DOI:10.25092/baunfbed. 1706872 *J. BAUN Inst. Sci. Technol., 28(1), 299-312, (2026)*

A multi-feature approach for musical instrument classification using machine learning

Abdurrahim Hüseyin EZİRMİK[1,*], Birol ÇİLOĞLUGİL[2]

[1] Balıkesir University Faculty of Engineering, Department of Computer Engineering, Cagis Campus, Balikesir.
[1]Ege University Institute of Natural and Applied Sciences, Department of Computer Engineering, Izmir
[2] Ege University Computer and Informatics Sciences Faculty, Department of Computer Engineering, Izmir.

Geliş Tarihi (Received Date): 26.05.2025
Kabul Tarihi (Accepted Date):17.11.2025

Abstract

This study examines the performance of a collection of spectral audio features, including RMS Energy, Zero Crossing Rate (ZCR), and Spectral Centroid, for musical instrument classification by using the Random Forest and XGBoost classifiers. These machine learning algorithms demonstrate enhanced precision in complex classification scenarios and improve the ability to discriminate among highly correlated instrument classes. Machine learning approaches were employed in this study due to being explainable, computationally efficient, and suitable when deep learning is not feasible under the constraints of hardware or data. As part of the experimental setup, the audio features were obtained from the Philharmonia dataset, which includes 20 instrument classes. Seven different configurations were evaluated, including each feature set individually, as well as their pairwise and triplet combinations. The highest performance in terms of accuracy was obtained when all attributes were utilized: 0.91 with Random Forest and 0.93 with XGBoost. These machine learning algorithms were particularly well adapted to distinguish acoustic differences in music. Confusion matrix analysis indicated that both models worked best for instruments with clear acoustic characteristics, such as guitar and banjo. The findings suggested that the combination of multiple complementary features improves the classification performance of musical instruments.

Keywords: *MIR, instrument classification, machine learning, feature extraction.*

[*]Abdurrahim Hüseyin EZİRMİK, huseyin.ezirmik@balikesir.edu.tr, https://orcid.org/0000-0002-1154-1537
Birol ÇİLOĞLUGİL, birol.ciloglugil@ege.edu.tr, https://orcid.org/0000-0003-3589-9135
[**]*Bu çalışma Abdurrahim Hüseyin EZİRMİK'in Birol ÇİLOĞLUGİL danışmanlığında yürütülen "Derin Öğrenme Tabanlı Enstrüman Tespit ve Müzik Öneri Sistemi" başlıklı doktora tezi kapsamında üretilmiştir.*

Makine öğrenmesi ile müzik enstrüman sınıflandırması için bir çoklu-öznitelik yaklaşımı

Öz

Bu çalışma, RMS Energy, Zero Crossing Rate (ZCR) ve Spectral Centroid gibi spektral ses özniteliklerinin, Random Forest ve XGBoost algoritmaları kullanılarak müzik aleti sınıflandırmasındaki performansını incelemektedir. Bu makine öğrenmesi algoritmaları, karmaşık sınıflandırma senaryolarında artırılmış doğruluk sağlama ve yüksek korelasyonlu enstrüman sınıfları arasında ayrım yapabilme yeteneğine sahiptir. Bu çalışmada makine öğrenmesi yaklaşımları açıklanabilir ve hesaplama açısından verimli olmaları ve derin öğrenmenin donanım veya veri kısıtlamaları nedeniyle uygulanamadığı durumlarda kullanışlı olmaları sebebiyle tercih edilmiştir. Deneysel çalışmada, ses öznitelikleri 20 enstrüman sınıfını içeren Philharmonia veri setinden elde edilmiştir. Her öznitelik seti tek başına ve ikili ile üçlü kombinasyonları da dahil olmak üzere toplam yedi farklı konfigürasyon değerlendirilmiştir. Doğruluk açısından en yüksek performans, tüm öznitelikler bir arada kullanıldığında elde edilmiş olup Random Forest için 0,91 ve XGBoost için 0,93'tür. Bu makine öğrenmesi algoritmaları, müzikteki akustik farklılıkları ayırt etmede özellikle başarılı olmuştur. Karışıklık matrisi analizi, her iki modelin gitar ve banjo gibi belirgin akustik özelliklere sahip enstrümanlarda en iyi performansı gösterdiğini ortaya koymuştur. Bulgular, birden fazla tamamlayıcı özelliğin kombinasyonunun müzik aleti sınıflandırma performansını artırdığını göstermiştir.

Anahtar kelimeler: *MIR, enstrüman sınıflandırma, makine öğrenmesi, öznitelik çıkarımı.*

1. Introduction

Automatic musical instrument identification has become an increasingly key area of interest in music information retrieval (MIR) and machine learning as well [1]. With the amount of digital audio content growing every day, automatic musical instrument identification from audio recordings becomes important for music recommendation, audio indexing, and content-based retrieval [2]. Traditional approaches have been largely driven by signal processing techniques that are domain specific, but recent development in machine learning has enabled better classification with very large and diverse sets of features.

Spectral audio features such as Root Mean Square (RMS) Energy, Zero Crossing Rate (ZCR), and Spectral Centroid are commonly used in audio signal processing due to their capabilities to detect dominant features of timbre and dynamics [3]. These features, while individually informative, provide improved performance when combined into multi-dimensional representations that reflect complementary aspects of the audio signals. However, the variability in audio sample lengths poses a significant challenge for traditional machine learning models that require inputs of fixed size [4]. This issue is commonly addressed using preprocessing techniques such as zero padding, where shorter samples are extended with zeros, or truncation, where longer samples are cut to a predefined length, in order to ensure uniform feature dimensionality across all inputs [5].

We use the Philharmonia dataset, an ensemble of high-quality, isolated recordings, and manually extract RMS Energy, ZCR, and Spectral Centroid features from each sample.

We explore in this paper the performance of Random Forest and XGBoost machine learning models on instrument classification from spectral audio features. The classification performance of each model is evaluated on separate sets of features as well as on combined sets in order to measure the contribution of each feature to model accuracy. Through the union of lightweight audio features with traditional machine learning classifiers, this study aims to demonstrate the promise of explainable low-complexity models for musical instrument recognition. The results provide the foundation for further research in cost-effective MIR systems capable of performing well even when computational power is limited.

The remaining part of this article is structured as follows. The Related Works section provides an overview of the current literature in the research field. Then, the Material and Methods section provides the dataset, feature extraction techniques, and machine learning models utilized in this study. The results are presented and evaluated in the Results and Discussion section. The Conclusion section finally provides an overview of the study with recommendations for further research.

2. Related works

Machine learning models have been successfully used in instrument identification tasks, as demonstrated by studies presented in this section. It is important to note that these studies focus on the same dataset as the one used in this study, namely the Philharmonia.

Toghiani-Rizi and Windmark (2017) used an Artificial Neural Network (ANN) trained with audio samples transformed to the frequency domain [6]. To verify the informative potential of the provided data, the authors experimented with a set of features from both the temporal and frequency domains. Their experimental results registered an accuracy of 93.5%, which verifies that ANN can recognize complex audio patterns when different features are used.

Uruthiran and Ranathunga (2019) derived many features from the time and frequency domains (Spectral Centroid, Root Mean Square, Zero Crossing Rate, MFCC, etc.) to train the Decision Trees, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) classifiers [7]. Of these classifiers, the SVM had the maximum mean accuracy of 93.37%, which indicates its competence to handle mixed feature sets in instrument classification tasks.

In the study by Tu and Li (2023), Multi-Layer Perceptron (MLP) and Support Vector Machine (SVM) models were used for the classification of isolated orchestral notes [8]. The findings indicated that MLP outperformed SVM, with 87% accuracy rate. Based on the observation, it can be said that MLP performs optimally when dealing with isolated audio samples.

Also, Su (2023) conducted a comparative analysis of six models for instrument classification including K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Gaussian Mixture Modeling (GMM), Artificial Neural Networks (ANN), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN) [9]. CNN deep learning model achieved the maximum accuracy of 96.82%. Among the machine learning models,

KNN, SVM, and GMM achieved accuracies of 93.34%, 73.35%, and 64.85%, respectively. In addition, SVM achieved the best computational time of 0.53 seconds. These results indicate an imbalance between classification accuracy and processing time across model types.

In addition, Bhagyalakshmi and Anandaraju (2025) proposed a KNN model utilizing optimized and tuned MFCC features and compared it with the SVM [10]. It was identified that, the SVM achieved 90% instrument classification accuracy, whereas the proposed method utilizing KNN had better accuracy at 94%. The findings demonstrate the effectiveness of using MFCC features coupled with KNN for high classification performance.

3. Material and method

3.1.Dataset

Several well-known datasets are used in musical instrument recognition, including NSynth, IRMAS, MusicNet, RWC, Good-Sounds, MedleyDB, and Philharmonia. NSynth is synthetic-audio centered and thus particularly appropriate for research where generated or test sounds are of interest. IRMAS (Instrument Recognition in Musical Audio Signals) is a widely used dataset which has 11 classes of instruments [11]. MusicNet provides a diverse collection of classical music excerpts with rich contextual information for analysis. The RWC database contributes recordings of many musical genres of high audio fidelity, while Good-sounds excels at studio-quality recordings of solo instruments. MedleyDB consists of royalty-free, annotated multitrack recordings [12]. The Philharmonia dataset, developed by the London Philharmonic Orchestra, contains high-quality orchestral instrument recordings [13]. These were recorded in collaboration with professional sound engineers and musicians to ensure fidelity and consistency.

The Philharmonia dataset is employed by this study for instrument identification because of several advantages. It includes a wide variety of orchestral instruments, allowing identification of 20 instrument classes with their sample counts listed in Table 1. The dataset has 13,681 samples, making it feasible to train effectively, resist overfitting, and support generalization. There is an imbalance in sample counts across instruments, ranging from 74 (banjo) to 1502 (violin). However, its good sound quality facilitates the extraction of precise acoustic features, thus providing better classification accuracy. In addition, the Philharmonia project is enhanced by continued community interaction, which serves to keep the dataset in high quality. It also promotes reproducibility and stimulates ongoing research by being open-access.

Table 1. Number of samples per instrument class in the Philharmonia dataset.

Instrument class	**Sample count**
banjo	74
bass clarinet	944
bassoon	720
cello	889
clarinet	846
contrabassoon	710
cor anglais	691
double bass	852
flute	878
french horn	652
guitar	106
mandolin	80
oboe	596
percussion	148
saxophone	732
trombone	831
trumpet	485
tuba	972
viola	973
violin	1502
Total:	13.681

3.2. Spectral feature extraction

To facilitate the classification of musical instruments, this study employs spectral features known for their effectiveness in representing timbral and dynamic properties of audio signals. These features represent the frequency content of audio signals. One-dimensional (1D) features typically consist of feature vectors extracted from the audio, while two-dimensional (2D) features are time-frequency representations such as Mel spectrograms or Constant-Q transforms (CQT). Three types of 1D features are extracted in this study: Root Mean Square energy, Zero Crossing Rate, and Spectral Centroid. RMS Energy is used to describe the amplitude envelope and subjective loudness of the signal, ZCR is a measurement of the rate of sign change of the signal, i.e., how noisy or edgy the signal is, and Spectral Centroid is the "center of mass" of the spectrum, typically associated with the high-frequency content of the sound [14].

Feature extraction is performed using the LibROSA library, a widely used Python package for audio analysis [15]. The features are extracted from the Philharmonia dataset. Audio files in the dataset are first loaded at their natural sampling rate [16]. Every recording is then processed further to get frame-wise RMS Energy, ZCR, and Spectral Centroid, which generates time-dependent feature matrices.

The RMS Energy for a given frame is defined as:

$$\mathrm{RMS}(n) = \sqrt{\frac{1}{N}\sum_{i=0}^{N-1} x^2(n+i)} \tag{1}$$

where *x(n)* is the audio signal and *N* is the frame length.

The Zero Crossing Rate measures the rate at which the signal changes sign:

$$\mathrm{ZCR}(n) = \frac{1}{2N} \sum_{i=1}^{N} |\mathrm{sgn}(x(n+i)) - \mathrm{sgn}(x(n+i-1))| \tag{2}$$

The sign function *sgn(x)* is defined as:

$$\mathrm{sgn}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases} \tag{3}$$

The Spectral Centroid represents the center of mass of the spectrum:

$$\mathrm{Centroid}(n) = \frac{\sum_{k=0}^{K-1} f_k \cdot |X(n,k)|}{\sum_{k=0}^{K-1} |X(n,k)|} \tag{4}$$

where f_k is the frequency of the *k*-th bin, $|X(n, k)|$ is the magnitude spectrum at frame *n* and *K* is the total number of frequency bins.

In order to provide a uniform representation for all features, only the common part of the feature sequence is retained by trimming them down to the shortest number of frames of the utilized features. Each feature vector was resized to a fixed length of 100 to ensure uniformity across all samples. It makes temporal matching and subsequent fusions possible. The extracted features are saved in compressed .npz format to preserve their variable-length sequences. Three feature types (RMS Energy, ZCR, and Spectral Centroid) are saved separately as individual arrays, and the related instrument labels are saved in a .npy file [17]. All files in the dataset are imported in this pre-processing stage, where one sample from every audio recording is saved along with its related label. This feature extraction technique promises to effectively capture both energy-based and spectral characteristics of instrument sounds, resulting in a concise yet informative representation for subsequent classification.

3.3. Machine learning algorithms

Machine learning is a branch of artificial intelligence that enables computers to learn from data patterns and predict or decide without being explicitly programmed [18]. Many types of machine learning algorithms are used in different tasks, and successful results are obtained. In this study, Random Forest and XGBoost classifiers were preferred due to their ability to effectively handle high-dimensional feature sets and their widespread use in various machine learning tasks [19]. In initial experiments, other machine learning models such as Support Vector Machine (SVM), Decision Trees, and K-Nearest Neighbors (KNN) have also been tested. However, we decided to continue the experiments with the most successful machine learning models and reported their findings.

3.3.1. Random forest

Random Forest is an ensemble learning method that builds multiple decision trees during training and produces the final output by taking the majority vote of the classes in classification tasks or the average of the predictions in regression tasks [20]. In training each tree, a bootstrap sample (i.e., randomly selected sample with replacement) of the training dataset is used, and during tree building, only a random subset of features is employed to consider for splitting at each node [21]. This randomness increases model diversity and reduces variance, resulting in improved generalization.

Let $h_t(x)$ denote the prediction of the t-th decision tree in the ensemble for a given input x, where $t = 1, 2, \ldots, T$, and T is the total number of trees. In classification tasks, the final output $\hat{y}$ is determined by majority voting among the individual tree predictions:

$$\hat{y} = \text{mode}\left(\{h_t(x)\}_{t=1}^{T}\right) \tag{5}$$

Here, *mode()* denotes the statistical mode, which refers to the class label that appears most frequently among the predictions of all decision trees in the ensemble.

Random Forest is a noise-resistant, non-parametric algorithm that can handle missing values and is less prone to overfitting [22]. It is particularly suitable for cases where model interpretability is less of a concern compared to predictive performance.

3.3.2. XGBoost

XGBoost is an efficiency-oriented version of Gradient Boosting algorithms [23]. Unlike Random Forest, where the trees are built independently, XGBoost builds trees sequentially so that each tree attempts to minimize the residual errors of the cumulative ensemble. The method optimizes a regularized objective function that balances model complexity and prediction error.

Let the model prediction at iteration t be denoted by:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i), \quad f_t \in \mathcal{F} \tag{6}$$

where f_t is the new regression tree added at iteration t, and F is the space of all possible trees [24].

The objective function at iteration t is defined as:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t)}) + \sum_{k=1}^{t} \Omega(f_k) \tag{7}$$

where l is a loss function that measures the difference between the prediction $\hat{y}_i^{(t)}$ and the true label y_i [25].

$\Omega(f)$ is a regularization term that penalizes model complexity, which can be formulated as follows:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{8}$$

Here, T is the number of leaves in the tree, w_j is the weight of the j-th leaf, and γ and λ are regularization parameters controlling the model complexity.

XGBoost uses second-order derivatives of the loss function during optimization and supports parallel computation, which makes it both accurate and efficient.

4. Experimental procedure

The experimental process was designed to ensure a systematic and reproducible workflow. Figure 1 illustrates the instrument classification process based on machine learning. Initially, audio files are collected as input data and processed through a feature extraction stage, where key spectral features are generated. To standardize the input, each audio sample is trimmed to ensure a fixed data size and labelled according to its corresponding instrument class. The extracted and labelled features are compiled into a dataset, which is subsequently used to train two ensemble-based machine learning algorithms. Finally, the trained models are evaluated to assess their classification performance and identify the most effective approach for instrument recognition.

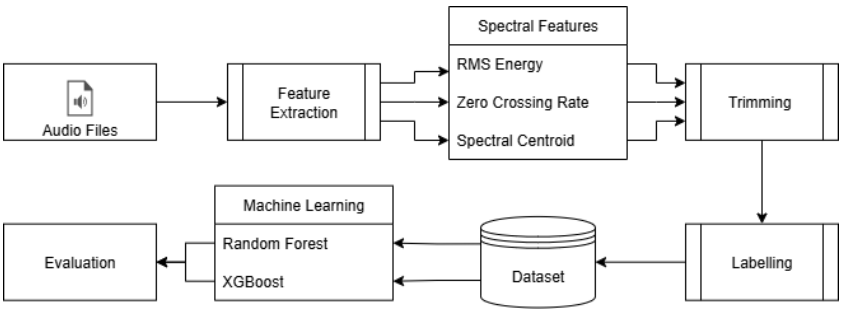

Figure 1. Overview of the experimental workflow for instrument classification.

The experiments were conducted on a personal computer equipped with an Intel Core i7-11700K processor, 32 GB of RAM, and an NVIDIA GeForce GTX 1660 Super GPU. Data processing and model implementation were carried out in the Anaconda Python environment using the Spyder IDE. The NumPy and Pandas libraries were employed for numerical computations and data handling, while the LibROSA library was used for spectral feature extraction. Scikit-learn and XGBoost libraries were utilized for model training, evaluation, and feature-based classification. Additional utilities, such as itertools and scikit-learn's metrics module, facilitated feature combination generation and performance assessment.

The Philharmonia dataset, consisting of 20 instrument classes, was collected, organized, and prepared for analysis. Spectral features, including Root Mean Square (RMS) Energy, Zero Crossing Rate (ZCR), and Spectral Centroid, were extracted from each audio sample to capture amplitude, frequency fluctuation, and spectral shape characteristics. Since the audio samples varied in length, feature arrays were standardized through truncation or zero-padding to achieve a uniform representation. All arrays were then flattened into one-dimensional vectors. The labels were numerically encoded. The dataset was randomly

divided into training and testing subsets with an 80/20 ratio to evaluate model generalization.

Two ensemble-based classifiers, Random Forest and XGBoost, were trained using the default settings for their hyperparameters. The Random Forest model was initialized with 100 decision trees (n_estimators=100) and a fixed random seed (random_state=42). The XGBoost model was trained using default hyperparameters, with the legacy label encoder disabled (use_label_encoder=False) and the multi-class evaluation metric set to log loss (eval_metric="mlogloss").

Seven configurations were tested to investigate the effect of different feature combinations, including single, pairwise, and triple feature sets. Classification accuracy was used as the primary evaluation metric due to its interpretability, broad acceptance, and suitability for comparison with previous studies. Additionally, confusion matrices were generated to analyze misclassifications among instrument classes.

5. Results and discussion

The performance of the musical instrument classification is presented in Table 2, using different feature sets for the Random Forest and XGBoost classifiers. The ZCR feature set yielded accuracies of 0.68 using Random Forest and 0.71 using XGBoost. For the Spectral Centroid feature set, accuracy was 0.74 using Random Forest and 0.76 using XGBoost. With the RMS Energy feature set, accuracy was 0.80 using Random Forest, whereas for XGBoost, the accuracy was slightly higher with 0.82.

Table 2. Classification accuracies of Random Forest and XGBoost for different feature sets.

Feature Set	Random Forest	XGBoost
ZCR	0.68	0.71
Spectral Centroid	0.74	0.76
RMS Energy	0.80	0.82
ZCR & Spectral Centroid	0.80	0.84
RMS Energy & ZCR	0.88	0.90
Spectral Centroid & RMS Energy	0.88	0.91
All Features Combined	**0.91**	**0.93**

When the pairwise combinations of the spectral features were analyzed, for the combination of ZCR and Spectral Centroid, the instrument classification accuracies were identified as 0.80 and 0.84 for Random Forest and XGBoost algorithms, respectively. The combination of RMS Energy and ZCR achieved a significant accuracy boost, reaching 0.88 with Random Forest and up to 0.90 with XGBoost. Pairing the Spectral Centroid with RMS Energy resulted in comparable improvements, achieving 0.88 and 0.91 accuracy scores. These findings point out that, all pairwise combinations outperformed the single feature baselines.

Finally, the highest performances for each machine learning algorithm were obtained when all of the features were combined. With the concatenated feature vector of RMS Energy, ZCR, and Spectral Centroid, Random Forest performed with an accuracy of 0.91, while XGBoost achieved the highest overall accuracy of 0.93. These results indicate that

combining triple acoustic features leads to better classification performance compared to utilizing individual features alone or pairwise combinations.

It is also noteworthy that XGBoost performed better then Random Forest across all feature settings as stated in Table 2, although by a small margin. This suggests that the gradient boosting nature of XGBoost may provide a more robust modelling of the decision boundaries, especially when dealing with heterogeneous features. To sum up, the experimental outcomes demonstrate the advantage of using a multi-feature approach in music instrument classification tasks. Moreover, the consistent improvement with XGBoost indicates its effectiveness as a classifier for machine learning-based audio applications.

On the other hand, confusion matrices were examined to provide an additional analysis of the classification results. The confusion matrices for Random Forest and XGBoost in Table 3 and Table 4 indicate that both models had performed quite well on most of the instruments. For example, banjo and guitar were well classified with little or no misclassifications in both models, indicating that these instruments do have definitive timbral features that are easily captured by the features selected.

Table 3. Confusion matrix for the Random Forest classifier trained with combined features. The rows represent the actual values, while the columns represent the predicted values.

	banjo	bass clarinet	bassoon	cello	clarinet	contrabassoon	cor anglais	double bass	flute	french horn	guitar	mandolin	oboe	percussion	saxophone	trombone	trumpet	tuba	viola	violin
banjo	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
bass clarinet	0	189	0	0	1	3	0	0	0	0	0	0	0	0	2	0	0	0	0	0
bassoon	0	0	132	1	1	2	5	0	0	5	0	0	0	0	0	3	0	0	1	0
cello	0	0	0	118	0	1	2	11	8	0	0	0	0	0	2	0	1	0	9	20
clarinet	0	0	0	1	166	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0
contrabassoon	0	0	0	2	4	140	0	0	1	1	0	0	0	0	1	2	0	3	0	2
cor anglais	0	0	0	1	0	0	142	0	0	0	0	0	0	0	2	1	1	0	1	0
double bass	0	0	1	4	0	0	1	149	0	0	0	0	0	0	0	0	0	1	10	1
flute	0	0	0	0	1	0	0	0	150	0	0	0	2	0	0	0	0	0	0	12
french horn	0	0	4	0	0	1	0	0	2	115	0	0	0	0	0	2	0	0	1	0
guitar	0	0	0	0	0	0	0	0	0	0	24	0	0	0	0	0	0	0	0	0
mandolin	3	0	0	0	0	0	0	0	2	0	0	10	0	0	0	0	0	0	2	0
oboe	0	0	0	0	3	0	1	0	5	1	0	0	87	0	0	1	0	0	2	1
percussion	0	1	0	2	0	0	0	2	2	1	0	0	0	11	1	0	0	1	0	6
saxophone	0	1	0	2	6	0	0	0	1	0	0	0	0	0	132	1	1	0	0	2
trombone	0	0	2	1	1	0	2	0	1	1	0	0	0	0	0	174	1	0	0	1
trumpet	0	0	1	0	1	0	5	0	1	0	0	0	0	0	1	0	87	0	0	0
tuba	0	0	0	0	0	3	0	1	0	1	0	0	0	0	0	0	0	185	0	0
viola	0	1	0	5	1	0	1	5	2	0	0	0	0	0	0	0	0	0	162	3
violin	0	0	0	4	1	0	2	2	5	0	0	0	0	0	0	1	0	0	5	284

Table 4. Confusion matrix for the XGBoost classifier trained with combined features. The rows correspond to the actual classes, and the columns correspond to the predicted classes.

	banjo	bass clarinet	bassoon	cello	clarinet	contrabassoon	cor anglais	double bass	flute	french horn	guitar	mandolin	oboe	percussion	saxophone	trombone	trumpet	tuba	viola	violin
banjo	20	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
bass clarinet	0	186	0	1	1	3	0	3	0	0	0	0	0	0	1	0	0	0	0	0
bassoon	0	0	137	3	1	1	2	1	0	4	0	0	0	0	0	0	0	0	1	0
cello	0	0	0	147	1	0	1	7	6	1	0	0	0	0	0	0	1	0	2	6
clarinet	0	0	0	1	165	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0
contrabassoon	0	1	0	1	2	148	0	0	1	0	0	0	0	0	0	0	1	0	0	2
cor anglais	0	0	1	2	1	0	140	0	2	0	0	0	0	0	1	0	0	0	1	0
double bass	0	1	0	3	1	2	0	148	0	0	0	0	0	0	0	0	1	0	11	0
flute	0	0	0	0	0	0	0	0	157	1	1	0	2	0	1	0	0	0	0	3
french horn	0	0	3	0	1	1	0	0	0	118	0	0	0	0	0	1	0	0	1	0
guitar	0	0	0	0	0	0	0	0	0	0	24	0	0	0	0	0	0	0	0	0
mandolin	0	0	0	0	0	0	0	0	0	0	0	15	0	0	1	0	0	0	1	0
oboe	0	0	0	1	0	0	1	0	4	1	0	0	92	0	0	1	0	0	1	0
percussion	0	0	0	0	0	0	0	0	1	0	0	0	0	22	1	1	0	1	0	1
saxophone	0	0	1	1	5	0	2	2	0	0	0	0	0	0	133	0	0	0	0	2
trombone	0	0	0	1	0	0	3	0	0	1	0	0	0	0	0	175	1	0	1	2
trumpet	0	0	3	0	0	0	2	0	0	0	0	0	1	0	1	0	88	0	0	1
tuba	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	184	1	0
viola	0	0	0	7	0	1	0	5	1	0	0	0	1	0	0	0	0	0	163	2
violin	0	0	0	4	0	0	0	3	1	0	0	0	0	0	0	0	0	0	3	293

Some instruments with similar spectral and temporal characteristics belong to the same instrument family, such as string, wind, or percussion instruments [26]. Both models showed difficulty in distinguishing between instruments that belong to same instrument family. For instance, wind instruments such as the bassoon and French horn were occasionally mixed up with each other, presumably due to the share of frequencies at low registers. Similarly, clarinet and saxophone, which are also wind instruments, were misclassified at times, particularly by Random Forest. Among string instruments, cello, double bass, viola, and violin were occasionally confused with each other by both algorithms. The highest confusion occurred when Random Forest misclassified 20 cello samples as violin. These findings suggest that although the models effectively generalize broad spectral patterns, distinguishing between instruments within the same family remains a challenge.

When performances of machine learning algorithms for each confusion matrices in Table 3 and Table 4 were compared, XGBoost demonstrated classification improvement in 13 out of 20 instruments, significantly reducing misclassifications compared to Random Forest. For example, the number of correctly classified cello samples increased from 118 to 147 with the XGBoost model, indicating a boost in accuracy. Both models provided precise classifications in their corresponding confusion matrices; however, the overall performance of XGBoost is higher by reducing the confusion between instruments with similar timbral characteristics, such as viola and violin.

When comparing the classification accuracy of combined feature approaches across various studies summarized in Table 5, it is evident that the findings obtained in this study align closely with the leading performances reported in the literature. The accuracy of 93% achieved in this study is very similar to the 93.5% accuracy achieved by Toghiani-Rizi and Windmark (2017) by utilizing the Artificial Neural Network (ANN) [6]. Uruthiran and Ranathunga (2019) also reported a very similar accuracy of 93.37% using the Support Vector Machine (SVM) [7]. Tu and Li (2023) achieved 87% accuracy with MLP and SVM models for isolated orchestral notes [8]. Su (2023) reported that the KNN model achieved an accuracy of 93.34%, which was the highest among the machine

learning models evaluated in the study [9]. Bhagyalakshmi and Anandaraju (2025) reported that their KNN model with optimized MFCC features achieved an accuracy of 94%, outperforming the SVM model, which achieved 90% [10]. Overall, the combination of different acoustic features and the effectiveness of the Random Forest and XGBoost algorithms in this study demonstrate a high and competitive classification performance compared to the state-of-the-art approaches in the literature. Our findings are in line with previous research, highlighting the benefits of multi-feature representations and machine learning methods.

Table 5. Comparison of classification accuracy results across different studies.

Study	Method(s)	Features	Accuracy
Toghiani-Rizi & Windmark (2017)	ANN	Frequency-domain features	93.5%
Uruthiran & Ranathunga (2019)	SVM	Time & frequency-domain features	93.37%
Su (2023)	KNN	Various features	93.34%
Tu & Li (2023)	MLP, SVM	Isolated orchestral notes	87%
Bhagyalakshmi & Anandaraju (2025)	KNN, SVM	MFCC features	94%, 90%
This study	Random Forest, XGBoost	RMS, ZCR, Spectral Centroid	**93%**

6. Conclusion

This paper demonstrates the power of combining spectral, lightweight audio features with machine learning models for automatic musical instrument classification. By employing features such as RMS Energy, Zero Crossing Rate, and Spectral Centroid, common descriptors of timbre and dynamic material, this paper demonstrates that simple audio descriptors can achieve competitive performance when combined with effective preprocessing and applied through suitable machine learning models. The Philharmonia dataset provided a great and varied source of single instrument recordings, and feature representation uniformity was ensured by using zero-padding and truncation.

The experimental results showed that ensemble-based classifiers like Random Forest and XGBoost achieved promising accuracy, particularly when using combined feature sets. These findings suggest that machine learning approaches, which are explainable and computationally efficient, have a significant contribution to offer in the broader context of music information retrieval, especially in situations where deep learning is not feasible under the constraints of hardware or data.

On the other hand, this study has some limitations such as the fixed-length pre-processing which can impact temporal integrity. The manually balanced and high-fidelity characteristic of the dataset that contains studio recordings might not adequately reflect variability of music in real life which includes environmental factors such as noise, reverberation, and room acoustics. Furthermore, the study is limited by the use of a restricted set of audio features, which lack other potentially informative acoustic descriptors. Additionally, models based on artificial neural networks were not utilized, as the study focused solely on traditional machine learning methods.

Future work may include utilizing temporal features to encode the time-varying dynamics of instruments and employing sequential modeling techniques, such as recurrent neural

networks. Further evaluation with more varied and noisy datasets can make the utilized machine learning models more robust and generalizable in practice. Hybrid methods which combine different feature extraction techniques and deep learning models are an attractive direction that balances the trade-off between accuracy and interpretability. Additional work can also consider feature selection and dimension reduction techniques for optimizing performance with a minimal increase in computational complexity, particularly for resource-limited scenarios.

References

[1] J. McKay, Automatic musical instrument identification, Master's thesis, Dublin Institute of Technology, 2011.

[2] S. Murthy and S. G. Koolagudi, Content-based music information retrieval (cb-mir) and its applications toward the music industry: A review, **ACM Computing Surveys (CSUR),** vol. 51, no. 3, pp. 1–46, 2018.

[3] C. Constantinescu and R. Brad, An overview of sound features in time and frequency domain, **International Journal of Advanced Statistics and IT&C for Economics and Life Sciences**, vol. 13, no. 1, 2023.

[4] J. Pons, O. Nieto, M. Prockup, E. Schmidt, A. Ehmann, and X. Serra, End-to-end learning for music audio tagging at scale, arXiv preprint arXiv:1711.02520, 2017.

[5] W. Qin and B. Yin, Environmental sound classification algorithm based on adaptive data padding, in **2022 International Seminar on Computer Science and Engineering Technology (SCSET)**, pp. 84–88, IEEE, 2022.

[6] B. Toghiani-Rizi and M. Windmark, Musical instrument recognition using their distinctive characteristics in artificial neural networks, arXiv preprint arXiv:1705.04971, 2017.

[7] P. Uruthiran and L. Ranathunga, Optimization of feature selection and classification of oriental music instruments identification, in **1st International Conference on Artificial Intelligence and Data Sciences (AiDAS)**, pp. 120–125, IEEE, 2019.

[8] H. Tu and Y. Li, Neural network for music instrument identification, CS 229 Machine Learning Final Project, Stanford University, 2023.

[9] Y. Su, Instrument classification using different machine learning and deep learning methods, **Highlights in Science, Engineering and Technology**, vol. 34, pp. 136–142, 2023.

[10] R. Bhagyalakshmi and M. Anandaraju, Identification of specific musical instruments using machine learning models, **Journal of Integrated Science and Technology**, vol. 13, no. 5, pp. 1108–1108, 2025.

[11] J. J. Bosch, J. Janer, F. Fuhrmann, and P. Herrera, A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals, in **13th International Society for Music Information Retrieval Conference (ISMIR 2012)**, pp. 559–564, 2012.

[12] R. M. Bittner, J. Salamon, M. Tierney, M. Mauch, C. Cannam, and J. P. Bello, MedleyDB: A multitrack dataset for annotation-intensive MIR research, in **15th International Society for Music Information Retrieval Conference (ISMIR 2014)**, vol. 14, pp. 155–160, 2014.

[13] S. K. Mahanta, A. F. U. R. Khilji, and P. Pakray, Deep neural network for musical instrument recognition using MFCCs, **Computación y Sistemas**, vol. 25, no. 2, pp. 351–360, 2021.

[14] G. M. Bhandari, Different audio feature extraction using segmentation, **International Journal for Innovative Research in Science Technology**, vol. 2, no. 9, pp. 1–5, 2016.
[15] P. A. Babu, V. S. Nagaraju, and R. R. Vallabhuni, Speech emotion recognition system with “, in **10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)**, pp. 421–424, IEEE, 2021.
[16] D. Lavry, Sampling theory for digital audio, Lavry Engineering, Inc., 2004.
[17] E. Bressert, **SciPy and NumPy: An overview for developers**, O’Reilly Media, Inc., 2012.
[18] I. H. Sarker, Machine learning: Algorithms, real-world applications and research directions, **SN Computer Science**, vol. 2, no. 3, p. 160, 2021.
[19] S. Fatima, A. Hussain, S. B. Amir, S. H. Ahmed, and S. M. H. Aslam, et al., XGBoost and random forest algorithms: An in-depth analysis, **Pakistan Journal of Scientific Research**, vol. 3, no. 1, pp. 26–31, 2023.
[20] G. Kunapuli, **Ensemble methods for machine learning**, Simon and Schuster, 2023.
[21] K. Friedrichs, N. Bauer, R. Martin, and C. Weihs, A computational study of auditory models in music recognition tasks for normal-hearing and hearing-impaired listeners, **EURASIP Journal on Audio, Speech, and Music Processing**, vol. 2017, pp. 1–22, 2017.
[22] P. Bonissone, J. M. Cadenas, M. C. Garrido, and R. A. Díaz-Valladares, A fuzzy random forest, **International Journal of Approximate Reasoning**, vol. 51, no. 7, pp. 729–747, 2010.
[23] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, A comparative analysis of gradient boosting algorithms, **Artificial Intelligence Review**, vol. 54, pp. 1937–1967, 2021.
[24] D. Nielsen, Tree boosting with XGBoost – why does XGBoost win every machine learning competition?, Master’s thesis, NTNU, 2016.
[25] A. Samat, E. Li, W. Wang, S. Liu, C. Lin, and J. Abuduwaili, Meta-XGBoost for hyperspectral image classification using extended MSER-guided morphological profiles, **Remote Sensing**, vol. 12, no. 12, p. 1973, 2020.
[26] G. Agostini, M. Longari, and E. Pollastri, Musical instrument timbres classification with spectral features, **EURASIP Journal on Advances in Signal Processing**, vol. 2003, pp. 1–10, 2003.