Technical Deep Dive: Audio Feature Extraction Program Analysis

Overview

Tremor Check uses a sophisticated Swift implementation for extracting acoustic features from audio recordings, particularly focused on voice analysis parameters. It uses the Accelerate and AVFoundation frameworks to process audio data and calculate various acoustic measurements commonly used in voice analysis and speech pathology.

Core Architecture

Main Entry Point

The primary function `extractAudioFeatures(from:)` serves as the entry point, accepting an audio file URL and returning a dictionary of computed features. The function:

1. Loads the audio file using AVAudioFile
2. Converts it to a standardized format (44.1kHz, mono, 32-bit float)
3. Processes the audio buffer
4. Calculates multiple acoustic features
5. Returns a comprehensive feature set

Feature Categories

The app extracts four main categories of acoustic features:

1. **Fundamental Frequency Features**
   - F0 (fundamental frequency)
   - Fhi (highest fundamental frequency)
   - Flo (lowest fundamental frequency)
2. **Perturbation Features**
   - **Jitter Measurements**
     - Jitter (%)
     - Absolute Jitter
     - RAP (Relative Average Perturbation)
     - PPQ (Period Perturbation Quotient)
     - DDP (Difference of Differences of Periods)
   - **Shimmer Measurements**
     - Shimmer (%)
     - Shimmer (dB)

    - APQ3/APQ5 (Amplitude Perturbation Quotients)
    - DDA (Difference of Differences of Amplitudes)
3. **Spectral Features**
    - HNR (Harmonics-to-Noise Ratio)
    - NHR (Noise-to-Harmonics Ratio)
    - RPDE (Recurrence Period Density Entropy)
    - DFA (Detrended Fluctuation Analysis)
    - Spread1 and Spread2 (Spectral Spread Measures)
4. **Nonlinear Dynamics Features**
    - D2 (Correlation Dimension)
    - PPE (Pitch Period Entropy)

Key Algorithms

1. Fundamental Frequency Analysis

The app uses autocorrelation for pitch detection:

```
func calculateFundamentalFrequencies(audioData: [Float], sampleRate:
Double)
```

- Divides audio into overlapping frames
- Applies autocorrelation using vDSP_conv
- Identifies peaks to determine pitch period
- Converts period to frequency
- Validates frequency within human voice range (50-500 Hz)

2. Jitter Analysis

Multiple jitter measurements are implemented:

- Percentage jitter: Relative cycle-to-cycle variation
- Absolute jitter: Average absolute difference between consecutive periods
- RAP: Three-point average comparison
- PPQ: Five-point average comparison
- DDP: Second-order perturbation

3. Shimmer Analysis

The app implements various shimmer calculations:

- Peak amplitude detection in 10ms windows
- Percentage and dB calculations of amplitude variations

- Multi-period perturbation quotients (APQ3, APQ5)
- Difference of differences analysis (DDA)

4. Spectral Analysis

Advanced spectral features are computed using:

- Fast Fourier Transform (FFT) via vDSP
- Harmonic-to-noise calculations
- Power spectrum analysis
- Statistical measures of spectral spread

5. Nonlinear Dynamics

Sophisticated chaos theory metrics are implemented:

- Phase space reconstruction for D2
- Entropy calculations for PPE
- Recurrence analysis for RPDE
- Fractality analysis via DFA

Feature Calculation Methods

FFT-Based Analysis

```
func calculateHarmonicEnergy(audioData: [Float]) -> Double
```

- Creates FFT setup with appropriate radix
- Performs forward FFT transform
- Calculates magnitude spectrum
- Analyzes harmonic components

Statistical Analysis

```
func calculateDFA(audioData: [Float]) -> Double
```

- Implements detrended fluctuation analysis
- Uses windowing technique
- Performs linear regression
- Calculates scaling exponent

Entropy Calculations

```
func calculateRPDE(audioData: [Float], m: Int = 2, r: Double = 0.2) ->
Double
```

- Performs phase space reconstruction
- Calculates recurrence periods
- Computes entropy measure
- Returns normalized entropy value