных операторов соответственно.

Полные группы определяют связь между бинарными операторами V_i . Замкнутые группы в свою очередь определяют связи между полными группами.

Для объединения данных, полученных от разных сегментов анализируемого сигнала, вычисляется гистограмма элементов структур, использованных при создании описания сегмента сигнала. Вариантами данных гистограмм являются одномерная, двумерная, а также трехмерная гистограммы.

Таким образом результатом этапа формирования системы признаков является массив признаков, полученный с помощью операции U -преобразования, а также с помощью вычисления замкнутых групп на основе U -преобразования.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						26
Изм	Лист	№ докум.	Подп.	Дата		

2.2.3 Размещение сегментов сигнала в бинарном дереве

В качестве средства для хранения признакового описания сигнала используется бинарное дерево. На самый верхний уровень бинарного дерева помещается исходный сегмент сигнала. Процесс размещение сегментов узлах дерева является рекурсивным:

1) Инициализируется N -уровневое двоичное дерево, в качестве текущего уровня анализа устанавливается первый:

$$i = 1, (i \leq N), \quad (12)$$

где i – текущий уровень дерева, N – число уровней дерева.

2) Вычисляется U -преобразование сегмента сигнала S_k , результатом вычисления является вектор спектральных коэффициентов:

$$\mu'_k = \text{sgn}(\mu_k), \quad (13)$$

где μ_k – результат вычисления U -преобразования сегмента сигнала S_k .

3) Вычисляется расстояние Хэмминга между μ'_k и бинарными векторами [13]:

$$d_1 = \text{DIST}[\mu'_k, T_1],$$

$$d_2 = \text{DIST}[\mu'_k, T_2], \quad (14)$$

где $\text{DIST} []$ – оператор вычисления расстояния Хэмминга, T_1 и T_2 – бинарные вектора: $T_1 = \{1,1,1,1,1,1,1,0,0,0,0,0,0\}$, $T_2 = \{0,0,0,0,0,0,0,1,1,1,1,1,1\}$.

Отношение d_1 к d_2 определяет половину анализируемого сегмента, которая будет использоваться для дальнейшего анализа. При $d_1 < d_2$ выбирается левая половина сегмента, информация о сегменте помещается в левый дочерний узел текущей вершины, в обратном случае - для анализа используется правая часть сегмента и информация помещается в правый дочерний узел.

Чтобы произвести сравнение признакового описания сегмента с шаблонами, используемыми для вычисления расстояния Хэмминга, необходимо преобразовать все положительные значения коэффициентов признакового описания к значению равному 1, а все отрицательные к 0.

4) Переход на следующий уровень анализа исходного сегмента S_k . В качестве рассматриваемого сегмента S_k выбирается его левая или правая часть в зависимости от решения на предыдущем шаге. Увеличивается на один текущий уровень дерева и выполняется переход к шагу 2. В случае если $i > N$, сегмент считается обработанным.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						27
Изм	Лист	№ докум.	Подп.	Дата		

Работа алгоритма выполняется до тех пор, пока информация о всех сегментах исходного сигнала не будет помещена в двоичное дерево.

В качестве результата работы алгоритма выступает бинарное дерево. В каждом узле данного дерева располагается массив сегментов, содержащий признаковое описание сигнала (рисунок 9).

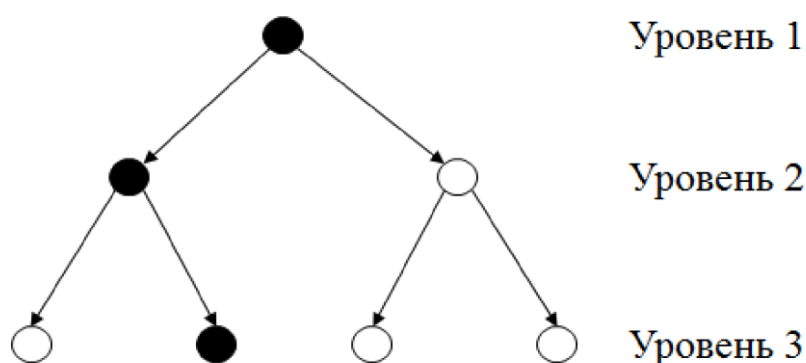


Рисунок 9 Размещение сегментов сигнала в узлах N-уровневого бинарного дерева

2.2.4 Поиск звукового сигнала в базе данных

Для того, чтобы произвести поиск некоторого музыкального фрагмента в базе данных звуковых сигналов необходимо сформировать бинарное дерево запроса, по алгоритму описанному ранее, и сравнить его с деревом этой базы данных. В результате сравнения признаковых описаний сегментов будет получен массив сегментов, на основании которого можно определить существует ли входной фрагмент в базе данных, или же нет.

Запрашиваемый сигнал подвергается предварительной обработке, формируется набор признаков по алгоритму, описанному выше, после чего происходит построение бинарного дерева запроса. Важным аспектом при построении дерева запроса является число уровней, на которые будет разбиваться данное дерево. Оно должно быть одинаковым с числом уровней дерева звуковой базы данных.

После того, как было построено бинарное дерево запроса, необходимо произвести сравнение искомого сигнала с сигналами из базы данных.

Процесс сравнения признаковых описаний входного сигнала и звуковой базы данных начинается с самого нижнего уровня дерева. После чего происходит, собственно,

сам процесс сравнения. Ранее было сказано, что число уровней дерева запроса и дерева звуковой базы данных должно совпадать. Это необходимо для того, чтобы сравнение признаков происходило в соответствующих узлах деревьев (рисунок 10).

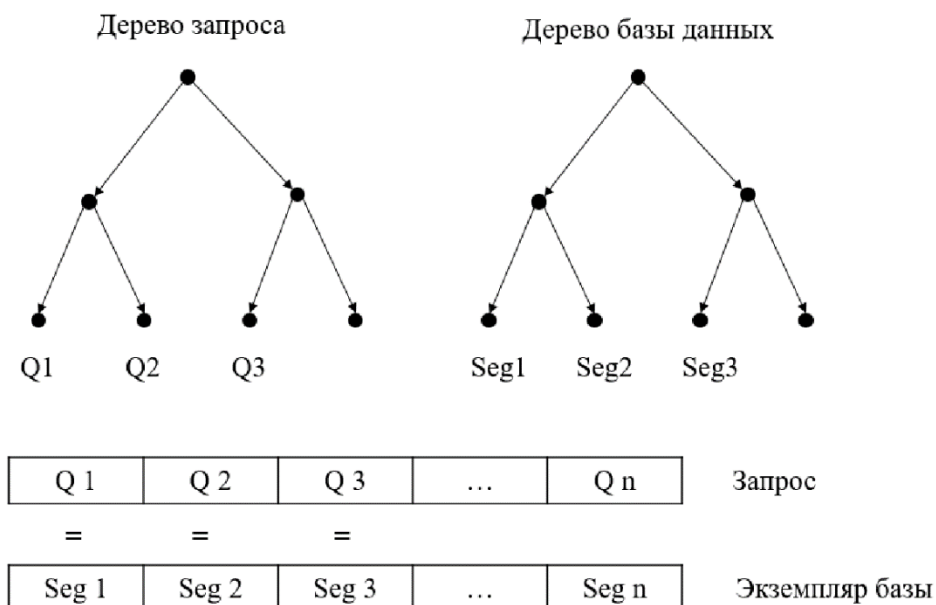


Рисунок 10 Сравнение сегментов деревьев сигнала запроса и звуковой базы данных

При обнаружении признаков схожих между собой происходит переход к соседнему элементу, а также их сравнение. Процесс завершится в случае нахождения полной последовательности схожих сегментов.

В результате произведенного сравнения сегментов будет получен массив сегментов, который необходимо отсортировать. На этапе вычисления признакового описания каждой композиции был присвоен номер, также были пронумерованы сегменты в рамках одной композиции. Полученная после сортировки максимальная последовательность сегментов является результатом поиска.

2.3 Выводы по главе 2

Данная глава содержит в себе теоретическое описание системы идентификации звукового сигнала в базе данных, а также подробное описание способа реализации всех этапов идентификации звукового сигнала в базе данных.

Ниже приведено краткое описание вариантов решения поставленных задач:

- Использование теории активного восприятия для формирования признакового описания сегментов сигнала;
- N -уровневого бинарное дерево в качестве метода хранения музыкальных композиций;
- Сформулирован и описан алгоритм поиска схожих сигналов в звуковой базе данных - бинарном дереве.

Подход к решению задач каждой из задач распознавания звукового сигнала, описанный в данной главе, свидетельствуют о научной новизне данного исследования. Далее глава 3 содержат в себе вычислительный эксперимент, анализ полученных данных, а также сравнение с уже существующими алгоритмами. На основании результатов экспериментов можно доказать, что разработанный метод является работоспособным.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						30
Изм	Лист	№ докум.	Подп.	Дата		

3 Вычислительный эксперимент

В рамках научного исследования был проведен вычислительный эксперимент. Были получены результаты, которые необходимо проанализировать. На основании анализа результатов можно сделать вывод о том, насколько эффективно представленный алгоритм справляется с поставленными задачами, а также произвести сравнение с уже существующими методами.

В данной главе представлена информация о проводимом вычислительном эксперименте и проведен анализ полученных результатов.

3.1 Описание эксперимента

Вычислительный эксперимент представляет собой тестирование алгоритма поиска звукового сигнала в базе данных на предмет эффективности, а также будет оцениваться устойчивость алгоритма к внешним искажениям.

Звуковая база данных представлена 1000 музыкальных композиций различных жанров. Каждая композиция представлена в формате MP3, а также записана в студийных условиях. В базе данных представлены музыкальные композиции длиной от 2,5 до 4,5 минут.

На вход алгоритма будут подаваться 1000 различных музыкальных композиций различных жанров записанные в MP3 формате. Испытания будут проводиться в нормальных условиях, то с использованием композиций, записанных в студийных условиях, а также с использованием наложений искажений.

Тестирование работы алгоритма в нормальных условиях обусловлено отсутствием шума, а все входные звукозаписи являются студийными. На данном этапе будут использоваться следующие изменяемые входные параметры:

1. Длительность входного сигнала.

В качестве длительности входного сигнала было принято использовать следующие значения – 1 сек., 2 сек., 4 сек. Небольшая длительность входного сигнала обусловлена скоростью работы алгоритма.

2. Величина сегмента сигнала, которая будет использоваться для дробления уже имеющихся звукозаписей в базе данных. Данная величина соответствует величине

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						31
Изм	Лист	№ докум.	Подп.	Дата		

длительности входного сигнала – 1 сек., 2 сек., 4 сек.

3. Число уровней дерева. Позволяет менять число уровней дерева признаков описаний, которое будет использоваться для хранения признаков описаний базы данных музыкальных композиций, а также входных музыкальных фрагментов. Большее число уровней дерева позволяет более детально хранить признаки описания музыкальных композиций. В качестве значений данного параметра будут использоваться - 3, 5 и 7 уровней разложения.

4. Величина смещения сегментов разложении сигнала. Будет использоваться смещение в 10, 100 и 1000 отсчетов.

В результате проведения эксперимента была получена наилучшая конфигурация параметров, при которых разработанная система показывает наибольшую эффективность. Данная конфигурация параметров также использовалась для анализа устойчивости системы к внешним шумам.

В качестве искажений будет использоваться шум величиной – 0 дБ, 5 дБ, 15 дБ.

В результате проведения вычислительного эксперимента будут анализироваться:

1. Точность распознавания сигнала запроса.
2. Время, потраченное на поиск одного звукового запроса.
3. Время формирования звуковой базы.

3.2 Описание программного продукта

В ходе выполнения исследовательской работы был разработан программный продукт для тестирования описанного алгоритма (см. Приложение А). В качестве языка программирования использовался язык *R*. Язык *R* чаще всего применяют в работах, связанных с обработкой данных, графическом отображении этих данных. Кроме того, данный язык программирования обладает большим набором библиотек, является простым в использовании и эффективным для выполнения поставленных задач.

Далее будут подробно описаны программные модули, которые реализуют функционал системы.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						32
Изм	Лист	№ докум.	Подп.	Дата		

Таблица 2 Описание функции startTest.

startTest	
Параметр	Описание
signalOffset	Величина смещения сегментов разложения сигнала
levelCount	Число уровней дерева
signalLength	Длина входного сигнала / сигналов из базы данных

startTest – функция предназначена для запуска тестирования. Принимает на вход конфигурацию параметров, которые будут меняться в ходе эксперимента. Данная функция отвечает также за вывод конечного результата тестирования.

Таблица 3 Описание функции createBaseTree.

createBaseTree	
Параметр	Описание
signalOffset	Величина смещения сегментов разложения сигнала
levelCount	Число уровней дерева
signalLength	Длина сигнала базы данных

createBaseTree – функция предназначена для создания бинарного дерева, содержащего признаки описания базы данных звуковых сигналов. Происходит обход все существующих музыкальных композиций из базы данных, формирование признакового описания по каждой из них, а также добавление в узлы дерева. Данная функция позволяет сохранить дерево базы данных в .Rdata файл для дальнейшей загрузки.

Таблица 4 Описание функции addNode.

addNode	
Параметр	Описание
currentNode	Текущий исследуемый узел дерева
levelCount	Число уровней дерева
signalsCount	Число музыкальных композиций в базе данных
searchIndex	Индекс сегмента сигнала

addNode – функция предназначена для добавления новых узлов в текущее дерево запроса или базы данных. Позволяет определить с помощью расстояния Хэмминга в каком из узлов, левом или правом, будет находиться признаковое описание. Данная функция вызывается рекурсивно пока бинарное дерево не будет полностью сформировано.

Таблица 5 Описание функции hammingDistance.

hammingDistance	
Параметр	Описание
inputVector	Вектор коэффициентов признакового описания

hammingDistance – функция предназначена для вычисления расстояния Хэмминга. Происходит преобразование коэффициентов к формату 1 или 0, а также сравнение с существующими шаблонами.

Таблица 6 Описание функции processSignal.

processSignal	
Параметр	Описание
inputSignal	Входной сигнал
signalLength	Длина входного сигнала

processSignal – функция предназначена для обработки входного сигнала, его

нормализации, обозначения фрагмента сигнала, заданной длины, а также для наложения шума при необходимости.

Таблица 7 Описание функции createSignalTree.

createSignalTree	
Параметр	Описание
signalOffset	Величина смещения сегментов разложения сигнала
levelCount	Число уровней дерева
signalLength	Длина входного сигнала

createSignalTree – функция предназначена для формирования бинарного дерева, содержащего признаковые описания входного сигнала.

Таблица 8 Описание функции searchSignal.

searchSignal	
Параметр	Описание
baseNode	Узел дерева базы данных
inputNode	Узел дерева входного сигнала
resultValue	Результат поиска

searchSignal – функция предназначена для поиска схожих сегментов в дереве запроса и дереве базы данных. Происходит сравнение признаковых описаний каждого сегмента из дерева запроса с каждым сегментом из дерева базы соответственно.

3.3 Анализ результатов

Изначально тестирование программного продукта производится в нормальных условиях, то есть используются звукозаписи студийного происхождения без наложения дополнительных шумов. Далее в таблице 9 приведены результаты работы системы в нормальных условиях.

Таблица 9 Результат работы алгоритма поиска звукового сигнала в базе данных в
нормальных условиях

Число уровней дерева	Длительность входного сигнала, с	Величина сегмента сигнала, с	Величина смещения сегментов	Точность поиска, %	Время поиска, с	Время формирования базы данных, с
3	1	1	1000	10,5	0,15	4,1
			100	22,3	0,29	
			10	39,5	0,34	
	2	1	1000	8,0	0,36	4,7
			100	15,3	0,41	
			10	16,1	0,44	
		2	1000	11,3	0,27	4,1
			100	30,8	0,25	
			10	44,1	0,32	
		4	1	1000	8,5	5,2
				100	9,3	
				10	13,1	
			2	1000	9,1	3,1
				100	14,4	
				10	25,7	
			4	1000	13,2	2,8
				100	27,7	
				10	43,8	

Таблица 9 Результат работы алгоритма поиска звукового сигнала в базе данных в нормальных условиях

Число уровней дерева	Длительность входного сигнала, с	Величина сегмента сигнала, с	Величина смещения сегментов	Точность поиска, %	Время поиска, с	Время формирования базы данных, с
5	1	1	1000	25,3	0,24	6,1
			100	47,8	0,29	
			10	61,5	0,33	
	2	1	1000	10,2	0,37	5,9
			100	24,9	0,39	
			10	30,4	0,42	
		2	1000	19,7	0,3	6,8
			100	50,1	0,34	
			10	64,3	0,37	
	4	1	1000	15,5	0,29	5,4
			100	21,4	0,3	
			10	34,8	0,36	
		2	1000	27,9	0,25	6,6
			100	35,8	0,3	
			10	51,2	0,34	
		4	1000	37,6	0,19	7,0
			100	61,4	0,24	
			10	81,3	0,31	

Таблица 9 Результат работы алгоритма поиска звукового сигнала в базе данных в нормальных условиях

Число уровней дерева	Длительность входного сигнала, с	Величина сегмента сигнала, с	Величина смещения сегментов	Точность поиска, %	Время поиска, с	Время формирования базы данных, с
7	1	1	1000	52,4	0,68	8,7
			100	68,6	0,7	
			10	86,5	0,77	
	2	1	1000	49,2	0,66	9,0
			100	64,3	0,69	
			10	80,5	0,7	
		2	1000	68,8	0,54	8,5
			100	81,4	0,6	
			10	92,7	0,63	
	4	1	1000	36,7	0,72	9,5
			100	43,6	0,77	
			10	64	0,75	
		2	1000	51,4	0,61	8,9
			100	63,5	0,66	
			10	81,1	0,69	
		4	1000	74,9	0,57	8,2
			100	85,5	0,59	
			10	94,3	0,63	

На основании данных, полученных в ходе вычислительного эксперимента, можно сделать выводы, что наилучший результат работы системы достигается при конфигурации системы со следующими параметрами:

- 1) Число уровней дерева – 7.
- 2) Длительность входного сигнала – 4 с.
- 3) Величина сегмента сигнала в базе данных – 4 с.

4) Величина смещения сегментов – 10 отсчетов.

Подобные результаты обусловлены в первую очередь количеством уровней дерева запроса, на которые разбивается входной сигнал. Происходит более детальное анализирование входного сигнала, которые также имеет большую длительность.

Кроме того, высокий процент точность распознавания достигается за счет увеличения времени поиска запроса, а также времени формирования базы данных.

Возможный вариант решения проблемы длительного формирования признакового описания базы данных заключается в формировании данной базы единожды и загрузки ее каждый раз, когда происходит процесс распознавания. При обновлении базы музыкальных композиций, признаковые описания будут пере вычисляться.

Для сравнения полученных результатов было выбрано три описанных в литературе подхода, предложенные следующими авторами:

- Подход с использованием спектрограмм и нейронных сетей, описанный в работе [17] (Wang A.)
- Подход с использованием скрытых марковских моделей и гауссовых смесей, описанный в работе [18] (Bello Dufaux A., Besacier L., Ansorge M.)
- Подход с использованием быстрого преобразования Фурье, а также хэш-таблицы [19] (Heinrich A. van Nieuwenhuizen, Willie C.)

В таблице 10 представлено сравнение разработанного алгоритма с уже существующими методами

Таблица 10 Сравнение с существующими методами

	Длительность сигнала	Время поиска, с	Время построения базы данных, с	Точность распознавания, %
Wang A.	5	0,59	-	91,4
Bello Dufaux A.	-	0,75	-	89,76
Heinrich A. van Nieuwenhuizen	3	0,71	255	96,35
Разработанный алгоритм	4	0,63	8,2	94,3

Из полученных данных можно сделать вывод, что разработанный метода является конкурентоспособным с существующими методами.

Для проведения вычислительного эксперимента с наложением шума использовалась наилучшая конфигурация параметров, полученная на предыдущем шаге. Количество уровней дерева – 5 и 7 уровней, длительность входного сигнала совпадает с величиной сегментов в базе данных, смещение сегментов – 10 отсчетов. Далее в таблице 11 приведены результаты работы системы в условиях наложения шума.

Таблица 11 Результат работы алгоритма поиска звукового сигнала в базе данных в условиях наложения шума

Число уровней дерева	Длительность входного сигнала, с	Уровень шума, дБ	Точность поиска, %
5	1	0	61,5
		5	44,9
		15	37,2
	2	0	64,3
		5	48,1
		15	40,7
	4	0	81,3
		5	67,9
		15	50,2
7	1	0	86,5
		5	67,2
		15	49,8
	2	0	92,7
		5	81,3
		15	75,5
	4	0	94,3
		5	91,4
		15	87,9

На основании полученных результатов, можно сделать вывод о том, что разработанный алгоритм является устойчивым к помехам. Наилучший результат распознавания по всем 3 случаям, при шуме 0 дБ, 5 дБ, 15 дБ, достигается при числе уровней дерева равному 7, а также длительности входного сигнала равной 4 секундам.

В таблице 12 представлено сравнение разработанного алгоритма с программным обеспечением *Shazam* и *Яндекс* при использовании одинаковых уровней шума.

Таблица 12 Результаты сравнение точностей распознавания в условиях наложения шума для разработанного алгоритма, алгоритма *Shazam* и алгоритма *Яндекс*

Уровень шума, дБ	Алгоритм <i>Shazam</i> , %	Алгоритм <i>Яндекс</i> , %	Разработанный алгоритм, %
0	98,7	96	94,3
5	95	91,2	91,4
15	87,5	85,3	87,9

Результаты, полученные в таблицах 9, 10, 11 и 12 говорят о том, что разработанный алгоритм поиска звуковых сигналов в базах данных является эффективным и полностью справляется с поставленной задачей.

3.4 Выводы по главе 3

Данная глава содержит в себе описание вычислительного эксперимента по исследованию системы поиска звукового сигнала в базе данных, анализ полученных результатов и краткое описание программного продукта, используемого для тестирования разработанного алгоритма.

Эксперимент проводился в 2 этапа: в нормальных условиях и в условиях наложения шума на музыкальную композицию.

В ходе проведения эксперимента были определены наилучшие конфигурации системы, при которых алгоритм показывает наибольшую точность распознавания. Данные конфигурации в последствии использовались в эксперименте с наложением шума на музыкальную композицию.

Полученные результаты были сравнены с результатами работы известных

алгоритмов. Результаты сравнения алгоритмов говорят о том, что разработанный метод имеет достаточно высокую точность распознавания, которая является схожей с результатами распознавания существующих методов. Кроме того, данный метод является устойчивым к внешним воздействиям. На основании этого можно сделать вывод о конкурентоспособности разработанного метода.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	<i>Лист</i>
						42
<i>Изм</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подп.</i>	<i>Дата</i>		

Заключение

В результате выполнения выпускной квалификационной работы был рассмотрен подход к построению модели и алгоритма поиска звукового сигнала в базе данных.

В качестве практического применения данная разработка может быть использована в музыкальной индустрии для выявления плагиата, а также в области голосовой биометрии, для удаленной идентификации пользователей.

Разработанный метод опирается на теорию активного восприятия на этапе вычисления признакового описания входного сигнала, а также звуковой базы данных. Для этого вычисления используются U -преобразования из теории активного восприятия совместно с алгеброй групп.

Для хранения полученных данных использовалось бинарное дерево, которое представляет собой упорядоченную структуру с признаковыми описаниями сигналов в каждом из своих узлов. Данный подход обеспечивает сравнительно быстрый доступ к каждому из элементов дерева и позволяет добиться высокой скорости поиска нужного элемента в базе данных.

Для подтверждения работоспособности и эффективности разработанного метода был написан программный продукт на языке R и проведен ряд испытаний.

В результате вычислительного эксперимента была получена наилучшая конфигурация работы системы поиска звукового сигнала в базе данных для нормальных условия, а также в случае зашумленного сигнала. Также было произведено сравнение результатов распознавания разработанного метода с результатами уже существующих алгоритмов. Полученная точность распознавания и результаты сравнения говорят о корректной работе разработанного алгоритма, а также о его конкурентоспособности.

На основании вышесказанного можно сделать вывод, что поставленные задачи в рамках научного исследования были выполнены в полной мере, а цель работы достигнута.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						43
Изм	Лист	№ докум.	Подп.	Дата		

Список литературы

1. Сорокин В. Н., Цыплихин А. И. Верификация диктора по спектрально-временным параметрам речевого сигнала // Информационные процессы, Том 10, №2, стр. 87 – 104
2. Маркина Ю. Ю., Белов Ю. С. Кепстральные коэффициенты как необходимая характеристика процесса создания системы имитации голоса человека с помощью методов глубокого обучения // Московский государственный технический университет имени Н. Э. Баумана
3. Jurafsky D, Martin J. H. Hidden Markov Models // MIT Lincoln Laboratory Lexington, Massachusetts, USA Johns Hopkins University Baltimore, Maryland, USA
4. Jansen A., Church K., Hermansky H. Towards Spoken Term Discovery At Scale With Zero Resources // Human Language Technology Center of Excellence, Department of Electrical and Computer Engineering, Department of Computer Science Johns Hopkins University, Baltimore, Maryland - INTERSPEECH 2010, 1676-1679 с.
5. Классификация данных методом опорных векторов – Habrahub – URL: <https://habr.com/ru/post/105220/>
6. Пекарь Д. В., Тихоненко С. Г. Алгоритм использования гауссовых смесей для идентификации диктора по голосу в технических системах // Н.Г. Загоруйко. – М.: Советское радио, 1972. – 208 с.
7. Юрков К. В. Векторное квантование на основе кодов, исправляющих ошибки // Вестник Санкт-Петербургского политехнического института – 2011. - №6. – С. 194 -201
8. Shazam: алгоритмы распознавания музыки, сигнатуры, обработка данных – Habrahub – URL: <https://habrahub.ru/company/wunderfund/blog/275043/>
9. Как Яндекс распознаёт музыку с микрофона – Habrahub – URL: <https://habrahub.ru/company/yandex/blog/181219/>
10. Утробин В.А. Информационные модели системы зрительного восприятия для задач компьютерной обработки изображений: учеб. пособие / В.А. Утробин. – Нижний Новгород: НГТУ им. Р. Е. Алексеева, 2001. – 234 с.
11. Утробин В.А. Элементы теории активного восприятия изображений / В.А. Утробин // Труды Нижегородского государственного технического университета им. Р.Е. Алексеева. – 2010. – Т. 81. – № 2. – С. 61-69.

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						44
Изм	Лист	№ докум.	Подп.	Дата		

12. Гай В. Е. Информационный подход к описанию звукового сигнала // Труды МФТИ. – 2014. – Том 6, №2. – С. 167- 171.

13. Расстояние Хэмминга – Википедия – URL: https://ru.wikipedia.org/wiki/Расстояние_Хэмминга

14. Гай В.Е. Метод обработки биометрической информации в системах контроля и управления доступом // Труды Нижегородского государственного технического университет им. Р.Е. Алексеева: НГТУ им. Р.Е. Алексеева. – Нижний Новгород, 2013, № 8 (94). – С. 65-32.

15. Гай В.Е. Метод оценки частоты основного тона в условиях помех // Цифровая обработка сигналов, № 4, 2013, С. 65-71

16. Гай В.Е. Идентификация диктора по спектральным характеристикам речевого сигнала // Датчики и системы, № 7, 2013, С. 40-45

17. Wang A. An Industrial-Strength Audio Search Algorithm // Proc. 2003 ISMIR International Symposium on Music Information Retrieval, Baltimore, MD, Oct. 2003.

18. Dufaux A., Besacier L., Ansorge M. Automatic sound detection and recognition for noisy environment // Institute of Microtechnology, Breguet, Switzerland, 2000.

19. Heinrich A. van Nieuwenhuizen, Willie C. Venter The study and implementation of Shazam's audio fingerprinting algorithm for advertisement identification // School of Electrical, Electronic and Computer Engineering North-West University, Potchefstroom Campus, South Africa, 2013

					ВКР-НГТУ-09.04.01-(М18-ИВТ-3)-019-2020 (ПЗ)	Лист
						45
Изм	Лист	№ докум.	Подп.	Дата		

ПРИЛОЖЕНИЕ А

```
#startTest.r
```

```
load('fold.rdata')
load('db.RData')
source('addNode.R')
source('createBaseTree.R')
source('searchSignal.R')
source('processSignal.R')
source('createSignalTree.R')
```

```
startTest = function( signalOffset,
levelCount, signalLength)
{
  segmentSize <- 2^17
  files <-
list.files("C:/Users/sekosh/Documents/diss_s
harutin/second", pattern="*.mp3",
full.names=TRUE)
  signalList <- processSignal (files,
signalLength)
  successCount <- 0
  start.time <- Sys.time()
  signalTree <- 'createSignalTree
(signalList [[0]],segmentSize, signalOffset,
levelCount-1)
```

```
  results <- list()
  results <- 'searchSignal (baseTree,
signalTree, results)
```

```
successCount <- sortResults(results)
```

```
  end.time <- Sys.time()
  time.taken <- end.time - start.time
  print(paste('Success: ', successCount *
100), "% ,search time:",
as.numeric(time.taken, units='secs'))
}
```

```
#createBaseTree.r
```

```
library(tuneR)
library(data.tree)
source('addNode.R')
```

```
createBaseTree = function(signalOffset,
levelCount)
{
  segmentSize <- 2^17
  start.time <- Sys.time()
  filenames <-
list.files("C:/Users/sekosh/Documents/diss_s
harutin/first", pattern="*.mp3",
full.names=TRUE)
  base <- list()
  signalsList <- list()
  for(i in 1:length(filenames))
  {
    base[[i]] <- readMP3(filenames[[i]])
    signalsList [[i]] <-
base[[i]]@left[(2^15*47):(2^15*98)]
```

```
  num <- length(signalsList)
  signals <- list()
```

```
  for(i in 1:num)
  {
    currentSignal <- signalsList[[i]]
    count <-
length(currentSignal)%/%segmentSize
```

```
    start <- 1
    finish <- segmentSize
    segNumber <- 1
    for(j in 1:count)
    {
      u <-
getUTrans(currentSignal[start:finish],
segmentSize, segmentSize, flt)
      feat <- getfeatures(cls1d, u, fullGrp,
clsGrp, oper, gfullm2d, goper2d, goper3d,
gfulls2d, gpfullm2d, gpfulls2d)
```

```
      currentList <-
list(currentSignal[start:finish], u[[1]][[1]],
feat [[1]][[2]], filenames[[i]], "1",
segNumber)
      signals[[length(signals) + 1]] <-
currentList
      start <- start + segmentSize * offsetDB
      finish <- finish + segmentSize *
signalOffset
      segNumber <- segNumber + 1
    }
  }
```

					БКР-НГТУ-09.04.01-(М18-ИБТ-3)-019-2020 (ПЗ)	Лист
						46
Изм	Лист	№ докум.	Подп.	Дата		

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ А

```
tree <- Node$new("1", value = signals)
addNode(tree, levelCount, filenames, 1)
end.time <- Sys.time()
time.taken <- end.time - start.time
print(paste("Creating DB time:",
as.numeric(time.taken, units='secs')))
save(tree,file=paste("db", levelCount,
signalOffset, segmentSize, "RData",
sep="."))
return(tree)
}
```

```
#hammingDistance.r
library(e1071)
```

```
hammingDistance = function (inputVector)
{
  resultVector <- list()
  coef <- inputVector [[1]][1:35]

  for(i in 1:length(u))
  {
    if(coef [[i]]>0)
    {
      resultVector[[i]] <- 1
    }
    else
    {
      resultVector[[i]] <- 0
    }
  }

  template1 <- c(1,1,1,1,1,1,1,1,0,0,0,0,0,0)
  template2 <- c(0,0,0,0,0,0,0,0,1,1,1,1,1,1)

  dist1 <- hamming.distance(resultVector,
template1)
  dist2 <- hamming.distance(resultVector,
template2)

  if(dist1 <= dist2)
  {
    if(dist1 <= 8)
    {
      return(1)
    }
    else
    {

```

```
return(0)
  }
  }
  else if(dist2 <= dist1)
  {
    if(dist2 <= 8)
    {
      return(2)
    }
    else
    {
      return(0)
    }
  }
}
```

```
#processSignal.r
```

```
processSignal = function(inputSignal,
inputLength)
{
  base <- list()
  signalsList <- list()
  startNum <- 50*2^15
  stopNum <- (50*2^15) + inputLength
*2^15

  for(i in 1:length(inputSignal))
  {
    print(inputSignal [[i]])
    base[[i]] <- readMP3(inputSignal [[i]])
    print(base[i])
    signalsList [[i]] <- norm(signalsList [[i]])
    base[[i]]@left[startNum:stopNum]
    signalsList [[i]] <-
      return(signalsList)
  }
}
```

```
#createSignalTree.r
```

```
createSignalTree = function(signal,
signalOffset, levelCount)
{
  newList <- list()
  start <- 1
  finish <- 2^17
}
```

					БКР-НГТУ-09.04.01-(М18-ИБТ-3)-019-2020 (ПЗ)	Лист
						47
Изм	Лист	№ докум.	Подп.	Дата		

ПРОДОЛЖЕНИЕ ПРИЛОЖЕНИЯ А

```
while(finish <= length(signal))
{
  u <- getUTrans(signal[start:finish],
segmentSize, segmentSize, flt)
feat <- getfeatures(cls1d, u, fullGrp, clsGrp,
oper, gfullm2d, goper2d, goper3d, gfulls2d,
gpfullm2d, gpfulls2d)
  newList[[length(newList) + 1]] <-
list(signal[start:finish], u[[1]][[1]],
feat[[1]][[2]], "Unknown.mp3", "1")
  start <- start + offset
  finish <- finish + offset
}
```

```
newTree <- Node$new("1", value =
newList)
addNode(newTree, levelsCount)
return(newTree)
}
```

#searchSignal.r

```
findSignal = function(baseNode, inputNode,
resultValue)
{
  children <- node$children
  targetChidren <- inputNode $children
  if (!is.null(children) &&
!is.null(inputNode))
  {
    for(i in 1:length(inputNode))
    {
      child <- children[[i]]
      targetChild <- inputNode [[i]]
      results <- findSignal(child, inputNode,
resultValue)
    }
  } else {
    value <- baseNode$value
    targetValue <- inputNode $value
    if(length(value) > 0 &&
length(targetValue) > 0)
    {
      for(j in 1:length(value))
      {
        for(i in 1:length(targetValue))
        {
```

```
if(search(targetValue[[i]][[2]][1:35],
value[[j]][[2]][1:35]))
{
  k <- length(resultValue) + 1
  resultValue [[k]] <- value[[j]]
  break
}
}
}
}
}
return(resultValue)
}
```

					БКР-НГТУ-09.04.01-(М18-ИБТ-3)-019-2020 (ПЗ)	Лист
						48
Изм	Лист	№ докум.	Подп.	Дата		