

Лабораторная работа №10. Обработка голоса

Программное обеспечение:

- Утилита ffmpeg (сайт www.ffmpeg.org).
- Библиотеки python на выбор: numpy, pydub, scipy.signal, matplotlib.pyplot, pylab, pywt, pyWavelets, librosa и другие.

1. Каждый студент с помощью микрофона записывает звуковые дорожки в формате *.wav (можно и mp3, но не все библиотеки с ходу его понимают, и лучше один канал, чтоб стерео не мешало):
 - а) Образец собственного голоса для звука «А» с максимальным частотным диапазоном (протяжно от баса до визга и даже фальцета), так чтобы громкость была более-менее одинаковой. Длительность до 10 секунд.
 - б) Образец собственного голоса для звука «И», аналогично.
 - в) Имитация собачьего лая (молодые люди) и мяуканья (девушки), крик Тарзана¹ (самые смелые).
2. Построить спектрограммы и сохранить в файл, пользуясь оконным преобразованием Фурье с окном Ханна (например). Частоты имеет смысл визуализировать на *логарифмической* шкале.
3. Найти минимальную и максимальную частоту голоса.
4. Найти наиболее тембрально окрашенный основной тон — частота, для которой прослеживается наибольшее количество обертонов.
5. Найти три самые сильные форманты для издаваемых звуков — частоты с наибольшей энергией в некоторой окрестности: шаг $\Delta t = 0,1$ с. и $\Delta f = 40\text{--}50$ Гц. Убедиться, что форманты для «А» и «И» разные и как-то близки к теоретическим. *Будьте осторожны: если в начале шаг по частоте у вас большой (например, 128 Гц), то все форманты будут одинаково грубо посчитаны и совпадут. Шаг нужен маленький, скажем, 10 Гц.*

¹ https://upload.wikimedia.org/wikipedia/en/9/97/John_Weissmuller%27s_MGM_Tarzan_Yell.ogg