МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное
учреждение высшего образования
«Национальный исследовательский университет «МЭИ»
Институт информационных и вычислительных технологий

Специальность: Прикладная математика и информатика (01.03.02)
Кафедра: ПМИИ

Курсовая работа

по дисциплине

“Программная инженерия”

Тема: “Проектирование программных средств аутентификации пользователей компьютерных систем по голосу”

Выполнил: Шамриков А.И.

Группа: А-05-19

Руководитель: Хорев П.Б.

Москва

2021

Оглавление

Список рисунков

Список таблиц

Введение

Задание:

Разработка прототипа приложения, выполняющего аутентификацию пользователей по их голосу.

Для достижения требуемого результата необходимо:

- изучить методы биометрической аутентификации;
- изучить особенности биометрической характеристики голоса
- программно реализовать выбранный метод, а также его протестировать.

Аутентификация – это процедура проверки подлинности, например проверка принадлежности голоса конкретному пользователю.

Один из методов аутентификации, набирающий популярность, — это биометрическая аутентификация. К этому понятию относятся методы, основанные на физических и поведенческих особенностях человека. К таким методам относятся: распознавание лица, отпечатков пальцев, глаз. Однако в данной работе я рассмотрю метод голосовой биометрии.

Аутентификация пользователей по голосу – это один из методов биометрической аутентификации. Голос каждого человека имеет уникальные особенности, такие как обертона, смолы, каденции и другие. Именно благодаря им мы и имеем возможность аутентифицировать личность человека при помощи голосовой биометрии.

Примеры уже существующих систем, использующих голосовую биометрию: Voice Pay (прототип PayPal), VoiceKey Service, АРМО-Вокс и другие.

Под аутентификацией в этой работе мы будем понимать проверку принадлежности голоса пользователя к его уникальному идентификатору. Так, идентификация и аутентификация являются неразрывно связанными понятиями, определяющими распознавание и проверку подлинности пользователей.

При осуществлении таких технологий мы оцифровываем запись голоса человека и сверяем полученные результаты с записью, полученной при регистрации.

Аутентификация пользователей имеет множество преимуществ, среди которых простота схемы осуществления, включающая в себя микрофон, программное обеспечение и базу данных. Таким образом, метод может быть реализован на базе только ПК с подключенным к нему микрофоном, что делает голосовую аутентификацию одной из самых доступных среди всех биометрических методов. Кроме того, метод прост и понятен для пользователя, для которого запись голоса – это простой процесс, не вызывающий вопросов.

Еще одним крупным преимуществом рассматриваемого метода является его работа на большом расстоянии благодаря существующим средствам передачи

голоса. Так, например, голосовая аутентификация является единственным методом, позволяющим идентифицировать пользователя по телефону.

Если учитывать стремительность развития технологий, а также направленность этих технологий на удобство пользователей, то можно смело сказать, что, даже если и не в ближайшее время, то через несколько лет, все системы отойдут от традиционных печатных паролей в пользу более практичных методов, одним из которых, безусловно, является голосовая аутентификация.

В данной курсовой работе я разберу различные реализации аутентификации пользователей по голосу, а также создам приложение-эмулятор аутентификации пользователей. В нем будет реализован следующий функционал: возможность регистрации новых пользователей, а также эмуляция входа в систему, результатом которой будет сообщение, показывающее, пройдена ли аутентификация пользователем.

Анализ задания.

Проблемы, возникающие при аутентификации личности по голосу.

Для начала разберем негативные стороны голосовой биометрии. Несмотря на то, что уникальность голоса обусловлена множеством различных факторов, ни одна система голосовой аутентификации не может гарантировать 100%-ую аутентификацию. Это связано с тем, что факторов, вызывающих ошибки также очень много. Перечислим некоторые из них:

- Условия записи. Сюда относится уровень и тип шума, а также уровень реверберации. Чем сильнее будут разниться условия при создании карты пользователя и при аутентификации, тем выше шанс ошибки.
- Изменения в голосе человека. Ясно, что голос больного человека сильно отличается от голоса здорового, причем как для человека, так и для компьютерных систем. Кроме того, известно, что тон голоса может меняться в течении жизни, более того, голос человека спросонья может иметь небольшие отличия от голоса бодрого человека. Голос может зависеть даже от настроения.
- Искажения. Влияние на точность аутентификации оказывают амплитудно-частотные характеристики микрофона (на сам микрофон, кстати, могут влиять даже температурные условия), канал передачи, вид кодирования в канале и т.д.

Для того чтобы снизить влияние этих факторов, создают методы и алгоритмы, обрабатывающие входной сигнал:

- Предварительная обработка речевого сигнала. Алгоритмы выделяют участки записи, где звучит голос.

- Сегментация голосов в фонограмме.
- Выделение главного голоса записи и удаление окружающих его шумов.

Различают два рода ошибок метода аутентификации по голосу.

Ошибка 1-го рода (FRR) – ошибка ложного отказа в доступе. Она заключается в том, что зарегистрированный пользователь, которому система должна предоставить доступ, получает отказ.

Ошибка 2-го рода (FAR) – ошибка ложного доступа. Она заключается в том, что система принимает чужого за своего и дает доступ тому, кто доступ получить не должен.

Таким образом, можно грубо назвать ошибку первого рода ошибкой удобства, а ошибку второго рода – ошибкой безопасности.

Метод голосовой аутентификации вообще не является самым безопасным среди биометрических методов. Приведем таблицу значений ошибок различных методов (Таблица 1. Ошибки биометрических методов).

Таблица 1. Ошибки биометрических методов

Биометрический признак	**Тест**	**Условия тестирования**	**FRR, %**	**FAR, %**
Отпечатки пальцев	FVC 2006	Неоднородная популяция (включает работников ручного труда)	2,2	2,2
Лицо	MBE 2010	Полицейская база фотографий База фотографий с документов	4,0 0,3	0,1 0,1
Голос	NIST 2010	Текстонезависимое распознавание	3,4	1,0
Радужная оболочка глаз	ICE 2006	Контролируемое освещение, широкой диапазон качества изображений	1,1 – 1,4	0,1

Как видим, взамен на удобство метода мы получаем посредственную безопасность.

Классификация методов голосовой биометрии.

Распознаватели речи:

- По типу архитектуры:
  - Клиент-серверные
  - Локальные
- По методу выделения признаков:
  - Спектральные анализ. Преобразование Фурье
  - Кодирование коэффициентами Линейного Предсказания (КЛП)
  - Кепстральный анализ
  - Вейвлет анализ
- По типу речи:
  - Распознавание изолированных слов
  - Распознавание слитной речи
- По назначению:
  - Командные системы
  - Системы диктовки текст в речь
  - Системы преобразования речь в речь
  - Системы идентификации и аутентификации
- По механизму функционирования:
  - Простейшие (корреляционные) методы
  - Вероятностно-сетевые модели принятия решений
    - Скрытое Марковское кодирование
    - Динамическое программирование (алгоритм Витерби)
    - Нейросетевой метод
  - ЭС (экспертные системы) с различным способом формирования БЗ (базы знаний)
- По потребительским качествам:
  - Дикторозависимые
  - Дикторонезависимые
- По типу структурной единицы:
  - Аллофон
  - Фонема
  - Дифон, трифон
  - Слово или фраза

Алгоритм работы методов речевой аутентификации.

Алгоритм условно можно разбить на две части: создание карты пользователя и сама процедура аутентификации.

Под созданием карты пользователя подразумевается следующее: создание уникального идентификатора, запись парольной фразы, обработка и оцифровка записи выбранным методом, сохранение пользователя.

Аутентификация, в свою очередь, подразумевает: считывание идентификатора пользователя, запись парольной фразы, обработка и оцифровка записи выбранным методом, сверка результатов работы метода из карты пользователя с введенным идентификатором и вывод ответа системы, полученного на основании сверки.

Для получения ответа системы необходимо задать пороговое значение. После получение результатов работы метода в аутентификации происходит просчет процесса схожести (по значению и относительному положению) фреймов. Если этот процент меньше или равен пороговому значению, то доступ разрешается, если же нет – система отказывает в доступе. Для оценки качества системы аутентификации удобно использовать ERR (Equal Error Rate), критерий, при котором обеспечивается равенство ошибок обеих родов.

Алгоритм аутентификации можно схематично изобразить следующим образом (Рисунок 1. Алгоритм аутентификации).

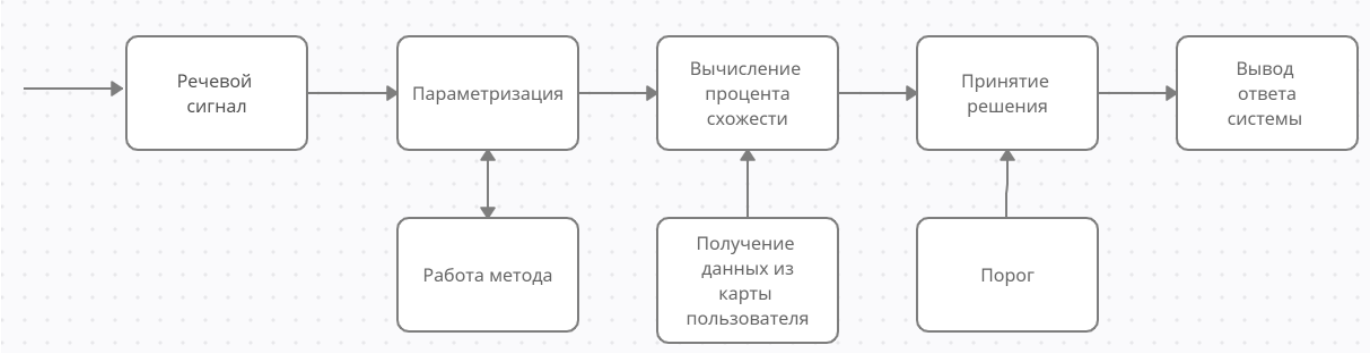


Рисунок 1. Алгоритм аутентификации

Предварительная обработка входного речевого сигнала.

При обработке речевого сигнала в первую очередь необходимо выделить из всего входного сигнала речевые фрагменты, убрав из записи все паузы и зашумленные участки, а также моменты тишины в начале и конце записи. Помехами, которые необходимо удалить, являются различные щелчки, гудки, DTMF-сигналы.

На Рисунок 2. Схема обработка входного сигнала. представлена схема обработки речевого сигнала.

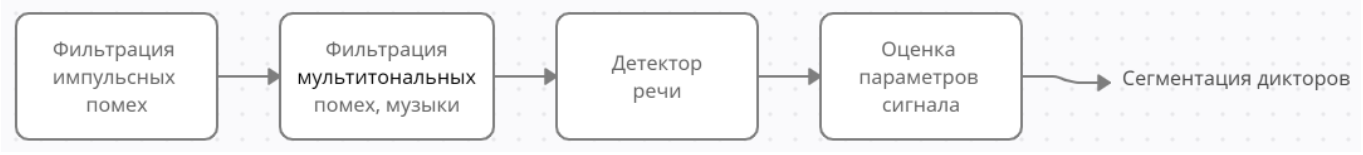


Рисунок 2. Схема обработка входного сигнала.

Важнейшим элементом обработки является детектор речевой активности (VAD). При разработке его алгоритма внимание уделяется выделению признаков, отвечающих требованию шумоустойчивости, и выбору правил классификации речь – не речь. Обычно используют алгоритмы на основе анализа энергии сигнала, спектрального и кепстрального анализа, обнаружения основного тона.

В ЦРТ, например, создан VAD, выделяющие вокализованные участки. Метод его работы заключается в выделении гласных и назализованных согласных. Таким образом алгоритм удаляет некоторые согласные, играющие наименьшую роль в аутентификации. Конечно, при этом удаляется и малые части важных участков, но их удаление компенсируется тем, что в записи остается только значимые для работы системы участки.

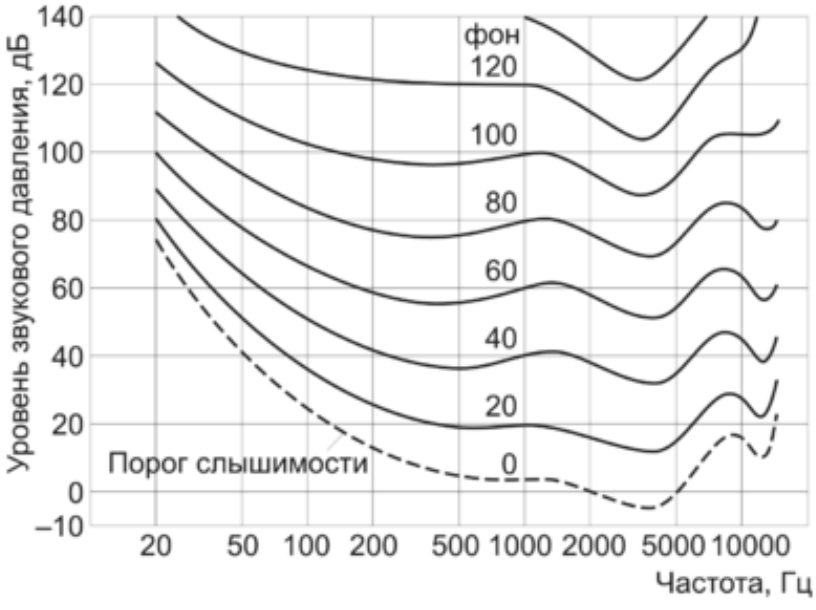
Обработка входного сигнала – это грандиозная работа, объем которой оценить довольно сложно.

Проектирование

Описание метода работы приложения.

При разработке приложения я буду использовать метод мел-частотных кепстральных коэффициентов.

Мел – это единица высоты звука. Однако высоты не фактической. Дело в том, что высота звука, при восприятии человеком, зависит не только от его частоты, сюда еще добавляются такие факторы, как громкость звука и тембр.



Так как АЧХ (амплитудно-частотная характеристика) человека совершенна не похожа на прямую, ввели подобранные опытным путем единицы громкости, например, фон.

Рисунок 3. Семейство кривых равных громкостей для разных уровней звукового давления

Точно также и тон звука воспринимается человеком нелинейно.

На Рисунок 4. Зависимость мел от частоты приведен график зависимости мел-шкалы от частоты звука.

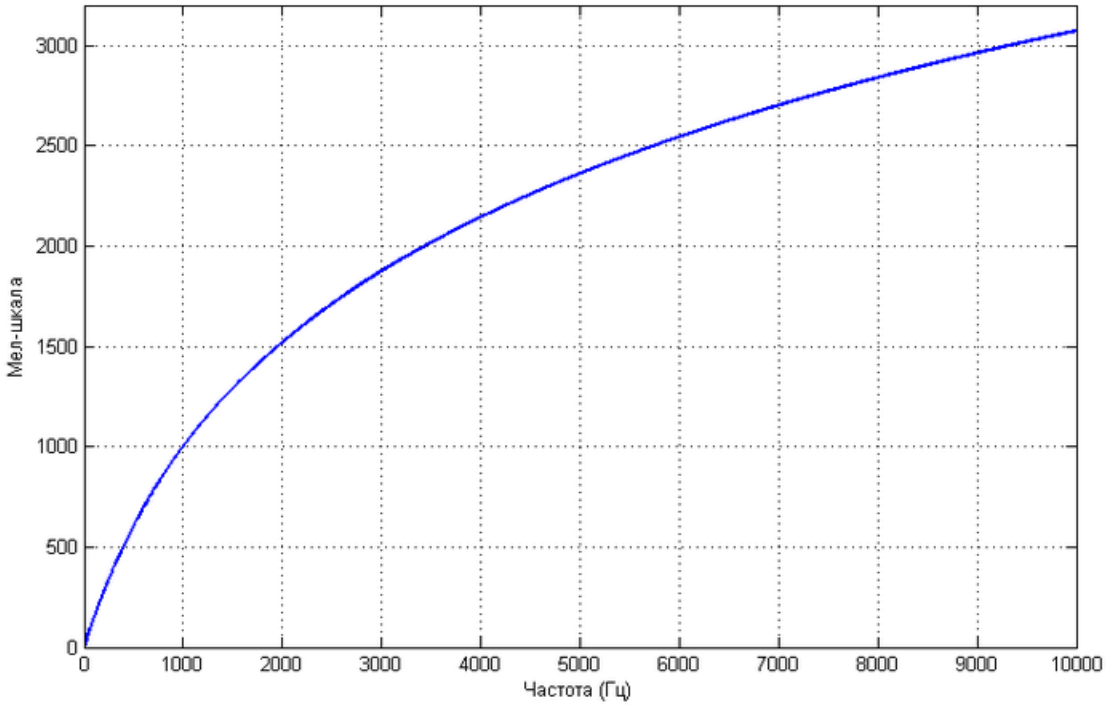


Рисунок 4. Зависимость мел от частоты

Такая зависимость не является очень точной, зато описывается простой формулой:

$$M(f) = 1127.01048\left(1 + \frac{f}{700}\right)$$

Метод мел-частотных коэффициентов имеет следующие преимущества: из-за использования спектра сигнала при последующем анализе учитывается его волновая природа, мел-шкала учитывает воспринимаемые человеком особенности звука, что позволяет выделять самые значимые частоты речи, таким образом позволяет хранить наиболее наполненные информацией частоты (на Рисунок 5. Зависимость размера односекундной записи от частоты дискретизации для различных значений глубины звучания приведена зависимость размера односекундной записи от частоты дискретизации для различных значений глубины звучания). Кроме того, такой метод позволяет программисту ограничить количество вычисляемых коэффициентов, то есть контролировать объем данных.

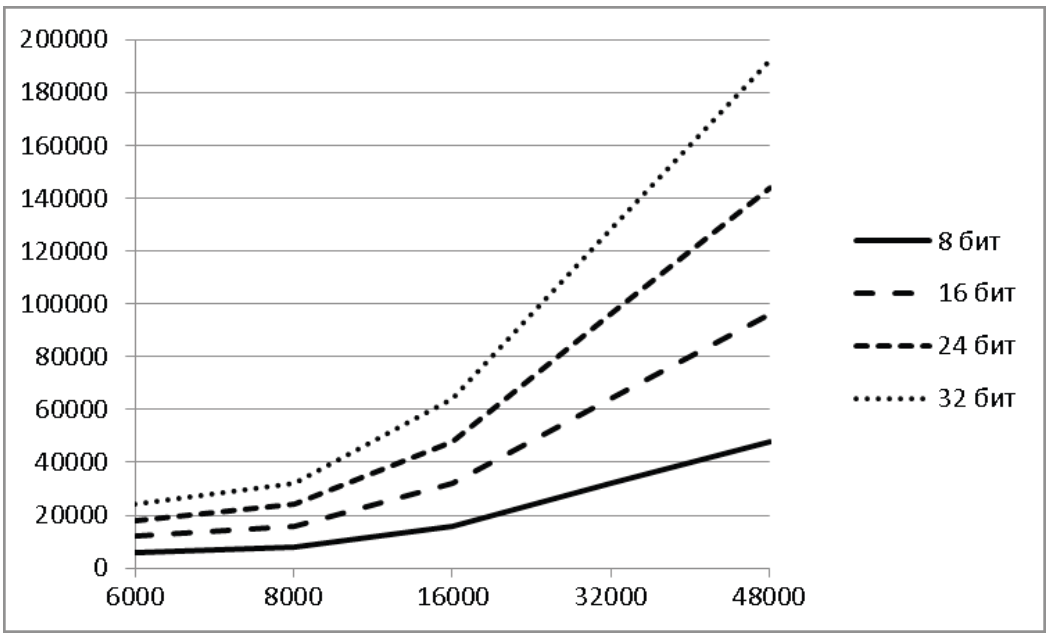


Рисунок 5. Зависимость размера односекундной записи от частоты дискретизации для различных значений глубины звучания

Реализацию метода разобьем на несколько шагов.

Разбиение исходного сигнала на фреймы.

Размер фрейма устанавливается на усмотрение разработчика. Чаще всего это значение от 20 до 40 мс. Считается, что за такой отрезок времени речевой сигнал не изменится сильно.

При разбиении сам сигнал записываем в виде:

$$x(n), 0 \leq n \leq N$$

где N – размер фрейма, $x_j(n)$ – j- ый фрейм.

Каждый следующий шаг выполняется для каждого отдельного фрейма.

Умножение кадров на функцию Хемминга.

Дело в том, что входной сигнал конечен и не периодический, из-за разрывов на его концах применение преобразования Фурье проявит эффект утечки. Чтобы этот эффект не так сильно влиял каждый кадр необходимо умножить на оконную функцию Хемминга:

$$\omega(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N-1$$

Преобразование Фурье.

К полученным результатам применяем дискретное преобразование Фурье:

$$X_j(k) = \sum_{n=0}^{N-1} x_j(n)\omega(n)e^{-\frac{2\pi i}{N}kn}, 0 \le k \le N$$

где j-номер фрейма, $i = \sqrt{-1}$.

Вычисление периодограммы.

Для каждого фрейма необходимо вычислить периодограмму:

$$P_j(k) = \frac{\left|X_j(k)\right|^2}{N}$$

Вычисление блока мел-фильтров.

Вообще говоря, мел-фильтр – это треугольная оконная функция, позволяющая суммировать количество энергии на определенном диапазоне частот. То есть позволяющая получить мел-коэффициент.

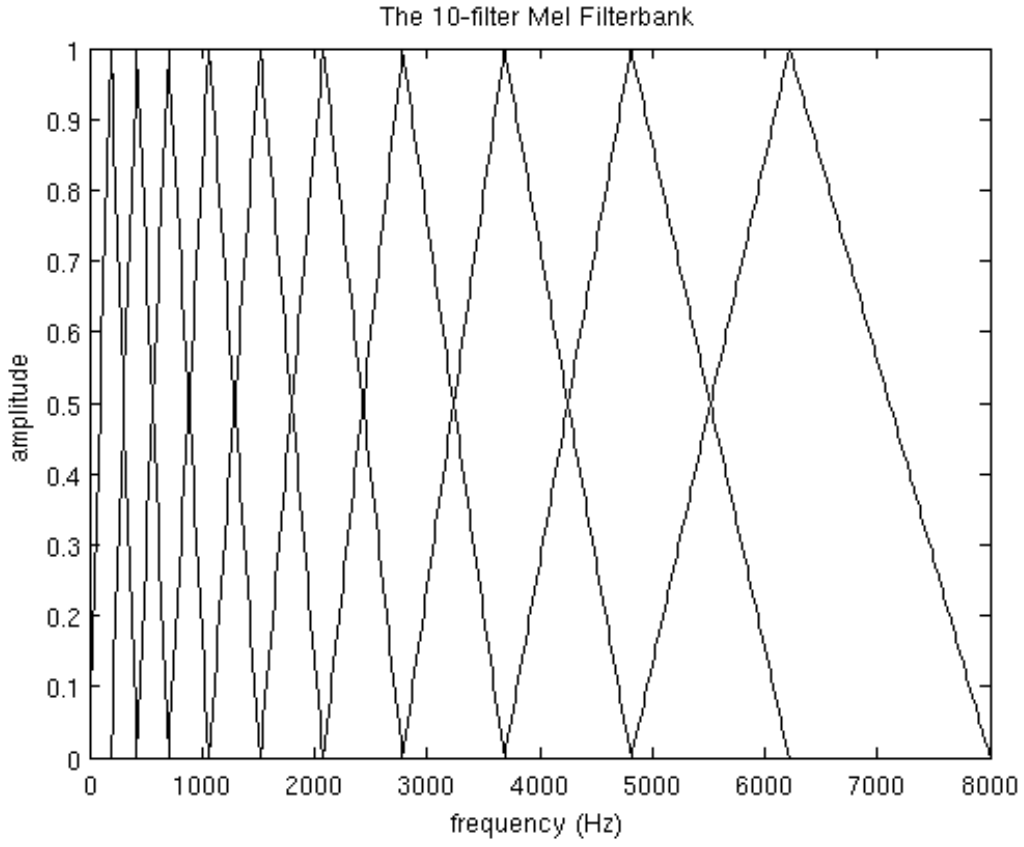


Рисунок 6. Мел-фильтры

Знание количества коэффициентов и анализируемого диапазона частот позволяет построить набор фильтров (Рисунок 6. Мел-фильтры).

Видно, что чем выше номер мел-коэффициента, тем шире основание. Это связано с тем, что шкала мел, как уже говорилось, нелинейная.

Сами фильтры строятся по следующей формуле:

$$H_m(k) = \begin{cases} 0, & k < f(m-1) \\ \dfrac{k - f(m-1)}{f(m) - f(m-1)}, & f(m-1) \le k < f(m) \\ \dfrac{f(m+1) - k}{f(m+1) - f(m)}, & f(m) \le k \le f(m+1) \\ 0, & k > f(m+1) \end{cases}$$

где m – это частота “опорных точек” треугольников. Причем их количество равно количеству треугольников + 2.

Логарифмирование полученных энергий.

Как уже говорилось, человек слышит громкость не в линейном масштабе. Большие колебания могут не сильно отличаться, если звук изначально громкий. Благодаря логарифмированию, мы можем приблизить наши функции к тому, что слышит человек. Делаем это по следующей формуле:

$$S_j(m) = ln \sum_{k=0}^{N-1} P_j(k) H_m(k)\,, 0 \le m < M$$

Вычисление мел-кепстральных коэффициентов.

Используем дискретное косинусное преобразование:

$$c_j(n) = \sum_{m=0}^{M-1} S_j(m) \cos\left(\frac{\pi n \left(m + \frac{1}{2}\right)}{M}\right), 0 \le n \le M$$

Смысл такого преобразования в том, чтобы “сжать” полученный результат так, что повысится значимость первых коэффициентов и понизится значимость последних. Таким образом, мы декоррелируем энергии фильтров. Кроме того, сохраняются только 12 из 20 коэффициентов. Это происходит из-за того, что более высокие коэффициенты представляют быстрые изменения энергий набора фильтров, которые усложняют распознавание речи, поэтому отбрасывая их, мы получаем небольшое улучшение.

Вот таким алгоритмом мы и получаем мел-кепстральные коэффициенты.

Поиск порогового значения.

Теперь, когда у нас уже есть цифровое представление голосового пароля, необходимо сравнить его с картой пользователя. Для этого нам понадобится пороговое значение. Так как выпускать приложение в оборот я не буду, я буду использовать порог, хорошо подходящий для тестирования, то есть порог EER, о котором говорилось выше.

На самом деле порог не высчитывается, он оценивается эмпирически. Мы просто берем диапазон тестов и прогоняем их для различных пороговых значений, а затем просто выбираем то, которое ближе всего к требованию.

Если бы мне требовалось сделать приложение, готовое к выпуску, то порог, очевидно, пришлось бы менять. Но и в таком случае я искал бы значение “на ощупь”.

Описание реализации

Так как я делаю лишь прототип приложения, позволяющий увидеть работоспособность метода, я решил не усложнять работу так, как если бы приложение делалось под выпуск.

Во-первых, я не добавлял работу с микрофоном. Входной сигнал для регистрации пользователя и для его входа в систему должен быть предзаписан в формате WAV. Кроме того, он должен содержать не более двух каналов, иметь тип PMC и иметь 16 бит на канал. Для хранения аудиофайлов, которые будут использоваться при работе приложения создана папка “Records”, в ней созданы две подпапки “Register” и “LogIn” для файлов регистрации и входа в систему соответственно.

Во-вторых, при регистрации используется лишь один аудиофайл.

В-третьих, кодовая фраза выбирается самим пользователем. При входе в систему пользователь, разумеется, должен использовать эту же фразу.

В-четвертых, я не стал создавать базу данных или каким-либо образом сохранять данные при закрытии приложения. Соответственно, после выхода из приложения при следующем его открытии придется регистрировать пользователей заново.

Также стоит отметить, что я установил пороговое значение равное нулю. Так я могу продемонстрировать метода, не отвлекаясь его погрешность.

В остальном метод, описанный выше, реализован полностью. Для его реализации я создал c++/clr приложение.

Принцип взаимодействия пользователя с приложением следующий: мы имеем окно (Рисунок 7. Главное окно приложения), в котором есть поля “Name” и “Record file”. В

этих полях указывается имя пользователя (которое служит также его уникальным идентификатором) и имя файла, в котором записан голосовой пароль. Для регистрации пользователь вводит в поле “Name” свой логин, а в поле “Record file” имя записанного файла, а затем нажимает на кнопку “Register”. После этого происходит регистрация или вывод сообщения об ошибке (например, о том, что ваш логин занят). Для входа в систему необходимо совершить те же действия, но нажать на кнопку “Log In”. Причем программа сама выберет путь к файлу. Для регистрации это будет “Records/Register”, а для входа “Records/LogIn”.

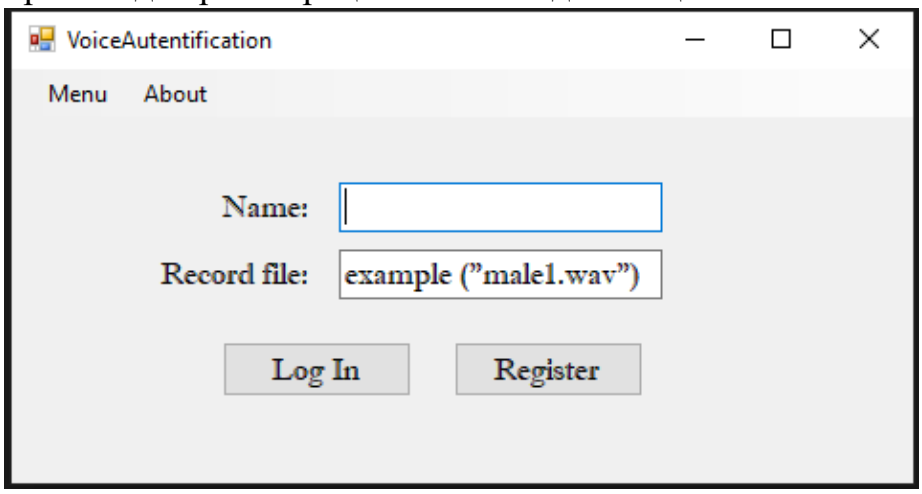


Рисунок 7. Главное окно приложения

Тестирование

Создание входного файла

Для создания файлов я использовал приложение FL Studio, позволяющее записывать аудиофайлы с различными настройками.

После установки приложения и его запуска мы видим следующее окно:

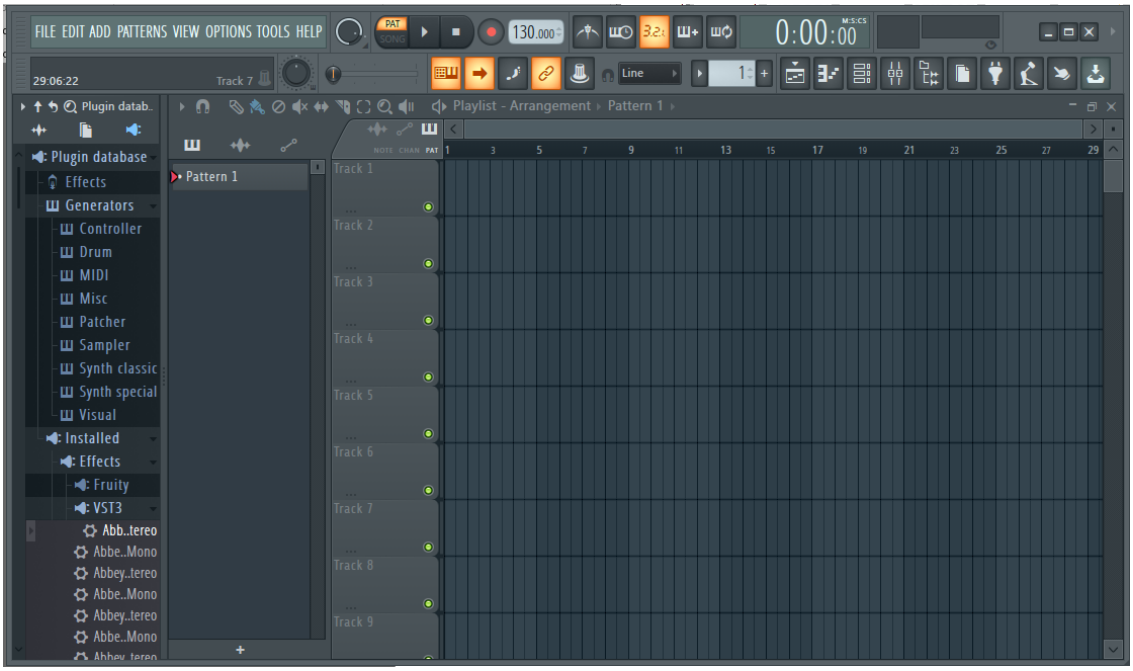


Рисунок 8. FL Studio

Сначала нужно установить опцию ввода на один из треков. Это делается так: кликаем правой кнопкой мыши по одному из треков (например Track 1), в выпадающем меню выбираем Track Mode -> Audio Track -> Insert 1:

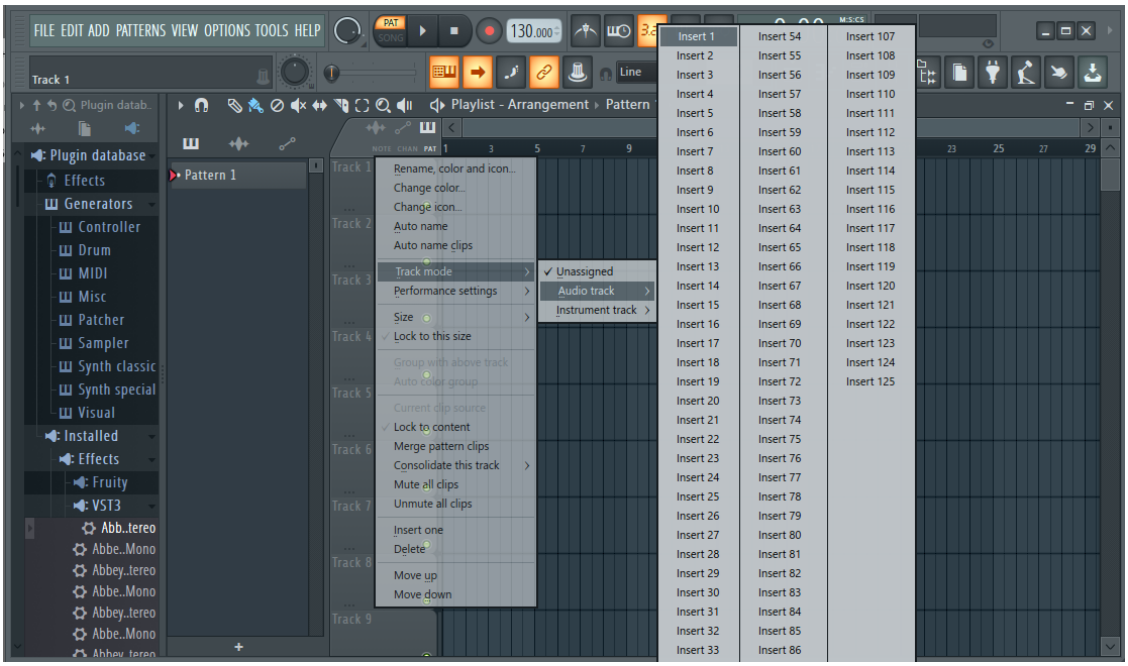


Рисунок 9. Выбор трека для ввода

Теперь в верхнем меню выбираем Mixer. Он обозначается значком (подчеркнуто красным)

Рисунок 10. Панель Mixer

В нем находим созданный Insert 1 и выбираем для него входное устройство:

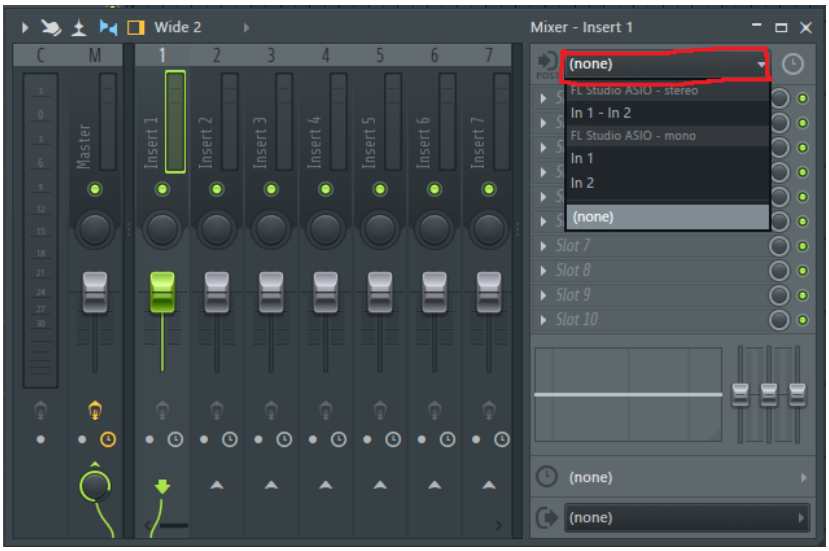


Рисунок 11. Выбор входного устройства

После этого трек автоматически перейдет в состояние записи. Чтобы начать запись закрываем миксер и нажимаем на кнопку записи.

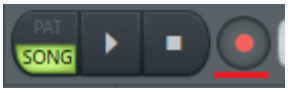


Рисунок 12. Начало записи

В открывшемся окне выбираем запись в плейлист:

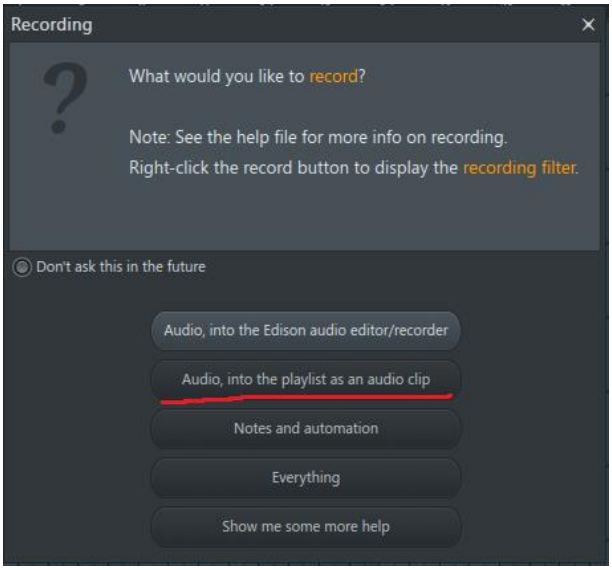


Рисунок 13. Выбор размещения записи

Затем запускается запись. Для окончания записи достаточно нажать пробел.

Теперь необходимо экспортировать запись. Для этого выбираем File -> Export -> Wav file.

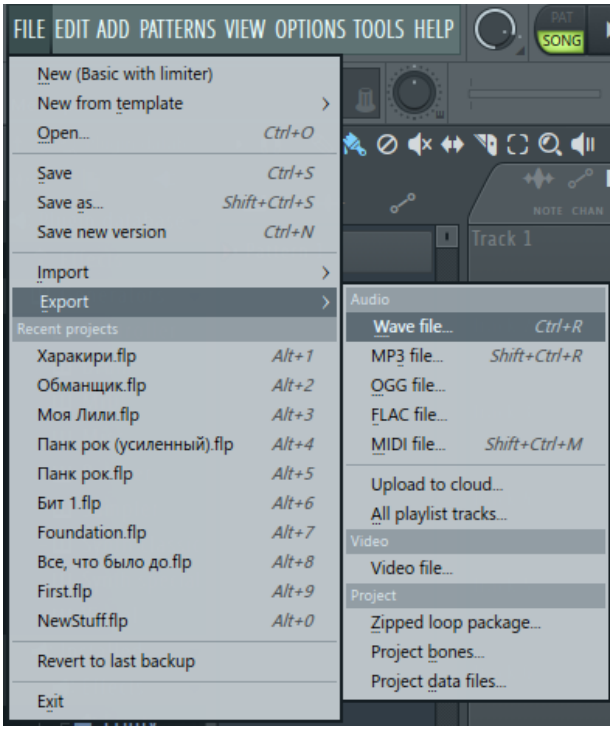


Рисунок 14. Экспорт файла

Теперь выбираем место, куда экспортированный файл сохраниться, и в открывшемся окне устанавливаем следующие настройки (достаточно поменять Stereo на Mono merged (выделено красным), остальные настройки изначально установлены правильно):

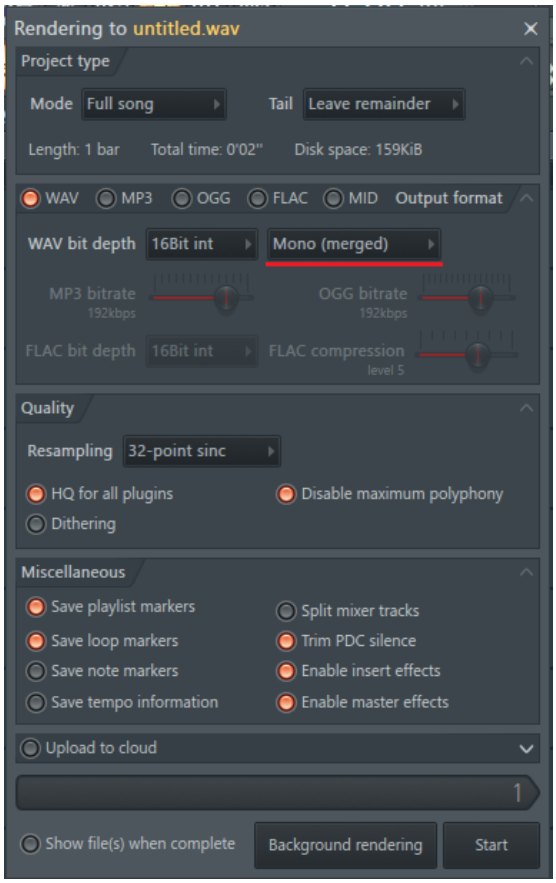


Рисунок 15. Настройки экспорта

Нажимаем кнопку Start. Просто запись голоса, длящуюся не так уж и долго, программа зарендерит почти мгновенно. После этого приложение можно закрывать. Сохранять проект необязательно.

Конечно, существуют и другие способы записи требуемого файла (возможно даже более простые), однако их я не тестировал. Кроме того, далеко не все звукозаписывающие программы создают файл нужного формата.

Тест

Попробуем зарегистрировать нового пользователя. Введем имя “Alexey” и выберем файл “Records/Register/male1.wav” (Рисунок 16. Регистрация пользователя).

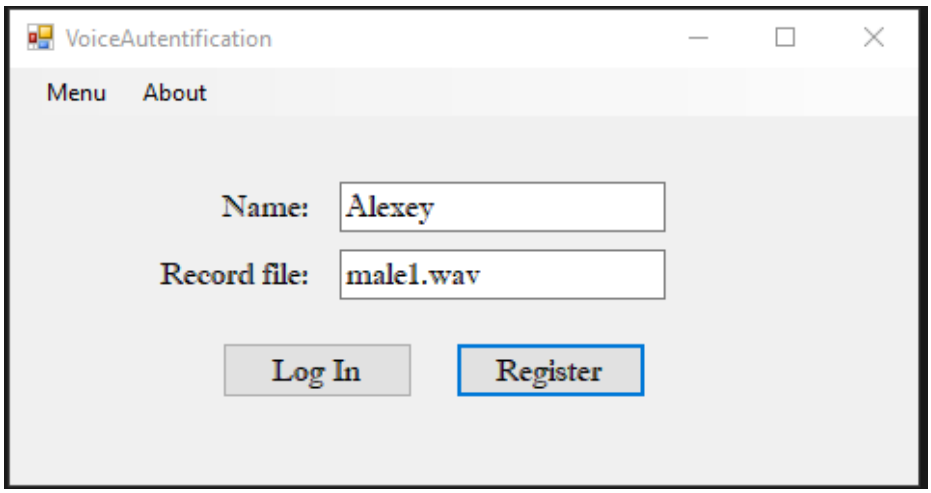


Рисунок 16. Регистрация пользователя

Обработка входного сигнала занимает довольно много времени (2-3 минуты). Как только программа заполнила карту нового пользователя, мы получаем сообщение об успешной регистрации (Рисунок 17. Успешная регистрация).

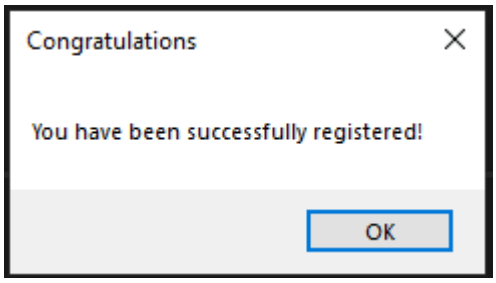


Рисунок 17. Успешная регистрация

Если сейчас мы попробуем создать такого же пользователя мы получим сообщение о неудаче (Рисунок 18. Неудача при регистрации).

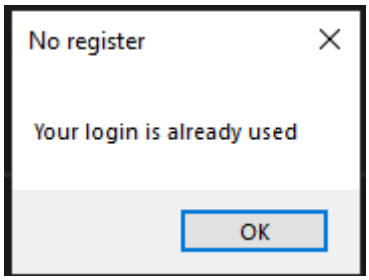


Рисунок 18. Неудача при регистрации

Теперь попробуем войти в систему под только что зарегистрированным пользователем, но используя запись другого голоса (Рисунок 19. Неверный пароль при входе). Получим неудачу (Рисунок 20. Отказ в доступе). Файл “Records/LogIn/male2.wav”.

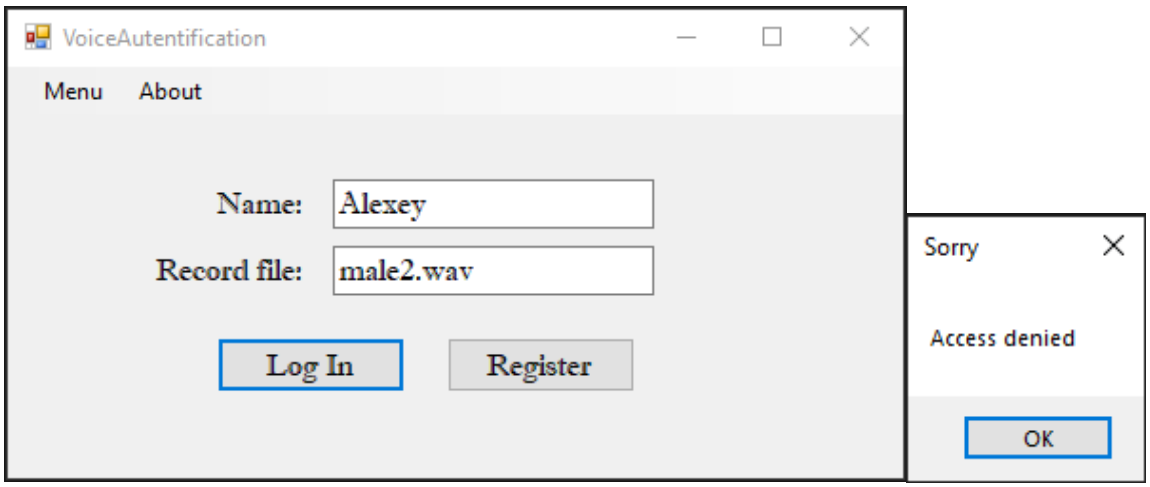


Рисунок 19. Неверный пароль при входе

Рисунок 20. Отказ в доступе

Однако, при попытке войти с записью правильного голоса (Рисунок 21. Вход с правильным паролем Рисунок 22. Доступ разрешен) система предоставляет доступ (Рисунок 21. Вход с правильным паролем Рисунок 22. Доступ разрешен). “Records/LogIn/male1.wav”.

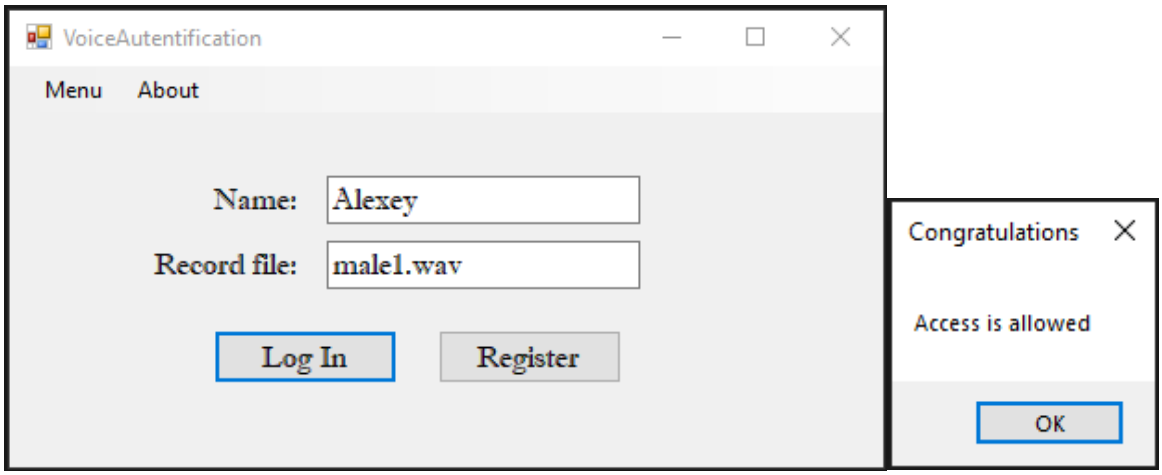


Рисунок 21. Вход с правильным паролем

Рисунок 22. Доступ разрешен

Как видим, приложение проходит тестирование!

Заключение

В данной работе я разобрал средства аутентификации пользователей компьютерных систем по голосу. Такие средства являются крайне удобными для пользователя, поскольку содержат в себе множество возможностей, которых нет ни у одного другого биометрического метода. Однако это удобство, к сожалению, компенсируется плохой надежностью. В связи с этим, голосовые методы редко можно встретить в виде единственного средства защиты. Чаще всего, используют многофакторную аутентификацию, применяя несколько различных биометрических методов.

Тем не менее, голосовая биометрия имеет уникальные особенности, которые позволяют ей при развитии технологий не только не терять популярность, но даже более того, становиться все более и более выгодной на фоне других методов. Я считаю, что однажды технологический прогресс дойдет до того, что мы сможем назвать аутентификацию по голосу абсолютно безопасным методом, а сам метод станет повсеместно главным средством защиты компьютерных сетей.

Список литературы

1. Евсеев В.Л., Козлов Ю.Е. Современные методы речевой аутентификации в приложениях мобильных устройств.

   https://bit.mephi.ru/index.php/bit/article/viewFile/25/35

2. Иванов Д.А., Никитин А.П. Метод текстозависимой аутентификации по голосу.
   https://documentation.rsuh.ru/jour/article/view/67

3. “Альфацефей”. Первичный анализ речевых сигналов. Лекция 1.
   https://alphacephei.com/ru/lecture1.pdf

4. “Habr”. Мел-кепстральные коэффициенты (MFCC) и распознавание речи.
   https://habr.com/ru/post/140828/

5. Ю.Н. Матвеев. Технологии биометрической идентификации личности по голосу и другим модальностям.
   https://www.researchgate.net/publication/236142366_Tehnologii_biometriceskoj_identifikacii_licnosti_po_golosu_i_drugim_modalnostam_BIOMETRIC_TECHNOLOGIES_OF_PERSON_IDENTIFICATION_BY_VOICE_AND_OTHER_MODALITIES