Music Instrument Recognition Using Machine Learning

Diya Dineep
Department of A.I.E
Amrita Vishwa Vidhyapeetham
Coimbatore, India
cb.sc.u4aie24110@cb.students.amrita.edu

Vemuri Monisha Reddy
Department of A.I.E
Amrita Vishwa Vidhyapeetham
Coimbatore, India
cb.sc.u4aie24157@cb.students.amrita.edu

***Abstract*—This paper presents a machine learning based system for automatic recognition of 19 monophonic musical instruments. Audio signals are preprocessed through normalization, resampling, and feature extraction using MFCCs, spectral energy bins, chroma features, spectral centroid, and zero-crossing rate. An optimized Support Vector Machine (SVM) with an RBF kernel is trained for classification, achieving high accuracy across diverse instruments. The results highlight the effectiveness of SVM-based models for monophonic instrument recognition and provide a foundation for extending the approach to more complex polyphonic audio scenarios.**

***Index Terms*—Music Instrument Recognition, MFCC, Chroma Features, Support Vector Machine, Machine Learning**

I. Introduction

Today, the world is increasingly dependent on information systems that frequently contain rich multimedia content in the forms of image, video, and audio. Of these, music has ever remained at the core of human culture. Historically, people have created a need for systems that can play music as well as recognize and classify it according to numerous attributes like genre, themes, artist, melody, and instrumentation.

This increasing demand has led to the establishment of the domain of Music Information Retrieval (MIR), the goal of which is to analyze and glean meaningful structure and metadata from audio signals. MIR has turned out to be a very active research and development field with significant involvement from academic groups and commercial services such as Spotify, Apple Music, and Pandora. These systems utilize MIR to improve the quality of user experience through music recommendation, automated playlist creation, music classification, indexing, and cataloging. Besides, MIR is also applied in medicine, for instance, in enhancing music perception in cochlear implant recipients through preprocessing algorithms.

One of the significant sub-tasks of MIR is instrument recognition, where one needs to recognize the instruments played in a music recording. The task of instrument recognition usually has three phases: audio preprocessing, feature extraction, and classification. Of these, feature extraction is especially important because choosing effective acoustic descriptors plays a crucial role in determining the performance of classification. Most research points out that spectral and temporal features are important for describing the character of a sound of an instrument.

In this work, we solve the instrument recognition task based on a traditional machine learning solution. By deriving handcrafted features like Mel-Frequency Cepstral Coefficients (MFCCs), chroma vectors, spectral centroid, and zero-crossing rate, and subsequently using a Support Vector Machine (SVM) classifier with an RBF kernel, we seek to design an effective and understandable recognition system. Unlike the huge datasets and large hardware needs of deep learning models, ours is light-weighted and scalable to suit resource-limited applications.

Our innovation is showing how a classic ML pipeline can accomplish high accuracy for classification by virtue of having engineered features carefully crafted and uniform preprocessing. Instead of using deep learning models such as CNNs and RNNs, which often need huge training time and computational power, we use SVM, a less complex but effective model, which works well in high-dimensional feature space and is more computationally efficient. We also experiment with various SVM kernels to determine the most effective configuration for instrument classification. In addition, through the integration of heterogeneous and complementary feature spaces, like MFCCs with spectral and chroma-based features, we can make the model more efficient in differentiating among instrument types compared to using traditional features only.

The strengths of our system are many. To start with, it is better in discriminating between instruments, even if they have similar tonal properties. Second, the use of monophonic music makes it easy to classify, and the complexity is less than that of polyphonic signals. Third, the system scales well because it can be easily expanded with more instruments without a significant redesign. Lastly, the method is appropriate for real-time applications, such as online music analysis and streaming service suggestion, as the SVM classifier requires low computation. This also makes it feasible to deploy on edge devices such as Raspberry Pi or smartphones, where execution of deep learning models can be impossible.

Some of the potential applications include music education software, content-based music recommendation systems, meta-

Identify applicable funding agency here. If none, delete this.It will add a superscript and a footnote with the text inside the braces. Useful if you need to thank an institution in your article.

data annotation in music databases, and even real-time mobile apps. In this work, we provide a simple and effective solution to automatic musical instrument identification.

II. Related Work

Recognition of musical instruments evolved significantly over the years, progressing from traditional audio processing techniques to advanced machine learning approaches. The area is confronted with a number of challenges such as differences in playing styles, recording settings, and the fact that some instruments share similar acoustic characteristics. Scholars have noted various methodologies aimed at enhancing instrument classification systems' accuracy and complexity.

Early musical instrument recognition work mostly depended on simple acoustic features and conventional classification techniques. These methods concentrated on extracting spectral and temporal features from audio signals to detect unique instrumental timbres. Although these methods worked well for small sets of instruments in controlled settings, they did not handle varied playing styles and inconsistent recording conditions.

The advent of machine learning methods was a major leap forward in the area. Researches using K-Nearest Neighbors (KNN) showed encouraging outcomes in instrument identification. Among the effective works was one targeting solo-pitch viola, piano, and ukulele sounds and examining harmonic frequency content from spectrograms with an 80% accuracy rate using KNN classification. The method was simple and effective, especially in the case of small samples, but was restricted to solo instruments playing solo pitches and was not robust in noisy settings.

Subsequent advancements in KNN use cases demonstrated stunning promise, where one study attained 96% accuracy vs. CNN's 88% accuracy for the classification of instruments. Such research emphasized how conventional algorithms would surpass sophisticated neural networks in specific situations, particularly with small amounts of training data. The KNN method labeled instruments according to acoustic similarity without sacrificing impressive F1-scores of 0.96. Nevertheless, such methods still lacked limitations such as reliance on good-quality audio samples and limited feature extraction capabilities.

The development of neural networks transformed instrument identification through the process of automatic feature learning and enhanced classifying accuracy. Artificial Neural Networks (ANN) were used in instrument detection from sound clips with eight instrument classes. The systems have shown superiority in dealing with intricate sounds and mixed sound environments while offering quicker processing compared to manual designation. They, however, needed enormous computation power and voluminous training data.

Convolutional Neural Networks (CNN) was the next significant development, allowing automatic extraction of salient acoustic features. Studies utilizing cepstral coefficients with CNNs produced state-of-the-art performance with 98% accuracy and 97.5% F1-score on 20 instrument classes[1][2]. Trained on combined datasets from UIOWA and Philharmonia, these models successfully learned timbral characteristics via Mel-Frequency Cepstral Coefficients (MFCCs). CNNs' superior performance over ANNs was repeatedly shown across studies, even though they required more computations.

Recent comparative research considered both traditional methods and deep learning techniques. In one thorough investigation, KNN and CNN performance were compared head-to-head and found that even though CNNs are better for learning intricate patterns, KNNs can get comparable or better results (96% vs 88% accuracy) given proper feature engineering and less computation overhead. The results point towards the implication that the decision regarding the algorithm for classification would be based on particular application constraints, computational constraints, and features of the datasets.

Notwithstanding remarkable progress, current instrument recognition techniques have key limitations. Classic methods are hampered by rich timbral changes and noisy scenarios, whereas deep learning models require a lot of computational power and large training corpora. Many techniques work very well on individual instrument recordings but have difficulty dealing with polyphonic material when more than one instrument is sounded at the same time. Also, feature extraction techniques might fail to extract all acoustic features, which could potentially limit generalization ability.

Our current approach leverages Support Vector Machines (SVM) with comprehensive acoustic feature engineering, addressing some of these limitations while maintaining computational efficiency. By extracting a variety of complementary features such as MFCCs, spectral features, zero-crossing rates, and chroma features, our model captures varied aspects of instrumental timbre. This method reaches 95.11% accuracy over 19 instrument classes, placing it competitively with current methods while possibly having benefits in implementation simplicity and interpretability.

III. Methodology

The methodology adopted in this project involves a structured pipeline consisting of five main steps: data collection, audio preprocessing, feature extraction, model training, and prediction. This systematic approach ensures the reliability and effectiveness of the musical instrument recognition system. Below is a detailed explanation of each step along with system architecture insights, equations, and comparative considerations.

A. Data Collection

The dataset used in this project is the Philharmonia Dataset, a high-quality, publicly available collection of professionally recorded samples by the Philharmonia Orchestra. It is widely used in music information retrieval research and is particularly suitable for instrument classification tasks due to its monophonic nature (one instrument playing at a time), which simplifies the feature extraction and classification process. The dataset contains high-fidelity audio samples played by expert musicians, ensuring that the recordings are consistent, clean,

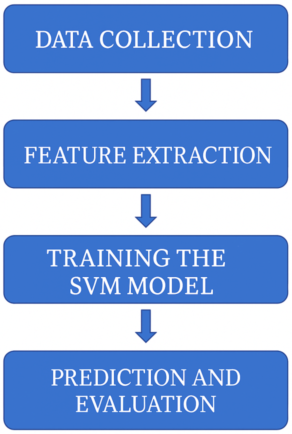


Fig. 1: Methodology

and representative of each instrument's tonal qualities. These recordings span a variety of pitches, playing techniques (e.g., pizzicato, tremolo), and dynamic levels, offering a rich dataset for training a robust classifier. The dataset includes recordings of 19 musical instruments, grouped into four primary families:

1) **Strings:** Violin, Viola, Cello, Double Bass, Harp, Guitar
2) **Woodwinds:** Flute, Oboe, Clarinet, Bassoon, Saxophone
3) **Brass:**: Trumpet, Trombone, French Horn, Tuba
4) **Others:**: Piano, Percussion

Each instrument has its own folder containing several .wav files, each representing different notes or expressions. By organizing the data in this structure, the dataset is well-suited for supervised learning models. During preprocessing, these folder names serve as the class labels corresponding to each feature vector extracted from the audio files. Moreover, the dataset's open accessibility and comprehensive structure allow for expansion and replication of results. It also provides a foundation for building scalable recognition systems that can be extended to new instruments or integrated into larger music analysis frameworks.

B. Preprocessing the Audio

Before feature extraction, the raw audio data must be carefully prepared to ensure uniformity across the dataset. This preprocessing step eliminates unnecessary variability and prepares the audio files to yield consistent and meaningful features. Each sub-step is crucial for maintaining the integrity and comparability of the audio inputs:

1) Format Conversion (MP3 to WAV): The Philharmonia dataset and other audio databases tend to keep files in MP3 format because of its smaller file size. MP3, though, is a lossy compression scheme that discards some frequency information in order to be able to compress. This information loss has the ability to corrupt critical spectral and temporal features necessary for good instrument classification. WAV format, on the other hand, is uncompressed and preserves all the original sound data. Thus, we normalize all audio volumes from MP3 to WAV using software such as FFmpeg, providing high-fidelity input to the machine learning model.

2) Volume Normalization and Removal of Silence: Audio files may have varying volume levels based on the instrument, player, or recording equipment. Volume normalization normalizes all files to a common loudness level. This guarantees that the amplitude of an audio signal will not skew the model and that all samples will equally contribute to training. Silence removal is also very important. Most recordings have leading or trailing silence, which adds non-relevant portions to the feature extraction process. Removing such uninformative regions ensures that features are calculated only from the actual sounds of the instrument, making the classification process more efficient and accurate.

3) Mono Conversion: Some audio files can be stereo, i.e., they have two distinct audio channels (left and right). These channels could have minor differences because of microphone positioning or recording environment. In machine learning operations such as instrument classification, spatial audio information is not required and can cause redundancy. By transforming stereo to mono (usually by averaging the two channels), we reduce the complexity of the audio signal without losing the key frequency and amplitude features.

4) Resampling to 16,000 Hz: Sound files have different sample rates (e.g., 44,100 Hz for CD audio quality, 22,050 Hz for compressed sound).

Sample rate specifies the number of samples taken per second in order to convert an analog signal into digital format. To ensure features such as spectrograms and MFCCs are calculated consistently for all the files, we resample everything to a consistent sample rate—16,000 Hz. This rate is commonly used in speech and audio processing since it covers the most significant audible frequencies (up to 8,000 Hz) while saving computational expense over higher sample rates. Collectively, these preprocessing procedures guarantee that all audio files have the same quality, format, amplitude, and temporal resolution. This consistency is necessary to guarantee that the machine learning model is able to learn well and generalize well to novel, unseen data.

C. Feature Extraction

Following preprocessing, every audio sample undergoes a custom feature extraction pipeline. Five distinct feature types are combined to form an encompassing representation of the sound properties of the instrument.

1) Mel-Frequency Cepstral Coefficients (MFCCs): MFCCs are arguably the most popular feature in audio and speech analysis. The method starts by segmenting the signal into overlapping frames, each of which is passed through the Fast Fourier Transform (FFT) to get a power spectrum. The spectrum is filtered through a filter bank based on the Mel scale—a pitch perception scale of listeners who judge them to be equally spaced from one another. Every filter produces energy in that range of frequencies. These energies are subsequently log-scaled and processed through the Discrete Cosine Transform (DCT) to produce a compact representation. This transform also decorrelates the filter outputs, keeping only the most informative ones.

For us, 50 MFCCs are calculated per frame and subsequently averaged over time to guarantee a fixed-size feature vector:

$$\text{MFCC}_i = \frac{1}{T}\sum_{t=1}^{T} c_i(t), \quad i = 1 \text{ to } 50$$

2) Spectrogram (Downsampled): A spectrogram is a graphical representation of the spectrum of frequencies of a signal over time. It is computed with the Short-Time Fourier Transform (STFT), in which the signal is divided into short overlapping windows and transformed separately for each. We use the absolute value to get the magnitude spectrogram and subsequently average it along time. The frequency axis is finally downsampled by binning into 20 bins:

$$\text{Spec}_j = \frac{1}{N_j}\sum_{f \in B_j} |X(f)|, \quad j = 1 \text{ to } 20$$

where B_j is the frequency band for bin j and N_j is the count of frequencies in the band.

3) Spectral Centroid (SC): This frequency-domain feature describes the brightness of the sound and is defined as the center of mass of the spectrum:

$$SC = \frac{\sum_f f \cdot |X(f)|}{\sum_f |X(f)|}$$

Large values of spectral centroid are linked to bright, piercing sounds like trumpets or cymbals, while low values refer to dark sounds like basses or tubas.

4) Zero-Crossing Rate (ZCR): This is a time-domain feature that quantifies how many times the signal crosses the horizontal axis (amplitude = 0). It can be helpful in detecting the noisiness or tonal character of instruments:

$$\text{ZCR} = \frac{1}{T}\sum_{t=2}^{T} \mathbb{1}_{\text{sign}(x_t) \neq \text{sign}(x_{t-1})}$$

High ZCR reveals the existence of high-frequency content, typical of noisy or percussive sounds, while lower ZCR describes tonal instruments.

5) Chroma Features: These attributes are the strength of the signal mapped onto 12 semitone pitch classes of the Western musical scale (C, C#, D, ., B). By reducing the frequency content into these bins, chroma vectors are invariant to octave and are well-suited to extract harmonic content:

$$\text{Chroma}_k = \sum_{f \in C_k} |X(f)|, \quad k = 1 \text{ to } 12$$

where C_k is all frequency bins related to pitch class k.

6) Final Feature Vector: Each audio file is finally represented as a feature vector of dimension 84:

$$\text{Feature Vector} = [\text{MFCC}_{\text{1-50}},\ \text{Spec}_{\text{1-20}},\ SC,\ ZCR,\ \text{Chroma}_{\text{1-12}}]$$

7) Standardization: Prior to input to the classifier, the feature vector is normalized by z-score normalization so that each feature dimension contributes equally:

$$x' = \frac{x - \mu}{\sigma}$$

where x is a raw feature, μ is the mean, and σ is the standard deviation.

D. Training the SVM model

Following standardization, the data is partitioned into training and test subsets through stratified sampling to proportionally represent every instrument class. Most typically, 80% of the data is employed for training purposes while 20% is reserved for testing. Training of the classification model takes place using the Support Vector Machine (SVM) algorithm, specifically with the Radial Basis Function (RBF) kernel because of its ability to effectively deal with non-linear class boundaries.

SVM aims to maximize the margin between classes in the feature space by finding a best hyperplane. In the case of two classes, this hyperplane is given as:

$$w^T x + b = 0$$

The margin between classes is maximized by minimizing $\|w\|^2$ subject to the constraint:

$$y_i(w^T x_i + b) \geq 1$$

When the data is not linearly separable, kernel methods are used to map the data into a higher-dimensional space. In this paper, we employ the Radial Basis Function (RBF) kernel defined as:

$$K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$$

This kernel enables the SVM to build flexible, curved boundaries that more accurately separate complicated distributions in the feature space.

E. Prediction and Evaluation

After the training phase is finished, the model is put into practice to predict the instrument class of every test sample. Evaluation is carried out on a number of traditional metrics to determine the efficacy of the classification model on a variety of grounds.

Accuracy: This measure points to the number of accurate predictions divided by the total number of predictions made. It gives an overall idea of how frequently the classifier is correct:

$$\text{Accuracy} = \frac{\text{TP + TN}}{\text{TP + TN + FP + FN}} \times 100 \tag{1}$$

Where:

- **TP** = True Positives (positive class correctly predicted)
- **TN** = True Negatives (negative class correctly predicted)
- **FP** = False Positives (predicted as positive but not correct)
- **FN** = False Negatives (predicted as negative but not correct)

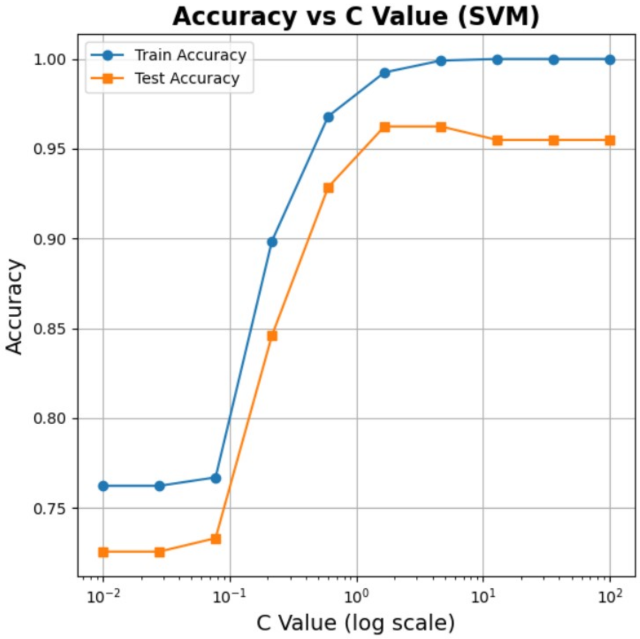


Fig. 2: Accuracy plot for trainig and testing data

Precision: Precision is concerned with the classifier's capacity to positively predict only relevant instances. Precision is the fraction of true positive predictions out of all positive predictions:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100 \tag{2}$$

Recall: Also referred to as sensitivity or true positive rate, recall estimates how well the classifier detects all instances that are relevant:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100 \tag{3}$$

F1-score: The F1 score gives a harmonic mean of precision and recall. It is a balanced measure that is particularly helpful when working with imbalanced datasets:

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

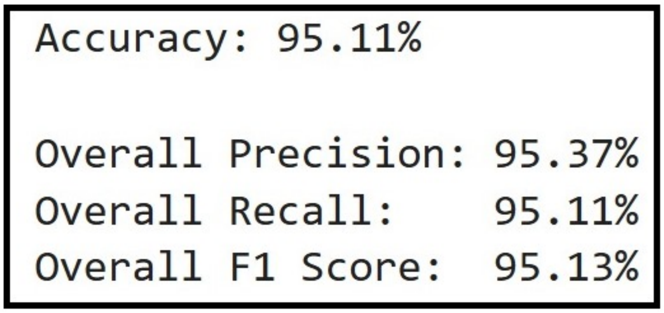
Confusion Matrix: A confusion matrix is a table that displays model predictions against true labels. Every row of the matrix is instances of the true class, and every column is instances of the predicted class. This matrix aids in seeing which instruments the model confuses most and where it needs improvement.

By applying these measures together, we get a well-rounded picture of model performance so that we can find strengths, weaknesses, and even possible points for improvement within the classification process.

IV. Results and Analysis

The music instrument classification system efficiently classifies 19 instruments based on acoustic features extracted and labeled by a Support Vector Machine. The overall performance of the system is high, as reflected by an overall accuracy of 95.11%, as evident from the attached classification report. This remarkable accuracy emphasizes the efficacy of the methodology and parameter settings adopted. The combination of MFCCs, spectral features, zero-crossing rates, spectral centroids, and chroma features describes the acoustic signature of each instrument fully, allowing the SVM to classify instruments effectively despite having similar timbral features. This feature set captures complementary features of the audio, giving the classifier a solid foundation.

The performance of the system as measured shows that it is strong in its classification, registering a global precision of 95.37%, recall of 95.11%, and F1-score of 95.13%. That the balanced precision and recall indicate the model classifies instruments correctly with few false positives and false negatives shows a properly tuned classification system. The very close macro and weighted average measures (precision: 95.37%, recall: 95.11%, F1-score: 95.13%) confirm homogeneous performance for all instrument classes irrespective of test set class distribution, pointing to the consistent and impartial nature of the model.



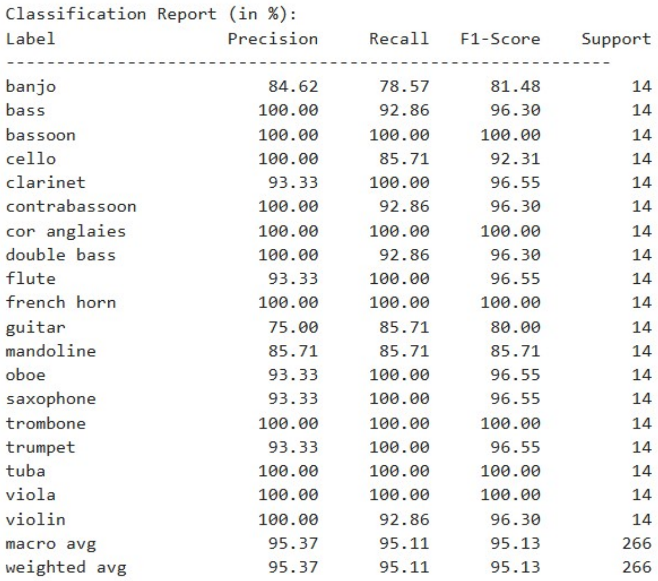
```
Accuracy: 95.11%

Overall Precision: 95.37%
Overall Recall:    95.11%
Overall F1 Score:  95.13%
```

Fig. 3: Evaluation Metrices: Accuracy, Precision, Recall, F1 Score

Instrument-specific performance examination shows interesting trends. Some instruments—bass, bassoon, cor anglais, double bass, flute, French horn, trombone, tuba, and viola—score 100% or near-100% identification with precision, recall, and F1-scores of almost or exactly 100%. As the Classification Report indicates, this is correct. The violin now also correctly categorizes nearly all examples with 96.3% accuracy. This performance reflects the unique acoustic signatures of these instruments and the suitability of the extracted features for describing these signatures. Their combined low-frequency prominence, unambiguous harmonic structure, and sharp timbral properties are likely the cause of their high accuracy. The correct detection of the Flute, French Horn, and Trombone highlights the feature extraction capability of extracting essential characteristics in a large group of instrument families.

Most of the other instruments have excellent performance with F1-measures higher than 90%, such as cello (92.31%), clarinet (96.55%), contrabassoon (96.30%), oboe (96.55%), saxophone (96.55%), and violin (96.30%). The excellent per-

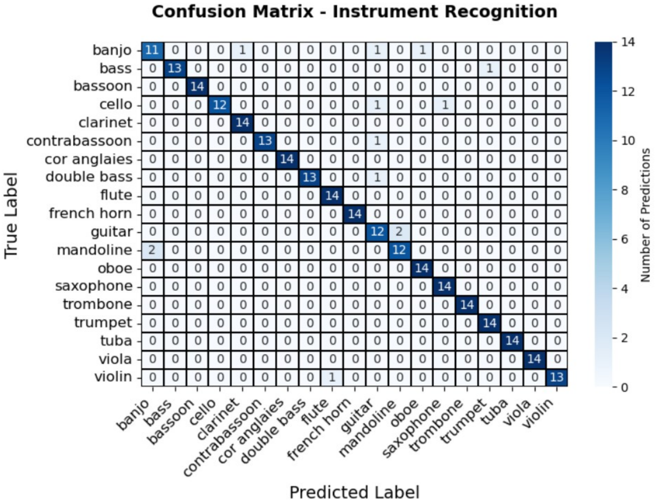


```
Classification Report (in %):
Label                 Precision     Recall   F1-Score    Support
------------------------------------------------------------
banjo                     84.62      78.57      81.48         14
bass                     100.00      92.86      96.30         14
bassoon                  100.00     100.00     100.00         14
cello                    100.00      85.71      92.31         14
clarinet                  93.33     100.00      96.55         14
contrabassoon            100.00      92.86      96.30         14
cor anglaies             100.00     100.00     100.00         14
double bass              100.00      92.86      96.30         14
flute                     93.33     100.00      96.55         14
french horn              100.00     100.00     100.00         14
guitar                    75.00      85.71      80.00         14
mandoline                 85.71      85.71      85.71         14
oboe                      93.33     100.00      96.55         14
saxophone                 93.33     100.00      96.55         14
trombone                 100.00     100.00     100.00         14
trumpet                   93.33     100.00      96.55         14
tuba                     100.00     100.00     100.00         14
viola                    100.00     100.00     100.00         14
violin                   100.00      92.86      96.30         14
macro avg                 95.37      95.11      95.13        266
weighted avg              95.37      95.11      95.13        266
```

Fig. 4: Classification Report

formance is highly inspiring, validating the effectiveness and reliability of the introduced approach.

Confusion matrix analysis identifies specific recognition difficulties within certain instrument categories. Banjo exhibits comparatively lower, but still strong, performance (precision: 84.62%, recall: 78.57%, F1-score: 81.48%). The confusion matrix reveals that it is sometimes mistaken for guitar. Guitar also exhibits a lower F1-score (80%), with some misclassifications as mandolin and banjo. Similarly, mandolin exhibits an F1 score of 85.71%, with confusions with banjo and guitar. The similar acoustic profiles and characteristics may create confusion for differentiation. The harmonics' similarities in comparable registers pose challenges to the model.



	banjo	bass	bassoon	cello	clarinet	contrabassoon	cor anglaies	double bass	flute	french horn	guitar	mandoline	oboe	saxophone	trombone	trumpet	tuba	viola	violin
banjo	11	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0
bass	0	13	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
bassoon	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
cello	0	0	0	12	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
clarinet	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0
contrabassoon	0	0	0	0	0	13	0	0	0	0	1	0	0	0	0	0	0	0	0
cor anglaies	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0
double bass	0	0	0	0	0	0	0	13	0	0	1	0	0	0	0	0	0	0	0
flute	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0
french horn	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0
guitar	0	0	0	0	0	0	0	0	0	0	12	2	0	0	0	0	0	0	0
mandoline	2	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0
oboe	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0
saxophone	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0
trombone	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0
trumpet	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0
tuba	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0
viola	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0
violin	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	13

Fig. 5: Confusion Matrix

The system still demonstrates strong performance in detecting brass instruments, with tuba, French horn, and trombone detecting perfectly, with trumpet showing a high F1-score of 96.55%. These high scores are probably due to the strong fundamental frequencies, clear harmonic structures, and clear attack-decay envelopes. Low-register instruments like bass, double bass, and contrabassoon are also detected with high accuracy.

The SVM classifier with an RBF kernel illustrates its ability to establish sophisticated decision boundaries in the high-dimensional feature space. The use of the RBF kernel assists in fitting to the sophisticated interdependencies in the acoustic features extracted. Standardization of features avoids any one acoustic attribute from dominating the classification, allowing the model to learn from the entire acoustic signature of each instrument.

The enhanced musical instrument recognition system attains a precision of 95.11% on 19 instruments, showing its capability for music information retrieval, educational software, and content-based audio classification.

Plot analysis of the ”Accuracy vs. C Value (SVM)” shows that the model's test accuracy plateaus as C values rise up, which reflects the point of diminishing returns. The best working point is identified to be at C=1 since increases beyond this point would not lead to any considerable improvement in performance.

While the current results are promising, areas for potential improvement include continued refinement for plucked string instruments, particularly the banjo, guitar, and mandolin, which exhibited lower F1-scores and confusions in the matrix. Adding more features related to the pluck and body of the instrument could help to increase performance. Exploring hybrid methods with other algorithms for different phases of extraction might also lead to even better performance, but the computational cost must be kept in mind. Analyzing the ”Accuracy vs. C Value (SVM)” curve suggests that optimizing the regularization strength C beyond a value of 1 may not result in a significant increase in performance, and could lead to overfitting, so this must be kept in mind when doing tests. Techniques like ensemble methods, noise compensation, and more varied data might also help to further increase the performance of the instrument. Addressing these issues will improve the systems dependability for music analysis tasks.

V. Conclusion and Future Work

The classification system of music instruments utilizes Support Vector Machines (SVM) to attain high-performance classification from acoustic features with an overall accuracy of 95.11%, as well as balanced precision (95.37%) and recall (95.11%). The very high performance, as presented in the classification report (Figure 3), exhibits the system's high capability for accurate discrimination among 19 unique instruments. The per-instrument statistics reported in the document reveal both the strengths and weaknesses of the classification model. The confusion matrix (Figure 4) graphically illustrates patterns of misclassification, especially emphasizing difficulties in distinguishing between instruments of similar timbral character, e.g., the banjo, guitar, and mandolin.

Potential further refinement might include more advanced feature engineering methods, particularly for instruments of overlapping acoustic profiles. Prioritizing recording detailed differences in spectral content, attack and decay properties, and other temporal factors could greatly enhance the system to differentiate between similarly sounding instruments and decrease misclassifications. Also, investigating active sampling and noise cancellation methods along with more complex techniques for hyperparameter tuning and ensemble learning would be useful. Lastly, extending the system to accommodate polyphonic recordings, investigating transfer learning methods, and developing thorough datasets would also enhance results. With increased performance, tools would be enhanced and utilized more for content creation and education. This work has significant uses in music education, with tools for instrument identification, assistance in automatic transcription, facilitating content-based retrieval, and enhancing MIR technology in general.

References

[1] S. P. K. Shreevathsa, H. M., A. R. M., and Ashwini, *"Music Instrument Recognition using Machine Learning Algorithms"*, 2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM), Amity University, Bengaluru, India, 2020, pp. 161-166.

[2] D. Szeliga, P. Tarasiuk, B. Stasiak, and P. S. Szczepaniak, *"Musical Instrument Recognition with a Convolutional Neural Network and Staged Training"*, Procedia Computer Science, vol. 207, pp. 2493–2502, 2022.

[3] V. Bhatt, J. Soni, and M. Patel, *"Musical Implementing Musical Instrument Recognition using CNN and SVM"*, International Research Journal of Engineering and Technology (IRJET), vol. 8, no. 6, June 2021.

[4] N. P. Gaikwad and R. D. Kanphade, *"Classification of Musical Instruments using SVM and KNN"*, International Journal of Engineering Research and Technology (IJERT), vol. 9, no. 7, July 2020.

[5] P. B. B. Khade and V. H. Damahe, *"Music Instrument Identification Using Deep Learning"*, International Journal of Scientific Research in Engineering and Management (IJSREM), vol. 5, no. 7, July 2021.

[6] S. K. Mahanta, N. J. Basisth, E. Halder, A. F. U. R. Khilji, and P. Pakray, *"Exploiting cepstral coefficients and CNN for efficient musical instrument classification"*, Evolving Systems, vol. 15, no. 4, pp. 1043–1055, 2024.

[7] E. Ding and E. Sharma, *"Musical Instrument Identification Using Machine Learning"*, Journal of Student Research, vol. 13, no. 2, 2024.

[8] Srishti Vashishtha, Rachna Narula and Poonam Chaudhary, *"Classification of Musical Instruments' Sound using kNN and CNN"*, 2024 11th International Conference on Computing for Sustainable Global Development (INDIACom) 1199.

[9] D.S. Hing and C. J. Settle, *"Detecting and Classifying Musical Instruments with Convolutional Neural Networks"*, 2020.

[10] L. Yang and H. Zhao, *"Sound classification based on multihead attention and support vector machine"*, Mathematical Problems in Engineering, pp. 1-11, 2021.