**in the report first include all the part one details and next enter the part 02 details**

Certainly! Below is a **detailed report template** for your assignment, with all **Part 01** (Task 1) details first, followed by **Part 02** (Task 2), as required by your PDF ([Simple-Audio-Classification-Using-Feature-Extraction-1-1.pdf][[1]](#fn1)).

**Simple Audio Classification Using Feature Extraction**

**A. Introduction**

This project addresses the problem of classifying audio signals using digital signal processing and feature extraction. The work is divided into two parts:

* **Part 1:** Classify unknown audio files as belonging to either class\_1 or class\_2 using feature extraction and distance-based comparison.
* **Part 2:** Design a filter-based DSP pipeline to distinguish ambulance and firetruck sounds, using spectral analysis, filter design, and energy-based features.

Our approach leverages Mel-Frequency Cepstral Coefficients (MFCCs) for robust feature extraction and applies both distance-based and filter-based classification as outlined in the assignment.

**B. Methodology**

**Part 1: Audio Classification Using Feature Extraction**

**1. Feature Extraction**

We extracted features from all audio files using **MFCCs** (Mel-Frequency Cepstral Coefficients), as recommended in the assignment. MFCCs are computed by:

* Framing the audio signal and applying a window function.
* Computing the FFT to obtain the spectrum.
* Passing the spectrum through a Mel-scaled filter bank.
* Taking the log energies and applying the Discrete Cosine Transform (DCT) to obtain the final coefficients.

**Mathematical Details:**

* For a frame $ x[n] $, the spectrum is:
* MFCCs are:where $ E\_n $ is the energy in the $ n $-th Mel filter.

**2. Similarity/Distance Measures**

We used **Euclidean distance** to compare feature vectors:

**3. Classification Task**

* Extract MFCCs for all training and unknown files.
* For each unknown file, compute the distance to all files in class\_1 and class\_2.
* Assign the file to the class with the closest match (minimum distance).

**4. Justification**

MFCCs are robust to noise and capture perceptual characteristics. Euclidean distance is effective and simple for comparing feature vectors. Both are recommended in the assignment PDF.

**Part 2: Filter-Based Classification of Emergency Vehicle Sounds**

**1. Spectral Analysis**

* We analyzed the FFT spectrum of ambulance and firetruck training files.
* Ambulance signals showed strong energy in the 300–700 Hz and 700–1200 Hz bands.
* Firetruck signals showed dominant energy in the 1200–2500 Hz band.

**2. Filter Design**

* We designed **three bandpass filters** using MATLAB’s designfilt:
  + Low: 300–700 Hz (ambulance wail)
  + Mid: 700–1200 Hz (ambulance yelp, firetruck overlap)
  + High: 1200–2500 Hz (firetruck horn)
* All filters are 8th-order IIR filters.

**Mathematical Formulation:**

**3. Feature Extraction**

* For each file, filter the signal using all three filters.
* Compute the energy in each band:
* Form a feature vector: [energy\_low, energy\_mid, energy\_high, ratios, stds].

**4. Threshold/Classification**

* Initially, we used a threshold on the energy ratio (ambulance/firetruck).
* For higher accuracy, we used a **KNN classifier** trained on the filter-bank feature vectors.

**5. Justification**

A filter bank captures more discriminative information than a single ratio. KNN can find optimal boundaries in multi-dimensional feature space.

**C. Implementation Details**

**Code Structure**

* Both parts are implemented in MATLAB as Part\_01.m and Part\_02.m.
* Each script is organized as:
  1. Data loading and preprocessing
  2. Feature extraction
  3. Classifier training (KNN)
  4. Testing and results output

**Classification Logic**

* **Part 1:** For each unknown file, extract MFCCs, compare to training samples using KNN, assign to the closest class.
* **Part 2:** For each test file, extract filter-bank features, use the trained classifier to predict the class.

**Tools Used**

* MATLAB (with Audio Toolbox for mfcc, designfilt, etc.)

**D. Results**

**Part 1: Unknown File Classification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| File Name | Predicted Class | Confidence | Min Dist to Class 1 | Min Dist to Class 2 |
| unknown1.wav | class\_1 | 0.93 | 1.12 | 2.45 |
| unknown2.wav | class\_2 | 0.88 | 2.05 | 1.34 |
| ... | ... | ... | ... | ... |

**Accuracy:** XX% (e.g., 95%)

**Part 2: Ambulance vs Firetruck Classification**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| File Name | True Class | Predicted Class | Confidence | Feature Vector (summary) |
| test1.wav | ambulance | ambulance | 0.91 | [0.8, 0.3, 0.1, ...] |
| test2.wav | firetruck | firetruck | 0.89 | [0.2, 0.5, 0.7, ...] |
| ... | ... | ... | ... | ... |

**Accuracy:** YY% (e.g., 92%)

**E. Conclusion**

* **What worked well:**
  + MFCCs with KNN provided robust classification for unknown files.
  + Filter-bank features with a KNN classifier significantly outperformed simple energy ratios in distinguishing ambulance and firetruck sounds.
* **Challenges:**
  + Initial attempts using only energy ratios and thresholds yielded low accuracy.
  + Overlapping frequency content and noisy recordings reduced separability.
* **Improvements/Alternatives:**
  + Using a richer feature vector (multiple bands, MFCCs) and a classifier like SVM/KNN greatly improved performance.
  + In future work, deep learning or more advanced feature selection could be explored.

**References**

* Assignment PDF: [Simple-Audio-Classification-Using-Feature-Extraction-1-1.pdf][[1]](#fn1)
* [MFCC Theory and Applications](https://en.wikipedia.org/wiki/Mel-frequency_cepstrum)
* [MATLAB MFCC Documentation](https://www.mathworks.com/help/audio/ref/mfcc.html)
* [Bandpass filter - Wikipedia](https://en.wikipedia.org/wiki/Band-pass_filter)

**This report structure is ready for you to fill in your actual results (tables, accuracy numbers) and any additional insights. It is fully compliant with your assignment PDF and includes all required mathematical and methodological details, with Part 1 first and Part 2 second as requested.**

⁂

1. Simple-Audio-Classification-Using-Feature-Extraction-1-1.pdf