

Белорусский государственный университет
факультет Радиофизики и Электроники
Минск, Беларусь, 2009



Использование технологий XML и COM для решения задач статистической радиофизики

Выпускная работа студента магистратуры Труса Александра Александровича
Руководитель: Хейдоров Игорь Эдуардович, доцент, кандидат физ.-мат. наук.



Содержание

- [Цель работы](#)
- [Задача индексации](#)
- [Вектора признаков](#)
- [Кепстр](#)
- [Формирования вектора признаков на основе вейвлет-преобразования](#)
- [Скрытые Марковские модели](#)
- [Метод опорных векторов](#)
- [Структура ядра системы индексации](#)
- [Структура ядра системы индексации](#)
- [Процессоры](#)
- [Компоненты чтения](#)
- [Пользовательский интерфейс](#)
- [Эксперимент](#)
- [Результаты эксперимента](#)
- [Заключение](#)



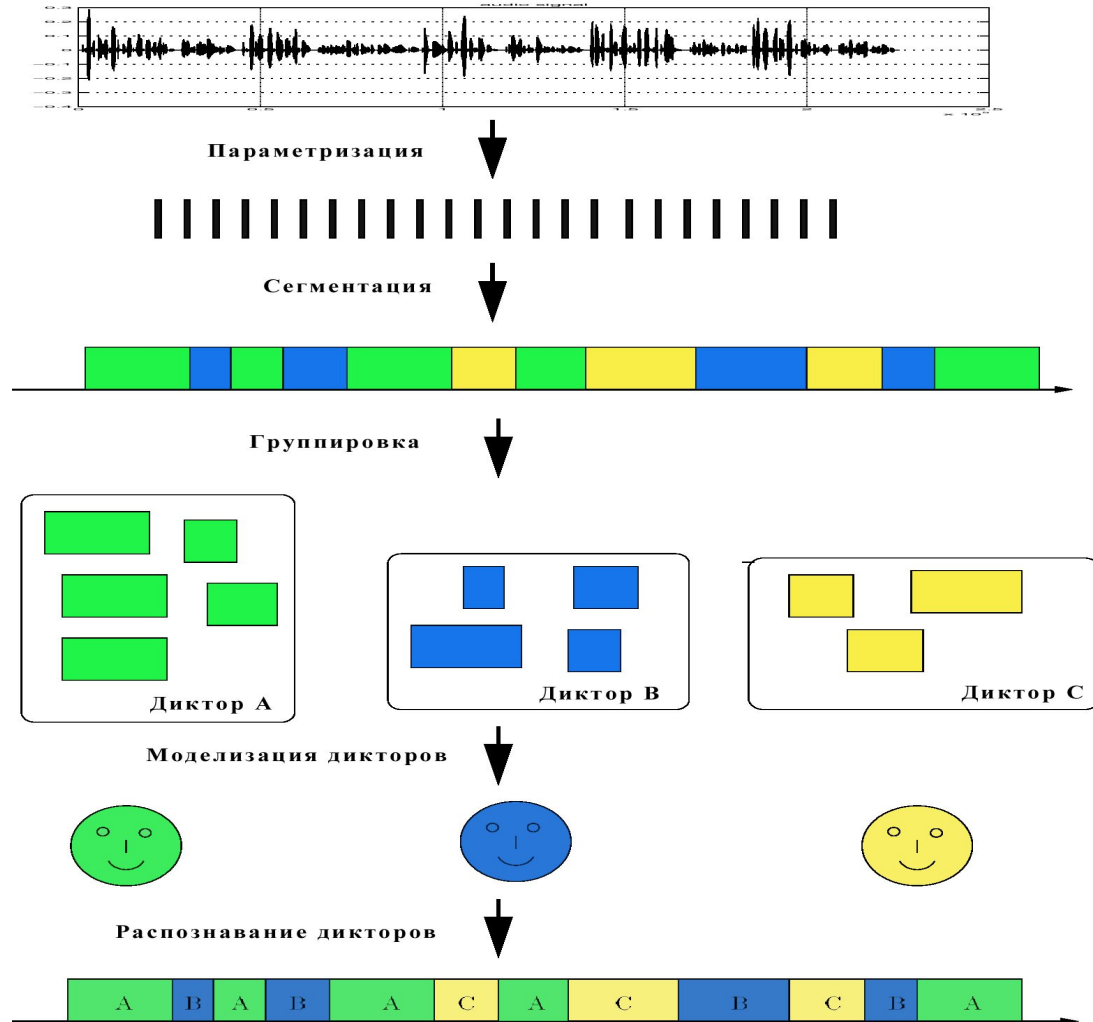
Цель работы

Целью данной работы является разработка системы индексации аудиопоследовательностей на основе СММ и МОВ. Для достижения поставленной цели необходимо решить следующие задачи:

- Изучить общую проблему индексации аудиодокументов и существующие подходы.
- Разработать структуру системы Разработать структуру системы индексации аудиопоследовательностей на основе Скрытой Марковской Модели Разработать структуру системы индексации аудиопоследовательностей на основе Скрытой Марковской Модели и Метода Опорных Векторов.
- Разработать алгоритм извлечения вектора признаков из аудиосигналов на основе вейвлет преобразования.
- Разработать структуру и реализовать систему индексации аудиопоследовательностей на языке C++.
- Разработать структуру компонентов системы Разработать структуру компонентов системы и реализовать графический интерфейс системы.

Реализовать базовые компоненты обработки аудиосигналов.

Задача индексации



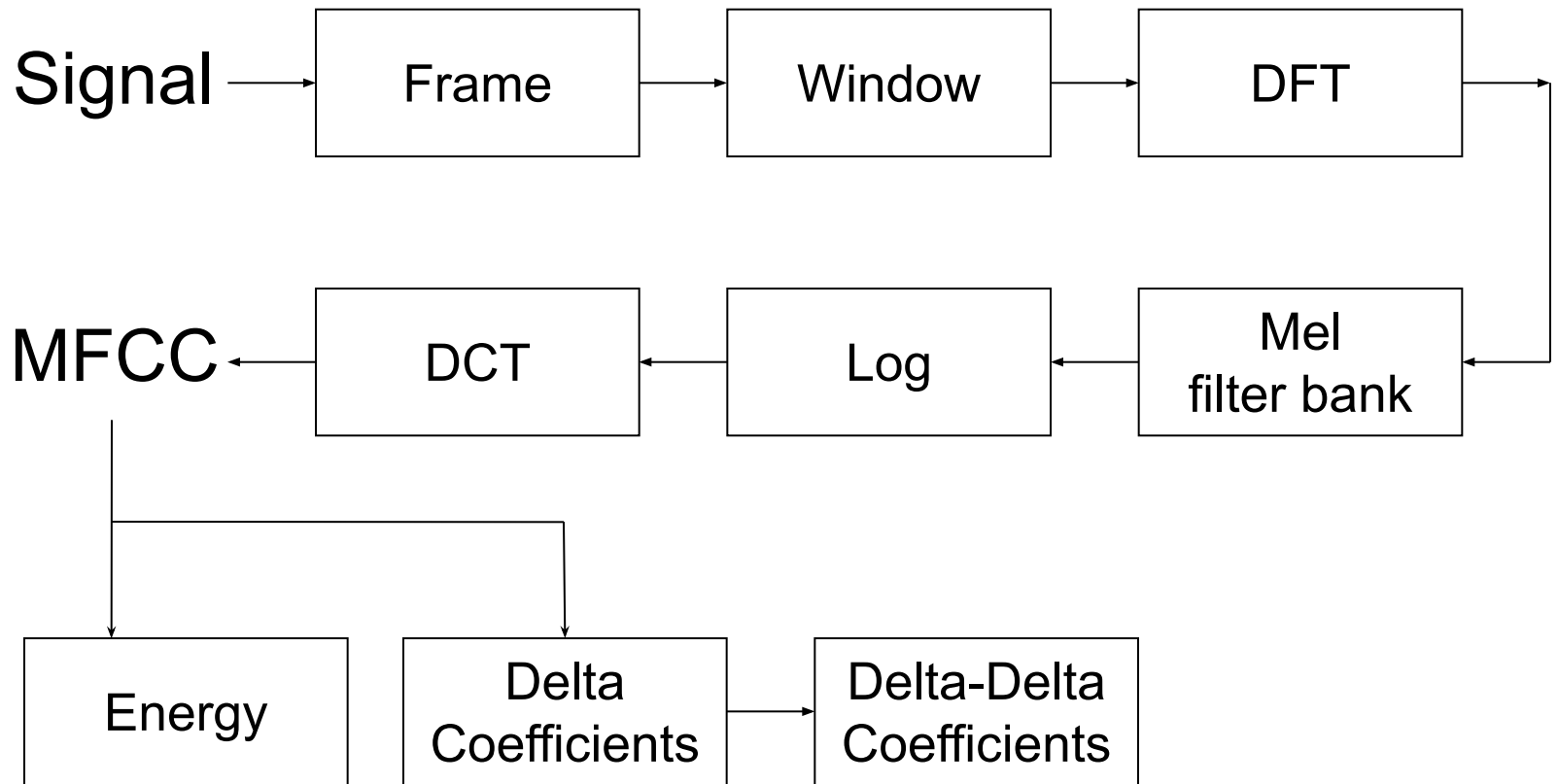
Вектора признаков

В данной работе использовалось 2 вида векторов признаков:

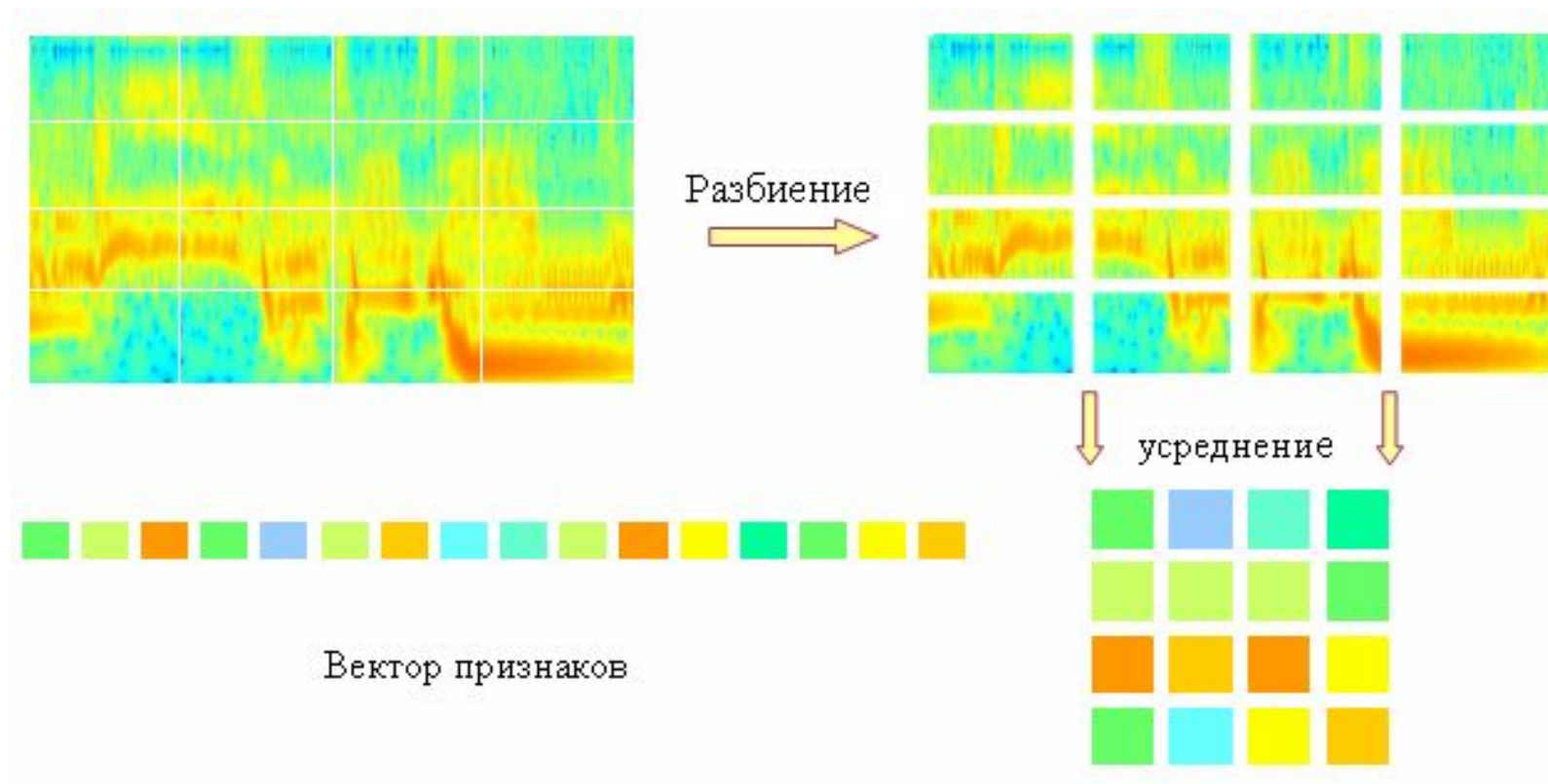
- Вектор признаков кепстральных коэффициентов
- Вектор признаков на основе вейвлет преобразования



Кепстр



Формирование вектора признаков на основе вейвлет-преобразования



Скрытые марковские модели

Скрытая марковская модель – это дважды стохастический случайный процесс в котором наблюдения являются вероятностной функцией состояния.

СММ характеризуются следующими параметрами:

- набор состояний $Q = q_1, q_2, \dots, q_N$;
- вариационная матрица переходов $A = \{a_{ij}\}, \sum_{j=1}^N a_{ij} = 1$;
- набор событий $O = o_1, o_2, \dots, o_T$;
- матрица вероятностей наблюдения $B = b_i(o_t), \sum_{t=1}^T b_i(o_t) = 1$;
- начальное и конечное состояния q_0, q_{end} ;
- матрица начальных вероятностей $\pi = \pi_1, \pi_2, \dots, \pi_N, \sum_{i=1}^N \pi_i = 1$.

Приближения СММ

1) $P(q_i | q_1 \dots q_{i-1}) = P(q_i | q_{i-1})$

2) $P(o_i | q_1, \dots, q_i, \dots, q_n, o_1, \dots, o_i, \dots, o_n) = P(o_i | q_i)$

Три фундаментальных проблемы СММ

- **Проблема Оценки** – при данных модели Φ и последовательности наблюдений $X = X_1, X_2, \dots, X_T$, какой будет вероятность появления модели, сгенерировавшей наблюдение?

Прямой алгоритм: $\alpha_t(j) = \sum_{i=1}^{N-1} \alpha_{t-1}(i) a_{ij} b_j(o_t)$

- **Проблема Декодирования** - при данных модели Φ и последовательности наблюдений $X = X_1, X_2, \dots, X_T$ какой будет наиболее вероятная последовательность состояний S в модели, сгенерировавшей наблюдения?

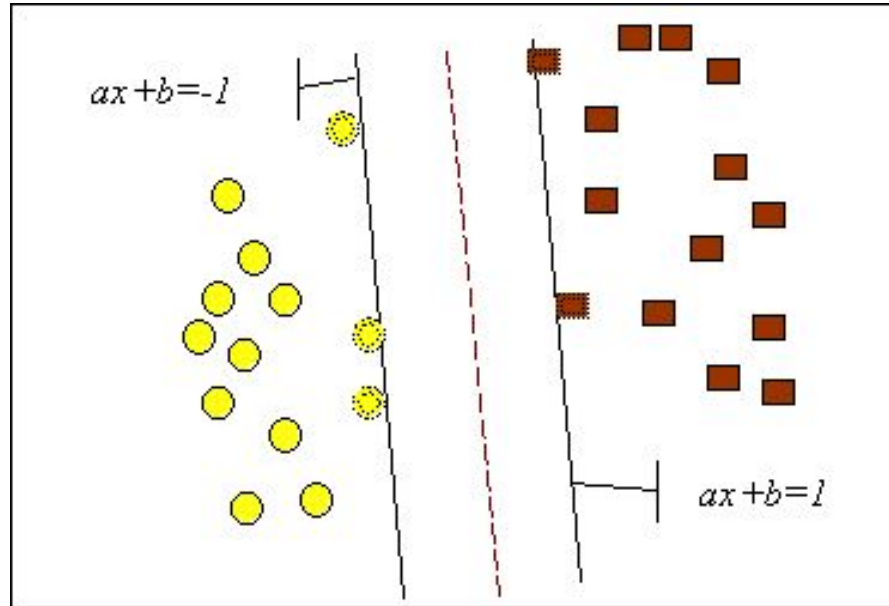
Алгоритм Витерби $v_t(j) = \max_{1 \leq i \leq N-1} v_{t-1}(i) a_{ij} b_j(o_t)$

- **Проблема Обучения** - при данных модели Φ и последовательности наблюдений $X = X_1, X_2, \dots, X_T$ как можно настроить параметры модели так, чтобы максимизировать совместную вероятность?

Прямой-обратный алгоритм



Метод опорных векторов

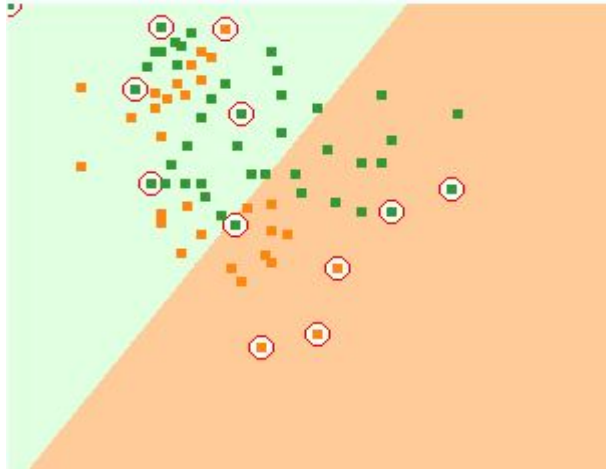


$$a(x) = \text{sign}\left(\sum_{j=1}^n w_j x^j - w_0\right) = \text{sign}(\langle w, x \rangle - w_0)$$

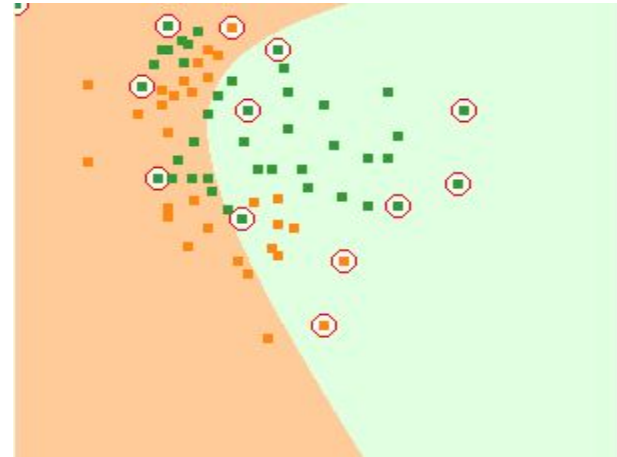
$$\langle w, x \rangle = w_0$$



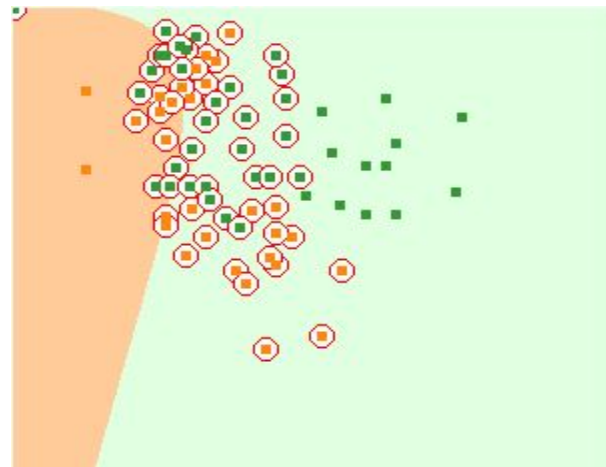
Метод опорных векторов



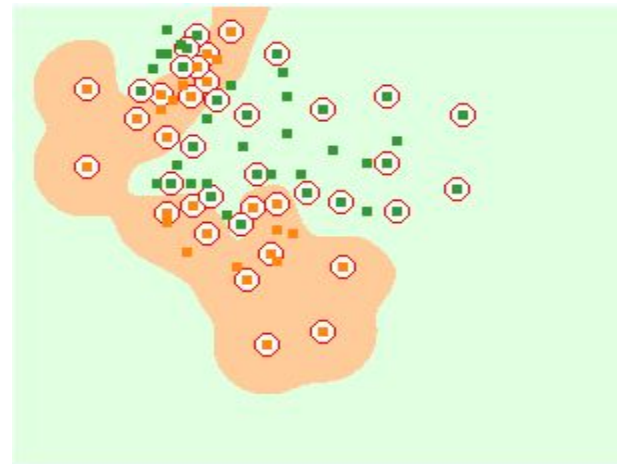
$$K(x_1, x_2) = x_1 \cdot x_2$$



$$K(x_1, x_2) = (1 + x_1 \cdot x_2)^p$$



$$K(x_1, x_2) = k \cdot \tanh(x_1 \cdot x_2)$$



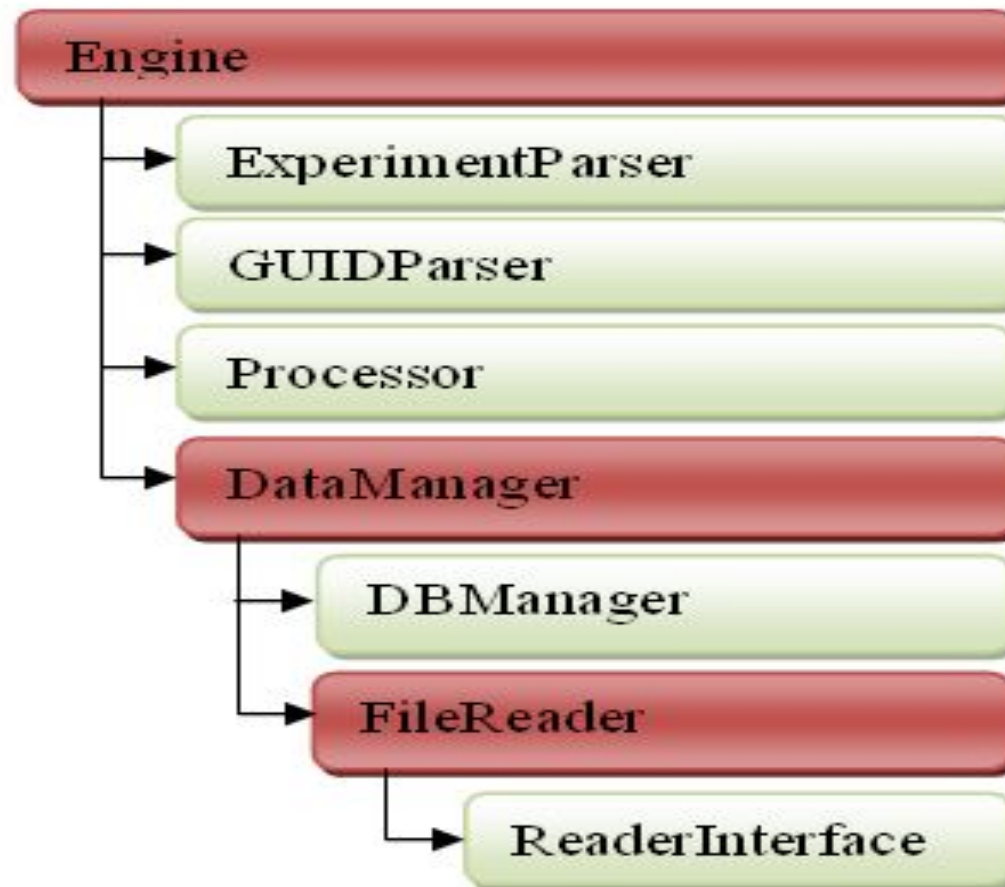
$$K(x, y) = \exp(-\gamma \|x - y\|^2)$$



Структура системы индексации аудио



Структура ядра системы индексации



Процессоры

```
class Processor : public Idispatch
{
public:
    virtual HRESULT STDMETHODCALLTYPE loadSettings(const
        char*settings) = 0;
    virtual HRESULT STDMETHODCALLTYPE work(DataStruct** data) = 0;
    virtual HRESULT STDMETHODCALLTYPE getProcessorId(char** id) =
        0;
    virtual HRESULT STDMETHODCALLTYPE getSize(int* val) = 0;
    virtual HRESULT STDMETHODCALLTYPE getStep(int* val) = 0;
    virtual HRESULT STDMETHODCALLTYPE getSettings(char**) = 0;
};
```

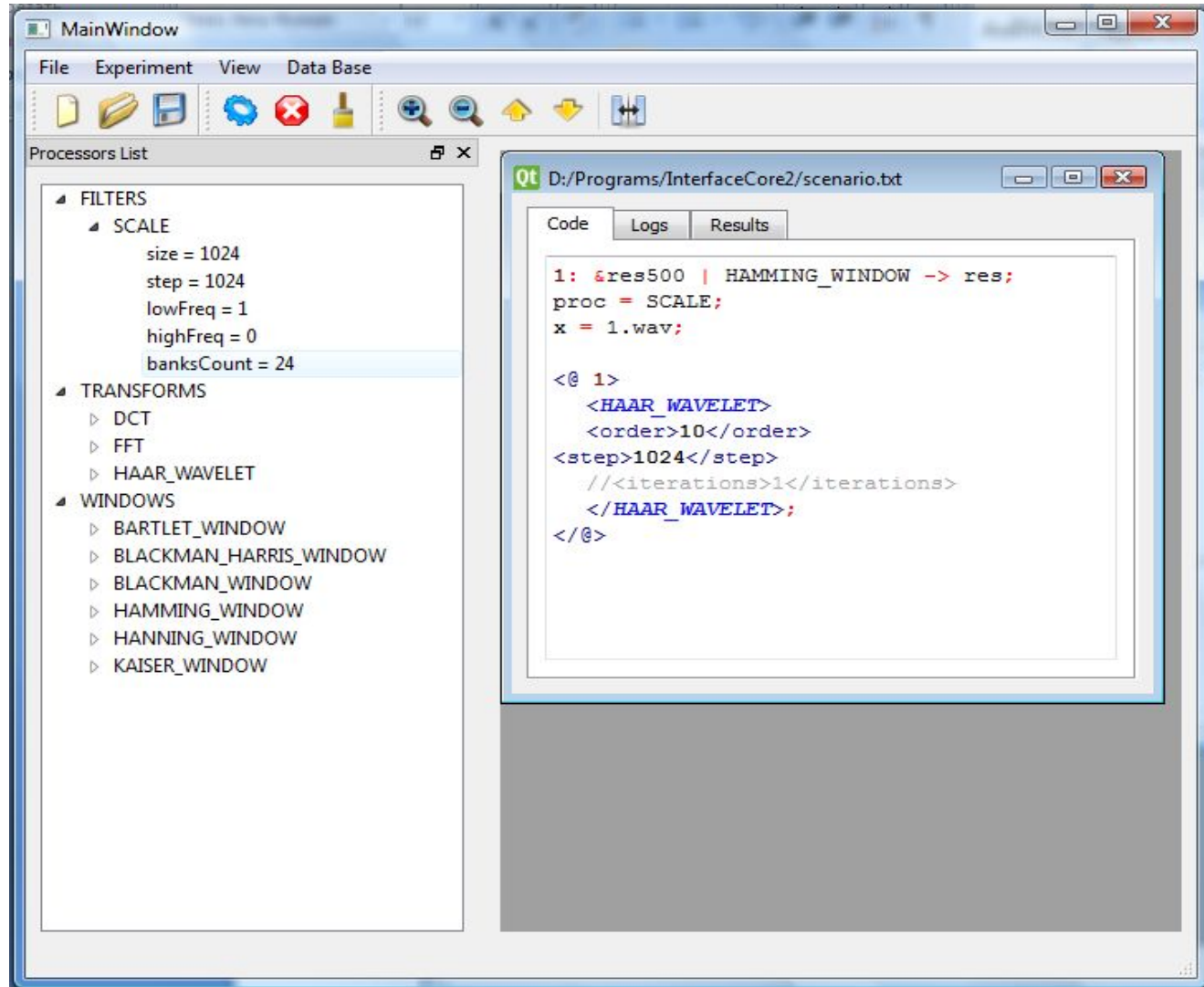


Компоненты чтения

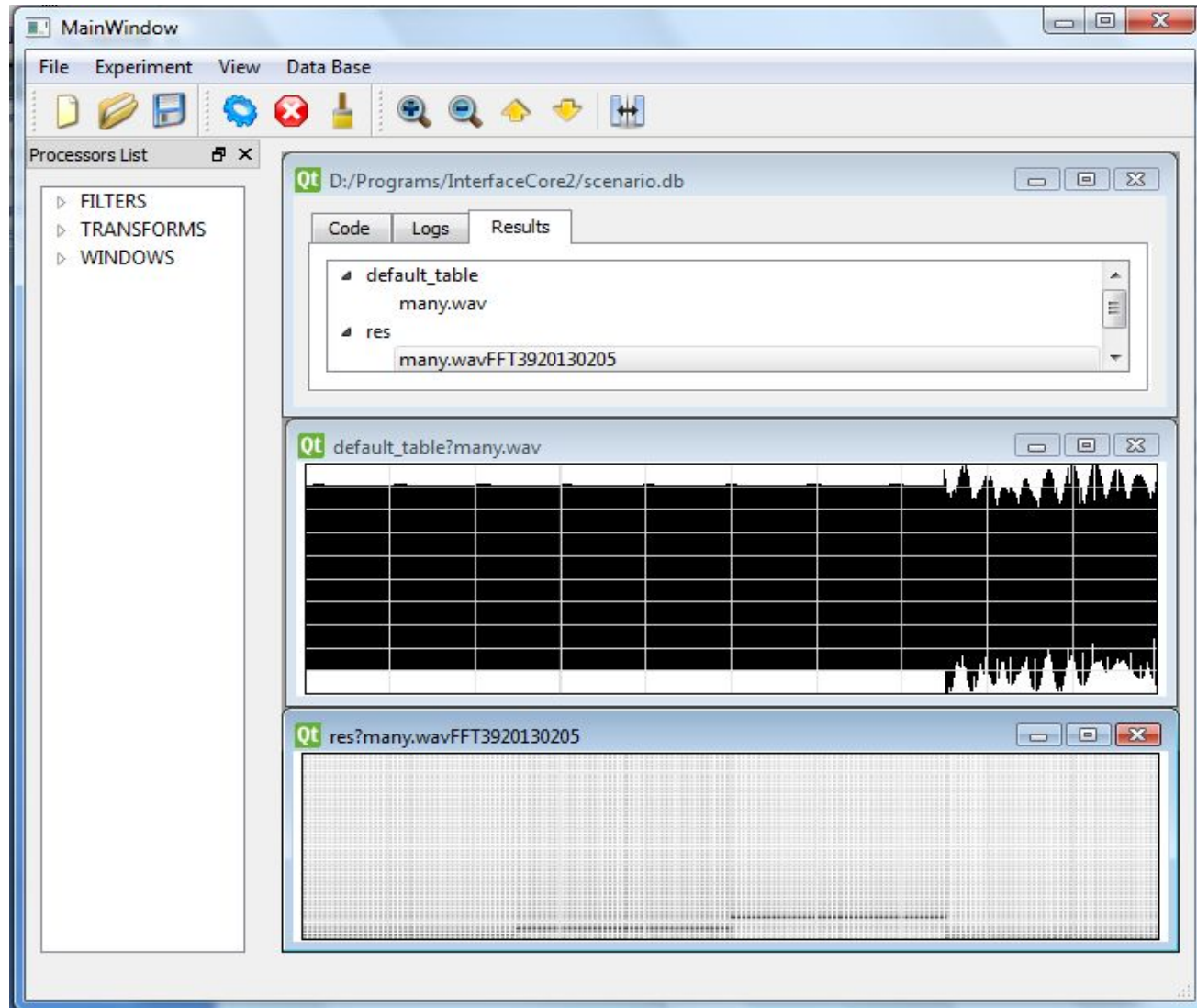
```
class ReaderInterface : public IDispatch
{
public:
    virtual HRESULT STDMETHODCALLTYPE open(const char*, ResultStruct*)
        = 0;
    virtual HRESULT STDMETHODCALLTYPE read(DataStruct**,
        ResultStruct*) = 0;
};
```



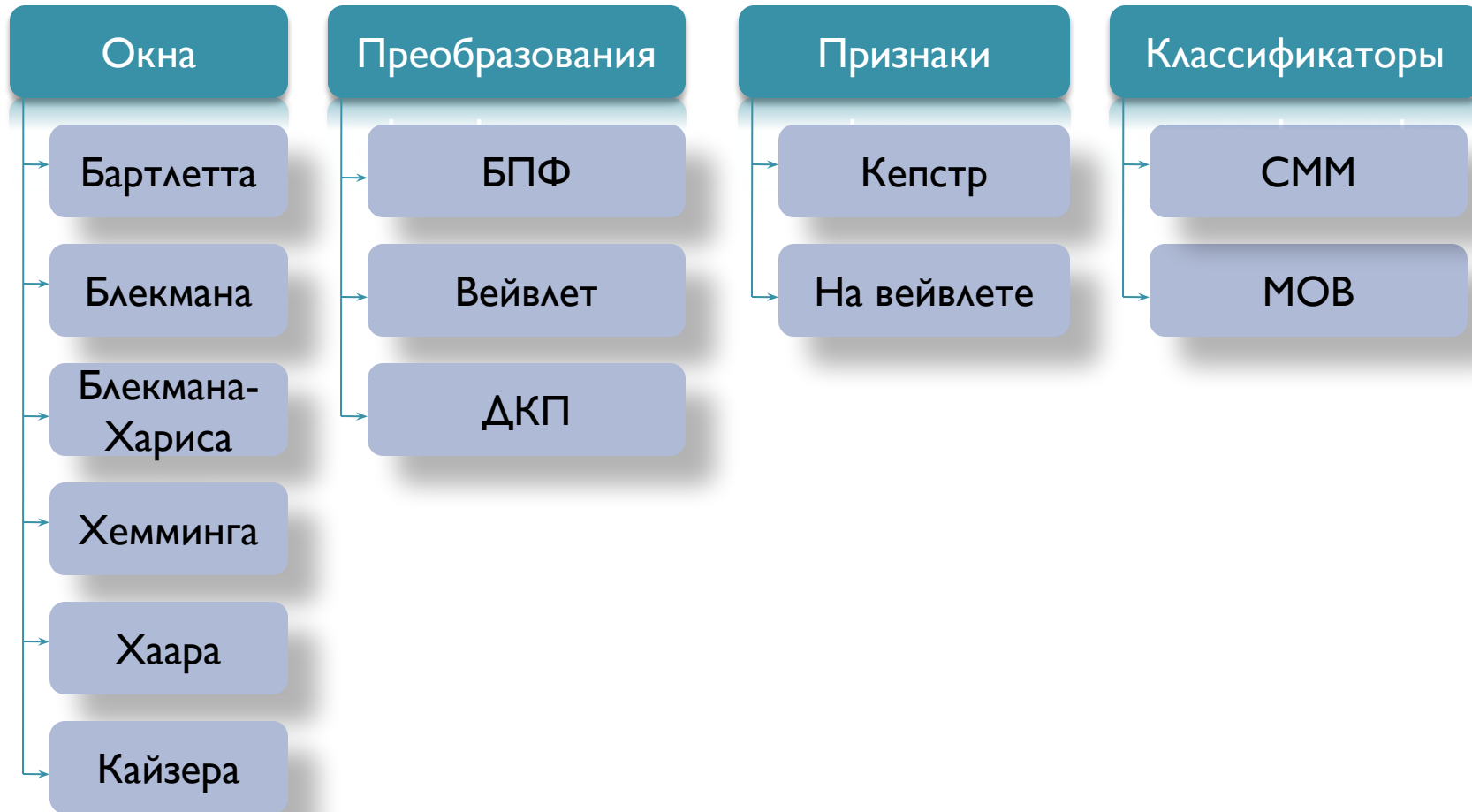
Пользовательский интерфейс



Пользовательский интерфейс



Базовые процессоры



Эксперимент

В рамках данного проекта, на основе созданной системы были проведены 3 эксперимента по индексации аудиопоследовательностей:

- сегментация речь/музыка/тишина
- распознавание эмоций на основе МОВ
- определение жанров музыки на основе СММ

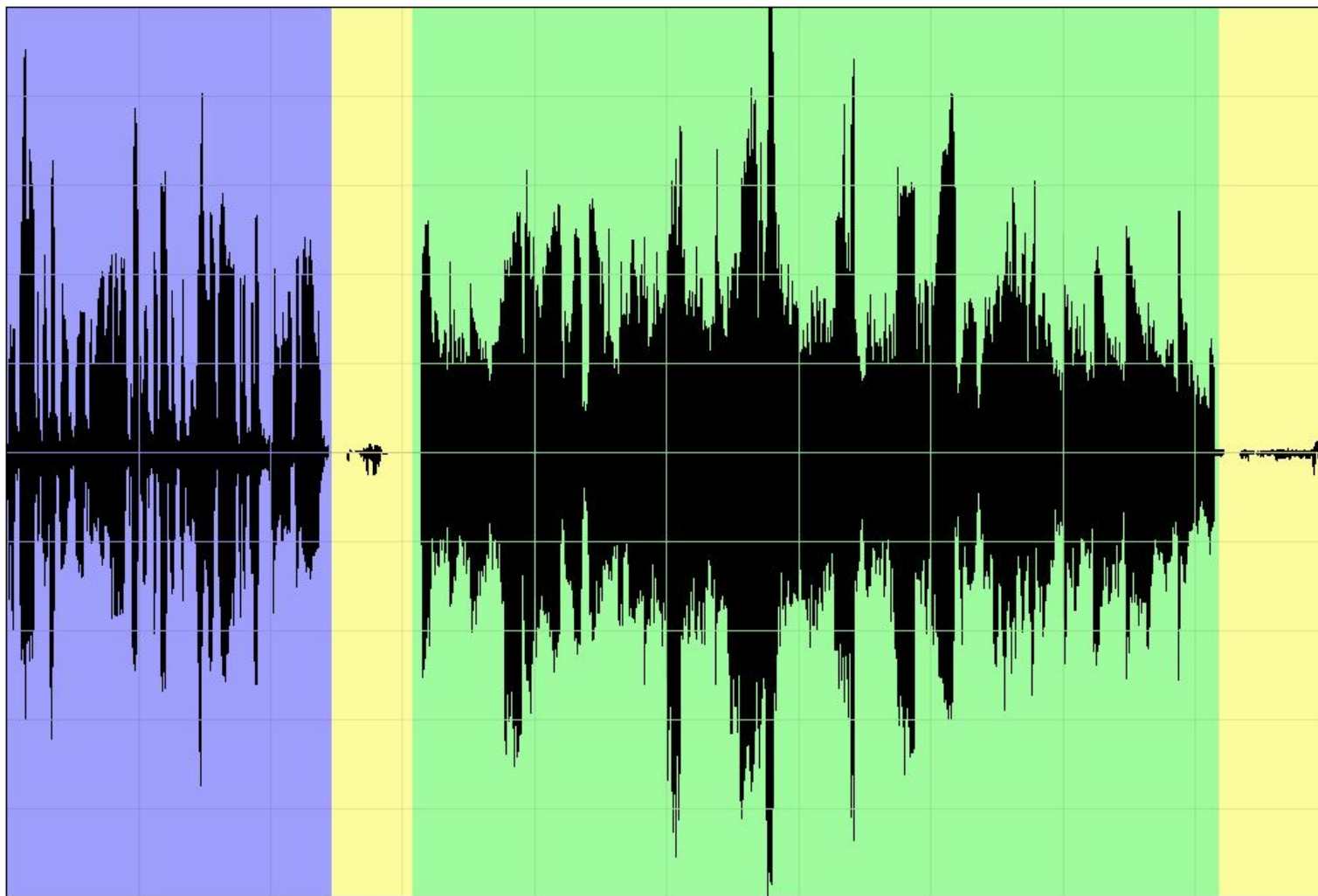


Сценарий эксперимента

```
MainWindow - [D:/Диплом/scenario.bt*]  
Qt File Experiment View Data Base  
Code Logs Results  
1: fileList | CEPSTRUM, S_M_S_CLASSIFIER -> speech_music;  
  
2: learnAnger | WAVELET_FEATURES, EMOTIONS;  
3: learnDisgust | WAVELET_FEATURES, EMOTIONS;  
4: learnJoy | WAVELET_FEATURES, EMOTIONS;  
5: learnSorrow | WAVELET_FEATURES, EMOTIONS;  
6: learnBoredom | WAVELET_FEATURES, EMOTIONS;  
7: learnFear | WAVELET_FEATURES, EMOTIONS;  
8: learnNeutral | WAVELET_FEATURES, EMOTIONS;  
  
9: &speech_music | WAVELET_FEATURES, EMOTIONS -> emotions;  
  
10: learnClassic | WAVELET_FEATURES, GENRES;  
11: learnRock | WAVELET_FEATURES, GENRES;  
12: learnPostRock | WAVELET_FEATURES, GENRES;  
13: learnFolk | WAVELET_FEATURES, GENRES;  
14: learnElectronic | WAVELET_FEATURES, GENRES;  
  
15: &speech_music | WAVELET_FEATURES, GENRES -> genres;  
  
<@ 1>  
<CEPSTRUM>  
  <size>1024</size>  
  <step>512</step>  
  <banksCount>12</banksCount>  
</CEPSTRUM>;  
  
<S_M_S_CLASSIFIER>  
  <vectSize>12</vectSize>  
</S_M_S_CLASSIFIER>;  
<@>  
  
//Setting for emotions  
<@ 2>  
<WAVELET_FEATURES>  
  <segmentSize>1024</segmentSize>
```



Сегментация речь/музыка/тишина



речь

тишина

музыка

тишина

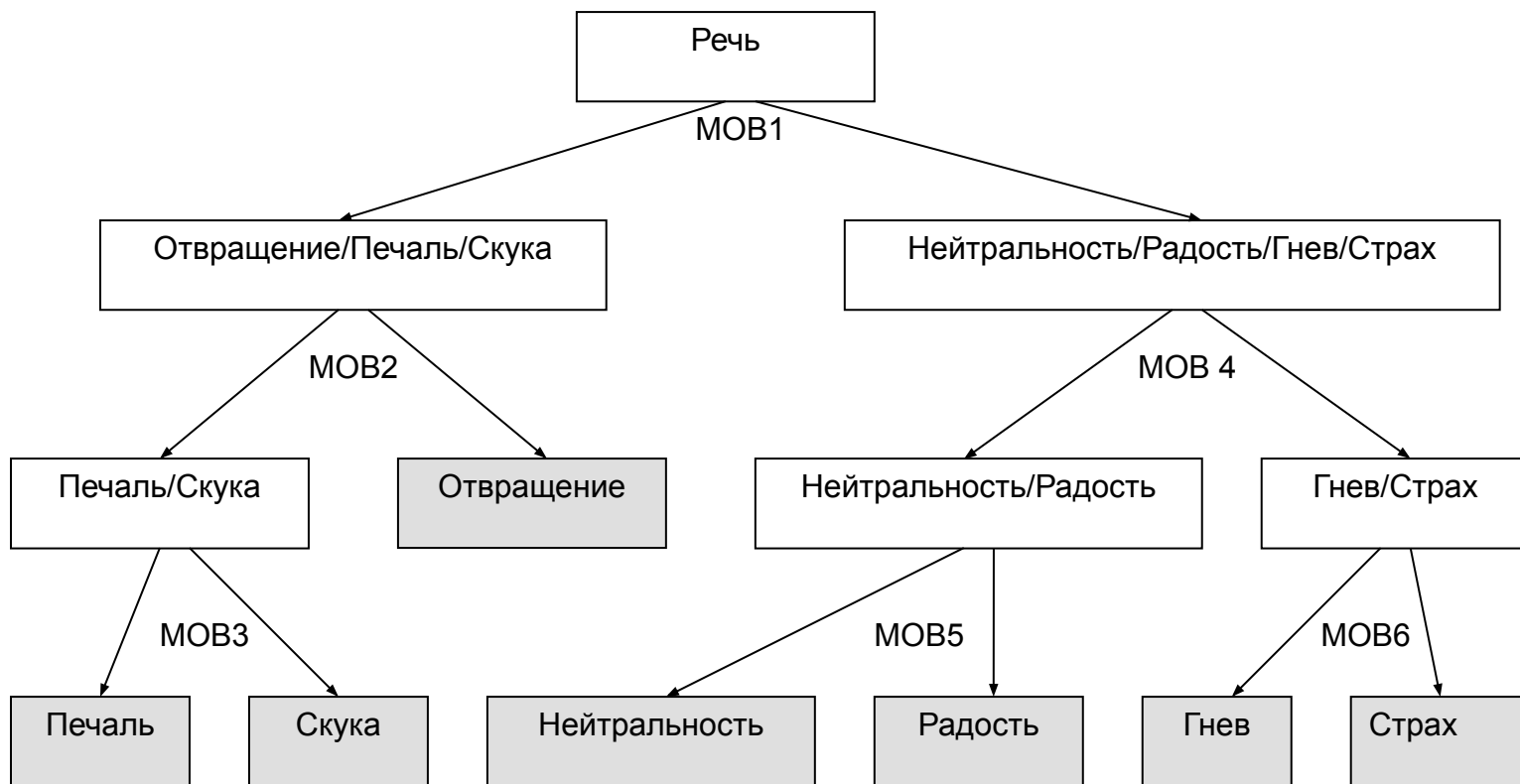


Сегментация речь/музыка/тишина

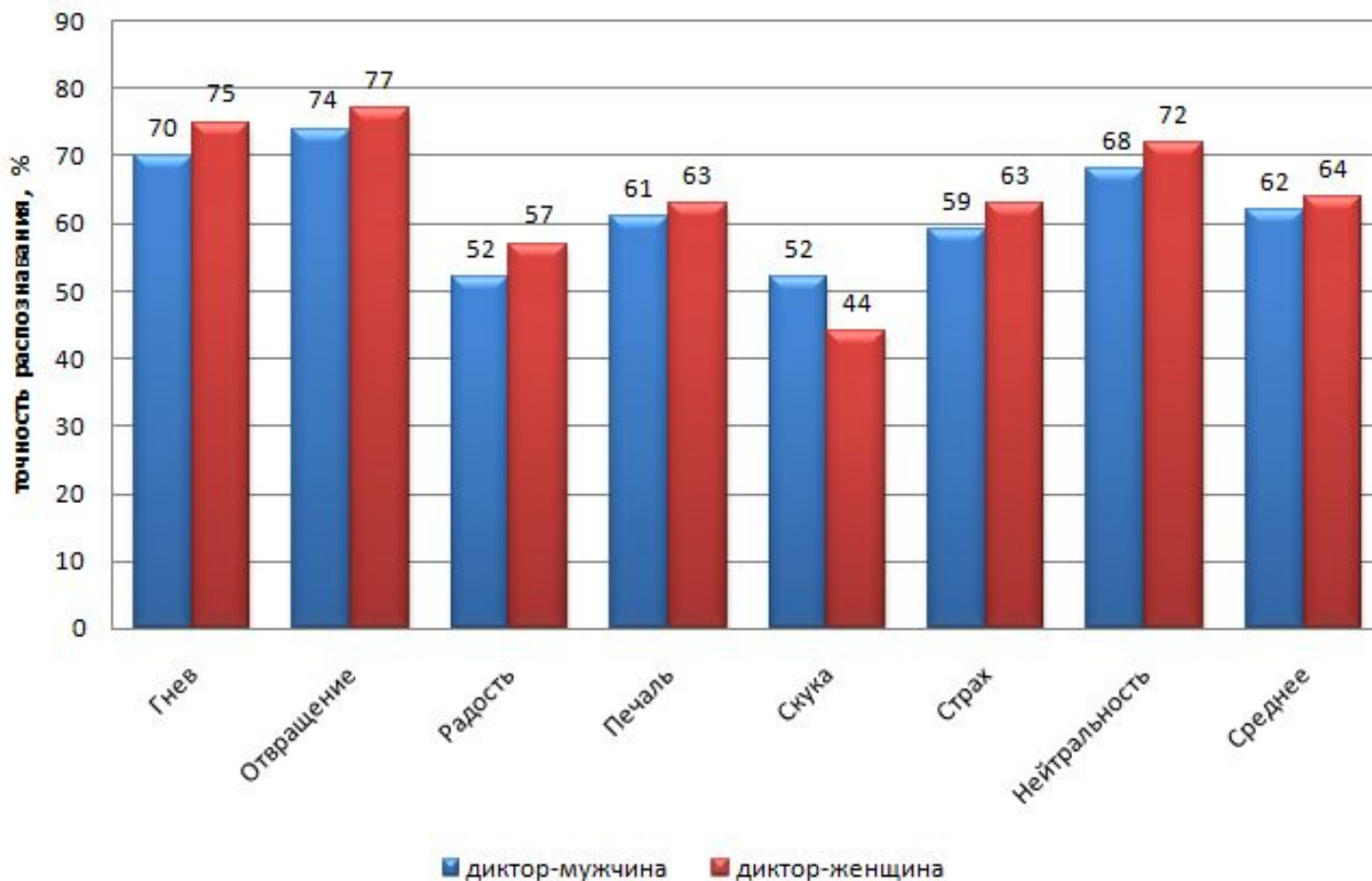
	Корректно	Критические ошибки	Некритические ошибки
Музыка(музыкальная база)	89.2%	3.7%	7.1%
Речь (языковая база)	92.3%	2.9%	4.8%
Музыка (языковая база)	87.4%	4.2%	8.4%



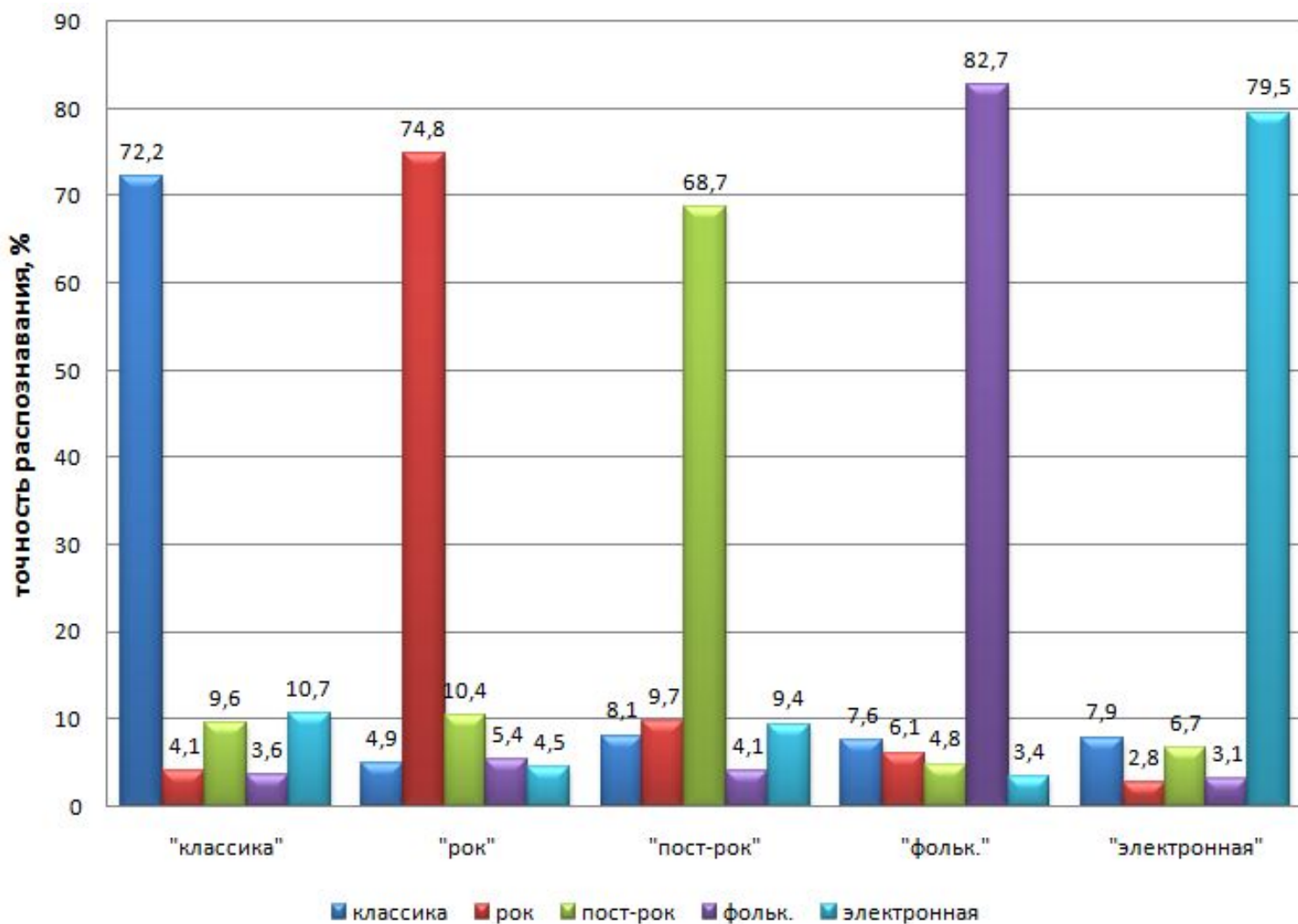
Распознавание эмоций на основе МОВ



Результаты распознавания эмоций на основе МОВ



Результаты определения жанров музыки на основе СММ



Заключение

В рамках данного проекта была выполнена следующая работа:

- разработаны структуры системы индексации аудиопоследовательностей и ее компонент, позволяющие легко модифицировать параметры системы для конкретной задачи индексации.
- разработан интерфейс компонент системы, позволяющий легко подключать к системе компоненты сторонних разработчиков с целью расширения возможностей системы.
- реализована система индексации аудиопоследовательностей на языке C++, графический пользовательский интерфейс системы и базовые компоненты индексации аудиопоследовательностей.
- На основе разработанной системы проведены эксперименты по разделению аудиопоследовательности на речь/музыку/тишину, распознаванию эмоций в речи, определению жанров музыки.



Спасибо за внимание

