From Oral Tradition to Digital Sheet: A Machine Learning Approach to Kulintang Music Transcription

Mary Robyn Bernabe
British School Manila
Manila, Philippines
e-mail: bernaberobyn@gmail.com

Atul Dubey
AIClub Research Institute, Mountain View
CA, USA
e-mail: atul.dubey@aiclub.world

***Abstract*—Kulintang music is a central component of the indigenous musical heritage of the southern Philippines, yet its transmission is increasingly at risk due to the absence of standardized written notation and its reliance on oral pedagogy. This limitation makes long-term preservation and global accessibility challenging. To address this gap, the first automated framework for converting kulintang performance videos into musical notes using machine learning algorithms is presented. A novel data set of 1,920 audio segments was constructed, derived from recordings of nine kulintang instruments, each exhibiting distinct tunings, timbral properties, and culturally specific acoustic characteristics. Audio was segmented through silence detection and transformed into spectral features optimized for per-note classification. Three supervised learning models (k-Nearest Neighbors (KNNs), Random Forest, and Multilayer Perceptron) were benchmarked to evaluate their effectiveness in recognizing kulintang notes across heterogeneous instruments. Random Forest achieved the strongest performance, with 90.03% cross-validation accuracy and 87% test accuracy, indicating the feasibility of data-driven transcription for this musical form. This work represents one of the earliest computational approaches for automated kulintang transcription and contributes new resources, methodologies, and tools that support preservation, education, and broader engagement with Filipino cultural heritage.**

Keywords-Kulintang; KNN; Random Forest; MLP.

I. Introduction

In the context of Southeast Asian indigenous and musical culture, the kulintang represents both an instrument and an ensemble as one of the most culturally significant musical traditions. The kulintang refers to a set of eight small gongs, arranged in increasing size and laid horizontally on a wooden rack, as shown in Figure 1.

Figure 1. Kulintang (source: ta.wikipedia.org).

Beyond its own instrument, the kulintang is typically accompanied by four other instruments that complete the ensemble. These include gandingan, the agung, babandil, and dabakan. Together, these instruments form the standard kulintang ensemble [1]. Although the kulintang ensemble includes several accompanying instruments, the following work specifically focuses on the kulintang as the primary subject and focus of the investigation.

Historically, kulintang music has been performed at community gatherings, weddings, and ritual events, serving as a form of entertainment and social identity. In addition, the kulintang and its music have been closely related to oral traditions, storytelling, and the cultural identity of indigenous societies in the Philippines. Specifically, the Maguindanao people of Cotabato, Sultan Kudarat, and Maguindanao provinces have the kulintang as a key aspect of their indigenous culture. In addition, other groups such as the Maranao of Lanao del Sur, Lanao del Norte and the Tausug of Jolo Island continue traditions regarding the kulintang, implementing this instrument into their dances, rituals and celebrations [2].

Currently, only a limited number of communities continue to actively perform kulintang music. As the tradition is primarily shared through oral tradition, this makes individual learning and sharing the kulintang music of the instrument difficult. Furthermore, standardized notation and written music for the kulintang can be rare and scarce, restricting access for musicians and learners to embrace the culture of the kulintang. Developing written forms of this music could address these difficulties by simplifying the learning and transmission of the repertoire. In this regard, Lucrecia Kasilag's cipher notation provides a culturally appropriate system in which kulintang music may be represented in written form [3].

For the following work, original data collection for the dataset was conducted by filming several kulintang players from several universities in the Philippines performing different kulintangs, with singular note strikes of each going from lowest to highest pitch. This resulted in a full dataset of 1920 audio segments across nine different kulintangs, allowing a comprehensive dataset for model development. In addition, the pitches of the kulintang gongs are not standardized and can vary based on the specific instrument and the player, and changes in unique tunings for different performances.

To address the limited data and documentation of kulintang music, this study proposes a model that identifies sequences of kulintang music notes from video recordings of performances. The process starts with the extraction of audio segments from video using silence detection to divide a performance into each of its notes, which are then processed into features through spectrogram analysis. These features are then used with machine learning techniques, including Random Forest, k-Nearest Neighbours (KNN), and Multi-Layer Perceptron (MLP) to build a model that can correctly identify kulintang music notes for each gong. Though this is just one part of the music transcription process, in future, it can be extended by capturing other finer details of pauses, tempo and rhythms in order to get the complete music sheet.

This work is motivated by the challenges of documenting kulintang music in a form that can endure and last beyond oral transmission. By applying machine learning techniques to produce culturally appropriate musical transcriptions, the study aims to support preservation and wider access without reducing the kulintang's cultural traditions to Western musical conventions.

The rest of the paper is laid out as follows. Section II presents the related work on music transcription and machine learning relevant to this study. Section III presents the techniques and related steps followed for building and evaluating the model. Section IV presents the validation and test results obtained from the various experiments. Section V discusses the results and behavior of the models. It also presents the shortcomings and challenges of the research. Finally, Section VI concludes the research by mentioning future possibilities.

II. Related Work

There has been research that has explored automatic music transcription for modern and Western instruments, such as piano and guitar, using machine learning techniques. Firstly, a study by Ru [4] investigates automatic music transcription and audio analysis within the context of the Western musical system, focusing on machine learning methods for extracting symbolic musical information from audio. They have converted audio signals into a set of feature representations using spectral and temporal analysis, and built a framework that integrates Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to extract temporal and spectral features from piano audio recordings. Moreover, attention mechanisms and reinforcement learning techniques were introduced to fully convert music to music sheets.

Similarly, Chieppa, Brutti, and Paiva [5] evaluated a note-level automatic guitar transcription model. They have created a custom GuitarSet dataset that includes MIDI transcription. To build a machine learning model on top of state-of-the-art note-level transcription models that use a self-attention mechanism, their model incorporates beat-informed quantization to accurately convert audio signals into tablature.

The above studies prove the effectiveness of machine learning and deep learning techniques on music transcription, but there have been no studies done on the kulintang so far. The objective of this paper is to explore the transcription of kulintang music using culturally specific notation systems and instrumentation.

III. Materials and Methods

This section describes the tools, techniques, and processes followed for performing the experiments.

A. Dataset

The original dataset for this study was collected from video recordings of several kulintang players (with different kulintangs) across different universities and schools in the Philippines. Each performer played a full scale of the instrument, striking each individual gong from the lowest to the highest pitch. As a result, this produced a total of 1920 audio segments recorded from nine different kulintangs to provide a varied dataset. Since kulintangs are created by hand, each instrument is not standardized, where each can have tonal variability and different pitches and tunings. When preparing the data for analysis, the audio was extracted and labelled manually according to its pitch number (1-8). The full dataset can be found [6].

B. Machine Learning Techniques

The **KNN** [7] algorithm is a classification method that determines the category of a new data point based on its proximity to previously known examples. It was selected to find the effectiveness of a simple neighbor based model on notes identification.

Random Forest [8] is an ensemble learning method that operates by constructing multiple decision trees and combining their outputs to produce a final prediction. This technique was used to explore the effectiveness of tree-based ensemble techniques on kulintang notes detection.

The **MLP** [9] is the simplest artificial neural network architecture composed of interconnected layers of neurons. In this study, it was used to explore the impact of simple deep learning techniques in distinguishing kulintang notes.

C. Data Processing and Featurization

Audio was extracted from each video using MoviePy. The Librosa library was used to identify non-silent intervals in the audio by setting a decibel threshold of 0.7 and removing the transients and background silence. Based on the detected intervals, each video was segmented into shorter clips corresponding to non-silent segments. These snippets represent isolated note events. For every segmented video snippet, the associated audio track was extracted and stored in .wav format, ensuring that the audio features could be processed independently of the video data.

Each audio snippet was converted into 25 numerical audio features describing its temporal and spectral properties. These included Root Mean Square Energy (RMSE), spectral centroid, spectral bandwidth, spectral rolloff, zero-crossing rate, and 20 Mel-Frequency Cepstral Coefficients (MFCCs). The mean value of each feature across the duration of the audio segment was calculated to produce a consistent numerical representation for each note. Metadata such as video name, snippet identifier, and file path were preserved, and each snippet was manually labeled with its note number (1-8, where 1 represents the lowest note and 8 the highest). The complete set of features and metadata of the entire dataset was compiled into a structured CSV file.

The dataset was divided into training and testing subsets using an 80:20 split using a stratified sampling method, ensuring that the distribution of the note classes was preserved across both subsets. The training set was used for model development, while hyperparameter tuning was performed through 5-fold cross-validation. The testing set was used for final performance assessment.

D. Model Training

As seen in Figure 2, the featurized dataset was used to train and evaluate three supervised machine learning models. The training dataset was used to train the models to recognize audio patterns that were associated with each note, while hyperparameter tuning was performed using 5-fold cross-validation. Finally, the testing dataset was used to evaluate the performance of the best-performing model. For KNN, the k-values were changed from 1 to 19 to find the best-performing value. For Random Forest, the number of trees was varied between 10-100, while the depth was varied between 1-14. For MLP, epochs were varied between 20-200, and learning rates between 0.0001-0.01.

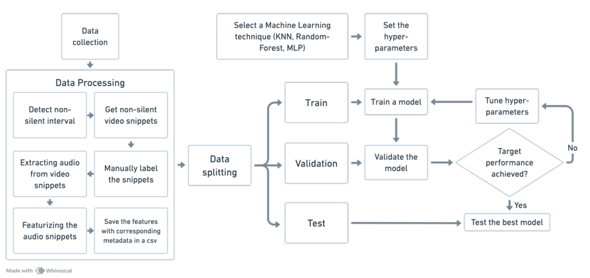

Figure 2. Overview of the proposed machine learning pipeline, including data processing, dataset splitting, model training, hyperparameter tuning, and evaluation.

E. Web Application

To test the model in real-world scenarios and make the model accessible, a web application was developed. In the web application, once a video is uploaded, it is processed to detect silent intervals, create snippets out of non-silent intervals, and extract audio from the snippets. The audio snippets are finally featurized and the resulting features are then passed through the model for prediction. In the predictions, if the confidence threshold is below 0.7, the snippet is ignored. Then, the rest of the predictions are then saved in a CSV file with timestamps that are relative to the original video. Finally, an output video is created by overlaying the predictions on the original video.

All the code required to reproduce data processing, featurization and model training can be found in [6].

IV. Results

In total, 209 experiments were performed with KNN, Random Forest, and MLP. The results obtained from the experiments are presented in this section.

A. KNN

A total of 19 experiments were performed by varying the k-values from 1-19 and the best validation accuracy of 59.43% was achieved at a k-value of 1. Figure 3 shows the variation of the validation accuracy based on k-values.

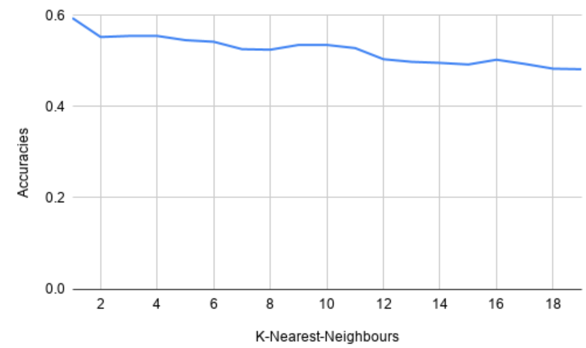

Figure 3. Cross-validation accuracy of the KNN model for different values of k.

B. Random Forest

A total of 140 experiments were performed by varying the number of estimators from 10-100 and the maximum depth from 1-14. The best cross-validation accuracy of 90.03% was achieved with 50 estimators and a maximum depth of 10. Figure 4 shows the variation of cross-validation accuracies with respect to maximum depths for various numbers of estimators.

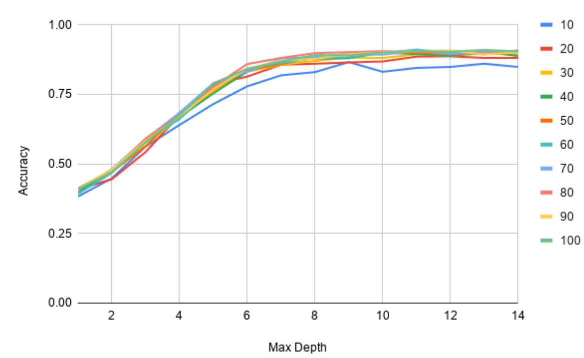

Figure 4. Cross-validation accuracy of the Random Forest model for different maximum depths and numbers of estimators.

C. MLP

A total of 50 experiments were performed by varying the number of epochs from 20-200 and learning rates from 0.0001 to 0.01. The best cross-validation accuracy of 49.24% was achieved at 200 epochs and a learning rate of 0.001. Figure 5 shows the variation of cross-validation accuracies with respect to epoch for various learning rates.

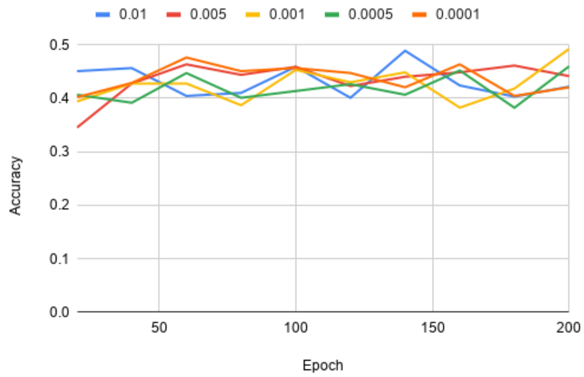

Figure 5. Cross-validation accuracy of the MLP model for different numbers of epochs and learning rates.

D. Results Summary and Test Results

Table I shows the best cross-validation accuracies for different models.

TABLE I. CROSS-VALIDATION ACCURACY OF DIFFERENT MODELS.

Model	Cross-validation accuracy (%)
KNN	59.43
Random Forest	90.03
MLP	49.24

Out of all the models, Random Forest achieved the highest performance, obtaining an accuracy of 87% on the testing dataset. Figure 6 shows the normalized confusion matrix for the best-performing Random Forest model evaluated on the testing dataset.

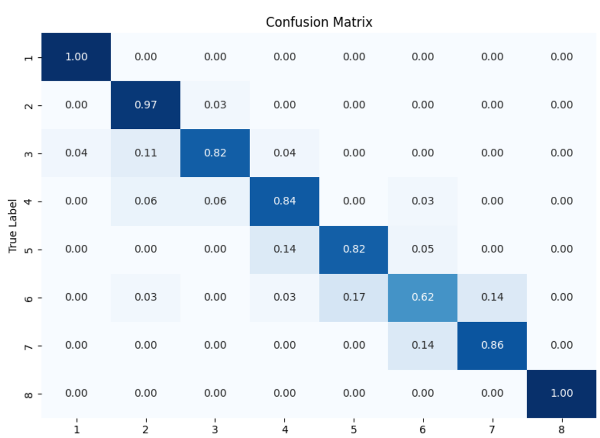

Figure 6. Normalized confusion matrix for the best-performing Random Forest model evaluated on the testing dataset.

As seen in Table II, the classification model demonstrated strong overall performance, achieving high precision, recall, and F1-scores across all eight classes, with particularly outstanding results for Class 1 and Class 8, both attaining an F1-score of 0.99 or higher.

TABLE II. TEST CLASSIFICATION REPORT FOR THE BEST MODEL

Class	Precision	Recall	F1-score	Support
1	0.97	1.00	0.99	35
2	0.83	0.97	0.90	31
3	0.88	0.82	0.85	28
4	0.84	0.84	0.84	31
5	0.78	0.82	0.80	22
6	0.78	0.62	0.69	29
7	0.82	0.86	0.84	21
8	1.00	1.00	1.00	19

V. DISCUSSION

In this study, the analysis of the KNN model indicates that accuracy starts at 59.43% with k = 1 and generally decreases as k increases. For example, at k = 2, the accuracy is 55.27%, and it continues to decline, reaching 48.20% by k = 19. The reasons for the low performance could be the number and nature of the extracted features.

In the case of Random Forest, increasing both parameters generally led to improved model accuracy. For a maximum tree depth of 10, the accuracy began at 38.23% with 1 estimator and peaked at 86.56% with 9 estimators, demonstrating early gains from increasing the complexity of the model. At deeper tree levels such as 50 and 70, accuracy improvements were observed up to around 90.03%, showing how additional depth and estimators can enhance the ability to capture complex patterns in the data. However, although higher values provided a more consistent performance boost, the returns diminished with very high depths and estimator counts, suggesting a plateauing effect.

MLP results showed that for a learning rate of 0.01, accuracy starts at 45.07% with 20 epochs, fluctuating, but reaches 48.89% at 140 epochs, suggesting some sensitivity to the number of epochs due to potentially more dynamic updates. Similarly, a learning rate of 0.005 shows improvement from 34.53% at 20 epochs to 46.15% at 180 epochs, indicating that a lower learning rate may benefit from more epochs as it updates weights more gradually. Conversely, at a very low learning rate of 0.0001 and increasing epochs, the model's performance displayed less variability, with accuracy reaching up to 47.63% at 60 epochs but not exceeding this substantially thereafter. Overall, the Random Forest model demonstrated its robust capability to capture complex patterns in the data through its ensemble approach, which leverages multiple decision trees to enhance predictive performance. While both KNN and MLP showed potential under specific conditions, they were more sensitive to their respective parameter choices and struggled to maintain the same level of accuracy across a broader range of configurations.

VI. CONCLUSION AND FUTURE WORK

This study concludes that by featurizing audio and employing classical machine learning techniques, kulintang music notes can be effectively identified. The Random Forest model achieved the highest performance, with a test accuracy of 87%, indicating the validity of this approach. However, the model encountered challenges in real-world applications, primarily due to external noise not being accounted for in the initial dataset. In addition, the proposed segmentation approach relies on silence detection to isolate individual notes. In faster kulintang performances, the decay of one gong may overlap with the attack of the next note, reducing the amount of detectable silence and potentially resulting in inaccurate note segmentation. Another limitation of this study is that note segments extracted from the same recording may be present in both the training and testing datasets, potentially affecting the model's ability to generalize to unseen recordings. Addressing these limitations by augmenting the dataset with background noise, investigating more robust segmentation techniques such as onset detection, using separate recordings for training and testing sets, and retraining the model could enhance its robustness. Future research could explore deep learning techniques, such as CNNs, known for their strength in handling audio data more intricately. Additionally, practical applications of this work could include automated transcription systems that preserve and promote traditional music forms. Through these improvements and applications, this research contributes to the growing field of music transcription and audio signal processing.

References

[1] D. S. Kalanduyan, "Magindanaon kulintang music: Instruments, repertoire, performance contexts, and social functions", *Asian music*, vol. 27, no. 2, pp. 3–18, 1996.

[2] S. N. Abdullah, "Kulintang ensemble of maguindanaon: Cultural appreciation, reflection and preservation (carp) of their traditional practices", *Samsudin N. Abdullah, PhD*, 2020.

[3] A. R. Franquelli, *Lucrecia R. Kasilag: The Western and Oriental Influences in Her Compositions.* Peabody Institute of the Johns Hopkins University, 1979.

[4] T. Ru, "Deep learning-based automatic piano note recognition and performance generation system", *International Journal of High Speed Electronics and Systems*, p. 2 540 814, 2025.

[5] S. Chieppa, P. Brutti, and R. P. Paiva, "Automatic guitar transcription with deep neural networks", *IEEE Access*, 2025.

[6] M. R. Bernabe, *KulintangProject*, https : / / github . com / Chineebernabe/KulintangProject, GitHub repository. Accessed 28 Mar. 2026.

[7] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, "KNN model-based approach in classification", in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, Springer, 2003, pp. 986–996.

[8] L. Breiman, "Random forests", *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[9] M.-C. Popescu, V. E. Balas, L. Perescu-Popescu, and N. Mastorakis, "Multilayer perceptron and neural networks", *WSEAS Transactions on Circuits and Systems*, vol. 8, no. 7, pp. 579–588, 2009.

 ISBN: 978-1-68558-403-0