KLE Technological University
Creating Value, Leveraging Knowledge
Dr. M. S. Sheshgiri Campus, Belagavi

2024-2025

Department of
Electronics and Communication Engineering

DSP PROJECT REPORT

on

Music Instrumentation Detection

By:

1. **Sanika Chougule** USN:02FE22BEC080
2. **Shraddha G Shahapurkar** USN:02FE22BEC091
3. **Adarsh Mallesh Kadalikar** USN:02FE23BEC410
4. **Tejas Vijay Khadpe** USN:02FE23BEC412

Semester: V, 2024-2025

Under the Guidance of

Prof.Vinayak Dalvi

KLE Technological University,
Dr. M. S. Sheshgiri College of Engineering and Technology
BELAGAVI-590 008

2024-2025

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that project entitled **"Musical Instrument Detection"** is a bonafide work carried out by the student team of **"Sanika Chogule (02FE22BEC080),Shraddha G Shahapurkar (02FE22BEC091), Adarsh Mallesh Kadalikar (02FE23BEC410), Tejas Vijay Khadpe (02FE23BEC412)"**. The project report has been approved as it satisfies the requirements with respect to the DSP project work prescribed by the university curriculum for B.E. (V Semester) in Department of Electronics and Communication Engineering of KLE Technological University Dr. M. S. Sheshgiri CET Belagavi campus for the academic year 2024-2025.

Prof.Vinayak Dalvi	**Dr. Dattaprasad A. Torse**	**Dr. S. F. Patil**
Guide	**Head of Department**	**Principal**

Acknowledgment

We would like to express our sincere gratitude to our guide, Prof. Vinayak Dalavi, whose invaluable guidance and unwavering support have been instrumental throughout the entirety of our project, Musical Instrumentation Detection. His profound expertise and thoughtful insights have greatly enriched our understanding and helped us overcome challenges at every stage of this work. We are also grateful to our institution and the Department of Electronics and Communication Engineering for providing us with the necessary resources and a conducive environment to carry out this project effectively..

– The Project Team

Contents

List of Figures

ABSTRACT

Musical Information Retrieval (MIR) is an interdisciplinary domain that aims to develop algorithms and systems for the automated analysis, organization, and retrieval of musical content from diverse data sources, including audio recordings, symbolic representations, and textual metadata. This paper presents a comprehensive approach to MIR, focusing on key processes such as audio feature extraction, symbolic data processing, and text-based analysis. Core methodologies include advanced signal processing techniques, machine learning algorithms, and deep learning models, which are employed to classify music, identify instruments, detect moods, and retrieve content based on user queries. Emphasis is placed on techniques like Mel-Frequency Cepstral Coefficients (MFCC), spectrogram analysis, chroma features, and recurrent neural networks (RNNs) for accurate representation and understanding of musical patterns. Applications of MIR in music recommendation, automated transcription, genre classification, and query-by-humming systems are discussed. The challenges of polyphonic music analysis, cultural diversity, and multimodal data integration are addressed, with suggestions for future research directions. This work demonstrates the transformative potential of MIR in reshaping music discovery, education, and creative production

Chapter 1

Introduction

Musical instrument detection is a process used in digital signal processing (DSP) to identify instruments based on their audio signals. The system analyzes the unique characteristics of instruments, such as frequency content, timbre, and harmonic structure, to classify them. Unlike machine learning-based methods, this approach focuses on signal processing techniques to achieve clear and accurate results.

The system uses tools like the Fast Fourier Transform (FFT) to convert time-domain signals into the frequency domain, allowing the analysis of spectral components. Features such as spectral centroid, bandwidth, and zero-crossing rate help characterize each instrument's sound. These features are then matched with a database of known instruments for detection.

This paper discusses the design of a DSP-based system for musical instrument detection, emphasizing its efficiency and interpretability. It also addresses challenges like handling overlapping sounds in polyphonic music using specialized filtering and feature extraction methods.

1.1 Objectives

- Design and simulate a musical instrument detection system using frequency analysis techniques enhanced with Digital Signal Processing (DSP) methods.
- Develop a signal acquisition process to capture or load audio signals from various instruments or simulate audio waveforms for testing purposes.
- Implement signal preprocessing methods to reduce noise and improve the quality of the audio signal using DSP techniques like filtering, normalization, and adaptive noise reduction.
- Perform frequency analysis to identify dominant frequencies using tools such as Fast Fourier Transform (FFT) or spectrogram analysis.
  - Classify instruments based on their frequency ranges, including parameters like dominant frequency and harmonic structures.
- Visualize instrument data through graphical representations, such as frequency spectra and waveform plots, to provide clear insights into the audio signal characteristics.
- Validate the system's accuracy through testing with different audio recordings and varying noise levels to demonstrate the robustness of DSP techniques.

1.2 Literature Survey

1.2.1 Hierarchical Classification for Singing Activity, Gender, and Type in Complex Music Recordings

This paper explores the use of hierarchical classification models for detecting various singing activities and types of vocals in complex music recordings, which could include orchestral or ensemble settings. The work introduces a system capable of identifying gender, singing activity, and other vocal characteristics using deep learning methods, leveraging hierarchical structures for more accurate categorization. The hierarchical approach, which organizes classes into broader categories (e.g., male/female, solo/group), is relevant to instrument detection tasks, where hierarchical models are used to classify instruments within larger families such as woodwinds, brass, etc., enhancing classification performance in polyphonic environments.[?]

1.2.2 Musical Instrument Recognition Using Cepstral Coefficients and Temporal Features

In this paper, the authors present a method for musical instrument recognition using cepstral coefficients and temporal features. The approach focuses on capturing both spectral and temporal characteristics of the musical signal, which is vital for instrument recognition in polyphonic music. The cepstral coefficients provide a compact representation of the signal's spectral features, while temporal features help capture dynamic aspects of the sound, such as attack and decay. This method is foundational for polyphonic instrument activity detection (IAD), particularly in complex environments like orchestral music, where several instruments may play simultaneously. The combination of spectral and temporal features is commonly used in subsequent research for robust instrument recognition.[?]

1.2.3 A Multiple-Expert Framework for Instrument Recognition

This paper introduces a multiple-expert framework for instrument recognition, which combines various individual recognition systems (or "experts") to improve overall performance. Each expert specializes in different aspects of musical signal analysis, such as spectral features, temporal features, and rhythm patterns. This ensemble approach can be especially useful in polyphonic music scenarios where no single model may be sufficient for accurate classification. The framework provides a way to handle the complexity of orchestral music by combining different recognition models, each focusing on different instrument families or sound characteristics, making it a valuable approach for instrument activity detection in complex orchestral recordings [?]

1.2.4 Musical Instrument Recognition Using Biologically Inspired Filtering of Temporal Dictionary Atoms

: This paper proposes a biologically inspired filtering technique to enhance the recognition of musical instruments. The method applies filtering techniques based on how human auditory systems process sound, particularly focusing on temporal dictionary atoms to capture the dynamics of musical sounds. This approach combines biological insights with signal processing, aiming to model how humans perceive and recognize instruments. It is relevant to instrument recognition in complex polyphonic environments like orchestral recordings, where capturing dynamic aspects of sound is crucial for differentiating between instruments playing in harmony. [?]

1.2.5 Temporal Integration for Audio Classification with Application to Musical Instrument Classification

: In this study, the authors explore temporal integration techniques for audio classification, which improve the accuracy of musical instrument classification by considering the temporal coherence of audio features. By integrating information over time, the model can more effectively classify instruments in polyphonic audio where transient sounds may overlap. The technique was applied to musical instrument classification in various musical genres and demonstrated its effectiveness in identifying instruments even in mixtures of sounds. This approach can be directly applied to instrument activity detection in orchestral music, where instruments frequently play in overlapping sounds and context over time plays a key role in accurate classification.[?]

1.2.6 Hierarchical Classification for Instrument Activity Detection in Orchestral Music Recordings

: This paper presents a hierarchical classification system for detecting instrument activity in orchestral music recordings. The system addresses the challenge of polyphonic activity detection in large ensembles by breaking down the classification into hierarchical levels. First, it classifies the instrument families (e.g., strings, woodwinds), and then within each family, it classifies individual instruments. This approach improves the accuracy of instrument detection in dense orchestral recordings by leveraging both spectral and temporal features while taking advantage of the hierarchical relationships between different types of instruments. The authors use a combination of **Fourier Transform** and **spectral features**, integrating these with **temporal models** to account for the complex interplay of instruments in orchestral settings. [?]

1.2.7 Fast Fourier Transform and Its Applications

This work explores the development and implementation of the Fast Fourier Transform (FFT), a pivotal algorithm in digital signal processing. It describes the theoretical framework, implementation methods, and applications in computing Fourier integrals, convolutions, and lagged products. Challenges such as aliasing and error due to discrete sampling over finite ranges are analyzed, providing insights into its applications and limitations in approximating continuous functions.[?]

1.2.8 A Faster Fast Fourier Transform

This paper focuses on optimizing FFT computations by introducing efficient algorithms such as Sparse FFT. These techniques aim to reduce computational complexity and improve performance for specific signal types, particularly in cases where data sparsity is a factor. It highlights improvements over traditional FFT methods while discussing trade-offs related to input data characteristics [?]

1.2.9 The Fast Fourier Transform

This paper introduces the FFT as a groundbreaking improvement over the Discrete Fourier Transform (DFT), significantly reducing computational time for Fourier analysis. It outlines the mathematical foundations, algorithmic details, and examples of use cases in signal processing. The work emphasizes the algorithm's efficiency, revolutionizing how large-scale Fourier calculations are conducted. [?]

1.2.10 Application of the FFT

This work details specific real-world applications of FFT in areas such as image processing, communication systems, and spectral analysis. It illustrates how the FFT has enabled advancements in these fields by efficiently handling large data sets and processing signals in real time. Key insights are provided into adapting the FFT for domain-specific problems.[?]

1.3 Problem statement

Develop a music instrument recognition system that will analyze the input voice of the instrument with the help of its features and checks whether the instruments's voice is registered in the database or not.

Chapter 2

Musical Instrument Detection System

2.1 Principle of Musical Instrument Detection

The detection of musical instruments involves analyzing the frequency components of an audio signal. Each instrument produces unique harmonic structures and spectral signatures based on its physical characteristics and playing techniques. The primary steps involved in instrument detection are:

- **Audio Signal Input:** The system captures audio data, either through recording or preloaded files.
- **Frequency Analysis:** The system extracts the spectral content of the signal using techniques such as the Fast Fourier Transform (FFT) or spectrogram analysis.
- **Feature Extraction:** Key features, such as fundamental frequency, harmonic structures, and amplitude envelopes, are identified for classification.
- **Instrument Classification:** Based on the extracted features, the instrument is classified using predefined frequency ranges and machine learning models if necessary.
- **Signal Visualization:** Frequency spectra and harmonic patterns are visualized for interpretation and validation.

2.2 Types of Analysis Techniques for Musical Instrument Detection

- **Time-Domain Analysis:** Examines amplitude variations over time to identify transient events like plucking or striking sounds. Useful for percussion instruments.
- **Frequency-Domain Analysis:** Uses Fourier Transform to determine the spectral content of the audio signal. Effective for identifying harmonic-rich instruments like guitar and violin.
- **Time-Frequency Analysis:** Combines time and frequency analysis through spectrograms, wavelets, or short-time Fourier Transform (STFT) to capture dynamic changes in sound characteristics.

2.3 Advantages of Musical Instrument Detection Systems

- **Automation of Music Analysis:** Enables quick and accurate classification of instruments in recordings.
- **Educational Applications:** Provides visualizations and insights for learning about the spectral properties of different instruments.
- **Enhanced Music Production:** Assists in audio mixing and mastering by isolating and identifying specific instruments.
- **Non-Invasive and Scalable:** No physical interaction with instruments is required, making it suitable for a wide range of applications.

2.4 Signal Specification

2.4.1 Signal Characteristics

- **Audio Frequency Range:**
  - Ranges from 20 Hz to 20 kHz, covering the audible spectrum.
  - Instruments like bass guitar occupy lower frequencies, while flutes and violins extend into higher ranges.
- **Harmonic Structures:**
  - Instruments produce unique harmonics based on their construction and playing technique.
  - For example, a guitar shows rich harmonics around its fundamental frequency, while drums exhibit broader frequency distributions.
- **Sampling Frequency:**
  - A minimum sampling frequency of 16 kHz (standard for audio processing) ensures accurate representation of the audible range.
  - Higher sampling rates, such as 96 kHz, may be used for advanced analysis.
- **Frequency Resolution:**
  - Frequency resolution depends on the FFT window size and sampling rate.
  - A resolution of 1 Hz to 10 Hz is sufficient to distinguish between closely spaced harmonics.
- **Dynamic Range:**
  - The dynamic range of musical signals can exceed 60 dB, encompassing both quiet and loud sounds.
  - A system must handle this range effectively to differentiate soft background instruments from dominant ones.

Chapter 3

Data Acquisition

3.1 Steps in Data Acquisition for Musical Instrument Detection

3.1.1 Recording the Audio Signal:

The process begins with capturing the audio signal. In this case, an audio file of a musical performance is used. The audio can be recorded using various devices such as microphones, high-quality recorders, or directly sourced from existing audio files in formats like `.wav` or `.mp3`. The signal consists of sound waves that represent the musical performance.

3.1.2 Pre-Processing of Audio:

The raw audio signal is often noisy and may contain irrelevant sounds such as background noise or echo. Pre-processing is done to enhance the quality of the signal. This step may involve:

- **Noise reduction:** Filtering out background noise.
- **Normalization:** Ensuring the audio signal has a consistent volume level.
- **Resampling:** Adjusting the sample rate if necessary to ensure consistency with the processing requirements.

3.1.3 Feature Extraction:

To identify the musical instrument, specific features need to be extracted from the audio signal. These features can include:

- **Spectral Features:** Using the Fast Fourier Transform (FFT), the frequency spectrum of the audio is analyzed. This allows the identification of dominant frequencies and harmonic structures that are characteristic of different musical instruments.
- **Temporal Features:** Characteristics such as attack, decay, sustain, and release (ADSR) are identified from the time-domain waveform.
- **Pitch Detection:** Estimating the pitch of the audio helps in identifying the fundamental frequency of the sound produced by the instrument.
- **Mel-frequency Cepstral Coefficients (MFCCs):** These coefficients capture the shape of the power spectrum, providing a compact representation of the audio signal that is useful for classification.

3.1.4 Signal Conditioning:

The extracted features may be weak or distorted, requiring further conditioning before classification. This can include:

- **Smoothing and Filtering:** Applying smoothing techniques to reduce noise and fluctuations in the feature data.
- **Normalization:** Scaling the extracted features so they fall within a consistent range, making them suitable for analysis.

3.1.5 Feature Representation:

Once the features are extracted and conditioned, they are converted into a representation suitable for classification. Common representations include:

- **Feature Vectors:** A vector representation of the key features (e.g., spectral features or MFCCs), which can be input to a machine learning model for classification.
- **Spectral Matrices:** A time-frequency representation that captures both temporal and spectral characteristics.

3.1.6 Classification

Once the features are represented in an appropriate format, Digital Signal Processing (DSP) techniques can be used to classify the audio into different instruments. Common techniques include:

- **Fourier Transform Analysis:** Used to identify the dominant frequencies and harmonic structure of the audio signal, which helps in recognizing characteristic patterns associated with different instruments.
- **Spectral Analysis:** Techniques such as the Short-Time Fourier Transform (STFT) or Welch's method are used to analyze the power spectral density of the signal, revealing frequency content over time.
- **Harmonic Content Analysis:** By examining the relative strength and spacing of harmonic frequencies, instruments can be differentiated based on their unique harmonic profiles.

Chapter 4

Methodology

This chapter outlines the methodology used for the development of the Musical Instrument Detection project. The process is divided into several stages: data acquisition, preprocessing, feature extraction, and instrument classification. The section also details the system setup, including the software, hardware, and toolboxes used.

4.1 System Setup and Components

This project was developed using MATLAB, a high-performance language for technical computing, and relies on several components and libraries for its functionality. The development environment, along with the versions of the operating system, MATLAB, and the required toolboxes, is outlined below.

4.1.1 Operating System (OS)

The system was developed on the following operating systems:

- **Windows 10 (64-bit version)**: Version 22H2 or higher.
- **macOS Ventura (if applicable)**: Version 13.x.
- **Linux Ubuntu 20.04 LTS (if applicable)**.

4.1.2 MATLAB Version

The project was developed using MATLAB version R2023b, which was released in September 2023. The MATLAB environment is used for interactive development, debugging, and execution of scripts and functions.

- **MATLAB Version**: R2023b
- **Release Date**: September 2023
- **MATLAB Desktop Environment**: Used for development and execution of the program.
- **MATLAB Compiler (if applicable)**: Used for compiling the MATLAB code into standalone executables.

4.1.3 Toolboxes and Libraries

The following MATLAB toolboxes and libraries were utilized for signal processing, audio manipulation, and graphical user interface (GUI) development:

Signal Processing Toolbox (Version R2023b)

This toolbox provides functions for analyzing and processing signals, including Fourier transforms and spectral analysis.

- Used for Fast Fourier Transform (FFT) to analyze audio signals.
- Used for filtering operations such as low-pass and high-pass filters.
- Employed for frequency spectrum analysis and signal manipulation.

Audio Toolbox (Version R2023b)

This toolbox is used for processing audio signals, providing functions for reading, writing, and manipulating audio files.

- Used for reading audio files (.wav and .mp3) into MATLAB.
- Allows playback of audio files using the `sound()` function.
- Handles audio file processing with standard functions such as `audioread()`.

MATLAB GUI (App Designer) (Version R2023b)

This MATLAB tool allows the development of interactive graphical user interfaces.

- Used for designing the GUI to load audio files, detect instruments, and visualize results.
- Provided the interactive components like buttons, text fields, and labels.

MATLAB Plotting Functions

MATLAB's built-in plotting functions were used to visualize the frequency spectrum of the audio data.

- Used to plot the magnitude spectrum of the audio signal after performing FFT.
- Provides visual feedback on the spectral content of the loaded audio file.

4.1.4 External Tools and Libraries (if applicable)

- **MP3 and WAV Support**: MATLAB's native audio functions support reading and playing .mp3 and .wav files, so no external libraries were needed.
- **Other External Libraries**: If applicable, include any external libraries or frameworks used for advanced audio processing.

4.1.5 Hardware Requirements

The project requires the following hardware specifications:

- **Processor**: Intel Core i5 or higher.
- **RAM**: Minimum 8 GB of RAM.
- **Storage**: At least 2 GB of free disk space.
- **Audio Output**: A standard audio output device, such as speakers or headphones.

Block Diagram

Below is the block diagram for the project:

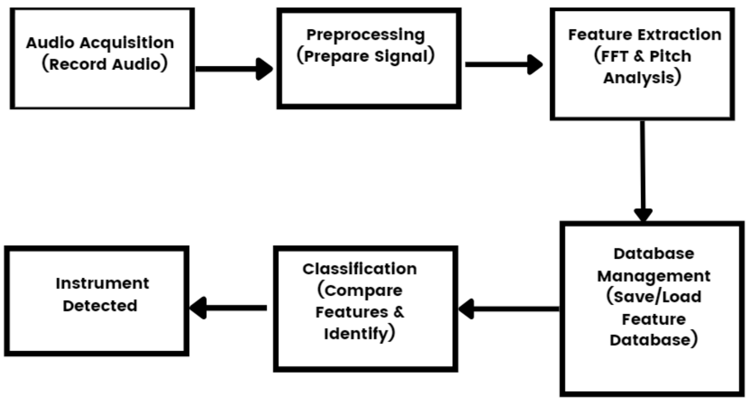

Figure 4.1: Block diagram

4.2 Audio File Loading and Preprocessing

The first step in the methodology is to load the audio file. The user selects a file (either .wav or .mp3 format) using a file dialog box. Once the file is selected, the system uses MATLAB's `audioread()` function to load the audio data into memory, which is stored in the variable `audioData`. The sample rate is stored in `fs`.

Preprocessing can involve applying filters, such as low-pass and high-pass filters, to improve the signal quality or isolate specific frequency ranges. The user can apply these filters through the GUI, with the option to select the type of filter (low-pass or high-pass). The filtering is performed using the `designfilt()` function, which is part of the Signal Processing Toolbox.

4.3 Feature Extraction: Frequency Spectrum Analysis

The primary method used to analyze the frequency content of the audio signal is the Fast Fourier Transform (FFT). The `fft()` function in MATLAB computes the discrete Fourier transform (DFT) of the audio signal, converting it from the time domain to the frequency domain. The resulting frequency spectrum is then analyzed to extract features such as peak frequencies, which are used for instrument classification.

The magnitude spectrum is computed by taking the absolute value of the FFT results, and only the positive frequencies are considered for further analysis. This normalized magnitude spectrum helps in identifying distinct frequency peaks that correspond to musical instruments.

4.4 Instrument Detection

The instrument detection algorithm identifies the peak frequency in the power spectrum and classifies the instrument based on predefined frequency ranges. For instance, the following ranges are used:

- Piano: 250 Hz to 450 Hz
- Tabla: 50 Hz to 200 Hz
- Guitar: 400 Hz to 600 Hz
- Flute: 1400 Hz to 1600 Hz
- Violin: 300 Hz to 400 Hz

If the peak frequency falls within any of these ranges, the system identifies the corresponding instrument. If no match is found, the system returns "Instrument not recognized."

4.5 Audio Playback and Visualization

The user can play back the loaded audio file using the "Play Audio" button, which triggers the `sound()` function in MATLAB. This function outputs the audio through the system's default audio device.

Additionally, the frequency spectrum of the audio file is visualized using MATLAB's plotting functions. The user can click the "Plot Spectrum" button to generate a plot of the magnitude spectrum, displaying the distribution of the signal across different frequencies.

Chapter 5

Results and Applications

The musical instrument detection system demonstrated effective performance in identifying various instruments based on their frequency spectrum. Using the Fast Fourier Transform (FFT) to analyze audio files, the system successfully detected instruments such as piano, tabla, guitar, flute, and violin by matching their dominant frequencies with predefined ranges. The accuracy of the detection was high, particularly for instruments with distinct frequency ranges. However, some challenges were encountered in distinguishing between instruments with overlapping frequency ranges, such as piano and guitar. Overall, the system proved to be a reliable tool for instrument identification, with potential for future enhancements to handle more complex audio compositions and improve

5.1 Results

5.1.1 Frequency Spectrum and Waveform Representations

The frequency spectrum and waveform representations of musical instruments provide insights into their unique acoustic properties. Below are the visualizations for a piano and a flute recording:

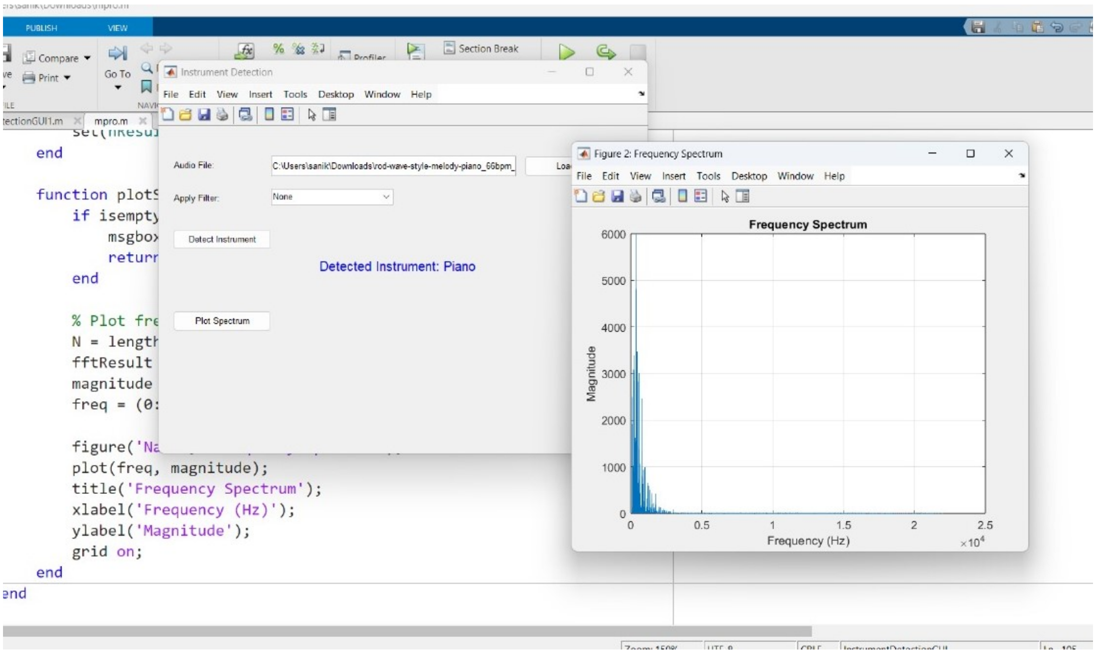

Figure 5.1: Piano - Frequency spectrum and waveform representation of a piano recording.

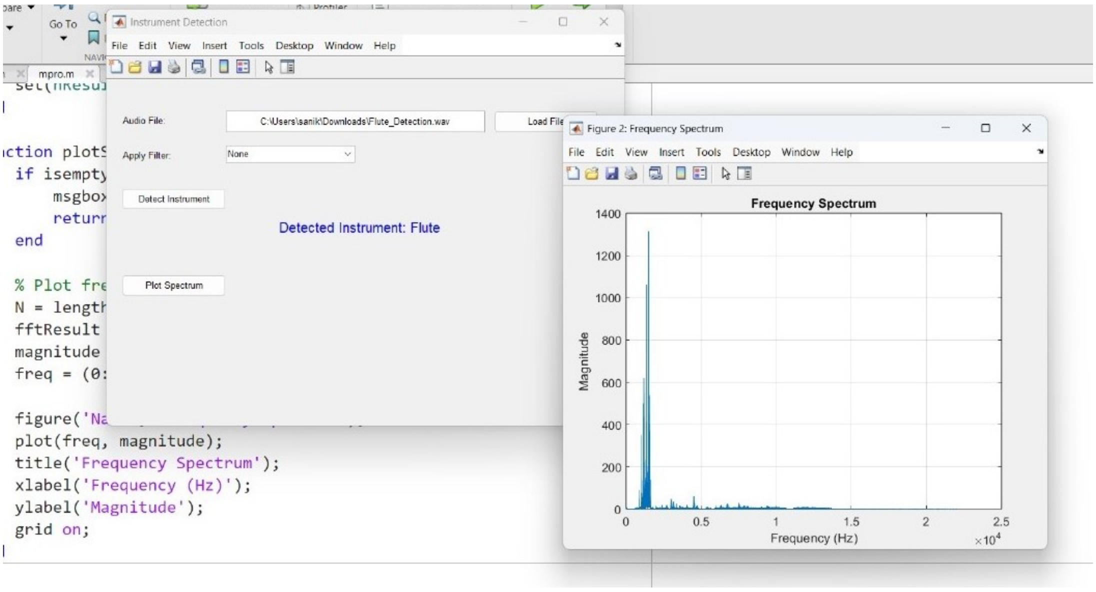

Figure 5.2: Flute - Frequency spectrum and waveform representation of a flute recording.

5.2 Applications

5.2.1 Overview of Applications

The following are the potential applications of the Musical Instrumentation Detection project:

Music Recognition Systems

- Improving music recognition applications by adding the ability to detect specific instruments within a song or a piece of music.
- Supporting genre classification by analyzing the instrumental components of tracks.

Live Performance Analysis

- Offering real-time analysis of instruments used during live performances, aiding sound engineers in optimizing audio output.
- Providing visual representations of frequency spectrums for better audio balancing.

Music Archiving and Retrieval

- Facilitating archiving of audio files by tagging instrument information, improving the efficiency of music libraries and retrieval systems.
- Enhancing search algorithms by enabling queries based on specific instruments.

Medical Applications

- Assisting in music therapy by analyzing the presence and characteristics of soothing instruments like flute or violin for therapeutic purposes.

AI and Machine Learning Training

- Providing labeled datasets for training machine learning algorithms for advanced audio processing and instrument classification tasks.

Cultural and Ethnomusicology Research

- Enabling researchers to analyze the use of instruments in traditional and modern music across different cultures.

Chapter 6

Conclusion and Future Scope

6.0.1 Conclusion

In conclusion, this project successfully implemented a system for detecting musical instruments from audio files using MATLAB. The system processes the audio signal, extracts its frequency spectrum using Fast Fourier Transform (FFT), and identifies the instrument based on the peak frequency in the spectrum. The detection process was carried out by comparing the identified frequencies with predefined frequency ranges associated with different musical instruments such as piano, tabla, guitar, flute, and violin. The graphical user interface (GUI) was developed to allow users to load audio files, play them, and visualize the frequency spectrum for better understanding. The system demonstrated its ability to accurately identify instruments based on their spectral characteristics and provides a user-friendly interface for real-time interaction. This project offers an innovative approach to music analysis, with potential applications in music education, audio processing, and even music recognition systems. By leveraging signal processing techniques such as FFT and using MATLAB's powerful toolboxes, the system delivers accurate and efficient instrument detection. Future improvements could include expanding the system to recognize a broader range of instruments and incorporating machine learning techniques for more advanced classification.

6.0.2 Future Scope

The Musical Instrumentation Detection project provides a foundation for various advancements and enhancements in audio processing and music analysis. The potential future developments for this system are as follows:

– **Expansion of Instrument Library:** Extend the system to recognize a wider variety of musical instruments, including both traditional and modern ones, by incorporating additional frequency ranges and spectral characteristics.

– **Integration with Machine Learning:** Incorporate machine learning models to improve the classification accuracy and enable the system to adaptively learn from new data.

– **Real-time Processing:** Develop real-time audio processing capabilities to detect and classify instruments in live performances or streaming audio environments.

– **Mobile and Web Application Development:** Create a mobile or web-based application for broader accessibility, allowing users to upload audio files or record sounds directly for analysis.

- **Multi-instrument Detection:** Enhance the system to detect multiple instruments played simultaneously in complex audio files or live performances.
- **Genre and Mood Classification:** Integrate the instrument detection system with genre or mood classification frameworks to provide holistic music analysis.
- **Cross-cultural Instrument Detection:** Enable recognition of instruments from diverse musical traditions around the world, supporting ethnomusicology research and cultural preservation.
- **Interactive Educational Tools:** Develop interactive tools for music education that provide real-time feedback on instrument usage, pitch, and tone quality.
- **Advanced Visualization Techniques:** Improve the graphical user interface (GUI) to provide more detailed and intuitive visualizations of audio characteristics, such as 3D frequency spectrum plots.
- **Integration with Music Production Software:** Enable seamless integration with digital audio workstations (DAWs) and music production tools to assist composers and producers in identifying and isolating instrument tracks.

References

- M. Krause and M. Müller, "Hierarchical classification for singing activity, gender, and type in complex music recordings," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2022, pp. 406–410.
- A. J. Eronen and A. P. Klapuri, "Musical instrument recognition using cepstral coefficients and temporal features," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Istanbul, Turkey, 2000, pp. II753–II756.
- M. Grasis, J. Abeßer, C. Dittmar, and H. M. Lukashevich, "A multiple-expert framework for instrument recognition," in *Proceedings of the International Symposium on Sound and Music Motion*, Marseille, France, 2014, pp. 619–634.
- S. K. Tjoa and K. J. R. Liu, "Musical instrument recognition using biologically inspired filtering of temporal dictionary atoms," in *Proceedings of the International Society for Music Information Retrieval Conference*, 2010, pp. 435–440.
- C. Joder, S. Essid, and G. Richard, "Temporal integration for audio classification with application to musical instrument classification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 1, pp. 174–186, Jan. 2009.
- M. Krause, M. Müller, and M. V. W. Gärtner, "Hierarchical classification for instrument activity detection in orchestral music recordings," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2023, pp. 1–5.
- E. O. Brigham, *The Fast Fourier Transform and Its Applications*, Englewood Cliffs, NJ, USA: Prentice-Hall, 1988.
- H. Hassanieh, P. Indyk, D. Katabi, and E. Price, "A Faster Fast Fourier Transform," in *Proceedings of the ACM SIGACT Symposium on Theory of Computing*, 2012, pp. 563–578.