Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Факультет математики и информатики
Кафедра информационных и управляющих систем
Направление подготовки /специальность 09.04.04 – Программная инженерия
Направленность (профиль) образовательной программы Управление разработкой программного обеспечения

ДОПУСТИТЬ К ЗАЩИТЕ
Зав. кафедрой
_____А.В. Бушманов
«_____»_____2023 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

на тему: Разработка модуля голосовой идентификации пользователя

Исполнитель
студент группы 157-ом _____ Г.М. Рзаева
(подпись, дата)

Руководитель
доцент, канд. техн. наук _____ С.Г. Самохвалова
(подпись, дата)

Руководитель научного
содержания программы
магистратуры
профессор, доктор техн. наук _____ И.Е. Еремин
(подпись, дата)

Нормоконтроль
доцент, канд. техн. наук _____ Л.В. Никифорова
(подпись, дата)

Рецензент
руководитель ООО «Зэт-Лабс»
_____ И.С. Вирта
(подпись, дата)

Благовещенск, 2023

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Факультет математики и информатики
Кафедра информационных и управляющих систем

УТВЕРЖДАЮ
Зав. кафедрой
_____А.В. Бушманов
«_____»_____2023 г.

З А Д А Н И Е

К магистерской диссертации студента Рзаевой Гюльнар Мушвиг-кызы

1.Тема магистерской диссертации: Разработка модуля голосовой идентификации пользователя

(Утверждено приказом от 21.02.2023 № 442-уч)

2.Срок сдачи студентом законченной работы (проекта) 23.06.2023 г.

3.Исходные данные к магистерской диссертации: предметная область, отчеты по практической подготовке, модуль голосовой идентификации

4.Содержание магистерской диссертации (перечень подлежащих разработке вопросов): анализ предметной области проводимого исследования, алгоритмическое и программное обеспечение решения поставленной задачи, разработка и реализация модуля голосовой идентификации пользователя

5.Дата выдачи задания: 30.01.2023 г.

6.Руководитель магистерской диссертации: Самохвалова Светлана Геннадьевна, доцент, канд. техн. наук

(фамилия, имя, отчество, должность, уч. степень, уч. звание)

Задание принял к исполнению (30.01.2023):_____

(Подпись студента)

РЕФЕРАТ

Магистерская диссертация содержит 85 страниц., 26 рисунков, 2 таблицы, 32 источника, 1 приложение.

МОДУЛЬ РАСПОЗНАВАНИЯ ПОЛЬЗОВАТЕЛЯ, ГОЛОСОВАЯ БИОМЕТРИЯ, ИДЕНТИФИКАЦИЯ ПО ГОЛОСУ, АНАЛИЗ РЕЧЕВОГО СИГНАЛА, МЕЛ-ЧАСТОТНЫЕ КЕПСТРАЛЬНЫЕ КОЭФФИЦИЕНТЫ.

Системы голосовой идентификации пользователя быстро развиваются в последнее время. Причиной развития данных систем является их востребованность в таких областях, как биометрический поиск, голосовая верификация пассажиров и водителя, разграничение прав доступа к информации с помощью голосовой биометрии. Важным достоинством систем голосовой идентификации по отношению к другим биометрическим системам идентификации является их дешевизна. Важно также, что современные системы распознавания по голосу по уровню надёжности идентификации не уступают, а иногда даже превосходят, например, системы идентификации человека по изображению. Усовершенствование систем голосовой биомтерии привело к созданию интеллектуальных систем, позволяющих не только распознавать, но и автоматически синтезировать человеческую речь.

Несмотря на уникальность голоса человека, ни одна из систем голосовой идентификации пользователя, как и любая другая биометрическая система, не может гарантировать 100% надёжность идентификации. Основными источниками ошибок являются: окружение (шум, отражение звука и т.д.); особенности речи (длительность, тембр, тональность); канал связи (искажения микрофона и канала передачи, погрешности кодирования аудио сигнала и т.д.)

Объектом исследования является идентификация пользователя по голосовым данным.

Предметом исследования являются акустические характеристики речевого сигнала.

Цель данной работы заключалась в проведении анализа существующих методов распознавания речи и разработке модуля голосовой идентификации пользователя с использованием мел-частотных кепстральных коэффициентов в качестве информативных акустических признаков входного речевого сигнала.

Практическое значение работы состоит в разработке программного модуля, предназначенного для идентификации пользователя по голосу. Результаты работы могут быть использованы в системах контроля и управления доступом в качестве основного или вспомогательного средства идентификации.

Результаты научно-исследовательской работы были опубликованы в 2 источниках.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Биометрическая идентификация – это предъявление пользователем своего уникального биометрического параметра и процесс сравнения его со всей базой имеющихся данных. Для извлечения такого рода персональных данных используются биометрические считыватели.

Системы биометрической идентификации человека основаны на принципе распознавания и сравнения уникальных характеристик человеческого организма: отпечатков пальцев, рисунка сетчатки глаза или индивидуальных особенностей голоса. Голосовая идентификация – это одна из наименее ресурсоемких технологий ограничения доступа. Голосовая идентификация является одной из ветвей развития технологии обработки речи и применяется при создании различных систем охраны и разграничения доступа.

Идентификация голоса – технология автоматического сравнения неизвестного голоса с фонотекой известных голосов, предназначенная для использования в системах разграничения и управления доступом.

Сам механизм речи основан на том, что в гортани на определенных частотах вибрируют голосовые складки, содержащие голосовые связки и мышцы. В результате воспроизводится уникальный звук, присущий только данному индивиду.

Актуальность темы исследования определяется тем, что рынок речевых технологий стремительно развивается, охватывая практически все сферы нашей жизни, в настоящее время многие ведущие компании усиливают работу в направлении развития голосовых интерфейсов и технологии распознавания речи.

В настоящее время – век мобильной электроники и высоких требований к защите информации, речевая аутентификация может иметь множество применений. Потенциальные пользователи этой технологии – это госструктуры, финансовые и медицинские учреждения, а также телекоммуникационная отрасль.

Сфера ее применения – от пресечения мошенничеств, в случае кражи удостоверений личности до защиты данных».

Обработка биометрических данных, в первую очередь, востребована для решения ряда важнейших задач с точки зрения обеспечения высокой безопасности и повышения качества обслуживания. Биометрические системы используют для идентификации набор неотъемлемых характеристик человека, что является предпочтительным с точки зрения защиты от краж, копирования или потери идентификационных признаков. Биометрические технологии данный момент внедряются в системы контроля и управления доступом в качестве основных или вспомогательных средств идентификации, внедряются в качестве вспомогательных идентификационных технологий в сферу обслуживания (в том числе, при обслуживании важных лиц) и в системы правоохранительных органов.

Голос – такая же неотъемлемая черта каждого человека, как и его лицо или отпечатки пальцев. Широкое распространение средств связи открывают большие возможности для применения данного идентификатора; кроме того, распознавание по голосу весьма удобно для пользователей и требует от них минимум усилий.

В рамках данной работы стояла задача в изучении рынка средств голосовой идентификации, существующих методов и классификаций голосового распознавания и разработке модуля голосовой идентификации пользователя.

1 ОБЩАЯ ХАРАКТЕРИСТИКА ИССЛЕДУЕМОЙ ЗАДАЧИ

1.1 Анализ рынка программного обеспечения для голосовой биометрии

Развитие рынка мобильной биометрии связано с активным использованием биометрических технологий в финансовом секторе. Существуют следующие направления использования биометрических технологий в финансовой сфере:

– банкоматы и терминалы самообслуживания: сенсоры, интегрированные прямо в банкоматы, снятие наличных из банкомата с помощью мобильного телефона с использованием биометрических технологий, биометрические пластиковые карты;

– совершение покупок с помощью биометрических технологий: как мобильные платежи, так платежи "на кассе", осуществляемые с помощью мобильного телефона или биометрических терминалов без использования карт;

– дистанционное обслуживание: удаленная идентификация, голосовая биометрическая идентификация в call-центрах и иное;

– корпоративное использование биометрических технологий: контроль за работой сотрудников, доступ к защищенным системам, банковские системы контроля и управления доступом.

Отдельным трендом на мировом рынке является внедрение биометрических технологий в платежных системах. В частности, PayPal начал сотрудничество с производителями электронной техники Lenovo и Intel в целях обеспечения возможности прохождения идентификации пользователями на персональных компьютерах с помощью отпечатка пальца при осуществлении платежей. Данный проект реализуется в сотрудничестве с разработчиком в сфере биометрических технологий Synaptics[1].

[1] Synaptics – разработчик многофункциональных, безопасных и привлекательных для новых пользователей интерфейсных решений на основе биометрических технологий. https://www.synaptics.com/company/news/Intel–LenovoPaypal–Synaptics–FIDO–Alliance

Настоящим прорывом на рынке стал запуск платежных сервисов Apple Pay, Samsung Pay и Android Pay (83% транзакций приходится на Apple Pay и Samsung Pay), где для совершения платежа используется мобильный телефон и встроенные в него биометрические технологии. Согласно оценкам компании Grand View Research, мировой рынок так называемых "мобильных кошельков" к 2024 г. составит 7,5 трлн долларов США, демонстрируя среднегодовые темпы роста на уровне почти 33%.

Банки по всему миру запускают пилотные проекты для тестирования разных биометрических технологий, и многие банки уже активно их применяют в бизнес-практике. Так, два крупных банка Сингапура (DBS и OCBC) используют системы распознавания голоса в своих call-центрах. CityGroup также интегрировала голосовую биометрию в свои процессы в Азиатском регионе. В Великобритании Barclays использует технологию идентификации по рисунку вен пальца (VeinID) для реализации доступа в мобильные приложения и авторизации платежей. Barclays также использует голосовую биометрию.

Российский рынок биометрических технологий развивается быстрыми темпами – за последние 4 года это в среднем почти 36%. Ожидается, что в следующие 4 года отрасль будет расти в среднем на 25,62%. К концу 2022 года российский рынок увеличился по сравнению с 2018 годом в 2,5 раза, говорится в исследовании J'son & Partners Consulting.[2] По мнению аналитиков, основным трендом является переход от внутрикорпоративного использования технологий к внедрению клиентских сервисов. Одним из самых главных драйверов любых инициатив, нацеленных на ускорение ВВП или отдельной отрасли, является реализация проектов, направленных на создание спроса – в том числе государственного – на ту или иную продукцию. Так, в кризисные годы многие государства активно строят дороги и другие инфраструктурные объекты.

[2] J'son & Partners Consulting – международная консалтинговая компания, специализируется на рынках телекоммуникаций, медиа, ИТ, инновационных технологиях в России, СНГ, Центральной Азии.

Одним из таких проектов-драйверов в России является платформа единой биометрической идентификации (ЕБС), направленная на расширение возможностей граждан в получении финансовых услуг. Сбор биометрии стал обязательным для всех банков, соответствующих установленным законом критериям. Тестирование системы запущено еще в июле 2018 года.

В список участников сейчас входят около 60 кредитных организаций, в том числе стратегически важные. К 1 января 2019 года подключилось к единой биометрической системе (ЕБС) 20% отделений кредитных организаций, к 1 июля – 60%. К концу 2019 года сбор данных происходит во всех отделениях банков, подключенных к ЕБС. Проект требует наличия и развития решений по информационной безопасности, включая HSM (программно-аппаратный криптографический модуль), мобильных приложений, защищенных каналов связи, а также непосредственно оборудования, делающего фотографии лица клиента и запись его голоса.

В январе 2021 года аналитическая компания J’son & Partners Consulting представила результаты исследования российского рынка биометрических технологий. Его объем за последние четыре года (2017–2021 гг.) увеличивался примерно на 35,74% ежегодно.

Российский рынок биометрии отличается от мирового по своей структуре: если в первом наблюдается активное проникновение систем распознавания лиц, то в глобальном масштабе доминируют разработки в области идентификации отпечатков пальцев. К концу 2021 года доля технологий распознавания лиц в общем объеме российского биометрического рынка составила почти 50%, а в течение четырех лет этот сегмент демонстрировал рост на уровне 106,7% в год.

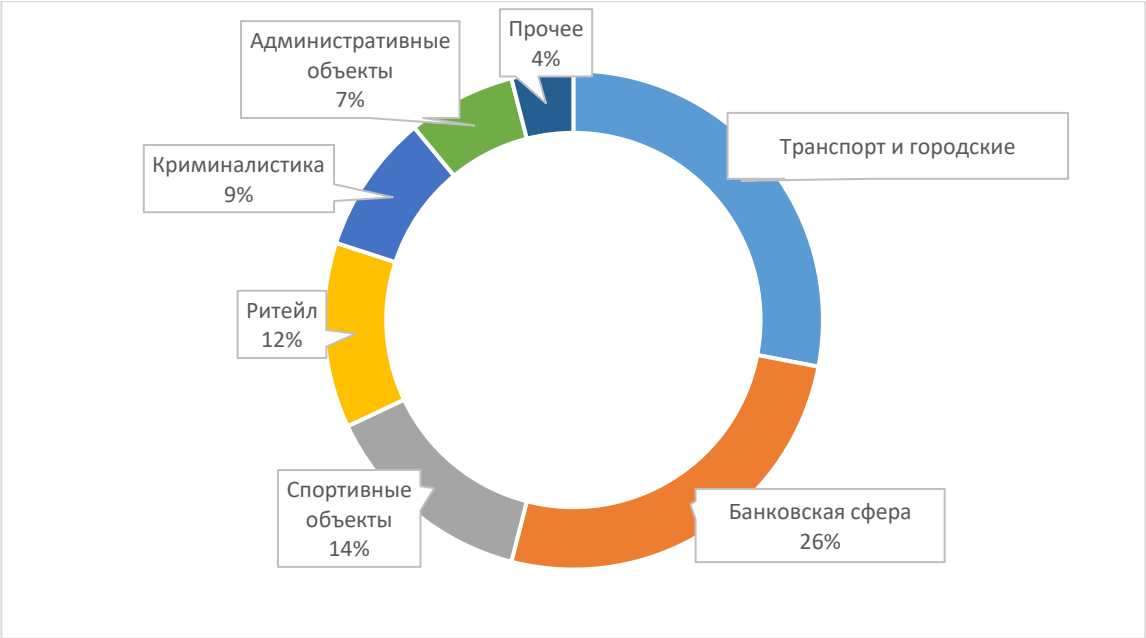

Рисунок 1 – Структура рынка биометрический технологий России в разрезе отраслей

С 2018 года использование биометрических персональных данных россиян в качестве основных идентификаторов для государственных систем и коммерческих сервисов активно продвигается в России.

Организована централизованная база данных – Единая биометрическая система, – оператором которой является ПАО «Ростелеком». Ведущие ИТ-вендоры указаны в таблице 1.

Таблица 1– ИТ-вендоры

№	Название продукта	Вендор
1	BioLink BioTime	Биолинк Солюшенс (BioLink Solutions)
2	BioLink FingerPass	Биолинк Солюшенс (BioLink Solutions)
3	Biosmart Studio	Прософт Биометрикс (ProSoft Biometrics)
4	Indeed Access Manager (Indeed AM)	Indeed ID, ранее Indeed Identity (Индид Компетенс Ай Ти)

Продолжение таблицы 1

5	СКУД BioSmart	Прософт Биометрикс (ProSoft Biometrics)
6	VisionLabs Luna	VisionLabs (ВижнЛабс)
7	BioSmart PV–WTC Терминал	Прософт Биометрикс (ProSoft Biometrics)
8	Единая биометрическая система (ЕБС)	Ростелеком
9	ЦРТ: Визирь	Группа компаний ЦРТ (Центр речевых технологий)
10	BioSmart–WTC2 Терминал	Прософт Биометрикс (ProSoft Biometrics)

Многие вендоры автоматизированных банковских систем активно включились в развитие технологий биометрической идентификации и реализовали ряд решений для взаимодействия банков с ЕБС и ЕСИА. Крупные ИТ-вендоры указаны в таблице 1. Однако на текущий момент сбор биологических атрибутов пока не носит массового характера и гражданами не востребован; к тому же для банков он является причиной дополнительных расходов, и его большая стоимость тормозит данный процесс.

Стоит дополнительно отметить, что тема применения биометрических технологий для идентификации граждан довольна сложна: она связана с защитой персональных данных и прав людей, с безопасностью бизнеса. После того как появилась информация о крупных утечках таких сведений из известных финансовых организаций, клиенты стали отказывать банкам в предоставлении биометрической информации из-за отсутствия уверенности в том, что она будет хорошо защищена и не попадёт в руки мошенников или недобросовестных сотрудников, торгующих базами данных.

1.2 Исследование предметной области распознавания речи

Подлинность диктора связана с физиологическими и поведенческими характеристиками системы производства речи конкретного диктора. Эти характеристики извлекаются из огибающей спектра (характеристики речевого тракта) и выше-сегментных признаков (характеристики голосового источника) речи. Обычно используют краткосрочные спектральные измерения кепстральных коэффициентов и их регрессионные коэффициенты.

В качестве регрессионных коэффициентов, обычно, использую коэффициенты первого и второго порядков, т.е производные временных функций кепстральных коэффициентов, извлеченные с каждого периода кадра, представляющие спектральную активность. Эти коэффициенты регрессии соответственно называются дельта кепстральные и дельта-дельта кепстральные коэффициенты. Распознавание дикторов может быть разделено на идентификацию и верификацию диктора. Идентификация диктора – это процесс определения, кто из зарегистрированных дикторов произнес фразу.

Верификация диктора – это процесс принятия или отклонения заявленной личности диктора. Большинство приложений, в которых используется голосовые данные подтверждают личность диктором, классифицируемым верификацией диктора.

Идентификация. В задаче идентификации диктора речевое высказывание неизвестного диктора анализируется и сравнивается с речевыми моделями известных дикторов. Неизвестный диктор идентифицируется как диктор, чья модель наиболее соответствует входному высказыванию. В верификации диктора, неизвестный диктор заявляет о своей подлинности, и высказывание этого неизвестного диктора сравнивается с моделью диктора, чью подлинность он заявил. Если соответствие достаточно хорошее, т.е. выше порога, заявленная личность подтверждается.

Высокое значение порога создает трудности для самозванцев быть принятыми системой, но с большим риском ложного отклонения правомерных пользователей. Низкое значение порога дает возможность правомерным пользователям быть однозначно принятыми, но с большим риском принятия самозванцев. Необходимо устанавливать порог на желаемый уровень клиентского отказа (ложный отказ) и принятия самозванца (ложный допуск), данные показывают распределение клиентов и самозванцев.

Основное различие между идентификацией и верификацией – это количество альтернативных решений. В идентификации количество альтернативных решений равно размеру популяции, тогда как в верификации только два выбора: принятие или отклонение, несмотря на размер популяции. Поэтому, эффективность идентификации диктора уменьшается при увеличении размера популяции, тогда как эффективность верификации диктора приближается к постоянной, независимо от размера популяции, но распределение физических характеристик крайне важно. Существует также случай, называемый «открытый выбор» идентификации, в котором относительной модели для неизвестного диктора может не существовать. В этом случае необходимо дополнительное альтернативное решение «неизвестный не соответствует ни одной модели».

Верификация. Верификация может быть рассмотрена частным случаем «открытого выбора» метода идентификации, в котором известен размер популяции равный единице. В верификации или идентификации дополнительный тестовый порог может быть применен для определения близко или соответствует принятое решения, если нет, то запрашивается новое испытание. Эффективность систем верификации диктора может быть оценена с помощью ROC кривой, принятой от психифизики. Кривая ROC получена путем определения двух вероятностей. Это вероятности правильного признания (процент ложного отклонения) и вероятности неправильного признания (процент ложного признания). По вертикальной и горизонтальной осям соответственно, и различные значения порога принятия решений. Также, компромисс обнаружения ошибки кривой, в которые

проценты ложного отклонения и ложного признания, определены на вертикальной и горизонтальной осях соответственно.

Кривая погрешности, как правило, наносится на нормальное отклонение масштаба. Равный уровень ошибок (ERR) является общепринятой мерой эффективности системы. Это соответствует порогу, в котором процент ложного признания равен проценту ложного отклонения.

Методы распознавания. Методы распознавания диктора часто делят на тексто-зависимые (фиксированные пароли) и тексто-независимые (без специальных паролей) методы. Первые требуют от диктора предоставления ключевых слов или предложений, один и тот же текст будет использован и для обучения, и для распознавания, тогда как последние не зависят от произнесения конкретного текста. Тексто-зависимые методы обычно основаны на шаблоне/модели последовательности соответствующих методов, в которых временная ось входящего речевого образца и связанных шаблонов или моделей записанных дикторов выровнена, и схожесть между ними накапливается с самого начала к концу высказывания. Так как этот метод может напрямую использовать голос личности, связанный с каждой фонемой или слогом, то он обычно достигает наибольшей эффективности распознавания, чем тексто-независимый метод.

Существуют различные применения, такие как судебно-экспертная экспертиза и наблюдение, в которых заранее заданные слова не могут быть изменены. Кроме того, человек может распознать дикторов, независимо от их содержания высказывания. Поэтому, тексто-независимые методы привлекают большее внимание. Другое достоинство тексто-независимого распознавания – это то, что оно может быть сделано последовательно, пока не будет достигнуто желаемое значение, без неприятного повторения слов диктором снова и снова.

Тексто-зависимые и тексто-независимые методы имеют значительные недостатки. Это то, что эти системы безопасности можно легко обойти, потому что кто-то может воспроизводить записанный голос зарегистрированного диктором выражения ключевых слов или предложений в микрофон и быть принятым как

зарегистрированный пользователь. Другая проблема – это то что людям часто не нравятся тексто-зависимые системы, потому что им не нравится их идентификационный номер, такой как номер социального страхования при прослушивании других людей. Для того, чтобы справиться с этими проблемами, некоторые методы используют маленькое множество слов, таких как цифры в качестве ключевых слов, и каждому пользователю будет предложено произнести последовательность ключевых слов, которые система случайно каждый раз выбирает. Однако даже этот метод не является достаточно надежным, так как он может быть взломан современными устройствами звукозаписи, которые могут произвести ключевые слова в заданном порядке.

Поэтому был предложен тексто-подсказочный метод распознавания диктора, в котором парольные предложения полностью заменяются через некоторое время. Тексто-зависимые методы распознавания дикторов делятся на методы DTW (динамическое искажение времени) и HMM (скрытые марковские модели).

1.3 Обзор существующих методов и средств обработки и анализа речевого сигнала

На сегодняшний день биометрические системы востребованы как никогда, они набирают популярность в самых различных отраслях. Иные способы контроля доступа, к сожалению, не приносят нужных результатов. Проведя анализ существующих программных продуктов, можно сказать следующее, что рынок достаточно насыщен. Рассмотрим наиболее приоритетные продукты по отдельности.

Программный продукт Apple, имеющий название «Siri», который был специально разработан, для продуктов, сходящих с конвейера компании «Apple», 28 ноября 2010 был представлен на мировом рынке, для всеобщего пользования. С данным приложение можно вести активный диалог, программа обрабатывает речь пользователя, проводит анализ и далее отвечает на вопросы, дает какие – либо рекомендации. За все время использования программного продукта, «Siri»

проводит анализ, делает логические заключение, с целью подстроиться под каждого пользователя индивидуально.

Проектирование ПО было начато в 2007 году Дагом Китлауссом (CEO), Адамом Чейером и Томом Грюбером, совместно с Норманом Винарским из SRI International. Разработки Siri контролировало Управление перспективных исследовательских программ. Сам программный продукт сошел с конвейера Международного центра искусственного интеллекта SRI.

Программный продукт от компании Microsoft, носящий название «Cortana», был представлен на мировом рынке 2 апреля 2014 года. Cortana – это ПО, с элементами искусственного интеллекта, которое будет использоваться в виде виртуального голосового помощника. Само ПО может быть интегрировано на такие платформы как Windows 10, Windows Phone,
Android, так же входит в планы интеграция на IOS, Xbox One. ПО призвано потребности пользователя предугадывать. Для более высокого уровня полезности программного продукта, следует разрешить доступ к личным данным. При интеграции Cortana, на свое устройство, она заменит стандартную поисковую систему, чтобы вызвать виртуального помощника следует нажать кнопку «Поиск». Нужный запрос можно как напечатать вручную или задать голосом. Интерфейс Cortana имеет гибкие настройки конфиденциальности.

Облачный сервис персонального ассистента, разработанный компанией Google и представленный на презентации Google I/O 18 мая 2016 года. Он считается Продолжением более раннего Google Now. Помощник может использоваться в смартфонах, также он включен в Google Allo – приложение для мгновенного обмена сообщениями, Google Home – умный голосовой Wi–Fi динамик для управления вашим домом, Android Wear – умные часы от Google.

Ассистент пришел на смену Google Now и запускается аналогичным образом – путем длительного нажатия на клавишу «Домой» или с помощью Voice Match.

Google Assistant подключается к Google Now и может извлекать из него информацию, выводя её в более привлекательном виде для пользователя, проверять погоду и много чего ещё. Однако, в отличие от своих аналогов, он может участвовать в двустороннем разговоре, используя алгоритм обработки естественного языка Google.

Приложение выдаёт информацию с учётом текущего местоположения пользователя, его личной информации и его личных предпочтений. Данный интерфейс является наиболее удобным для постоянного обновления информации, по словам самих разработчиков.

Алиса – виртуальный голосовой помощник, созданный компанией Яндекс. Распознаёт естественную речь, имитирует живой диалог, даёт ответы на вопросы пользователя и, благодаря запрограммированным навыкам, решает прикладные задачи. Алиса работает на смартфонах, компьютерах и автомобилях. По данным Яндекса, ежедневная аудитория голосового помощника Алисы составляет 8 млн пользователей.

Общение с ассистентом возможно голосом и вводом запросов с клавиатуры. Алиса отвечает или прямо в диалоговом интерфейсе, либо же показывает поисковую выдачу по запросу или нужное приложение. Кроме ответов на вопросы, Алиса умеет решать прикладные задачи: включить музыку, поставить будильник/таймер, вызвать такси или поиграть в игры.

Google, Apple, Microsoft и другие компании постоянно работают над оптимизацией своих голосовых ассистентов, чтобы научить их лучше понимать человеческую речь и запросы пользователей. Аналитическая компания Perficiend Digital решила провести независимое исследование и выяснить, какой из помощников стоит признать самым продвинутым.

По результатам исследования голосовые ассистенты справились более чем на 40%. Результаты представлены на графике (рисунок 2).

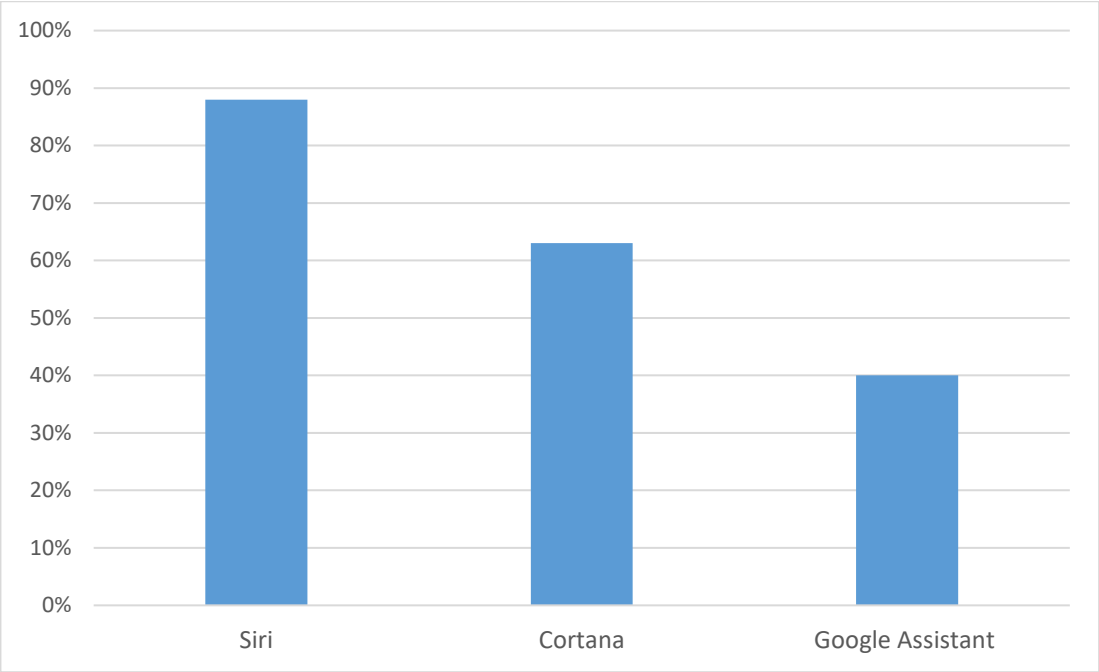

Рисунок 2 – Диаграмма сравнения голосовых помощников

Существует несколько способов преобразования сигнала, о самых распространённых из них будет описано ниже.

Преобразование Фурье.

Преобразование Фурье – это функция, которая описывает амплитуду и фазу каждой синусоиды, соответствующей определённой частоте. (Амплитуда представляет высоту кривой, а фаза представляет начальную точку синусоиды). Эта новая функция, описывающая коэффициенты («амплитуды») при разложении исходной функции на элементарные составляющие – гармонические колебания с разными частотами (подобно тому, как музыкальный аккорд может быть выражен в виде амплитуд нот, которые его составляют). Преобразование Фурье функции f вещественной переменной является интегральным и задаётся с помощью следующей формулы:

$$\hat{f}(w) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-xw}\, dx \qquad (1.1)$$

Хотя формула, которая задаёт преобразование Фурье, имеет понятный смысл только для функций класса $L_1(R)$, преобразование Фурье может использоваться и для более широкого класса функций и даже обобщённых функций. Это возможно благодаря особому ряду свойств преобразования Фурье: Преобразование Фурье является линейным оператор:

$$(\alpha f + \beta g) = \alpha \hat{f} + \beta \hat{g} \tag{1.2}$$

Справедливо равенство Парсеваля: $f \in L_1(R) \cap L_2(R)$ если, то преобразование Фурье сохраняет L_2-норму:

$$-\int_{\infty}^{\infty} |f(x)|^2 \, dx = \int_{-\infty}^{\infty} \left|\hat{f}(w)\right|^2 dw \tag{1.3}$$

Это свойство позволяет по непрерывности распространить определение преобразования Фурье на всё пространство $L_2(\mathbb{R})$. Равенство Парсеваля будет при этом справедливо для всех $f \in L_2(\mathbb{R})$. Формула обращения:

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (w) e^{xw} \, dw \tag{1.4}$$

Преобразование Фурье обобщённых функций. Преобразование Фурье можно определить для широкого класса обобщённых функций. Определим вначале пространство гладких быстро убывающих функций (пространство Шварца):

$$s(R) \coloneqq \left\{\varphi \in C^{\infty}(R) : \forall n, m \in N x^n \varphi^{(m)}\right\} \tag{1.5}$$

Ключевым свойством этого пространства является то, что это инвариантное подпространство по отношению к преобразованию Фурье.

Вейвлет-преобразование.

Широко используемое преобразование Фурье для анализа сигналов, как непрерывных, так и дискретных, оказывается недостаточно эффективным при обработке сложных сигналов. Например, Фурье спектры для сигналов из двух синусоид, которые с разными частотами, первый из которых, представляющий

собой сумму синусоид, а второй представляет собой, последовательно следующие друг за другом синусоиды, одинаковы и будут выглядеть как два пика на двух фиксированных частотах (рисунок 3). Из этого следует, преобразование Фурье в своём обычном виде не приспособлено для анализа нестационарных сигналов, так как теряется информация о временных характеристиках сигнала. Речевой сигнал является примером нестационарного процесса, в котором информативным является сам факт изменения его частотно-временных характеристик.

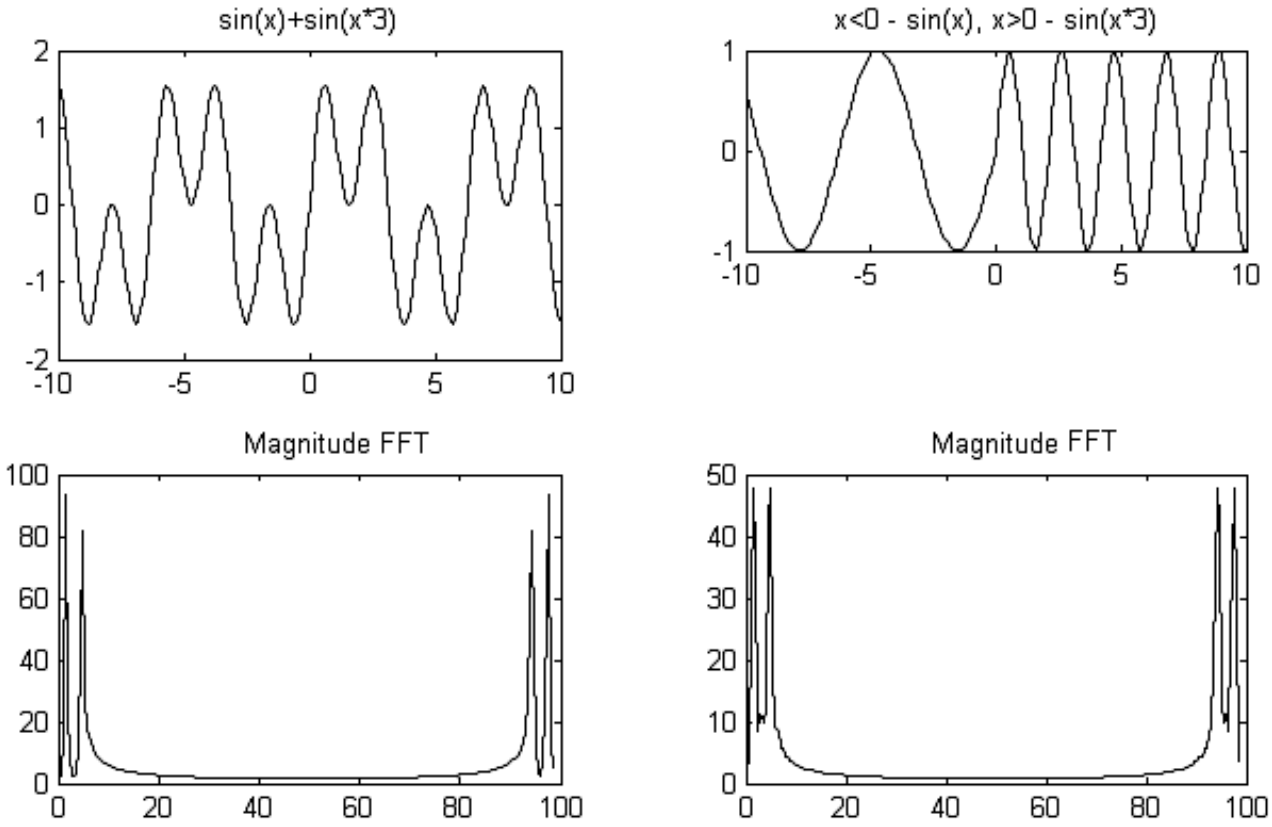

Рисунок 3 – Пример неинформативности преобразования Фурье

Для анализа таких процессов требуются базисные функции, способные выявлять в исследуемом сигнале как частотные, так и его временные характеристики, т. е. функции со свойствами частотно-временной локализации. Такие возможности предоставляют вейвлеты, являющиеся обобщением спектрального анализа.

Вейвлеты – функции двух аргументов – масштаба и сдвига. В отличие от стандартного преобразования Фурье, они позволяют обрабатывать сигнал одновременно в физическом пространстве – время, координата, и частотном пространстве

$$[w_\Psi f](x,a) = |a|^{-\frac{1}{2}} \int_{-\infty}^{+\infty} f(t)\psi\left(\frac{t-x}{a}\right) dt \qquad (1.6)$$

здесь $\Psi\left(\frac{t-x}{a}\right)$– вейвлет, a – масштабный коэффициент, x – параметры сдвига.

Таким образом, вейвлет-преобразование обеспечивает двумерное представление исследуемого сигнала в частотной области в плоскости частота-положение. Аналогом частоты при этом является масштаб аргумента базисной функции (чаще всего – времени), а положение характеризуется её сдвигом. Это позволяет найти особенности сигналов, одновременно локализуя их на временной шкале. Другими словами, вейвлет-анализ можно охарактеризовать как спектральный анализ локальных возмущений.

В теории вейвлет-анализа существует множество направлений. Например, используя многомасштабный (кратномасштабный) вейвлет-анализ сигнал можно представить, как последовательность образов с разной степенью детализации, что позволяет найти локальные особенности сигнала и классифицировать их по интенсивности.

Преобразование Гильберта-Хуанга.

Под преобразованием Гильберта-Хуанга (Hilbert-Huang transform – HHT) понимается метод эмпирической модовой декомпозиции (EMD) нелинейных и нестационарных процессов и Гильбертов спектральный анализ (HSA). Этот метод потенциально жизнеспособный для нелинейного и нестационарного анализа данных, специально для частотно-энергетических временных представлений. EMD-HSA предложил Норден Хуанг в 1995 в США (NASA) для изучения поверхностных волн тайфунов, включая возможность на анализ произвольных временных рядов коллективом соавторов в 1998 г. В последующие годы, активно

расширяя применения алгоритма для других новых отраслей науки и техники, взамен термина EMD-HSA был принят более короткий термин преобразования HHT.

Empirical Mode Decomposition

EMD (Empirical Mode Decomposition) – метод разложения сигналов на функции, получившие названия внутренних или «эмпирических мод». Метод представляет собой адаптивную итерационную вычислительную процедуру разложения исходных данных (непрерывных или дискретных сигналов) на эмпирические моды или внутренние колебания.

Огибающие сигналов. У каждого сигнала присутствуют локальные экстремумы: чередующиеся локальные максимумы и локальные минимумы с произвольным расположением по координатам (независимым переменным) сигналов. По этим экстремумам с использованием методов аппроксимации можно построить две огибающие сигналов: нижнюю – построенную по точкам локальных минимумов, и верхнюю – построенную по точкам локальных максимумов, а также функцию «среднего значения огибающих», которой отвечает срединная линия, расположенная в точности между нижней и верхней огибающими.

Функции внутренних мод сигналов. Модовая декомпозиция сигналов основана на предположении, что любые данные состоят из различных внутренних колебаний (intrinsic mode functions, IMF). В любой момент времени данные могут иметь множество сосуществующих внутренних колебаний – IMFs. Каждое колебание, линейное или нелинейное, представляет собой модовую функцию, которая имеет экстремумы и нулевые пересечения.

Кроме того, колебания в определенной степени «симметричны» относительно локального среднего значения. Конечные сложные данные образуются суммой модовых функций, наложенных на региональный тренд сигнала.

Эмпирическая мода – это такая функция, которая обладает следующими свойствами:

Количество экстремумов функции (максимумов и минимумов) и количество пересечений нуля не должны отличаться более чем на единицу.

В любой точке функции среднее значение огибающих, определенных локальными максимумами и локальными минимумами, должно быть нулевым.

Схема преобразования Гильберта-Хуанга может быть разделена на две части. В первом этапе, экспериментальные данные разлагаются в ряд внутренних модовых функций (IMFs). Эта декомпозиция рассматривается как расширение данных в терминах внутренних модовых функций. Иначе, эти внутренние модовые функции представлены как базис преобразования, которое может быть линейным или нелинейным, как диктуется по условиям. Так как IMFs имеют хорошие Гильбертовы преобразования, то могут быть вычислены соответствующие мгновенные частоты.

Таким образом, в следующем шаге можно с лёгкостью локализовать любое явление, как во времени, так и на частотной оси. Локальная энергия и мгновенная частота, выведенная из IMFs, дают дистрибутивные “энергетические время-частотные” данные, и такое представление, определяемое как Гильбертов спектр.

Алгоритм эмпирической декомпозиции сигнала складывается из следующих операций его преобразования:

Находим в сигнале положение всех локальных экстремумов, максимумов и минимумов процесса (номера точек экстремумов), и значения в этих точках (Рисунок 4) Между этими экстремумами сосредоточена вся информация сигнала. Группируем раздельно для максимумов и для минимумов массивы координат и соответствующих им амплитудных значений. Число строк в массивах максимумов и минимумов не должно отличаться более чем на 1.

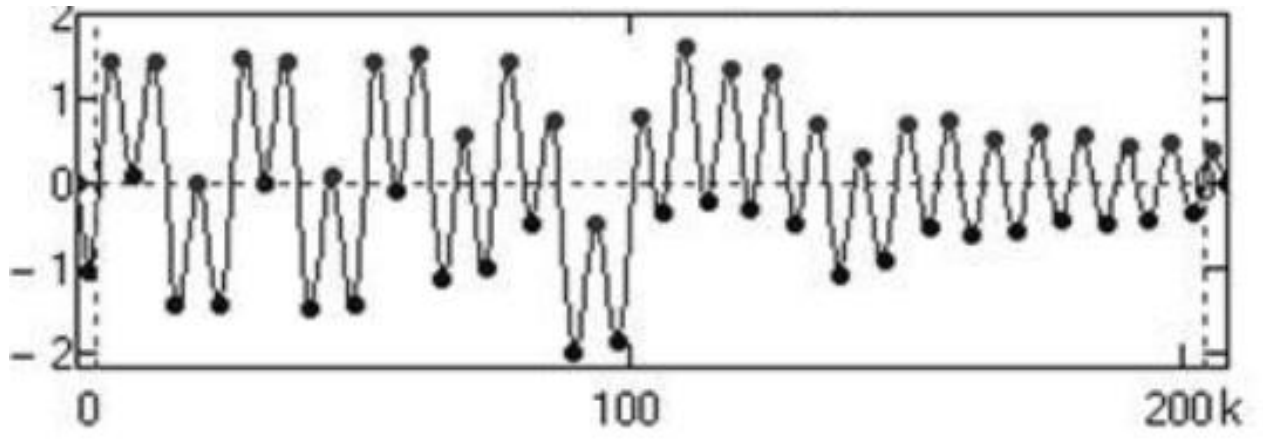

Рисунок 4 – Локализация экстремумов в сигнале

Применяя сплайны (или каким-либо другим методом) вычисляем верхнюю и нижнюю огибающие процесса соответственно, по максимумам и минимумам, как это показано на Рисунке 5. Определяем функцию средних значений между огибающими (Рисунок 5).

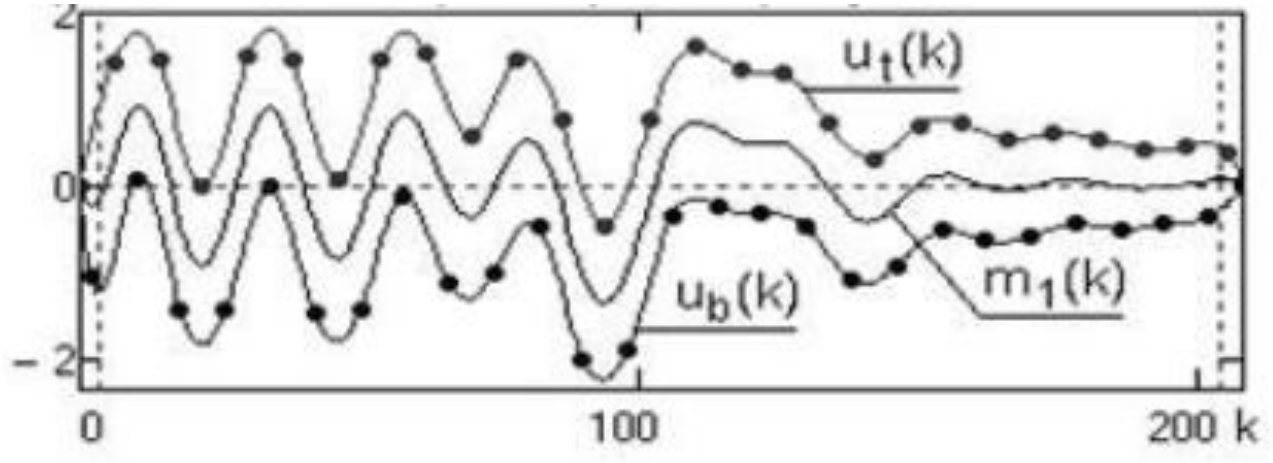

Рисунок 5 – Интерполяция экстремумов и построение огибающих

По мере увеличения количества итераций функция стремится к нулевому значению, а функция – к неизменяемой форме.

Метод EMD закончен, когда остаток, в идеале, не содержит экстремумов. Это означает, что остаток – или константа или монотонная функция. Извлечённые IMFs симметричны, имеют уникальные локальные частоты, различные IMFs не показывают ту же самую частоту в то же самое время. Другими словами, оста-

новка декомпозиции сигнала должна происходить при максимальном «выпрямлении» остатка, т.е. превращения его в тренд сигнала по интервалу задания с числом экстремумов не более 2-3.

В результате проделанной работы по данной главе было проведено предварительное исследование рынка системы распознавания речи, сформулирована актуальность проблемы, так же была исследована предметная область системы контроля доступа, выявлены все достоинства и недостатки; Были рассмотрены существующие методы обработки и анализа голоса.

Преобразование Фурье и вейвлет-преобразование заслуженно получили широкую известность благодаря использованию в них хорошо обоснованных математических методов и наличию эффективных алгоритмов их реализации.

Кроме того, оба этих преобразования, как показала практика, достаточно универсальны и могут с успехом применяться в различных областях. Но для практического использования хотелось бы иметь преобразование, не только позволяющее работать с нестационарными процессами, но и использующее адаптивный, определяемый исходными данными базис преобразования. Поэтому для данной задачи выбран метод Гильберта-Хуанга, так как он удовлетворяет всем необходимым условиям.

2 АЛГОРИТМИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ РЕШЕНИЯ ЗАДАЧИ

2.1 Алгоритм работы модуля

Использование модуля голосовой идентификации диктора в системе распознавания речи предполагает увеличения коэффициента качества распознавания, а также повышение дикторонезависимости системы.

Структура входных и выходных данных представлена на рисунке 6.

Входными данными являются:

– аудиосигнал;

– словарь эталонных фраз зарегистрированных дикторов;

Выходными данными являются:

– сообщения об идентификации диктора;

– отчеты о работе программы.

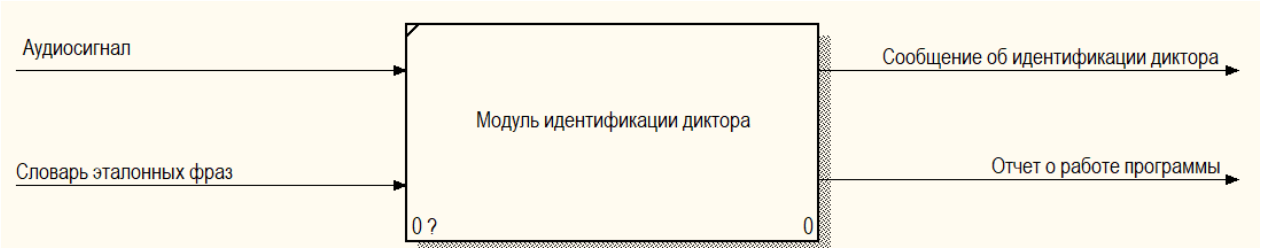

Рисунок 6 – Диаграмма взаимодействия модуля идентификации диктора

На вход модуля голосовой идентификации диктора поступает аудиосигнал, запись которого происходит через микрофон, подключённый к входу звуковой карты компьютера, или используется уже ранее записанный сигнал. Акустический компонент преобразует сигнал в цифровую форму с заданными параметрами частоты дискретизации. Затем осуществляется очистка сигнала от шума, удаление не несущих информации участков, проводится нормализация сигнала и его разбиение на фиксированные интервалы во временной области, на которых будут определяться характеристики. На блоке выделение характерных признаков происходит выделение характерных признаков сигнала с помощью преобразования Гильберта-Хуанга.

Под преобразованием Гильберта-Хуанга (Hilbert-Huang transform – HHT) понимается метод эмпирической модовой декомпозиции (EMD) нелинейных и нестационарных процессов и Гильбертов спектральный анализ (HSA). Этот метод потенциально жизнеспособный для нелинейного и нестационарного анализа данных, специально для частотно–энергетических временных представлений.

EMD (Empirical Mode Decomposition) – метод разложения сигналов на функции, получившие названия внутренних или «эмпирических мод». Метод представляет собой адаптивную итерационную вычислительную процедуру разложения исходных данных (непрерывных или дискретных сигналов) на эмпирические моды или внутренние колебания.

Эмпирическая мода – это такая функция, которая обладает следующими свойствами:

– количество экстремумов функции (максимумов и минимумов) и количество пересечений нуля не должны отличаться более чем на единицу;

– в любой точке функции среднее значение огибающих, определенных локальными максимумами и локальными минимумами, должно быть нулевым.

Затем данные переходят на блок идентификации, так же в этот блок приходят данные с блока база данных с уже сохранёнными данными.

Далее происходит идентификация и классификация (определение принадлежности к определённому классу), затем следует отправление результата об идентификации диктора. Декомпозиция диаграммы взаимодействия модуля идентификации диктора представлена на рисунке 7.

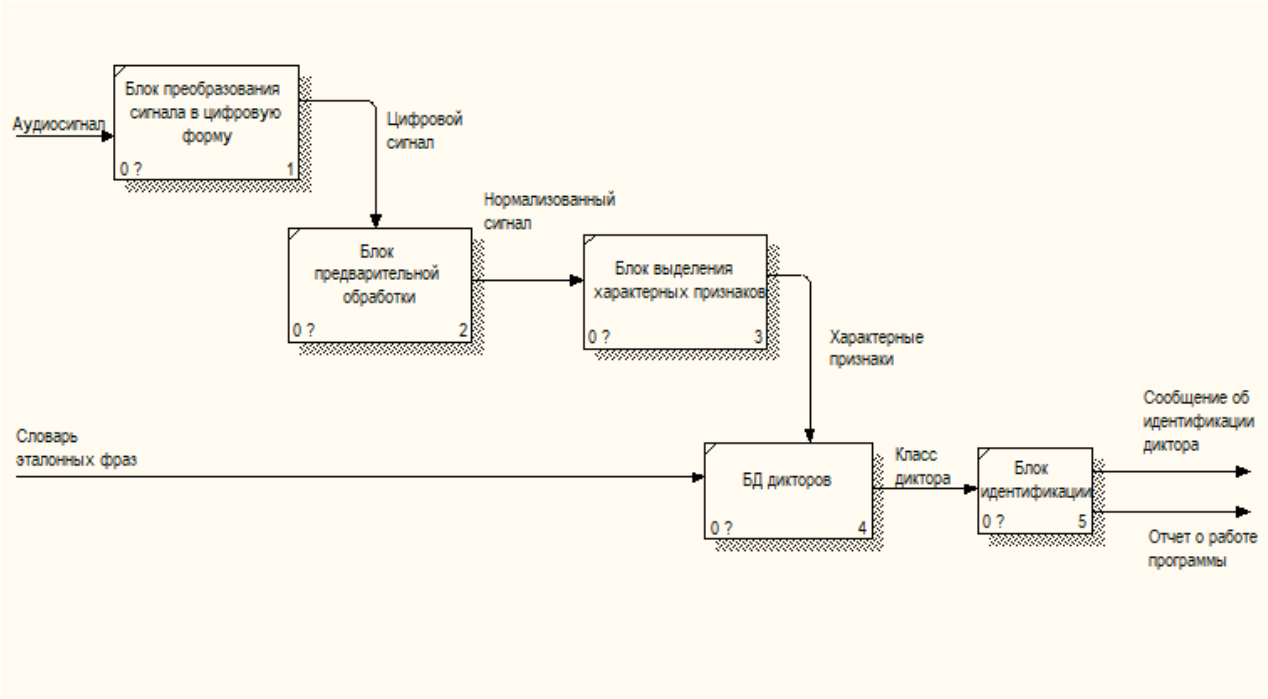

Рисунок 7 – Декомпозиция модуля идентификации диктора

Нейронная сеть. В любой системе распознавания речи всегда присутствует этап сравнения входного сигнала с имеющимися эталонами. Вне зависимости от наличия или отсутствия предварительной обработки сигнала (выделение основных признаков, преобразование в другую форму в новом параметрическом пространстве и т. д.) сигнал представляет собой вектор в установленном параметрическом пространстве, который в дальнейшем будет сравниваться с векторами, хранящимися в памяти для определения его принадлежности к определённому классу. Такая классификация образов является одной из основных задач, решаемых с помощью нейронной сети, среди которых наиболее распространены:

– распознавание зрительных, слуховых образов; огромная область применения: от распознавания текста и целей на экране радара до систем голосового управления;

– ассоциативный поиск информации и создание ассоциативных моделей;

– формирование моделей различных нелинейных и трудно описываемых математически систем, прогнозирование развития этих систем во времени; применение на производстве; прогнозирование природных процессов, изменений курсов и т.д;

– системы управления и регулирования с предсказанием; управление роботами, другими сложными устройствами – разнообразные конечные автоматы: системы массового обслуживания и коммутации, телекоммуникационные системы;

– принятие решений и диагностика, исключающие логический вывод; особенно в областях, где отсутствуют чёткие математические модели: в медицине, криминалистике, финансовой сфере;

Выделим характерные свойства искусственных нейронных сетей:

Обучаемость. Одним из этапов функционирования нейронной сети является обучение, в процессе которого на ее вход поочерёдно поступают данные из обучающего набора с целью корректировки весовых коэффициентов синаптических связей для получения наиболее адекватного сигнала на выходе нейронной сети;

Способность к обобщению. Отклик сети после обучения может быть до некоторой степени нечувствителен к небольшим изменениям входных сигналов (шуму или вариациям входных образов);

Способность к абстрагированию. Если при обучении предъявить сети несколько искажённых вариантов входного образа, то сеть может создать на выходе идеальный образ, который не входил в обучение;

Универсальность. Хотя почти для всех перечисленных задач существуют эффективные математические методы решения и, несмотря на то, что сети проигрывают специализированным методам; благодаря универсальности и перспективности они являются важным направлением исследования, требующим тщательного изучения.

2.2 Методы классификации речевого сигнала

В данном подразделе рассматриваются основные существующие решения задачи идентификации диктора по голосу. Несмотря на то, что методы во многом отличаются, в целом можно выделить следующие основные этапы, присущие каждому из рассматриваемых методов.

Извлечение признаков из входного речевого сигнала и построение модели (шаблона) диктора на основе полученных на предыдущем шаге векторов признаков. Процесс определения диктора, зарегистрированного в системе, по входному речевому сигналу во всех рассматриваемых методах состоит в поиске наиболее подходящей сохранённой модели на основе каких-либо критериев.

Dynamic Time Warping.

Dynamic Time Warping (DTW) – метод динамического программирования, позволяющий найти близость между двумя последовательностями измерений за некоторый промежуток времени. В общем случае эти последовательности могут быть разной длины, и измерения могут производиться с разной скоростью. В качестве сохраняемой модели в данном методе выступает последовательность векторов признаков входного речевого сигнала из обучающей выборки $Q = \{q_1,\ldots,q_n\}$. Пусть $C = \{c_1,\ldots,c_m\}$– последовательность векторов признаков входного речевого сигнала из тестовой выборки. Также вводятся понятия матрицы выравнивания двух последовательностей в позиции (i, j) которой содержится значение выравнивания между элементами c_i и q_j последовательностей C и Q соответственно, и набора индексов смежных элементов этой матрицы $W = \{W_1,\ldots,W_T\}$, определяющего соответствие между элементами сопоставляемых последовательностей. При этом элементы набора W должны удовлетворять следующим условиям:

$$w_1 = (1,1), w_T = (m, n)$$

$$\text{Если } w_{t-1} = (a', b'), \text{ то } w_t = (a, b), \text{ где } a - a' \leq 1, b - b' \leq 1 \quad (2.1)$$

Целью алгоритма DTW является нахождение такого набора W, удовлетворяющего условиям 1 и 2, при котором суммарное искажение последовательности C относительно последовательности Q было бы минимальным, то есть:

$$\mathrm{BTW}(\mathrm{Q},\mathrm{C}) = \min\left\{\frac{1}{\mathrm{T}}\sqrt{\sum_{\mathrm{t}=1}\mathrm{M}(\mathrm{w_t})}\right\} \quad (2.2)$$

Значение этого выражения и будет определять меру близости последовательностей Q и C. Для нахождения значения DTW(Q, C) применяется метод динамического программирования, где на каждом шаге вычисляется значение M(i, j) по формуле:

$$\mathrm{M}(\mathrm{i},\mathrm{j}) = \mathrm{d}(\mathrm{i},\mathrm{j}) + \min\{\mathrm{M}(\mathrm{i}-1,\mathrm{j}-1), \mathrm{M}(\mathrm{i}-1,\mathrm{j}), \mathrm{M}(\mathrm{i},\mathrm{j}-1)\} \quad (2.3)$$

Основным преимуществом алгоритма DTW является простота реализации. Тем не менее, данный алгоритм неприменим для решения задачи текстонезависимой идентификации диктора.

Hidden Markov Model.

HMM – статистическая модель, которая может использоваться для решения задачи классификации скрытых параметров на основе наблюдаемых. HMM представляет собой конечный автомат, в котором переходы между состояниями осуществляются с некоторой вероятностью, и задано стартовое состояние, с которого начинается процесс. Через дискретные моменты времени может осуществляться переход в новые состояния. При этом каждому скрытому состоянию с заданной вероятностью соответствует наблюдаемое состояние. Кроме того, текущее состояние автомата зависит только от конечного числа предыдущих, а закон смены состояний не меняется во времени.

Vector Quantization.

Задача векторного квантования с кодовыми векторами $W = \{W_1, W_2 \ldots, W_n\}$, для последовательности входных векторов $C=\{C_1, C_2 \ldots, C_m\}$, ставится как задача минимизации искажения при замещении каждого вектора из Q соответствующим кодовым вектором. Моделью диктора в данном методе является множество

кодовых векторов, получаемое из входной последовательности векторов признаков речевого сигнала. Для построения этого множества исходная последовательность векторов признаков разбивается на L кластеров, и в качестве кодовых векторов берутся их центры.

Метод векторного квантования прост в реализации, применим к задаче текстонезависимой идентификации диктора, однако не всегда дает высокую точность распознавания.

Нейронная сеть.

В любой системе распознавания речи всегда присутствует этап сравнения входного сигнала с имеющимися эталонами. Вне зависимости от наличия или отсутствия предварительной обработки сигнала (выделение основных признаков, преобразование в другую форму в новом параметрическом пространстве и т. д.) сигнал представляет собой вектор в установленном параметрическом пространстве, который в дальнейшем будет сравниваться с векторами, хранящимися в памяти для определения его принадлежности к определённому классу. Такая классификация образов является одной из основных задач решаемых с помощью нейронной сети, среди которых наиболее распространены:

– распознавание зрительных, слуховых образов; огромная область применения: от распознавания текста и целей на экране радара до систем голосового управления;

– ассоциативный поиск информации и создание ассоциативных моделей;

– формирование моделей различных нелинейных и трудно описываемых математически систем, прогнозирование развития этих систем во времени; применение на производстве; прогнозирование природных процессов, изменений курсов и т.д;

– системы управления и регулирования с предсказанием; управление роботами, другими сложными устройствами - разнообразные конечные автоматы: системы массового обслуживания и коммутации, телекоммуникационные системы;

– принятие решений и диагностика, исключающие логический вывод; особенно в областях, где отсутствуют чёткие математические модели: в медицине, криминалистике, финансовой сфере; Выделим характерные свойства искусственных нейронных сетей:

– Обучаемость. Одним из этапов функционирования нейронной сети является обучение, в процессе которого на ее вход поочерёдно поступают данные из обучающего набора с целью корректировки весовых коэффициентов синаптических связей для получения наиболее адекватного сигнала на выходе нейронной сети;

– Способность к обобщению. Отклик сети после обучения может быть до некоторой степени нечувствителен к небольшим изменениям входных сигналов (шуму или вариациям входных образов);

– Способность к абстрагированию. Если при обучении предъявить сети несколько искажённых вариантов входного образа, то сеть может создать на выходе идеальный образ, который не входил в обучение;

– Параллельность обработки и реализуемости нейронных сетей;

– Универсальность. Хотя почти для всех перечисленных задач существуют эффективные математические методы решения и, несмотря на то, что сети проигрывают специализированным методам; благодаря универсальности и перспективности они являются важным направлением исследования, требующим тщательного изучения.

На данный момент существуют методы классификации речевого сигнала, позволяющих решать задачу идентификации диктора по голосу. Многие из их ещё находятся в стадии разработки и закрыты от общего обзора, поэтому решено разработать свой модуль идентификации диктора. Для решения задачи классификации рационально использовать нейронную сеть с архитектурой трёхслойного персептрона. К её достоинствам можно отнести сравнительную простоту

анализа и достаточно высокую эффективность классификации. Благодаря использованию непрерывной функции возбуждения такие сети способны к обобщению обучающей выборки.

2.3 Обзор возможностей профильного программного обеспечения

Данное ПО предназначено для голосовой идентификации диктора в системе распознавания речи. Внедрение предполагает увеличение коэффициента качества распознавания, а также повышение дикторонезависимости системы.

При работе с модулем используется инструмент пользователя MATLAB, для моделирования структуры нейронной сети выбран пакет Neural Networks был.

В пакет входят более полутора десятка известных типов искусственных нейронных сетей и обучающих правил, которые позволяют пользователю выбрать наиболее подходящую для конкретного приложения или исследовательской задачи парадигму, которые предоставляют следующие возможности:

- Преобразование звукового сигнала в цифровую форму;
- Предварительная обработка сигнала, поступающего на вход системе идентификации диктора, удаление участков тишины – пауз до и после записи, а также между словами;
- Выделение характерных признаков. Для выделения таких характеристик выбран спектрально-формантный анализ, суть которого описана в работе;

Рассмотрим диаграмму вариантов использования, представленную на рисунке 8. Пользователь участвует в процессе предоставления речевого сигнала и проверке подлинности. Речевой сигнал, полученный от пользователя, поступает в блок идентификации, после чего происходит проверка на совпадение. Если голос распознан, то появляется сообщение об идентификации пользователя.

Оператор имеет возможность регистрации нового голоса, обновления БД и устранения ошибок работы системы. В первых двух процессах также используется база данных.

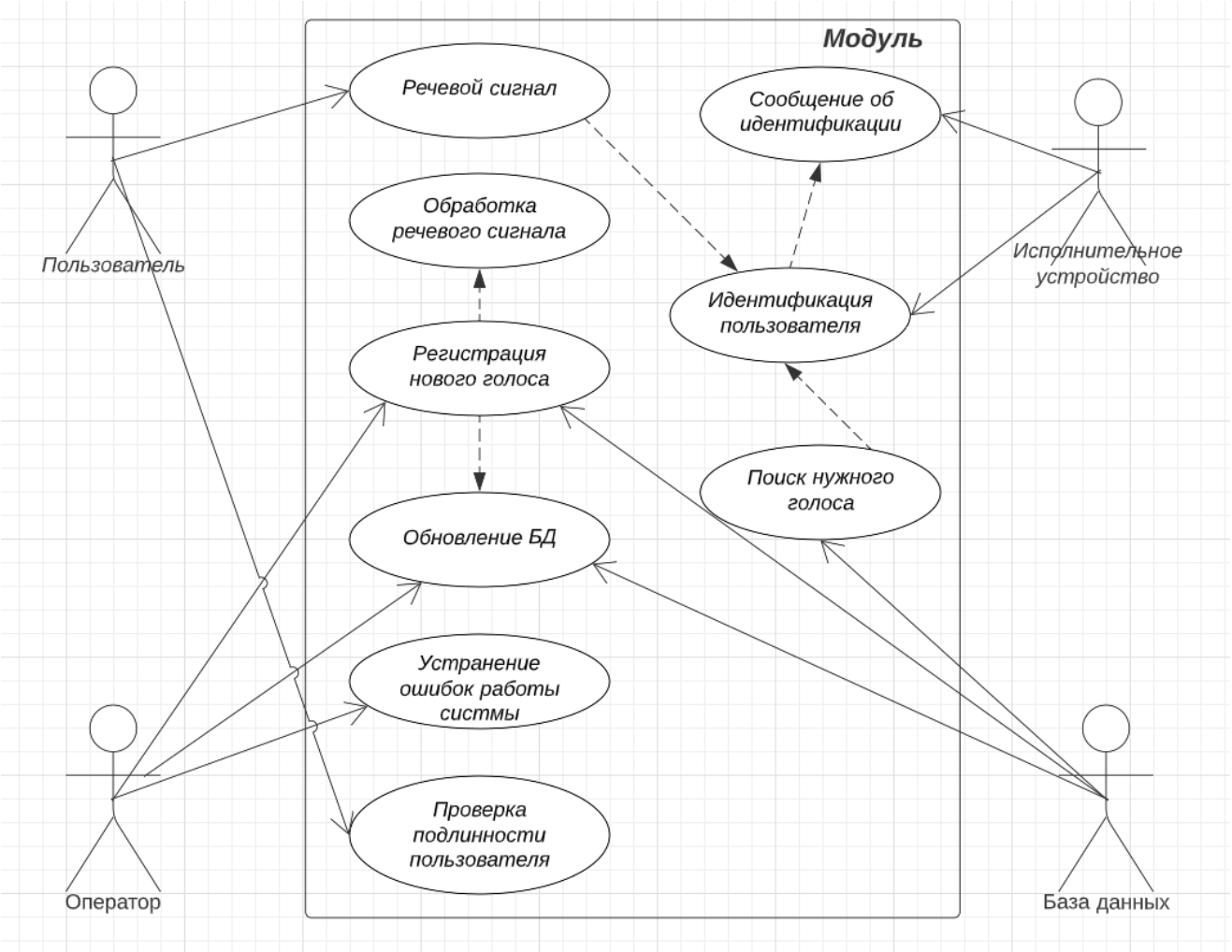

Рисунок 8 – Диаграмма вариантов использования

Программное обеспечение состоит из нескольких блоков, которые между собой взаимодействуют следующим образом: предоставленная пользователем голосовая информация попадает в блок преобразования сигнала в речевую форму. После чего обработанный сигнал попадает в блок предварительной обработки, где происходит выделение характерных признаков. После блока выделения характерных признаков происходит поиск совпадений в БД Дикторов, затем в блоке идентификации происходит идентификация пользователя и результат идентификации предоставляется пользователю.

На рисунке 9 изображена диаграмма последовательности, рассмотрим подробно каждый блок.

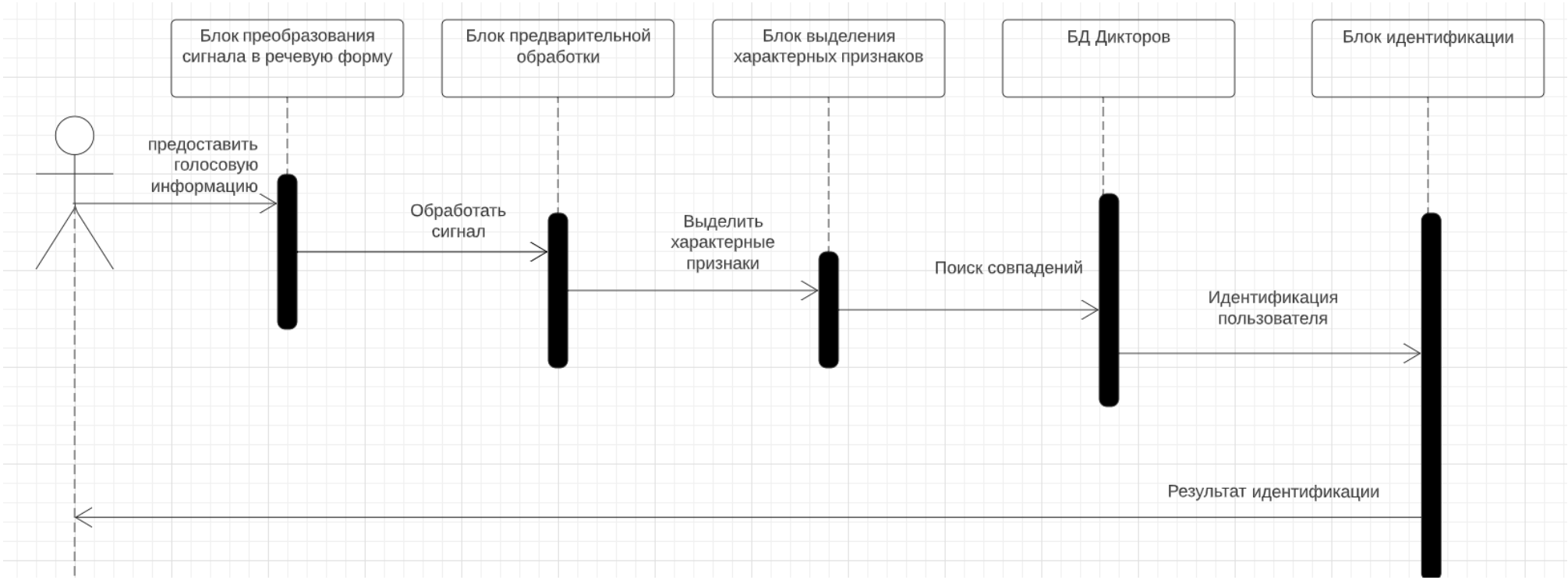

Рисунок 9 – Диаграмма последовательности

Блок преобразования сигнала в цифровую форму. На вход модуля голосовой идентификации диктора поступает аудиосигнал, запись которого происходит через микрофон, подключённый к входу звуковой карты компьютера, или используется уже ранее записанный сигнал. Акустический компонент преобразует сигнал в цифровую форму с заданными параметрами частоты дискретизации. Цифровой сигнал передается в следующий блок предварительной обработки.

Блок предварительной обработки сигнала. В данном блоке осуществляется очистка сигнала от шума, удаление не несущих информации участков. Проводится нормализация сигнала и его разбиение на фиксированные интервалы во временной области, на которых будут определяться характеристики. Далее нормализованный сигнал передается в блок выделения характерных признаков.

Блок выделения характерных признаков. На блоке выделение характерных признаков происходит выделение характерных признаков сигнала с помощью преобразования Гильберта-Хуанга.

Блок Идентификации. При выборе режима «распознавание» данные переходят на блок идентификации, так же в этот блок приходят данные с блока база данных с уже сохранёнными данными. Далее происходит идентификация, затем следует отправление результата об идентификации диктора.

База данных дикторов. В базе данных хранится словарь эталонных фраз, заранее записанные аудиофайлы для распознавания пользователей.

Данный модуль предназначен для улучшения работы системы распознавания речи. С помощью данного модуля система сможет распознавать дикторов по их характерным признакам голоса, что позволит увеличить надёжность распознавания диктора.

Разрабатываемое программное обеспечение должно обеспечивать выполнение следующих функциональных действий:

– преобразование звукового сигнала в цифровую форму;

– предварительная обработка сигнала, поступающего на вход системе идентификации диктора, удаление участков тишины – пауз до и после записи, а также между словами;

– выделение характерных признаков. Для выделения таких характеристик выбран спектрально-формантный анализ, суть которого описана в работе;

– анализ речевой информации со словарем эталонных фраз каждого диктора с использованием его обучающих акустических векторов.

– вывод информации об анализе и статусе распознавания пользователя, если диктор распознан, то система выводит ФИО диктора, если нет, то сообщение: «Пользователь не идентифицирован»;

Предоставление отчетов о работе программного обеспечения;

2.4 Характеристика выбранного программно-технического обеспечения

В качестве платформы для исследований и разработки модуля голосовой идентификации диктора был выбран пакет MATLAB. Данный выбор объясняется следующим:

MATLAB – пакет прикладных программ для решения задач сложных технических вычислений, а также используемый в этом пакете язык программирования. MATLAB используют более 1 000 000 научных и инженерных работников, он работает на большинстве современных операционных систем, включая GNU/Linux, Mac OS, Solaris и Microsoft Windows.

Язык MATLAB является высокоуровневым интерпретируемым языком программирования, включающим основанные на матрицах структуры данных, широкий спектр функций, интегрированную среду разработки, объектно–ориентированные возможности и интерфейсы к программам, написанным на других языках программирования.

MATLAB – это удобные средства для разработки алгоритмов, включающие высокоуровневые, с использованием концепций объектно–ориентирован-

ного программирования. В нём имеются все необходимые средства интегрированной среды разработки, включая отладчик и профайлер. Функции для работы с целыми типами данных облегчают создание алгоритмов для микроконтроллеров и других приложений, где это необходимо.

Встроенная среда разработки позволяет создавать графические интерфейсы пользователя с различными элементами управления, такими как кнопки, поля ввода и другими. С помощью компонента MATLAB Compiler эти графические интерфейсы могут быть преобразованы в самостоятельные приложения, для запуска которых на других компьютерах необходима установленная библиотека MATLAB Component Runtime.

В пакет MATLAB входят различные интерфейсы для получения доступа к внешним подпрограммам, написанным на других языках программирования, данным, клиентам и серверам, общающимся через технологии Component Object Model или Dynamic Data Exchange, а также периферийным устройствам, которые взаимодействуют напрямую с MATLAB. Многие из этих возможностей известны под названием MATLAB API.

Система MATLAB предоставляет мощный язык программирования, ориентированный на математические преобразования, который превосходит по возможности и скорости вычислений традиционные языки программирования.

Для решения проблемы классификации, сообщающиеся функции и процедуры системы собираются в специальные папки. Это создаёт концепцию пакетов прикладных программ, представляющих собой коллекции M-файлов для решения определённой задачи или проблемы. Именно пакеты прикладных программ – MATLAB Application Toolboxes, которые входят в состав семейства продуктов MATLAB, позволяют находиться этой системе на уровне самых современных приложений.

Для моделирования структуры нейронной сети выбран пакет Neural Networks. В пакет входят более полутора десятка известных типов искусственных нейронных сетей и обучающих правил, которые позволяют пользователю

выбрать наиболее подходящую для конкретного приложения или исследовательской задачи парадигму. Для каждого типа архитектуры и обучающего алгоритма имеются функции инициализации, обучения, адаптации, создания и моделирования, демонстрации и примеры применения.

Один из пакетов системы – Wavelet Toolbox, предоставляет разнообразные возможности обработки сигналов с помощью вейвлетов. С его помощью реализуются принципиально новые виды декомпозиции и реконструкции сигналов и изображений с повышенной эффективностью и новыми качественными возможностями – например, в идентификации тонких локальных особенностей функций и сигналов.

В стандартный пакет аудио поддержки системы MATLAB R15b входят функции, которые позволяют произвести запись звукового сигнала с возможностью настройки значений частоты дискретизации и разрядности.

Встроенная в MATLAB среда разработки пользовательского интерфейса MATLAB GUIDE позволяет реализовать элементы визуально-ориентированного программирования (инструментальные панели, кнопки, меню и т.д.). В системе MATLAB присутствует уникальная возможность сохранения значений всех обрабатываемых переменных в постоянную память из оперативной. Переменные, которые сохранены в специальных mat–файлы, могут быть в дальнейшем загружены т.к. одновременно файлы могут быть сохранены переменные разных типов, и структура самого хранения является упорядоченной, то данную возможность системы MATLAB удобнее применять для создания базы данных эталонов.

Разрабатываемое программное обеспечение главным образом должно обеспечить выполнение следующих функциональных действий:

– анализ речевой информации с последующим распознаванием диктора;

– вывод информации об анализе и статусе распознавания диктора;

Для обеспечения надежности предъявляются следующие требования:

– программное обеспечение должно стабильно функционировать при ограниченном объеме оперативной памяти и процессорного времени;

– сохранение информации о работе программного обеспечения в электронном виде;

– сохранение результатов работы программного обеспечения в электронном виде.

Требования к информационной программной совместимости.

Данное программное обеспечения должно функционировать в операционных системах Windows 8/8.1/10/11.

3 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ ПРЕДПОЛАГАЕМОГО АЛГОРИТМА РЕШЕНИЯ ЗАДАЧИ

3.1 Основные этапы практической разработки программного продукта

3.1.1 Общая структура программного продукта

Средой для разработки программы был выбран программный пакет MATLAB, т.к. в состав пакета входят готовые библиотеки для работы со звуковыми сигналами. Также в состав MATLAB входит среда Guide Builder для создания приложений с графическим интерфейсом пользователя. Для моделирования структуры нейронной сети был выбран пакет Neural Networks. Wavelet Toolbox, входящий в состав MATLAB является очень удобный инструментом для изучения и проведения вейвлет-преобразований. В стандартный пакет аудио поддержки системы MATLAB включены функции, позволяющие произвести запись звукового сигнала. Главное меню пакета вызывается из MATLAB с помощью команды wavemenu.

Для моделирования системы распознавания диктора была подготовлена текстонезависимая русскоязычная база голосов, состоящая из 8-ми дикторов.
Алгоритм моделирования системы распознавания диктора включает в себя 2 этапа: обучение и тестирование. И на первом и на втором этапе выполняется выделение уникальных характеристик диктора – MFCC-коэффициентов. Данный процесс сопровождается предварительной обработкой речевого сигнала.

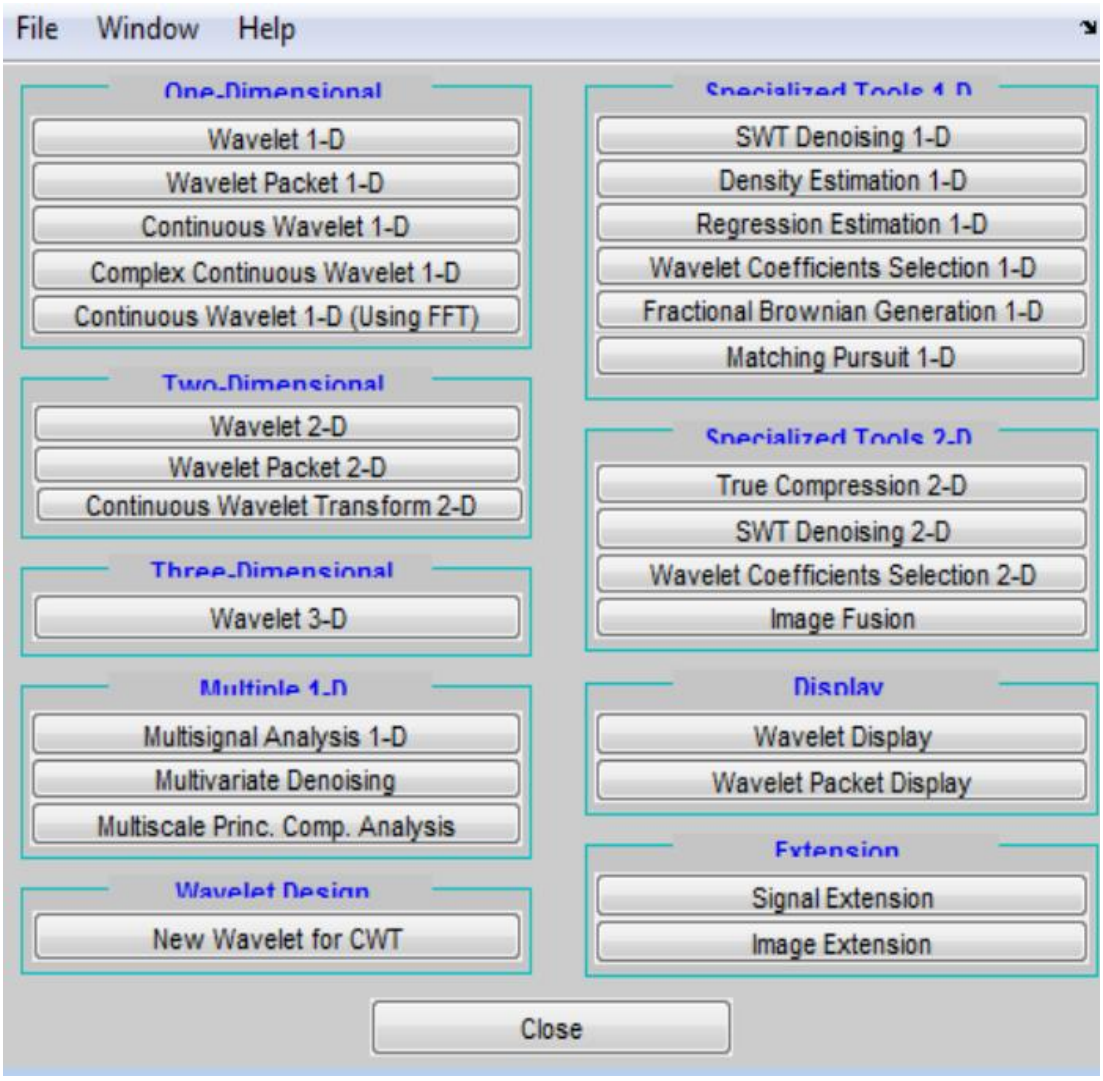

Рисунок 10 – Главное окно Wavelet Toolbox

В MATLAB предусмотрены средства для воспроизведения и записи звука (речи), а также для работы со звуковыми файлами формата wav.

Чтение wav-файлов. Для считывания wav-файлов в MATLAB используется функция wavread. В простейшем случае она может быть использована следующим образом: y = wavread ('filename'), где 'filename' - имя звукового файла (расширение wav указывать не обязательно). В имя файла необходимо включить полный путь, за исключением тех случаев, когда файл находится в текущем (для MATLAB) каталоге или в одном из каталогов, входящих в список поиска MATLAB.

Другой способ, не требующий указания имени файла, – полный путь, который заключается в определении местонахождения файла на жестком диске с помощью меню MATLAB.

В результате вызова функции в переменную y будет помещено все содержимое указанного файла. Строки матрицы y соответствуют отсчетам сигнала, столбцы - каналам, которых в wav-файле может быть один (моноканал) или два (стереоканал).

Помимо отсчетов сигнала в wav-файлах хранится и служебная информация, которая содержит следующие параметры:

– частоту дискретизации, для определения которой в указанную функцию необходимо включить второй выходной параметр:

[y, Fs] = wavread ('filename'),

где Fs – частота дискретизации, Гц;

– число бит на отсчет, для определения которого необходимо добавить еще один выходной параметр:

[y, Fs, bils] = wavread ('filename');

– число отсчетов и каналов записи. Для получения данной информации необходимо вызвать функцию wavread с двумя входными параметрами: именем файла и текстовой строкой 'size':

wavesize = wavread ('filename*, 'size').

При вызове такой функции из wav-файла извлекается служебная информация, которая возвращается в виде двухэлементного вектор-строки, первый элемент которого содержит число отсчетов, второй - число каналов;

– продолжительность звучания сигнала (в секундах), которую можно определить следующим образом:

wavesize (1) Fs, где 1 указывает на первый параметр вектора wavesize.

Имеются и возможности считывания данных из wav-файла не целиком, а отдельными фрагментами. Для этого используется второй входной параметр

функции wavread. Если этот параметр является числом, будет считано соответствующее количество отсчетов, начиная с первого:

y = wavread ('filename', N).

Если нужный фрагмент расположен не в начале файла, придется указать его начало и конец:

y = wavread ('filename', [nl, n2]).

В результате в переменную y будут считаны отсчеты с номерами от n1 до n2 включительно (нумерация отсчетов начинается с единицы).

Чтобы узнать объем памяти (в килобайтах), требуемый в MATLAB для хранения записи, необходимо использовать следующую функцию:

prod (wavesize)*8/1024.

Для просмотра речевого (звукового) сигнала выведем его в виде графика с помощью следующей функции: plot (y).

Если необходимо вывести график по каналам стереозаписи, то применяют следующие функции:

subplot (2, 1, 1); plot (:, 1); subplot (2, 1, 2); plot (:, 2) или просто plot(y).

Если сигнал имеет большую длину, то можно использовать следующую функцию (фрагменты выводятся друг иод другом): strips (x, N), где x - вектор отсчетов сигнала (двумерный массив не допускается), N – число отсчетов в каждом фрагменте (этот параметр можно опустить, по умолчанию размер фрагмента составляет 200 отсчетов).

Запись wav-файлов. Для записи вектора (или матрицы) на диск в виде wav-файла используется функция wavwrite (y, Fs, N, 'filename'), где y – записываемые данные, Fs - частота дискретизации, Гц, N – число бит на отсчет (8 или 16), 'filename' - имя создаваемого файла. Параметры Fs и N можно опускать, при этом используются значения по умолчанию:

Fs = 8 000 Гц, N= 16.

Записываемые данные должны быть вещественными и лежать в диапазоне от –1 до 1. Значения, выходящие из этого диапазона, будут обрезаны и сделаны равными.

Воспроизведение звуковых файлов. Помимо работы с wav-файлами можно воспроизводить вектор и матрицу в звуковом в виде с использованием следующих функций:

– sound, синтаксис которой записывается следующим образом: sound (y, Fs, bits), где y – вектор или двухстолбцовая матрица сигнала, Fs – частота дискретизации, Гц, bits - число бит на отсчет (8 или 16).

Параметры Fs и bits можно опускать, при этом их значения будут приниматься по умолчанию.

Выходных параметров у у функции нет. После вызова она передает вектор y звуковой карты для воспроизведения и сразу же, не дожидаясь окончания звука, возвращает управление MATLAB;

– wavplay, синтаксис которой имеет следующий вид: wavplay (y, Fs, 'mode'), где параметр mode управляет режимом воспроизведения, который может принимать два значения:

– 'sync'– синхронный режим, означающий что функция вернет управление интерпретатору MATLAB только после окончания звука;

– 'async' – асинхронный режим, при котором функция передает данные для воспроизведения звуковым драйверам Windows и сразу же возвращает управление системе MATLAB. не дожидаясь окончания звука.

Параметры Fs и mode можно опускать, при этом их значения принимаются по умолчанию: Fs =11025 Гц и 'mode' = 'async'.

Запись звука (речи). Функция wavrecord позволяет записать звук в переменную MATLAB с помощью звуковой карты компьютера:

y = wavrecord (n, Fs, ch, 'dtype'),

где n - число записываемых отсчетов, Fs - частота дискретизации, Гц. ch –число каналов записи, 'dtype' – тип записываемых данных.

Возвращаемый результат – матрица, каждый столбец которой соответствует одному каналу записи. При стереозаписи первый столбец – левый канал, второй – правый канал.

Для параметра dtype возможны следующие значения:

– 'double’ - 16-битная запись, данные масштабируются к диапазону от -1 до 1 и представляются в восьмибайтовом формате с плавающей запятой;

– 'single' - 16-битная запись, данные масштабируются к диапазону -1...1 и представляются в четырехбайтовом формате с плавающей запятой;

– ‘untl6’ - 16-битная запись, данные представляются в двухбайтовом целочисленном формате (диапазон от-32 768 до 32 767);

– 'unint8' - 8-битная запись, данные представляются в однобайтовом беззнаковом целочисленном формате (диапазон от 0 до 255, нулевому напряжению на входе соответствует значение «128»).

Входные параметры Fs, ch, dtype можно опускать, при этом их значения будут приниматься, но умолчанию: Fs =11 025 Гц, ch =1, dtype ='double’

3.1.2 Предварительная обработка сигнала

Предварительная обработка сигнала, поступающего на вход системе идентификации диктора, начинается с удаления участков тишины – пауз до и после записи, а также между словами.

На рисунке 11 представлен участок сигнала длительностью около 2 секунд, на рисунке 3 – этот же сигнал после исключения участков тишины.

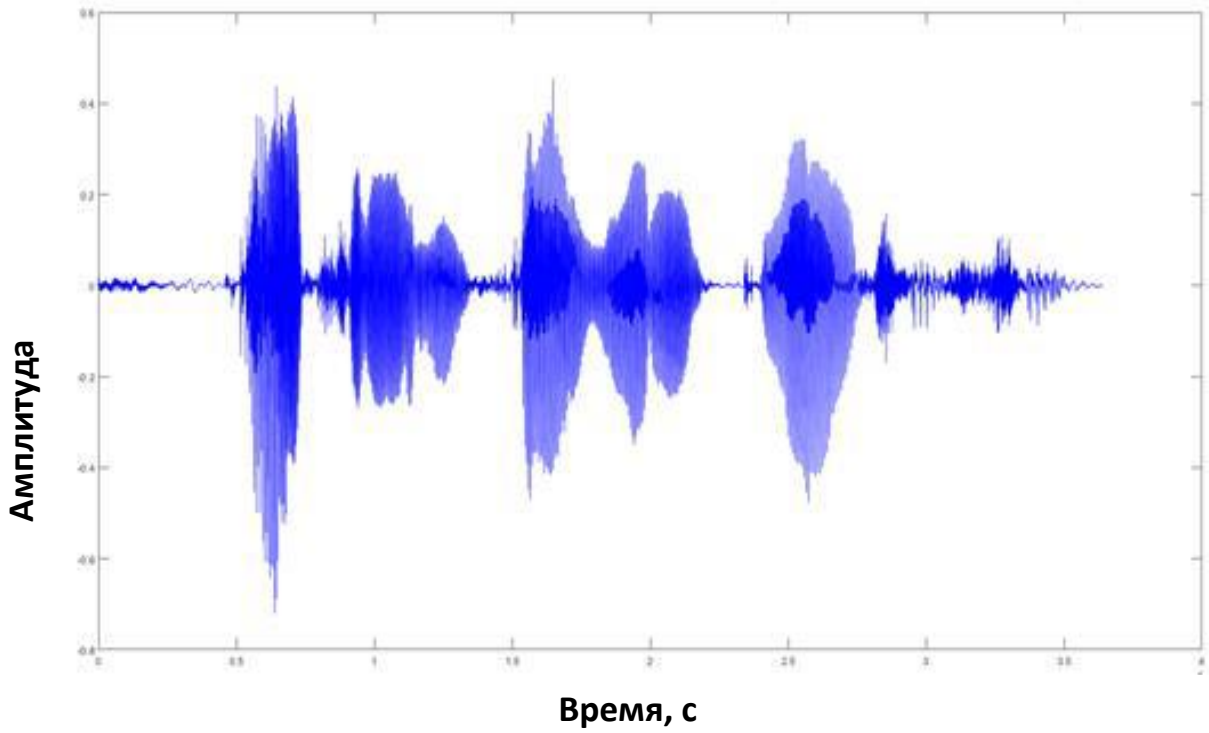

Рисунок 11 – Речевой сигнал, поступающий на вход системе идентификации

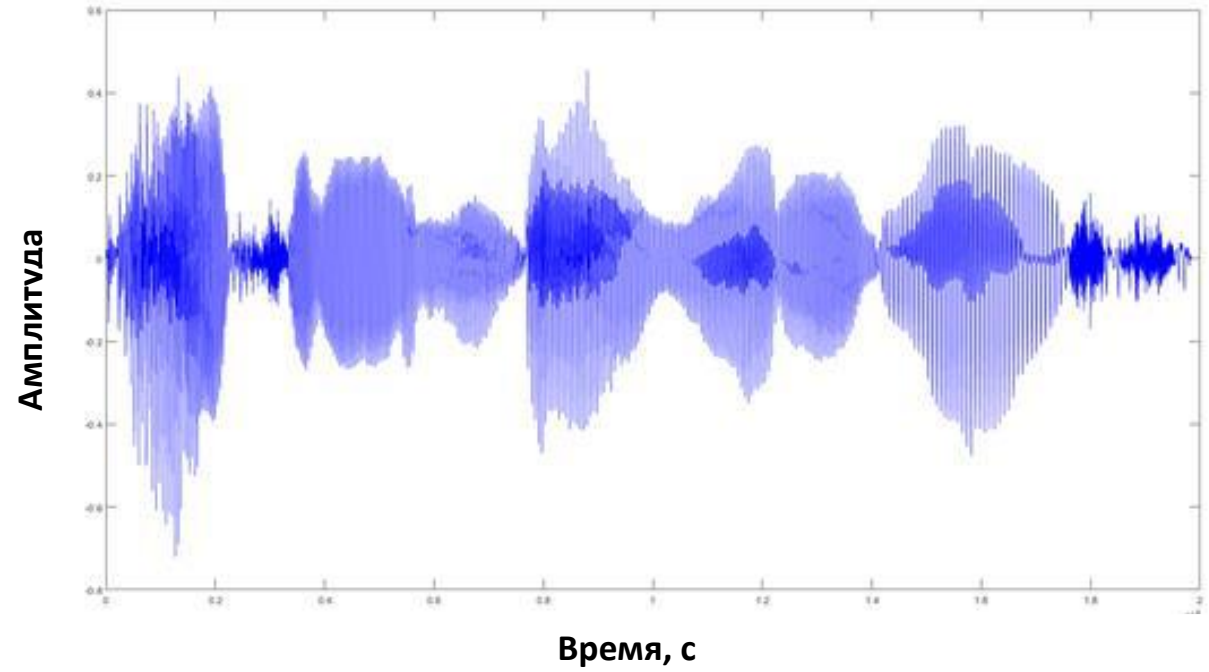

Рисунок 12 – Речевой сигнал после удаления участков тишины

С этой целью в среде MATLAB была написана функция PauseDelete, принимающая на вход значения амплитуды речевого сигнала и частоту его дискретизации:

function [ResSig] = PauseDelete(s,fs).

Возвращаемое значение – результирующий сигнал, из которого исключены участки пауз (тишины). Удаление участков тишины осуществлялось методом выделения границ речевого сигнала на основе нормального распределения.

Следующий этап – нормализация сигнала по амплитуде с применением КИХ-фильтра. На рисунке 13 синим цветом отмечен исходный сигнал, красным – нормализованный по амплитуде.

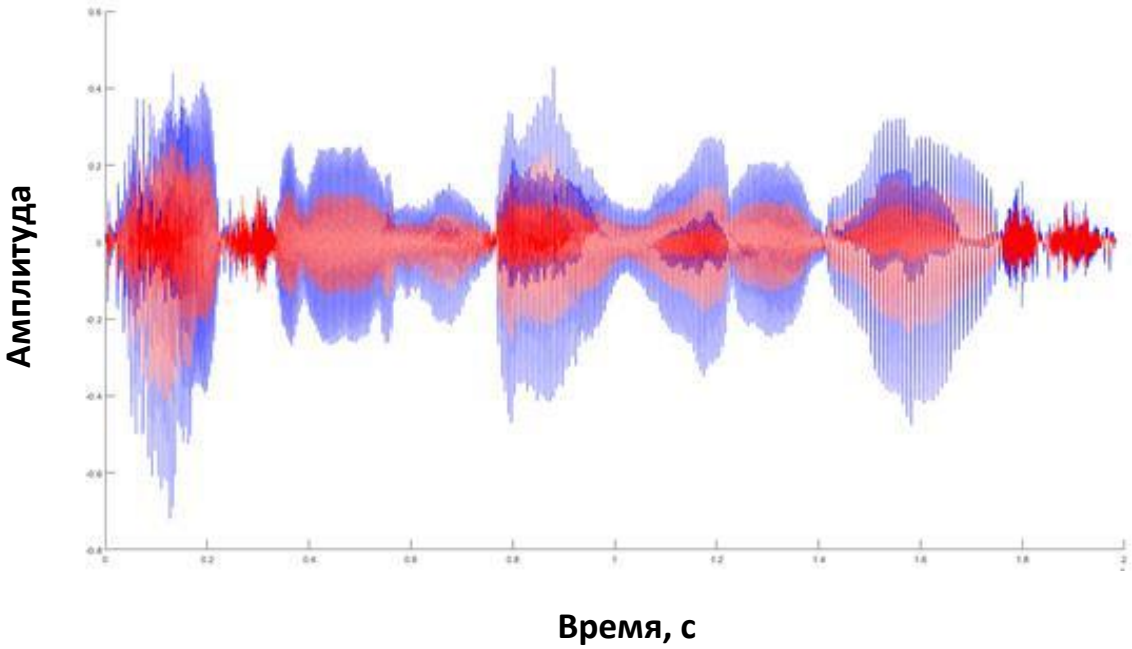

Рисунок 13 – Нормализованный сигнал

3.1.3 Извлечение мел-кепстральных коэффициентов

Под этапом обучения понимается создание кодовой книги каждого диктора с использованием его обучающих акустических векторов. Для выделения таких векторов выбран алгоритм MFCC.

Для получения мел-кепстральных коэффициентов в среде MATLAB была написана функция mfcc, входными данными которой являются речевой сигнал и

частота дискретизации, выходными – массив полученных мел-кепстральных коэффициентов:

function c = mfcc(s, fs).

Далее будут продемонстрированы некоторые промежуточные результаты работы данной функции.

Вначале функция выполняет разбиение сигнала на *L* кадров (фреймов) по *N*=512 отсчетов, что соответствует 32 мс при частоте дискретизации 16 кГц с перекрытием по *M*=256 отсчетов как показано на рисунке 14.

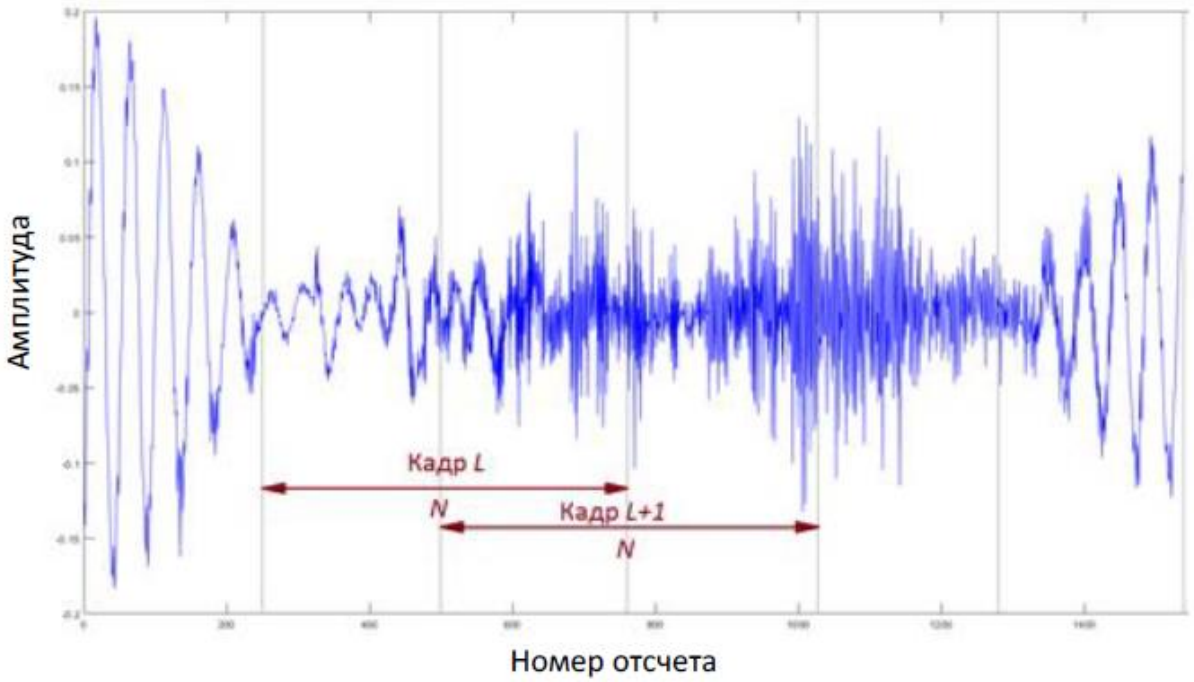

Рисунок 14 – Покадровое разделение речевого сигнала с перекрытием

Для минимизации спектральных искажений выполняется умножение каждого фрейма на оконную функцию Хэмминга. Результат такого умножения для единичного фрейма представлен на рисунках 15,16.

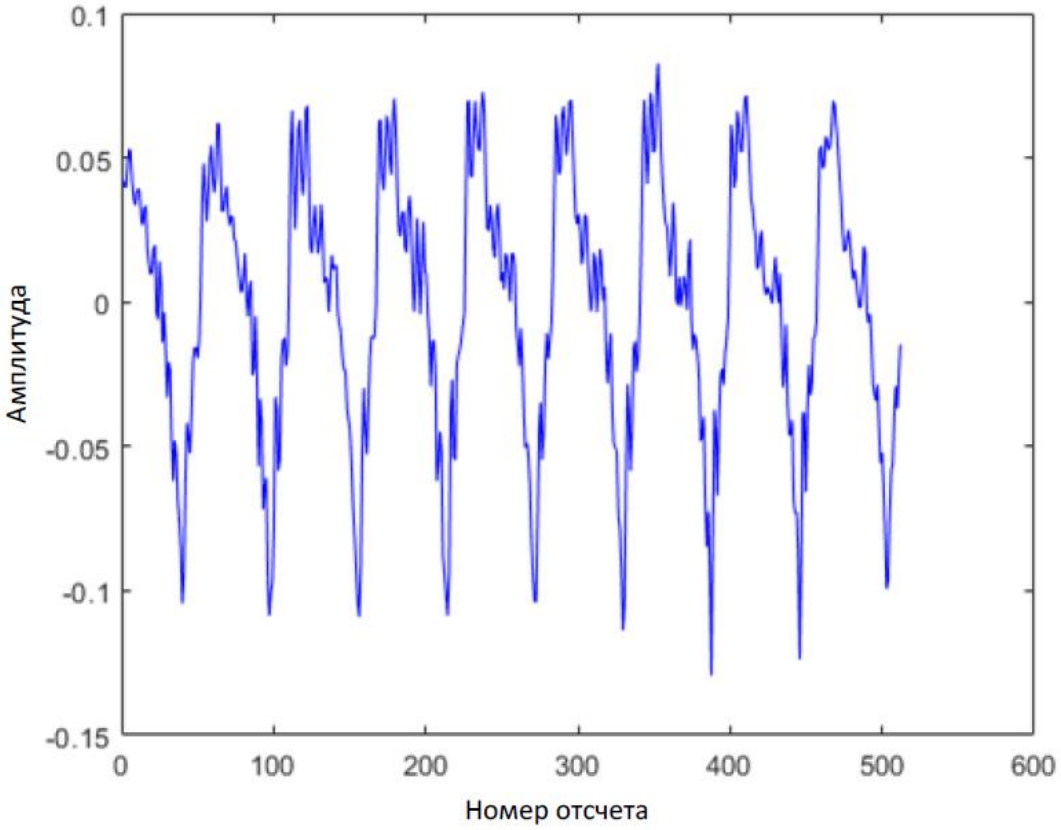

Рисунок 15 – Кадр сигнала до умножения на окно Хэмминга

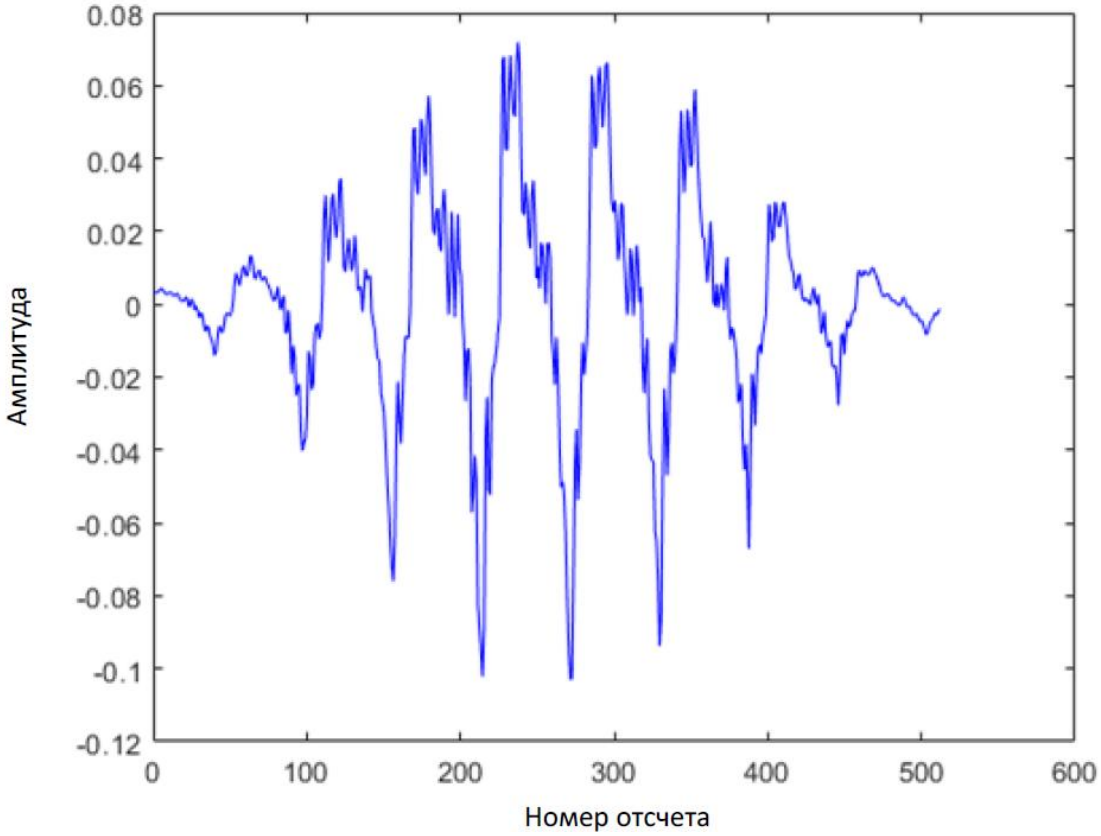

Рисунок 16 – Кадр сигнала после умножения на окно Хэмминга

Следующим этапом в теле функции mfcc для каждого фрейма выполняется ДПФ. Спектр сигнала (кадра), длительностью N=512 отсчетов, представлен на рисунке 8. На графике отображена только первая половина коэффициентов ДПФ, так как набор коэффициентов ДПФ симметричен относительно позиции с индексом $N/2$.

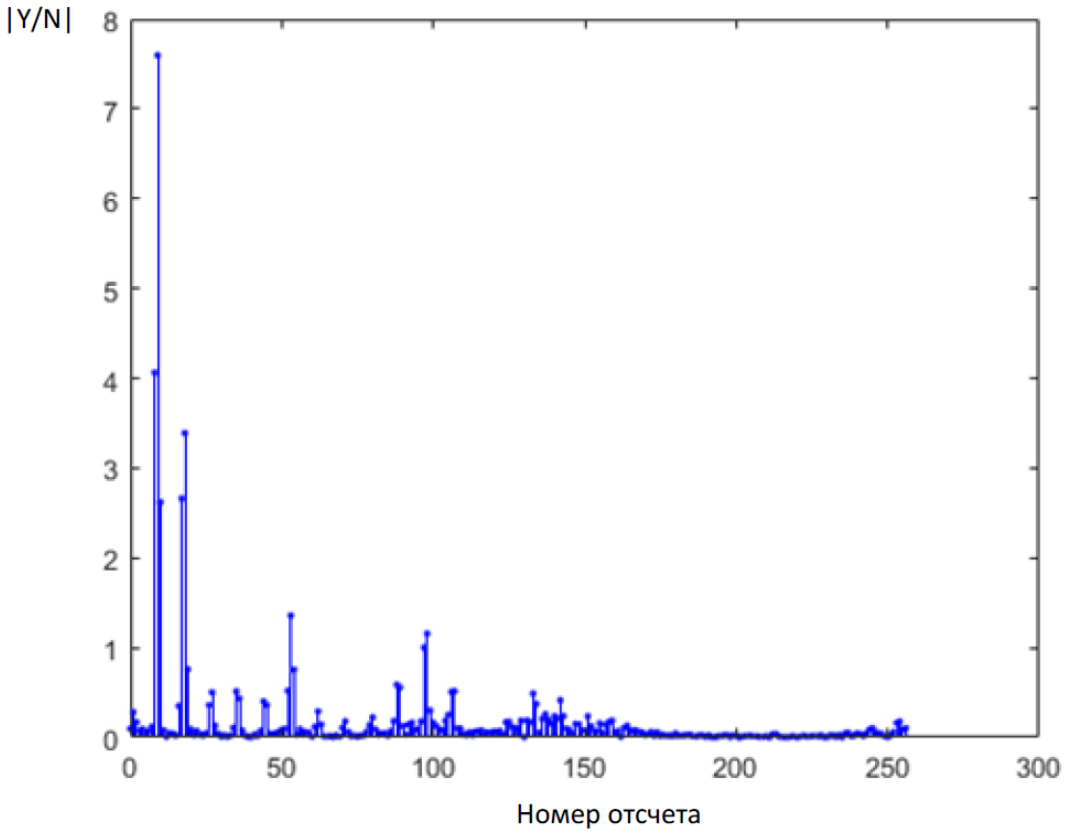

Рисунок 17 – Спектр кадра речевого сигнала

Следующим этапом является наложение гребенки треугольных фильтров. Вычисление весовых коэффициентов фильтра выполняется с использованием функции:

function m = getMelFilterBank(p, n, fs).

Функция принимает на вход 3 параметра: количество фильтров, количество отсчетов спектра сигнала, частота дискретизации. Выходными значениями являются значения весовых коэффициентов фильтров.

После наложения гребенки треугольных фильтров выполняется логарифмирование энергии частотной области, умноженной на весовой коэффициент, затем дискретное косинусное преобразование. В результате имеем 20 мел-кепстральных коэффициентов (Рисунок 18).

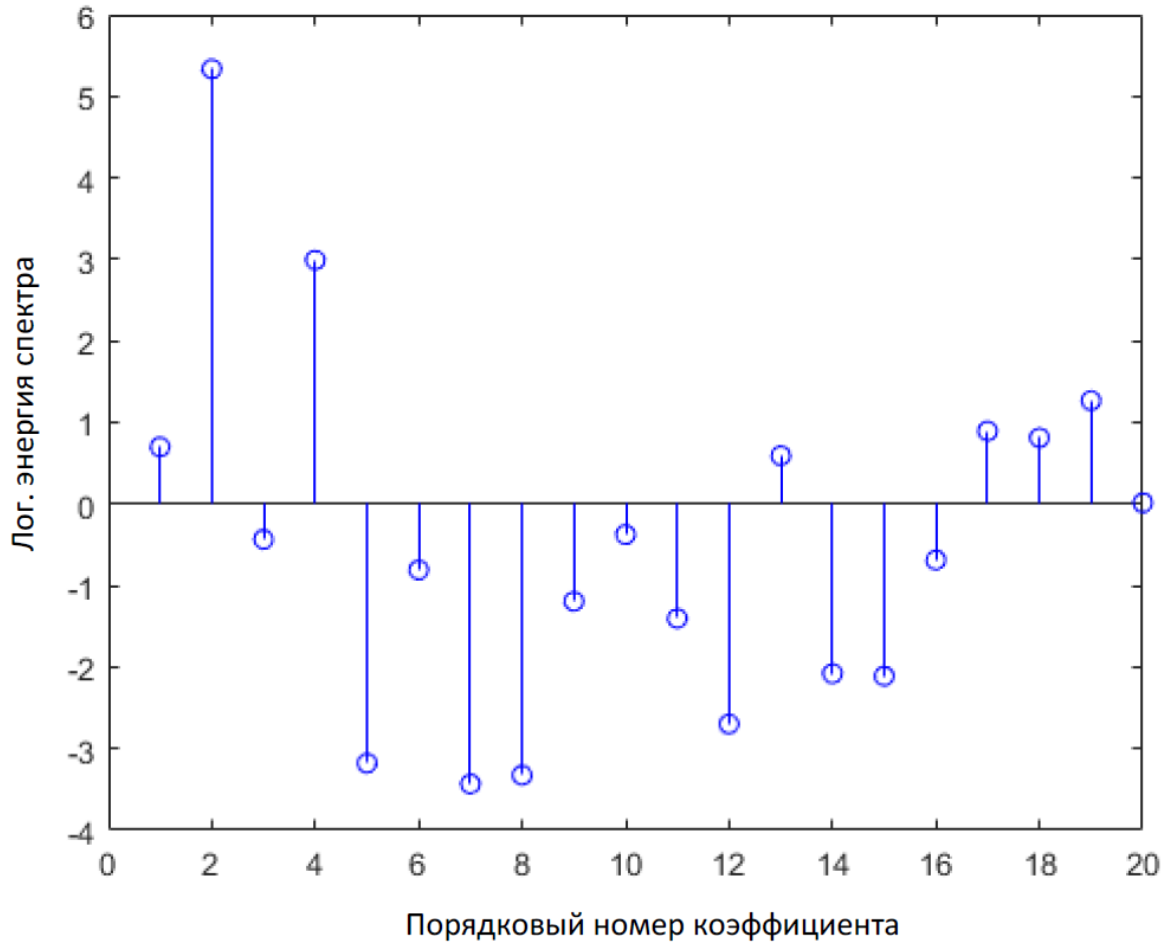

Рисунок 18 – Мел-частотные кепстральные коэффициенты кадра речевого сигнала

Первый полученный коэффициент исключается из результирующего множества мел-кепстральных коэффициентов, так как представляет энергию сигнала и не несет полезной информации. На рисунке 19 показаны значения мел-кепстральных коэффициентов для 4 разных фреймов одного сигнала.

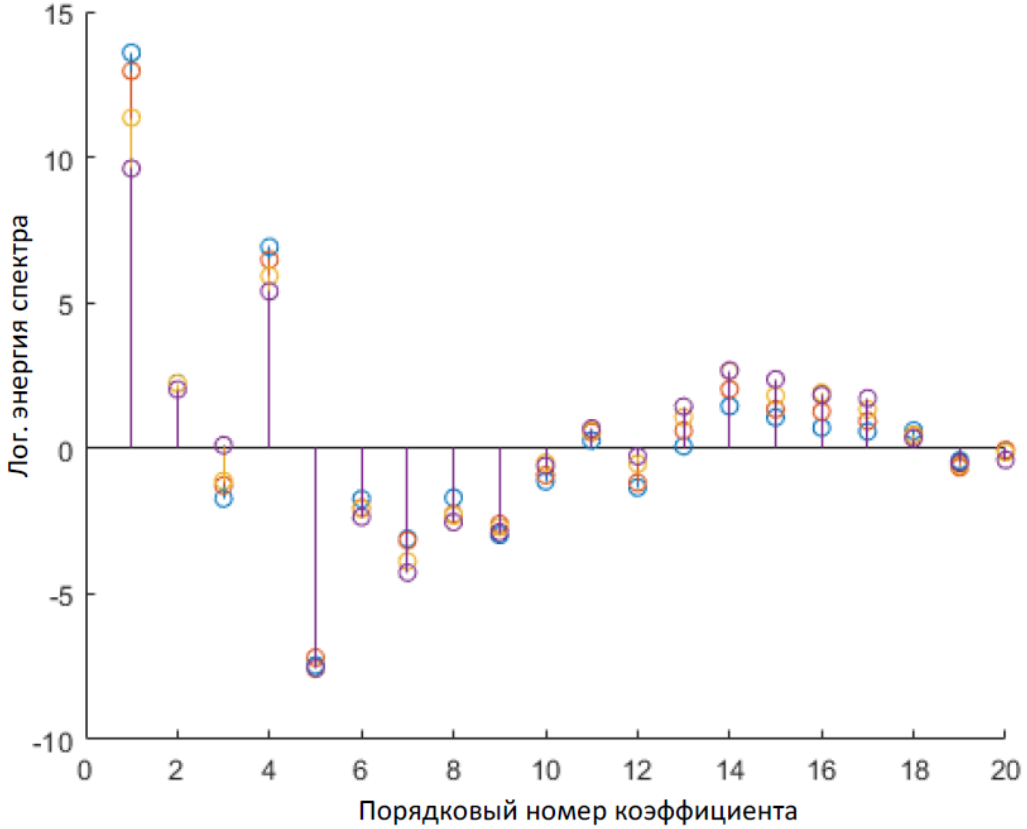

Рисунок 19 – Мел-частотные кепстральные коэффициенты, полученные для 4-х разных кадров одного речевого сигнала

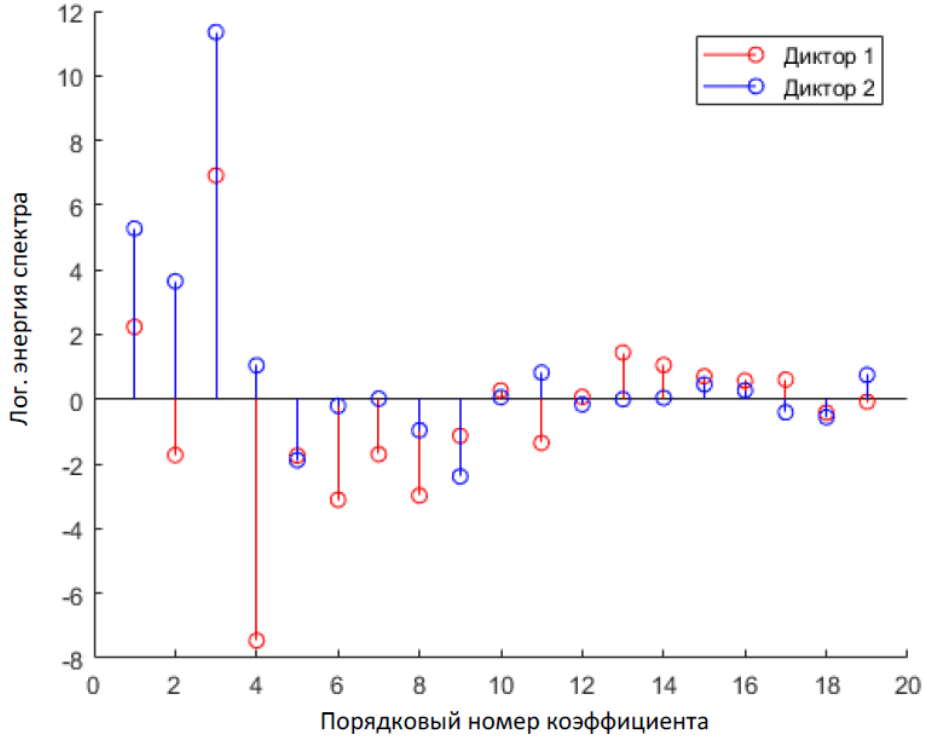

Рисунок 20 – Мел-частотные кепстральные коэффициенты, полученные для кадров речевых сигналов двух разных дикторов

Как видно, значения MFCC-коэффициентов двух разных дикторов различны.

3.1.4 Описание работы программы

Перед началом работы пользователь должен добавить БД и обучить нейронную сеть. Все эти операции осуществляются через графический интерфейс. Также есть возможность управления записью/воспроизведением звука, открытия и сохранения звуковых файлов с помощью диалоговых окон, а также графического отображения речевого сигнала.

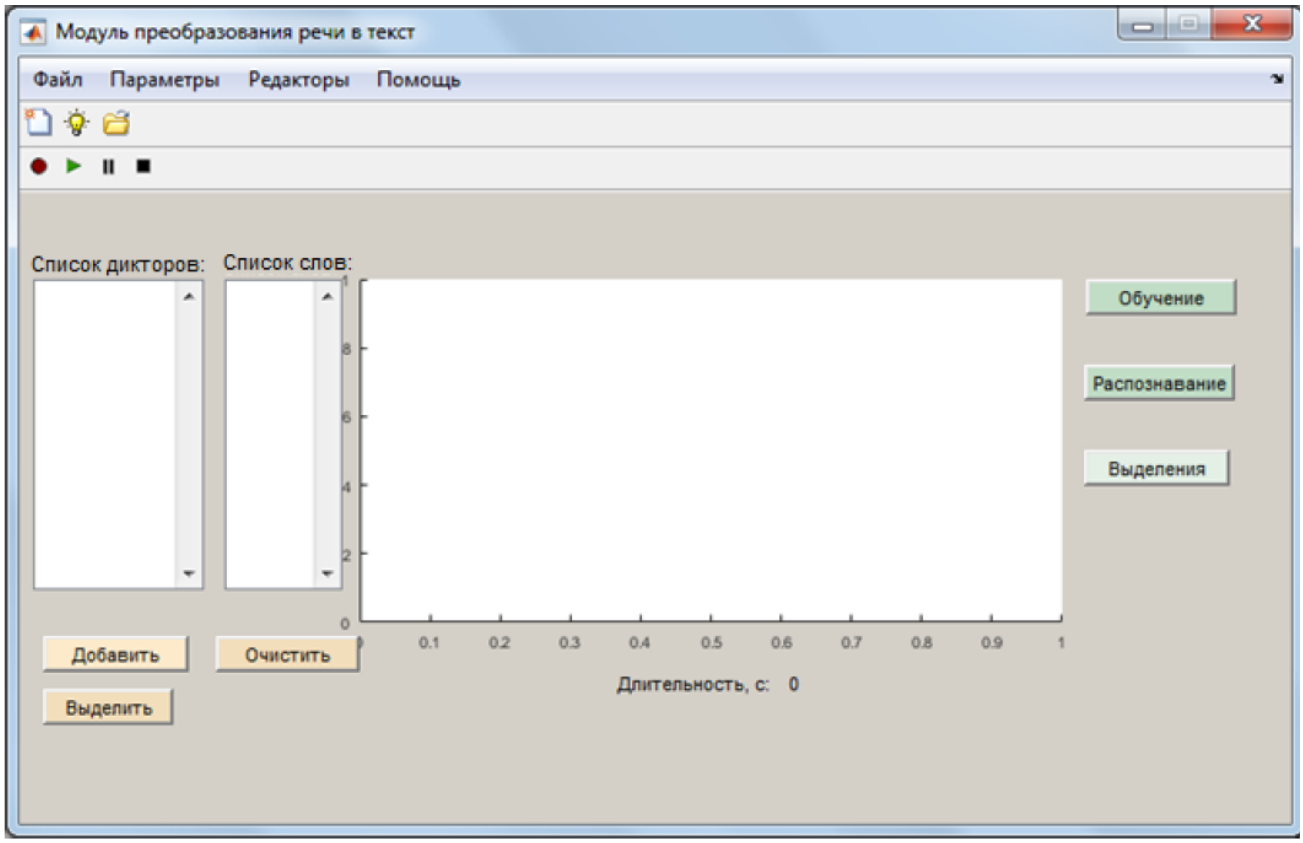

Рисунок 21 – Пользовательский интерфейс программы

На панели управления располагаются кнопки «Play», «Record», «Pause» и «Stop». При помощи которых можно записать и прослушать записанный сигнал. В графической области отображается записанный аудиосигнал или загруженный файл. При нажатии на кнопку «Режим выделения» графическая область поделиться на сектора, для более удобного выделения. Выделить нужный фрагмент из потока аудиосигнала, можно выделив нужный отрезок на графической области и нажать на кнопку «Добавить». Данный сегмент отобразится в списке входных сигналов.

Обучающая выборка формируется с помощью команды «Добавить», путём добавления нужных слов эталонов из «Списка сигналов» в «Список слов» с указанием содержания в диалоговом окне. При этом выполняется предварительная обработка, и рассчитываются признаки выбранных слов.

Сформированную обучающую выборку признаков слов можно сохранить на диск для последующего использования. Сохранение и загрузка выборки выполняется с помощью диалоговых окон, вызываемых через соответствующие команды в меню «Файл» (рисунок 22).

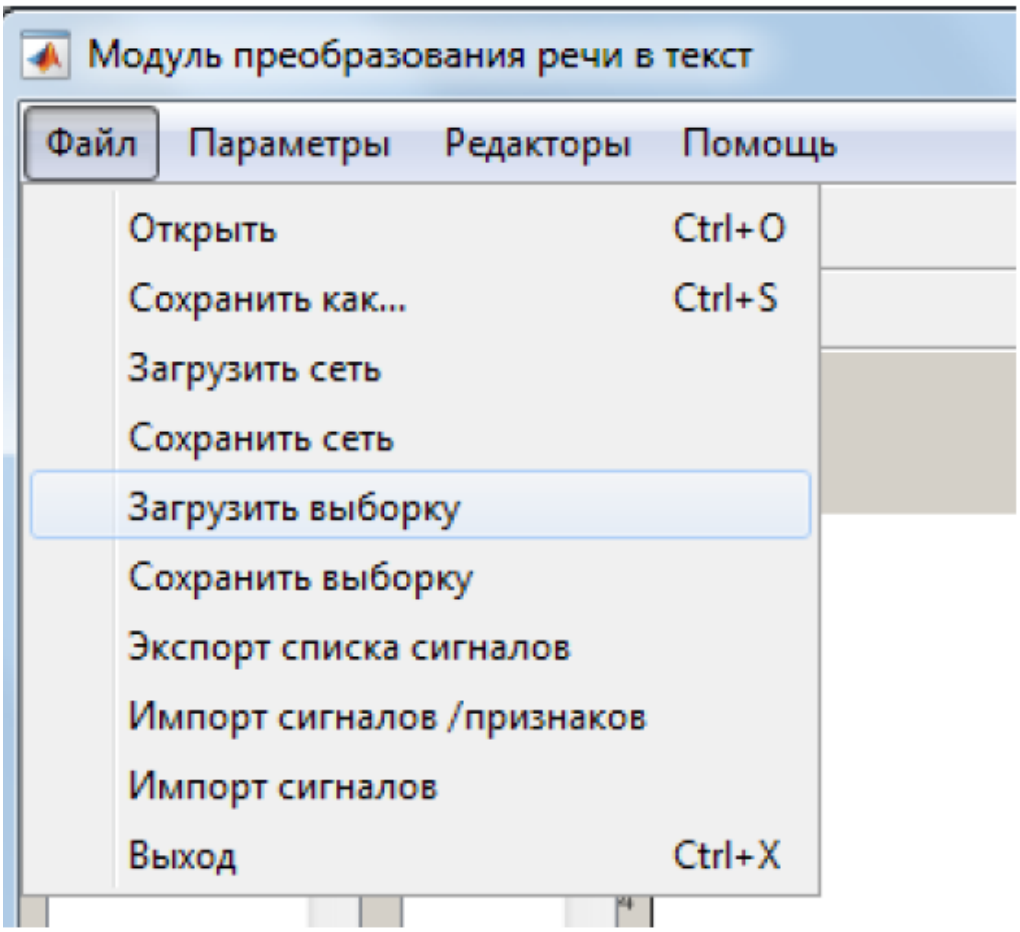

Рисунок 22 – Команды в меню «Файл»

Функция «Обучение» запускает процедуру обучения нейронной сети на сформированном обучающем множестве с заданными параметрами.

Обучение будет направленно на отношение характерных признаков голоса к одному из двух классов:

– Первый класс – мужской голос;

– Второй класс – женский голос.

В дальнейшем можно будет увеличить количество классов для более лучшего результата.

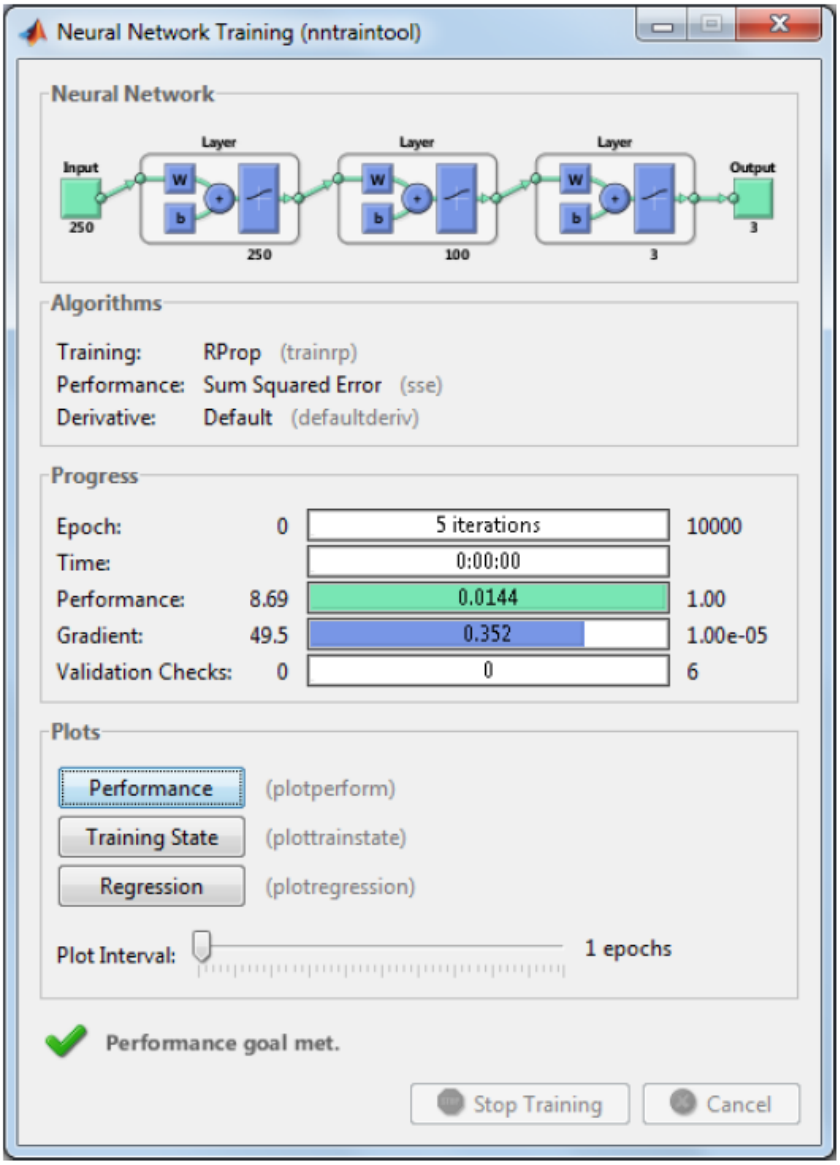

Рисунок 23 – Обучение нейронной сети

В «Neural Network» показано, что нейронная сеть состоит из трёх слоёв (трёхслойный персептрон). Первый слой состоит из 250 нейронов, второй слой из 100 нейронов, 3 слой это количество классов.

Предоставляется возможность сохранить значения весов связей нейронной сети, обученной под конкретного диктора, для дальнейшего использования при

распознавании. Импорт и загрузка сети осуществляется через меню «Файл» (рисунок 22).

Для распознавания необходимо записать аудиосигнал или открыть заранее записанный файл. При нажатии кнопки «Распознать» начнётся процесс распознавания. Загружается уже обученная нейронная сеть, совершается преобразование Гильберта-Хуанга и аудио-сигнал отправляется на нейронную сеть. Там сравниваются характерные признаки и в текстовом окне на главной форме отображается наименование распознанного диктора (рисунок 24).

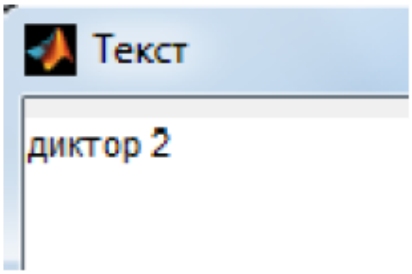

Рисунок 24 – распознанный диктор

3.2 Результаты фактического тестирования программного продукта

Для наполнения речевой базы было привлечено 10 человек. Каждый диктор предоставил аудио-записи пяти ключевых фраз.

Также в тестирование были использованы записи, полученные с помощью синтеза речи. Для решения данной задачи были выбраны системы Google cloud Speech API и Yandex SpeechKit. тем самым, количество распознаваемых голосов в тесте выросло до 20 единиц.

SpeechKit Cloud API – это HTTP API, который позволяет разработчикам приложений использовать речевые технологии Яндекса:

– распознавание речи;
– синтез речи.

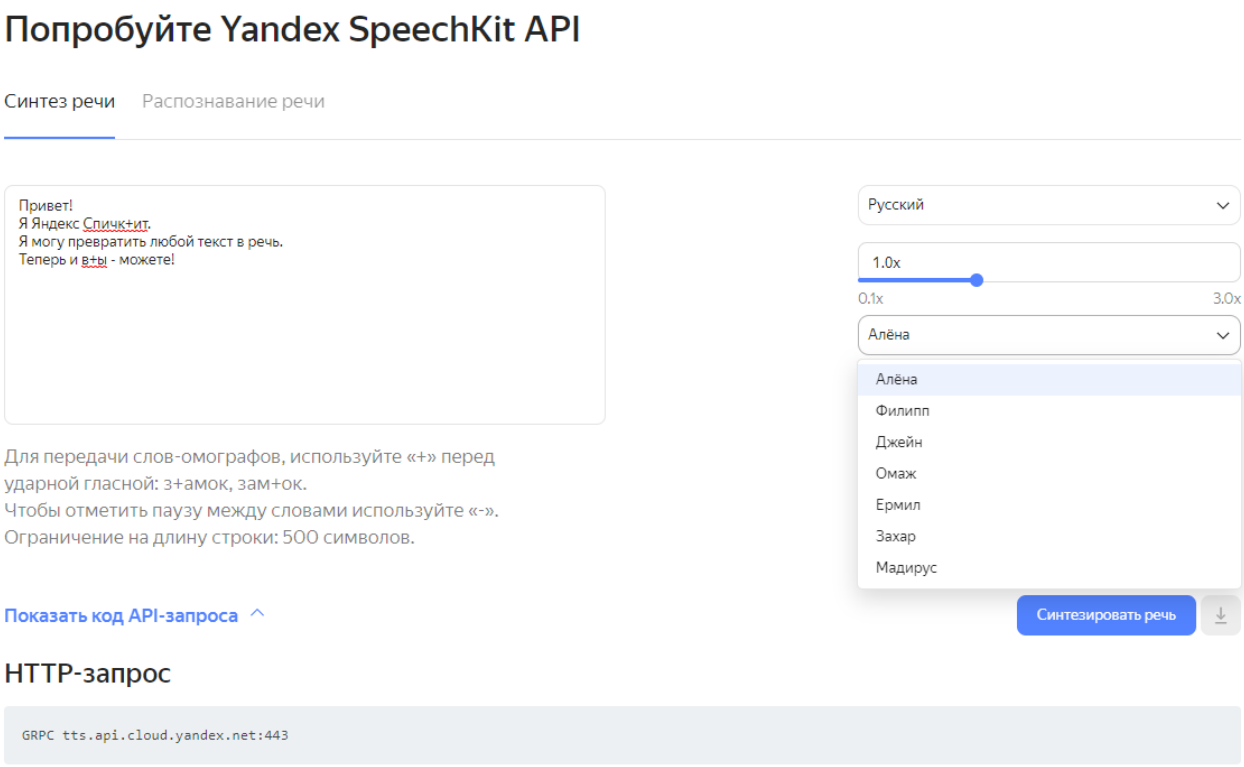

Рисунок 25 – Синтез речи в Yandex SpeechKit

Speech API – интерфейс программирования приложений, основанный на технологии COM (технологический стандарт от компании Microsoft, предназначенный для создания программного обеспечения на основе взаимодействующих компонентов объекта, каждый из которых может использоваться во многих программах одновременно), предназначенный для распознавания и синтеза речи. Speech API предоставил два женских и два мужских голоса, для наполнения голосовой базы.

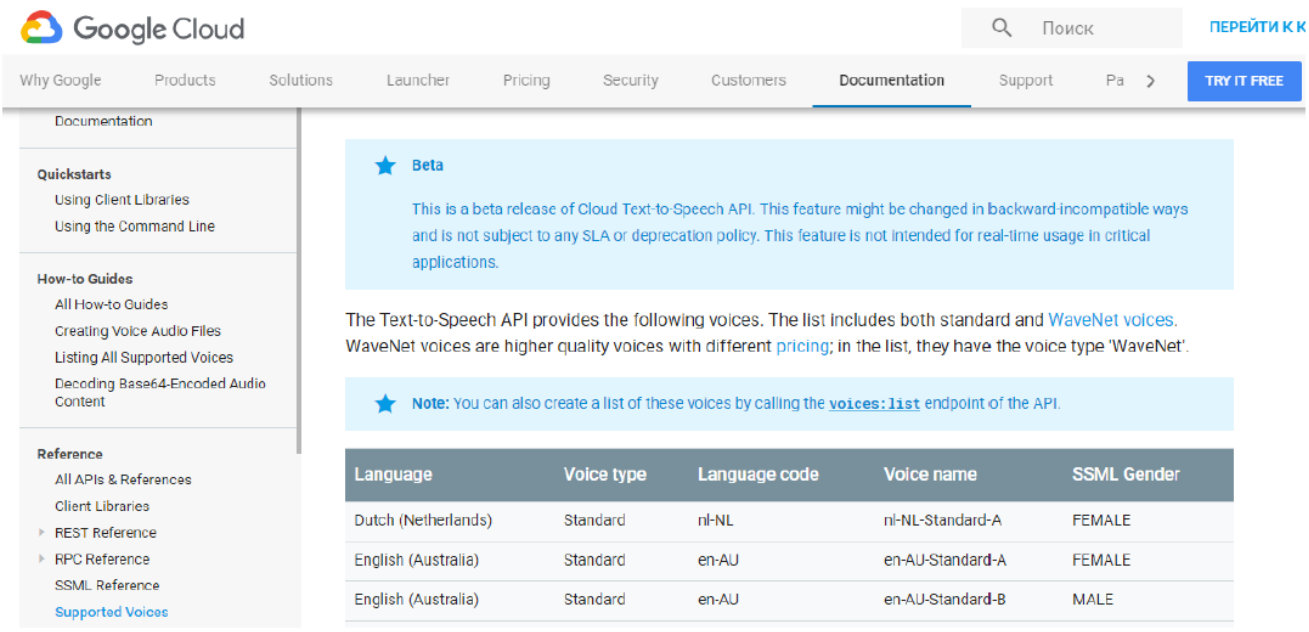

Рисунок 26 – Система Google cloud Speech API

Модуль голосовой идентификации диктора прошёл тестирование, для наблюдения различий было проведено сравнение с работой системы распознавания речи без модуля голосовой идентификации диктора (таблица 2).

Таблица 2 – Оценка качества работы модуля в системе распознавания речи

Количество дикторов	Количество тестов	Коэффициент распознавания без модуля идентификации, %	Коэффициент распознавания с модулем идентификации, %
2	20	95	100
5	50	82	97
10	100	78	94
20	200	65	92

Были проведены тесты с разным количеством задействованных дикторов, относящихся к разным классам. Из проведённых тестов видно, что распознавание диктора с помощью данного модуля в составе системы распознавания речи, повысило процент правильной идентификации диктора.

Данный модуль предназначен для улучшения работы системы распознавания речи. С помощью данного модуля система сможет распознавать дикторов по их характерным признакам голоса, что позволит увеличить надёжность распознавания диктора. Тесты показали, что надёжность распознавания диктора с модулем голосовой идентификации выросла на (20-30) %.

3.3 Анализ достоверности и практической значимости результатов

На сегодняшний день биометрические системы востребованы как никогда, они набирают популярность в самых различных отраслях. Иные способы контроля доступа, к сожалению, не приносят нужных результатов. Проведя анализ существующих программных продуктов, можно сказать следующее, что рынок недостаточно насыщен. Владельцам крупных и средних предприятий крайне необходим жёсткий контроль и отчётность, как в сфере безопасности, так и в сфере работы персонала.

Актуальность темы исследования определяется тем, что рынок речевых технологий стремительно развивается, охватывая практически все сферы нашей жизни, в настоящее время многие ведущие компании усиливают работу в направлении развития голосовых интерфейсов и технологии распознавания речи.

Практическое значение работы состоит в разработке модуля голосовой идентификации диктора. Обработка биометрических данных, в первую очередь, востребована для решения ряда важнейших задач с точки зрения обеспечения высокой безопасности и повышения качества обслуживания. Биометрические технологии в данный момент внедряются в системы контроля и управления доступом в качестве основных или вспомогательных средств идентификации, внедряются в качестве вспомогательных идентификационных технологий в сферу обслуживания и в системы правоохранительных органов.

ЗАКЛЮЧЕНИЕ

В результате выполнения работы были проанализированы существующие подходы и выбран алгоритм классификации пользователей по их голосовым характеристикам. Разработан метод выделения признаков речевого сигнала, позволяющий проводить идентификацию дикторов.

Была выполнена программная реализация модуля голосовой идентификации диктора с использованием среды MATLAB R15b. Определено повышение качества распознавания речи за счёт выбора оптимального классификатора, обученного на речевом материале, с схожими с диктором голосовыми характеристиками. В результате экспериментальных исследований разработанного модуля, отмечено увеличение надёжности распознавания на (20-30) %.

Результаты научно-исследовательской работы были опубликованы в 2 источниках:

– журнал XLVIII международной научно-практической конференции «EurasiaScience»;

– международный научный журнал «Молодой ученый».

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1 Аграновский, А.В. Метод идентификации диктора на основе индивидуальности произношения гласных звуков// Акустика и прикладная лингвистика: Ежегодник РАО. Вып. 3. – М.: 2002 – 115 с.

2 Алимурадов А. К., Чураков П. П. Помехоустойчивый адаптивный алгоритм сегментации «Сигнал/пауза» для систем распознавания речи //Известия высших учебных заведений. Поволжский регион. Технические науки. – 2015. – №. 2 (34).

3 Ануфриев И.Е. MATLAB 7 / И.Е. Ануфриев, А.Б. Смирнов, Е.Н. Смирнова. –СПб.: БХВ–Петербург, 2005. – 1104 с.

4 Ахмад Х.М., В. Ф. Жирков; Владим. Введение в цифровую обработку речевых сигналов: учеб. Пособие \ Гос. ун-т. – Владимир: Изд-во Владим. Гос. ун-та, 2007. – 192 с.

5 Болотский, А. В. Исследование операций и методы оптимизации: учебное пособие / А. В. Болотский, О. А. Кочеткова. – Санкт-Петербург: Лань, 2020. – 116 с.

6 Васильев, Р.А. Исследование особенностей фонетического строя речи и определение национальности диктора при проведении процедуры идентификации по голосу // Информация и безопасность. 2012. – 494 с.

7 Галунов В. И. О возможности определения эмоционального состояния говорящего по речи //Речевые технологии. – 2008 – С. 60−66.

8 Дашкевич, И. В. Использование Вейвлет-преобразования в задаче голосовой идентификации диктора. / И. В. Дашкевич, М. С. Медведев // Международная научно-практическая конференция «Новшества в области технических наук». – Тюмень; Секция №20.

9 Дьяконов, В. MATLAB: Учебный курс // В. Дьяконов. Санкт–Петербург.: Питер, 2001. – 560 с.

10 Исследование алгоритмов обработки сигналов в системе MATLAB: метод, указания к лабораторным работам / Владим. гос. ун-т; сост. Е. К. Левин. - Владимир : Изд-во Владим. гос. ун-та, 2011. – 78 с.

11 Каштанов, П.А. Распознавание речи (технологии, рынок) [Электронный ресурс] – Режим доступа: https://www.tadviser.ru/a/613523 – 29.01.2023.

12 Коваль, С. Л. Комплексная методика идентификации дикторов по голосу и речи // С. Л. Коваль. Информатизация и информационная безопасность правоохранительных органов: труды XX международной научной конференции. Москва.: Академия управления МВД России, 2011. С. 364-370.

13 Ле, Нгуен Виен. Распознавание речи на основе искусственных нейронных сетей / Нгуен Виен Ле, Д. П. Панченко. [Электронный ресурс] – Режим доступа: https://moluch.ru/conf/tech/archive/3/712/– 29.03.2023.

14 Ле Н. В., Панченко Д. П. Подходы к выделению речи из исходного сигнала для системы обработки речи //Молодой ученый. – 2011. – №. 5 – 1. – С. 77−79.

15 С. Макконнел. Совершенный код. Практическое руководство по разработке программного обеспечения / С. Макконнел. – СПб: «Питер», 2005. – 897с.

16 Мартынович, П.А., Свириденко, В.А.. Система верификации диктора для его надежного распознавания через телефонную сеть / П.А. Мартынович,В.А. Свириденко // - М.: Известия вузов, 2016. - с.211-216.

17 Матвеев, Ю. Н. Технологии биометрической идентификации личности по голосу и другим модальностям // Вестник МГТУ им. Н. Э. Баумана. Электронное научно–техническое издание. 2012. № 3 [Электронный ресурс] – Режим доступа: http://vestnik.bmstu.ru/catalog/it/biometric/91.html/ – 02.04.2023.

18 Первушин, Е.А. Обзор основных методов распознавания дикторов [Электронный ресурс] – Режим доступа: http://cyberleninka.ru/article/n/obzor–osnovnyh–metodov–raspoznavaniya–diktorov.pdf. – 22.03.2023.

19 Рахманенко И. А. Программный комплекс для идентификации диктора по голосу с применением параллельных вычислений на центральном и графическом процессорах //Доклады Томского государственного университета систем управления и радиоэлектроники. – 2017. – Т. 20. – №. 1 – С. 70−74.

20 Рзаева, Г. М. Разработка модуля голосовой идентификации пользователя / Г. М. Рзаева. — Текст : непосредственный // «EurasiaScience» XLVIII Международная научно-практическая конференция. — 2022. — № 48. — С. 68-69.

21 Рзаева, Г. М. Разработка модуля голосовой идентификации пользователя / Г. М. Рзаева, С. Г. Самохвалова. — Текст : непосредственный // Молодой ученый. — 2022. — № 43. — С. 18-19.

22 Ромашкин Ю. Н., Петров Ю. О. Распознавание пола диктора на основе GMM-модели голоса //Речевые технологии. – 2009 – С. 31−38.

23 Свириденко, В.А. Речевые технологии в биометрике: верификация и идентификация диктора / В.А. Свириденко // Доклад на конференции «Биометрия–2002» //– М.: 2002.

24 Сергиенко А.Б. Цифровая обработка сигналов: учебное пособие/ А.Б.Сергиенко. –М.: Форум, 2005. – 606 с.

25 Смоленцев, Н. Г. MATLAB. Программирование на C++, C#, Java и VBA / Н. Г. Смоленцев. – Москва: ДМК Пресс, 2015. – 498 с.

26 Сорокин В. Н., Макаров И. С. Определение пола диктора по голосу //Акустический журнал. – 2008. – Т. 54. – №. 4. – С. 659−668.

27 Тампель, И. Б. Автоматическое распознавание речи : учебное пособие / И. Б. Тампель, А. А. Карпов. – Санкт–Петербург : НИУ ИТМО, 2016. –138 с. – Текст : электронный // Лань : электронно–библиотечная система. –Режим доступа: https://e.lanbook.com/book/91466 – 12.04.2023.

28 Трусова. П.В. Введение в математическое моделирование / ред. Трусова. П.В. – Москва, Логос, 2007. – 440 с.

29 Центр речевых технологий [Электронный ресурс]: официальный сайт компании «Центр речевых технологий». Режим доступа: http://www.speechpro.ru – 04.05.2023.

30 Чен К. MATLAB в математических исследованиях / К. Чен, П. Джиблин, А. Ирвинг – М.: Мир, 2011. – 346 с.

31 Чесебиев И.А. Компьютерное распознавание и порождение речи: учебное пособие/ И.А. Чесебиев. – М.: Форум, 2008. - 128 с.

32 Шерхонов, В.С. Система исследования речевых компонентов [Электронный ресурс] – Режим доступа: http://www.stelani.ru/services/uslugi–po–napravleniyu–rechevye–tekhnologii/350/ – 03.04.2023.

ПРИЛОЖЕНИЕ А
ЛИСТИНГ ПРОГРАММЫ

Листинг wav_param.m

```
function [P,segnames] = wav_param(Y,phonames,levelwavelet,typewavelet)
xds=Y'
Fram=800;
Nfr=1; Pt=[];
cca=0;
xds=mapminmax(xds);
for ff=1:fix(length(xds)/Fram)
Nxds=xds(Nfr:Nfr+Fram);
[c,l]=wavedec(Nxds,levelwavelet,typewavelet);
DCELL=detcoef(c,l,'cells');
for kk=1:10
NRG(kk)= sum(abs(DCELL{kk}))';
end
Pt(:,ff)=[NRG]';
Nfr=Nfr+Fram+1;
end
cca=cca+1;
if cca>1
P=[P,Pt];
else P=Pt;
end;
segnames=0;
```

Листинг net_train.m

```
function [net] = net_train(WDS,S1,net_goal)
WDS=getappdata(gcf,'WDS')
alphab='';
count=0; cc=0;
ff=size(WDS,2);
```

```
WDS2=struct();
```

Продолжение ПРИЛОЖЕНИЯ А

```
for ii=1:ff
if ii>1
WDS2(ii-1).word=WDS(ii).word;
WDS2(ii-1).fts=WDS(ii).fts;
end
end;
WDS=WDS2;
ff=ff-1;
for ii=1:ff
cc=WDS(ii).word;
have=0;
for jj=1:size(alphab,1)
l1= alphab(jj,:)
if strcmp(alphab(jj,:),cc);
have=1 ;
end;
end;
if (have==0)
count=count+1;
alphab=strvcat(alphab,cc);
end;
end;
count
V=zeros(count,count);
for ii=1:count
V(ii,ii)=1;
end;
T=zeros(count,length(2));
for ii=1:size(WDS,2)
for jj=1:count
if (strncmp(WDS(ii).word,alphab(jj,:),size(alphab(jj,:),1)))
T(jj,ii)=1;
```

```
end;
```



```
end;
end;
Pm=250;
P=zeros(250,ff);
temp=0; dm=0; ztemp=0;
for ii=1:size(WDS,2)
nframes=size(WDS(ii).fts);
nframes=nframes(2)*10;
temp=reshape(WDS(ii).fts,1,nframes);
temp=temp(:);
dm=Pm-size(temp,1);
temp=[temp;zeros(dm,1)]
P(:,ii)=temp;
end
P pr=minmax(P);
S1= 250;
S2= 100;
S3=count;
net = newff( pr , [S1 S2 S3 ] ,{'logsig' 'logsig' 'logsig'},'trainrp');
net.performFcn = 'sse';
net.trainParam.goal = net_goal;
net.trainParam.show = 20;
net.trainParam.epochs = 10000;
net.trainParam.mc = 0.95;
net = init(net);
[net,tr] = train(net,P,T);
net.userdata=alphab
```

Листинг recognize.m

```
function [phonems] = recogniz(masW,fs,net,diction,levelwavelet,typewavelet)
cnt=1;
Fram=800;
Nfr=1;
```

```
alphabet=net.userdata;
P=[]; Pm=250;
xds=masW';
Ln=length(masW');
xds=mapminmax(xds);
while Nfr<=(Ln-Fram)
Nxds=xds(Nfr:Nfr+Fram);
[c,l]=wavedec(Nxds,levelwavelet,typewavelet);
DCELL2=detcoef(c,l,'cells');
Nfr=Nfr+Fram+1;
for kk=1:10
NRG2(kk,1)= sum(abs(DCELL2{kk}))';
end
NRG2;
P=[P;NRG2];
end
dm=Pm-size(P,1);
P=[P;zeros(dm,1)]
Q(:,cnt)=sim(net,P)
[M,ind] = max(Q(:,cnt)');
plot(P)
outword=alphabet(ind,:)
H=text_out;
data = guidata(gcf);
set(data.edit1,'string',outword);
end
```

Листинг wav.m

```
function varargout = wav(varargin)
if nargin == 0 % LAUNCH GUI
fig = openfig(mfilename,'reuse');
mysqldb.server = 'localhost';
mysqldb.user = 'root';
```

```
mysqldb.password = '';
```

Продолжение ПРИЛОЖЕНИЯ А

```
mysqldb.db = 'rech';
global polzovatel;
polzovatel = 'nobody';
setappdata(fig, 'server', mysqldb.server );
setappdata(fig, 'user', mysqldb.user );
setappdata(fig, 'password', mysqldb.password);
setappdata(fig, 'sqldb', mysqldb.db);
Y=0; fs=44100; nbits=16;
struct.inPoint=0;
struct.outPoint=1;
vops={10,'db8'};
fops={2048,'hamming',0.75};
net_opt=[20 , 1];  % инициализация данных приложения
setappdata(fig,'sampler',Y); % текущий обрабатываемый сигнал
setappdata(fig, 'fd',fs);
setappdata(fig, 'nbits',nbits); % разрядность
setappdata(fig, 'theAudioRecorder',audiorecorder(fs,nbits,1)); % объект записи
setappdata(fig, 'theAudioPlayer',audioplayer(Y,fs)); % объект воспр.
setappdata(fig,'audioSelection',struct);
setappdata(fig,'samples',struct);
segments=0;
samples=struct();
setappdata(fig,'segments',segments);% сегменты обрабатываемого сигнала
setappdata(fig,'fft_opt',fops); % ячейка с параметрами
setappdata(fig,'net',Y); % нейросеть
traindata=0;
setappdata(fig,'traindata',traindata); %обучающая выборка
phonames='';
setappdata(fig,'phonames',phonames);
setappdata(fig,'net_opt',net_opt); % параметры сети
setappdata(fig,'wav_param_opt',vops);
S={};
```

```
setappdata(fig,'S',S); diction={};
```

Продолжение ПРИЛОЖЕНИЯ А

```
setappdata(fig,'diction',diction); %словарь
%отображение панели управления (запись, воспр, пауза, стоп)
[htoolbar, haudiobtns]=render_audiotoolbar(fig);
handles = guihandles(fig);
guidata(fig, handles);
data = guidata(fig);
if nargout > 0
varargout{1} = fig;
end
elseif ischar(varargin{1}) % INVOKE NAMED SUBFUNCTION OR CALLBACK
try
if (nargout)
[varargout{1:nargout}] = feval(varargin{:}); % FEVAL switchyard
else
feval(varargin{:}); % FEVAL switchyard
end
catch
disp(lasterr);
end
end;
function varargout = pushbutton1_Callback(h, eventdata, handles,varargin )
v=0;
% определение количества входных параметров
if nargin > 3
v=varargin{1};
end
if v~=1
v=0;
end % Вызов текущего сигнала
Y=getappdata(gcbf,'sampler');
player=getappdata(gcf,'theAudioPlayer');
FS=get(player,'SampleRate');
```

NBITS=get(player,'BitsPerSample'); % определение установленных параметров

Продолжение ПРИЛОЖЕНИЯ А

```
fops=getappdata(gcbf,'fft_opt');
overlap = fops{3};
nfft=fops{1};
win=fops{2};
overlap = round(overlap*nfft); % Передача входных параметров функции определения частоты основного тона и формантного анализа
[fosn,n]=span(Y,FS,nfft,win,overlap,v);
% ----------------------------------------------------% Вызов функции сегментации
function varargout = pushbutton2_Callback(h, eventdata, handles, varargin)
% Вызов текущего сигнала
Y=getappdata(gcbf,'sampler');
player=getappdata(gcf,'theAudioPlayer');
FS=get(player,'SampleRate');
NBITS=get(player,'BitsPerSample');
% Вызов функции сегментации
[s1,C2] = words(Y,FS,NBITS);
C3(1)={Y}; % отображение полученных сегментов в виде списка
data = guidata(gcf);
name='сегмент';
names='речевой сигнал';
for i=1:s1
names=strvcat(names,strcat(name,int2str(i)));
C3(i+1)=C2(i);
end
set(data.listbox2,'string',names); % сохранение результата сегментации в виде данных приложения
setappdata(gcf,'segments',C3);
% ---------------------------------------------------------------- % Включение функции увеличения и разметки для графика отображения сигнала
function varargout = pushbutton3_Callback(h, eventdata, handles, varargin)
data = guidata(gcbf);
zoom xon;
```

```
grid on; % меню Файл
```

Продолжение ПРИЛОЖЕНИЯ А

```
function File_Callback(hObject, eventdata, handles)
function Open_Callback(hObject, eventdata, handles) % запуск диалогового окна открытия файла
[file,path,filt]=uigetfile('*.wav','wave');
% передача данных файла объекту воспоизведения
[Y,FS,NBITS]=wavread(cat(2,path,file));
rmappdata(gcf,'theAudioPlayer');
setappdata(gcf,'theAudioPlayer',audioplayer(Y, FS, NBITS)); % установка нового значения текущего обрабатываемого сигнала
rmappdata(gcf,'sampler');
setappdata(gcf,'sampler',Y);
C3(1)={Y};
rmappdata(gcf,'segments');
setappdata(gcf,'segments',C3); % отображение сигнала на временном графике
% рассчет длительности сигнала, с
data = guidata(gcf);
set(data.text4,'string',length(Y)/FS);
set(data.listbox2,'value',1);
set(data.listbox2,'string','речевой сигнал');
T=[0:(length(Y)-1)];
T=T'/FS;
plot(T,Y);
% ---------- % меню Параметры
function Options_Callback(hObject, eventdata, handles)
%Вызов меню настройки параметров записи
function Record_Callback(hObject, eventdata, handles)
H = rec_opt;
function SetRecParam(fdis,numbit)
% ---------- %Функция сохранения данных в виде звукового файла *.wav
function Save_as_Callback(hObject, eventdata, handles)
% запуск диалогового окна сохранения файла
```

```
[FILENAME, PATHNAME, FILTERINDEX]=uiputfile('*.wav', 'wav1');
```

Продолжение ПРИЛОЖЕНИЯ А

```
data = guidata(gcf);
Y=getappdata(gcbf,'sampler');
player=getappdata(gcf,'theAudioPlayer');
FS=get(player,'SampleRate')
NBITS=get(player,'BitsPerSample')
wavwrite(Y,FS,NBITS,cat(2,PATHNAME,FILENAME)); % --- Executes during object creation,
after setting all properties.
function edit1_CreateFcn(hObject, eventdata, handles)
if ispc
set(hObject,'BackgroundColor','white');
else
set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end % --- Executes during object creation, after setting all properties.
function listbox2_CreateFcn(hObject, eventdata, handles)
if ispc
set(hObject,'BackgroundColor','white');
else
set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end % --- Executes on selection change in listbox2.
%Функция выбора сегмента из списка----------------------------------
function listbox2_Callback(hObject, eventdata, handles)
n=get(hObject,'Value');
segments=getappdata(gcf,'segments');
player=getappdata(gcf,'theAudioPlayer');
FS=get(player,'SampleRate')
%построение временой диаграмы выбранного сигнала
T=[0:(length(segments{n})-1)];
T=T'/FS;
plot(T,segments{n});
%загрузка выбранного сигнала в объект воспроизведения
NBITS=get(player,'BitsPerSample')
rmappdata(gcf,'theAudioPlayer');
```

setappdata(gcf,'theAudioPlayer',audioplayer(segments{n}, FS, NBITS));

Продолжение ПРИЛОЖЕНИЯ А

```
rmappdata(gcf,'sampler');
setappdata(gcf,'sampler',segments{n});
data = guidata(gcf);
set(data.text4,'string',length(segments{n})/FS); % ------------------------------------------------------------
--------- %Выход из программы
function Exit_Callback(hObject, eventdata, handles)
selection = questdlg(['Закрыть ' get(handles.figure1,'Name') '?'],...
['Закрыть ' get(handles.figure1,'Name') '...'],...
'Да','Нет','Да');
if strcmp(selection,'Нет')
return;
end
close all;
% --------------------------------------------------------------------
function Help_Callback(hObject, eventdata, handles)
function about_Callback(hObject, eventdata, handles)
about; % --------------------------------------------------------------------- % Вызов меню установки па-
раметров БПФ
function fft_set_Callback(hObject, eventdata, handles)
H = fft_opt;
function mysql_set_Callback(hObject, eventdata, handles)
M = mysql_settings;
% Вызов меню установки пользователя
function polzovatel_set_Callback(hObject, eventdata, handles)
U = select_user;
% --- Executes on button press in pushbutton4.
%Запуск формантного анализа
function pushbutton4_Callback(hObject, eventdata, handles)
pushbutton1_Callback(gcbf,0,0,1)
% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
Y=getappdata(gcbf,'sampler');
```

```
player=getappdata(gcf,'theAudioPlayer');
```



```
FS=get(player,'SampleRate');
% определение установленных параметров БПФ
fops=getappdata(gcbf,'fft_opt');
overlap = fops{3};
nfft=fops{1};
win=fops{2};
overlap = round(overlap*nfft);
phonem(Y,FS,nfft,win,overlap);
%--------------------------------------------------------------------
function Import_Net_Callback(hObject, eventdata, handles)
[file,path,filt]=uigetfile('*.mat','Net');
% передача данных файла объекту воспоизведения
nnet=load(cat(2,path,file));
temp = struct2cell(nnet);
nnet=temp{1}
rmappdata(gcf,'net');
setappdata(gcf,'net',nnet);
%-------------------------------------------------------------------- % Сохранение сети
function Export_Net_Callback(hObject, eventdata, handles)
[FILENAME, PATHNAME, FILTERINDEX]=uiputfile('*.mat', 'mat1');
data = guidata(gcf);
NET=getappdata(gcf,'net');
fpath=cat(2,PATHNAME,FILENAME);
save(fpath, 'NET');
%-------------------------------------------------------------------- % распознавание
function pushbutton10_Callback(hObject, eventdata, handles)
Y=getappdata(gcbf,'sampler');
player=getappdata(gcf,'theAudioPlayer');
FS=get(player,'SampleRate');
vops=getappdata(gcf,'wav_param_opt');
levelwavelet = vops{1};
typewavelet = vops{2};
```

```
NET=getappdata(gcf,'net');
```

Продолжение ПРИЛОЖЕНИЯ А

```
diction=getappdata(gcf,'diction');
recogniz(Y,FS,NET,diction,levelwavelet,typewavelet);
%--------------------------------------------------------------------
function listbox3_Callback(hObject, eventdata, handles)
function pushbutton11_Callback(hObject, eventdata, handles)
H= phoname;
%-------------------------------------------------------------------- % Вычисление признаков
function pushbutton8_Callback(hObject, eventdata, handles)
P=getappdata(gcf,'traindata'); % получение обучающей выборки
net_opt=getappdata(gcf,'net_opt')
phonames=getappdata(gcf,'phonames');
data = guidata(gcf);
nnet=net_train(phonames,P,net_opt(1),net_opt(2)); % обучение сети
%close;
rmappdata(gcf,'net');
setappdata(gcf,'net',nnet); % сохранение сети
%-------------------------------------------------------------------- % очистка обучающей выборки
function pushbutton9_Callback(hObject, eventdata, handles)
data = guidata(gcf);
set(data.listbox3,'string','');
rmappdata(gcf,'traindata');
traindata=0;
setappdata(gcf,'traindata',traindata);
%-------------------------------------------------------------------- % Выделение из речевого сигнала
function pushbutton12_Callback(hObject, eventdata, handles)
data = guidata(gcf);
Y=getappdata(gcf,'sampler');
C=getappdata(gcf,'segments');
player=getappdata(gcf,'theAudioPlayer');
FS=get(player,'SampleRate')
names=get(data.listbox2,'string');
time_range=get(data.axes2,'XTick')
```

```
N1=fix(time_range(1)*FS)
```



```
N2=fix(time_range( numel(time_range) )*FS)
ph_sample=Y(N1:N2);
L=numel(C)
C{L+1}=ph_sample;
set(data.listbox2,'string',strvcat(names,'сегмент'));
setappdata(gcf,'segments',C);
%----- Запуск редактора ----------------------------------------
function Phon_edit(hObject, eventdata, handles)
data = guidata(gcf);
S=getappdata(gcf,'S')
H= phon_edit;
data=guidata(gcf);
set(data.uitable1,'data',S);
%----- Подключение сочетаний ----------------------------------
function Phon_open(hObject, eventdata, handles)
S=load('static_rus.mat');
S=getfield(S,'S');
rmappdata(gcf,'S');
setappdata(gcf,'S',S);
H=phon_connect;
%----- Сохранение ----------------
function Phon_save(hObject, eventdata, handles)
data = guidata(gcf);
S=getappdata(gcf,'S');
fpath='static_rus.mat';
save(fpath, 'S');
H=phon_save;
%----- Запуск редактора словаря ------------------------------------------
function Diction_edit(hObject, eventdata, handles)
data = guidata(gcf);
diction=getappdata(gcf,'diction')
H= dict_edit;
```

```
data = guidata(gcf);
```



```
set(data.listbox1,'string',diction);
%----- Подключение словаря сохранение словаря ----------------------------------------
function Dict_open(hObject, eventdata, handles)
[file,path,filt]=uigetfile('*.mat','Dict');
diction=load(cat(2,path,file));
diction=getfield(diction,'diction');
rmappdata(gcf,'diction');
setappdata(gcf,'diction',diction);
%----- Сохранение словаря ------------------------------------------
function Dict_save(hObject, eventdata, handles)
[FILENAME, PATHNAME, FILTERINDEX]=uiputfile('*.mat', 'mat1');
data = guidata(gcf);
diction=getappdata(gcf,'diction');
fpath=cat(2,PATHNAME,FILENAME);
save(fpath, 'diction');
%---------------- Сохранить выборку
function Par_save(hObject, eventdata, handles)
[FILENAME, PATHNAME, FILTERINDEX]=uiputfile('*.mat', 'parameters');
data = guidata(gcf);
traindata=getappdata(gcf,'traindata');
phonames=getappdata(gcf,'phonames');
params{1}=traindata;
params{2}=phonames;
fpath=cat(2,PATHNAME,FILENAME);
save(fpath, 'params');
%------------------- Загрузить выборку
function Par_open(hObject, eventdata, handles)
[file,path,filt]=uigetfile('*.mat','Parameters');
data = guidata(gcf);
params=load(cat(2,path,file));
params=getfield(params,'params');
rmappdata(gcf,'traindata');
```

```
rmappdata(gcf,'phonames');
```

Продолжение ПРИЛОЖЕНИЯ А

```
setappdata(gcf,'traindata',params{1});
setappdata(gcf,'phonames',params{2});
set(data.listbox3,'string',params{2});
%---------------- Экспорт списка сигналов ------------------
function Sample_export(hObject, eventdata, handles)
Y=0;
sname='';
path='';
folder=uigetdir('')
data = guidata(gcf);
samples=getappdata(gcf,'samples')
snum=size(samples,2)
for nn=1:snum
Y=samples(nn).sample;
sname=samples(nn).name;
path=strcat(folder,'\',sname,'_',num2str(nn),'.wav');
wavwrite(Y,22050,16,path);
end
function Sample_import(hObject, eventdata, handles)
Y=0;
sname='';
phname='';
path='';
sample=struct();
names='';
P=0;
folder=uigetdir('');
data = guidata(gcf);
list3names='';
d=dir(cat(2,folder,'\*.wav'));
snum=size(d,1);
for nn=1:snum
```

sname=d(nn).name;


```
phname = regexp(sname, '_', 'split');
path=cat(2,folder,'\',sname);
Y=wavread(path);
samples(nn).name=char(phname(1));
samples(nn).sample=Y;
names=strvcat(names,sname);
C3(nn)={Y};
[Pt,segnames]=wav_param(Y,char(phname(1)),6,'db8');
if P==0
P=Pt;
phonames=segnames;
else
P=[P,Pt];
phonames=strvcat(phonames,segnames);
end;
if length(list3names)==0
list3names=char(phname(1));
else
list3names=strvcat(list3names,char(phname(1)));
end
end
rmappdata(gcf,'traindata');
setappdata(gcf,'traindata',P);
rmappdata(gcf,'phonames');
setappdata(gcf,'phonames',phonames);
set(data.listbox3,'string',list3names);
setappdata(gcf,'samples',samples);
set(data.listbox2,'string',names);
setappdata(gcf,'segments',C3);
function Segment_import(hObject, eventdata, handles)
Y=0;
sname='';
```

```
phname='';
path='';
sample=struct();
names='';
P=0;
folder=uigetdir('');
data = guidata(gcf);
list3names='';
%d=dir(folder)
d=dir(cat(2,folder,'\*.wav'));
snum=size(d,1);
for nn=1:snum
sname=d(nn).name;
phname = regexp(sname, '_', 'split');
path=cat(2,folder,'\',sname);
Y=wavread(path);
samples(nn).name=char(phname(1));
samples(nn).sample=Y;
names=strvcat(names,sname);
C3(nn)={Y};
end;
setappdata(gcf,'samples',samples);
set(data.listbox2,'string',names);
setappdata(gcf,'segments',C3);
function test_Callback(hObject, eventdata, handles)
data = guidata(gcf);
C3=getappdata(gcf,'segments');
phonames=get(data.listbox2,'string');
n_net=getappdata(gcf,'net');
ph_test(C3,phonames,n_net)
%--------------------------------------------------------------
function net_opt(hObject, eventdata, handles)
```

H= net_opt;

Продолжение ПРИЛОЖЕНИЯ А

```
%-----------------Коннектимся к БД----------------------------
function use_mysql_db(hObject, eventdata, handles)
mysql('open', getappdata(gcf, 'server'), getappdata(gcf, 'user'), getappdata(gcf, 'password'));
mysql('use', getappdata(gcf, 'sqldb'));
function close_mysql_db(hObject, eventdata, handles)
mysql('closeall');
function check_query(hObject, eventdata, handles)
global polzovatel;
query = strcat(strcat('SELECT dir,file_name FROM files_words WHERE (user_id = (SELECT id
FROM users WHERE (fio = '',polzovatel), ''))))');
mysql(query);
function download_query(hObject, eventdata, handles)
F = ftp_download;
function wav_param_opt_Callback(hObject, eventdata, handles)
%
H = wav_param_opt;
```