Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547

Sound Classification of Musical Instruments with Sonogram Features using Machine Learning Algorithms

S. Prabavathy[1], R. Selvalakshmi[2]

[1, 2]Assistant Professor

[1]Department of Computer Applications, [2]Department of Computer Science

[1]Arulmigu Subramania Swamy Arts and Science College, Vilathikulam, Thoothukudi, Tamilnadu, India.

[2]Kamaraj College (Autonomous), Thoothukudi, Tamilnadu, India.

To Cite this Article

S. Prabavathy, R. Selvalakshmi" **Sound Classification of Musical Instruments with Sonogram Features using Machine Learning Algorithms**" *Musik In Bayern, Vol. 91, Issue 4, April 2026, pp1-22*

Article Info

Received: 12-02-2026 Revised: 13-03-2026 Accepted: 24-03-2026 Published: 06-04-2026

Abstract

Music can be generated by the interplay of a variety of instruments. Basically, humans identify the musical instrument from which it has been played; however, it is difficult for a machine to investigate it automatically. Moreover, different instruments produce various sounds in terms of factors like timbre, supremacy, and kind of playing, which makes the music identification more tedious. Initially, instrument data is highly significant and helpful for humans, and it is added in audio tags. Musical Instruments Sound analysis is carried out in different formats. The proposed method classifies the sound of the musical instruments using Sonogram with k-Nearest Neighbor and Support Vector Machine. 22-dimensional Sonogram features are used for the classification of musical instruments. KNN and SVM are used to identify the class label of the musical instrument sound for the respective music signal. Experiments are conducted for the acoustic feature namely, Sonogram, and the performance of machine learning algorithms are studied. In this paper, the performance of Sonogram with SVM yields 97.98 % and Sonogram with KNN yields 95.54%. Sonogram with SVM performs well and gives more accuracy when compared to Sonogram with KNN in this work

Keywords: Musical Instrument Classification, Musical Instruments Sound, Sonogram, k-Nearest Neighbor, Support Vector Machine.

1 INTRODUCTION

Digital audio systems have now dominated audio distribution, with CD players, internet radio, mp3 players, and iPods frequently being the systems of choice [1]. Mixing of desks for live events digital processing predominates is highly applicable in many applications like television and film studios. The requirement to automatically classify and categorize audio data has

caused audio classification to recently become a significant study subject. All types of audio transmissions, including voice, music, and more general sound signals and their combinations, are referred to as audio. Thousands of audio recordings can easily be found in multimedia databases or file systems. However, the audio is typically seen as a cryptic group of bytes with only the most basic fields—such as file type, name, sample rate, etc.—attached. Digital audio waveforms can be used to extract meaningful information that can be compared and categorised effectively.

REPRESENTATION OF AUDIO SIGNALS

Digital processing is now getting preference for managing audio and speech. This is because new audio applications and systems are predominantly digital. Speech, Audio, and Hearing related research or development are centered on digitalization these days. Since digitalization fosters platform independence, one can create and prototype using a digital processing platform, and then deploy it on another platform. Such a development platform would be for ease-of-use and testing, while the criteria for a deployment platform may be separate: low power, small size, high speed, low cost, etc. Representations of musical instrument sounds are not unique, it is very useful to represent such sound as a collection of sine waves (sinusoids) with time-varying amplitudes, frequencies and phases and possibly with an additive noise signal having certain time-varying spectral properties.

1.1. MUSICAL INSTRUMENTS SOUND CLASSIFICATION

Music analysis has been an extremely active research topic. Each style of instrument has its distinct sound quality and is associated with particular forms of music. Experienced musical listeners can usually identify which instruments are present in a music recording, although identification accuracy varies with the prominence of an instrument (in the music), familiarity, the number of instruments played, etc., Classification of music signals by humans is an implicit task, but a little challenging for computer systems. The use of digital audio signal processing includes a sequence of processes, for compressing the digitalized audio signal, productions of audio effects as well as classifying the audio. In today's world, one can observe the great importance of multimedia content management in audio segmentation and classification. The major problems are experienced in audio and visual data handling. It is occupied by audio classification with important applications in broad fields [2].

1.2. TYPES OF MUSICAL INSTRUMENTS

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2026-547**

Musical instruments are said to be global units of human culture. Archaeology has stated that, pipes and whistles are existed from Paleolithic Period, clay drums and shell trumpets are from Neolithic Period. It is strongly developed that the ancient city cultures of Mesopotamia, the Mediterranean, India, East Asia, and the Americas are composed of well-equipped assortments of musical instruments, which represents a longer and previous deployment of a model. In this work, twelve musical instruments from three different types of musical instruments family are taken for classification namely French horn, Trumpet, Trombone and Tuba from Brass, Banjo, Cello, Violin and Guitar from String, and Flute, Bass Clarinet, Clarinet and Bassoon from Woodwind are classified in this proposed work. Figure 1 shows the types of musical instruments.

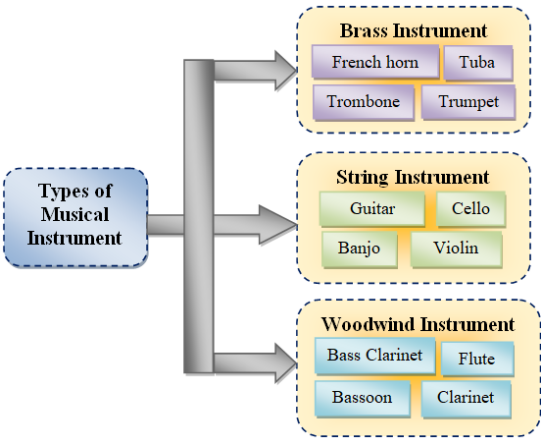

Figure 1: Types of Musical Instrument

In this paper, the Sonogram features [10] are used extraction and the classifier Support Vector Machine [11] and k-Nearest Neighbor [12] are used to classify the musical instruments. Figure 2 shows the block diagram for the proposed work.

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547

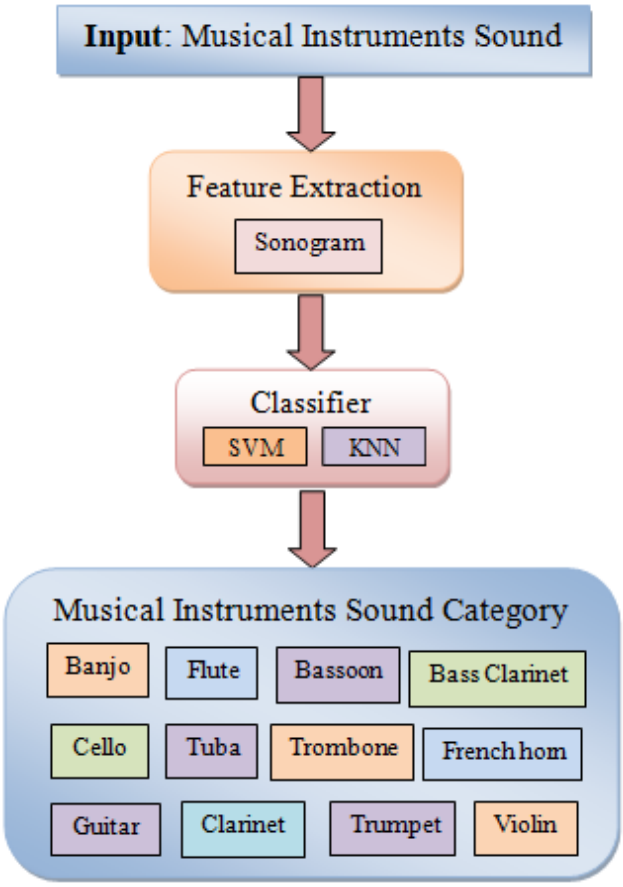

Figure 2: Block diagram of the proposed work

II LITERATURE REVIEW

In [3] a beat histogram determination model demonstrated to identify the beat strength that has been applied for finding tempo of music recording. It has been searched with automated classification of audio signals as hierarchy of musical genres. Also, it is presented with 3 feature sets to demonstrate timbre texture, rhythmic data as well as pitch information. The function as well as relative significance of the presented characteristics is examined by training statistical pattern analyzing classifiers with the help of actual audio collections. Under the application of presented feature sets, maximum classification rate has been achieved for 10 musical genres. Finally, the results are comparable and addressed for human musical genre classification.

In [6] the Epithelial Dysplasia images are collected and classified as normal or epithelial dysplasia. The image with RGB are converted into HSV Color space and SURF and SIFT features are extracted and the Support Vector Machine is used as a classifier for classification. 91.4% of accuracy obtained for SURF, SURF gives a better performance when compared to SIFT with SVM model.

In [8] the distinct notes applied for classifying musical instruments. The Time Encoded Signal Processing generates ordinary matrices to encode the notes and obtained from tedious sound

wave. The encoded signals are elegant and light weight in processing model. The final matrices are fed as an input of Fast Artificial Neural Network (FANN) for examining the musical instrument with effective outcomes and lower the processing expense than traditional system. The classification of musical instrument with the application of FFNN classification model [4] integrates features of temporal and spectral of musical instrument. It has been executed in 2 phases. Initially, filter the spectral features from musical signal which are applied for examining the instrument with the help of diverse frequency evaluation models. Secondly, a FFNN is applied for classifying the signals.

In [7] the musical instruments sound classification is done using Convolutional Neural Network which is one of the techniques in deep learning. The classification deals with sixteen types of musical instruments from four different families and yields satisfactory results.

Demonstrated a beat histogram determination model in [18] to identify the beat strength that has been applied for finding tempo of music recording. It has been searched with automated classification of audio signals as hierarchy of musical genres. Also, it is presented with 3 feature sets to demonstrate timbral texture, rhythmic data as well as pitch information. The function as well as relative significance of the presented characteristics is examined by training statistical pattern analyzing classifiers with the help of actual audio collections. Under the application of presented feature sets, maximum classification rate has been achieved for 10 musical genres. Finally, the results are comparable and addressed for human musical genre classification.

In [19] investigated the classification of musical instrument sounds using both k-Nearest Neighbor (kNN) and Convolutional Neural Networks (CNNs). In their study, audio signals were preprocessed and transformed into time–frequency representations, such as Mel-spectrograms, to capture spectral and temporal characteristics of the instruments. The kNN classifier served as a baseline to evaluate similarity-based classification, while CNNs were employed for automated feature extraction and higher-level pattern learning. Results demonstrated that CNNs significantly outperformed kNN in terms of accuracy and robustness, highlighting the advantage of deep learning for capturing complex timbral patterns. The study reinforces the trend of combining classical and deep learning methods for effective musical instrument recognition, particularly when using spectrogram or sonogram-based features

In [20] the authors of this paper developed and tested musical instrument detection for TurkishMarchPat-based feature engineering model. In this work, the kaggle online dataset were used for classification. Three seconds duration with the sampling frequency of 22.05kHz with 66,150 samples per sound were used. The proposed system achieved 97.87 % of accuracy.

In [21] investigated the identification of specific musical instruments using traditional machine learning models by extracting time–frequency audio features such as Mel-spectrograms and MFCCs. Their study evaluates the performance of classifiers including Support Vector Machine (SVM) and other learning models for distinguishing instrument sounds based on spectral characteristics. The results indicate that machine learning algorithms can effectively classify musical instruments when appropriate feature representations are used, particularly for monophonic audio signals. The work highlights the continued relevance of classical machine learning techniques for musical instrument sound classification, especially in scenarios with limited data and computational resources.

In [22] employed a supervised deep learning paradigm for musical instrument classification, where raw audio signals from the IRMAS dataset are first preprocessed and transformed into log-Mel spectrograms to capture essential spectral and temporal features. These spectrograms are then fed into convolutional neural networks (CNNs), including architectures like DenseNet121 and ResNet-50, to automatically learn discriminative patterns for each instrument class. The models are trained using labeled data, and their performance is evaluated with metrics such as accuracy, precision, recall, and F1-score. This paradigm demonstrates that deep CNN-based approaches effectively extract complex timbral features and outperform traditional machine learning methods, providing accurate and reliable classification of musical instruments.

In [23] the author proposes a deep-learning framework specifically designed for the classification of Chinese traditional musical instruments. It employs multi-feature fusion, combining spectral, temporal, and cepstral features such as spectrograms, MFCCs, and chroma features, to comprehensively capture the unique timbral characteristics of each instrument. To enhance the model's ability to identify discriminative patterns, an attention mechanism is incorporated, allowing the network to focus on the most relevant segments of the audio signal. The framework is evaluated on a dataset of Chinese traditional instrument sounds and demonstrates superior classification accuracy compared to traditional machine learning models like SVM and kNN, as well as standard CNN architectures. This study highlights the effectiveness of integrating feature fusion with attention mechanisms in deep learning models for complex instrument sound recognition tasks.

III PREPROCESSING

A musical instruments sound signal preprocessing takes place as follows. The preprocessing of raw musical data contains three phases they are pre-emphasis, segmentation and windowing

phases. Initially, the original music signal is pre-processed and the major use is for unifying the music format, make pre-emphasis and divides the music signal into music segments. Then, perform windowing and framing to all musical segments.

- Preemphasis Processing: On combined with the human ear hearing process, the audio frequency range which could be heard by the human ear is 60 Hz-20 kHz. If audio signal modeling is executed, the audio signal is pre-emphasized, and it eliminates minimum-frequency interference, particularly 50 Hz or 60 Hz power frequency interference. Pre-emphasis is usually performed by the digitalization of the audio signal with a pre-emphasis digital filter that is usually a 1st-order high-pass digital filter:

$$H(z) = 1 - \mu z^{-1} \quad (4)$$

With respect to time field, when the processed signal is $y(n)$, next $y(n)$ can be defined as:

$$y(n) = x(n) - \mu * x(n-1) \quad (5)$$

Where $x(n)$ is the original signal series and $y(n)$ is the pre-emphasized series. During the procedure, the pre-weighting coefficient μ is obtained as [0.97, 0.98]. A 1st-order high-pass digital filter. With pre-emphasis modeling, the result of sharp noise is diminished, and the maximum-frequency segment of the signal is boosted that creates the spectrum of the signal flat, and pre-emphasis coefficient is generally on 0.97 or 0.98. A signal namely pre-emphasized with the filter requires to be normalization. Figure 3 shows the structure of pre-emphasis filter.

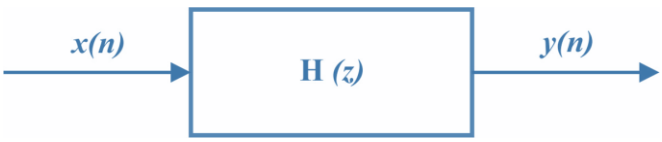

Figure 3: Structure of pre-emphasis filter

Windowed framing: Once the pre-emphasis digital filtering method is executed, the windowing and framing modeling is executed next. An audio signal characteristics modify very gradually over a short duration of time, thus removed audio features stay even in this slow transition. So, if modeling an audio signal, the separate audio signal is primary separated into a unit of length to modeling; i.e., the separate audio instance points are separated into audio frames. Usually, a "short-time" audio frame has time of about various tens of milliseconds. Based on the length of the separated audio unit, it split an audio unit into audio frame, audio clip, audio shot and audio maximum-level semantic unit.

While the framing implement the model of continuous segmentation, a system of overlapping segments is usually implemented to create a smooth transition among frames as well as continue its continuity. An overlapping portion of the preceding frame and the next frame is

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2026-547**
known as frame shift, and the frame shift is frequently obtained as half of the frame length. The framing is performed with weighting a restricted length window which is multiplied by $y(n)$ through a certain window function $w(n)$ to form a windowed audio signal $y_w(n) = w(n) * y(n)$. A signal in the time field is multiplied that is corresponding to the convolutional computation in the frequency field. So, the windowing computation is also being illustrated as follows:

$$Y_w(e^{j\omega}) = \frac{1}{2\pi}\int_{-\pi}^{\pi} Y\left(e^{j\theta}\right)W\left(e^{j(\omega-\theta)}\right)d\theta \tag{6}$$

where Y and W signified the spectrum, correspondingly. It is seen that the window function $w(n)$ not only influences the waveform of the original signal in the duration field, besides influences the shape of its frequency field. The 2 main generally utilized window functions are the rectangular window and Hamming window, as defined in the following Eqs. (7)-(8).

$$w(n) = \begin{cases} 1. & 0 \le n \le (n-1) \\ 0. & n = else \end{cases} \tag{7}$$

$$w(n) = \begin{cases} 0.54 - 0.46\cos\left[2\pi n/(N-1)\right], & 0 \le n(n-1) \\ 0, & n = else \end{cases} \tag{8}$$

The alternative of the shape and length of the window function $w(n)$ has a huge control on the features of the short-term analysis parameters. So, an appropriate window must be chosen for making the short-term parameters optimally return the characteristic alters of the speech signal. A rectangular window has optimal spectral smoothness, however the maximum-frequency module is missing, the waveform aspect is missing, and the rectangular window is reason leakage. A Hamming window is efficiently conquer the leakage (Gibbs) occurrence and has the widest function series. When the window length N is huge, it can be corresponding to a very slightly low pass filter.

If the audio signal passes, the high-frequency portion revealing the waveform facts is hindered, and its short-time energy modifies slightly with time. It does not truly reveal the amplitude difference of the speech signal. On the other hand, when N is too tiny, the passband of the filter becomes wider and short-term energy alters sharply with time, and a smooth energy function could not be attained. Thus, the length of the window might be selected properly, usually with a time of 15-30ms. Behind the modeling, the audio signal has been separated into short-time signals of a frame-by-frame plus window purpose, after that, all short-term audio frames is

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547
considered as a smooth arbitrary signal, and the digital signal methods are utilized for extracting the audio characteristic parameters. Figure 3 depicts the working frame blocking

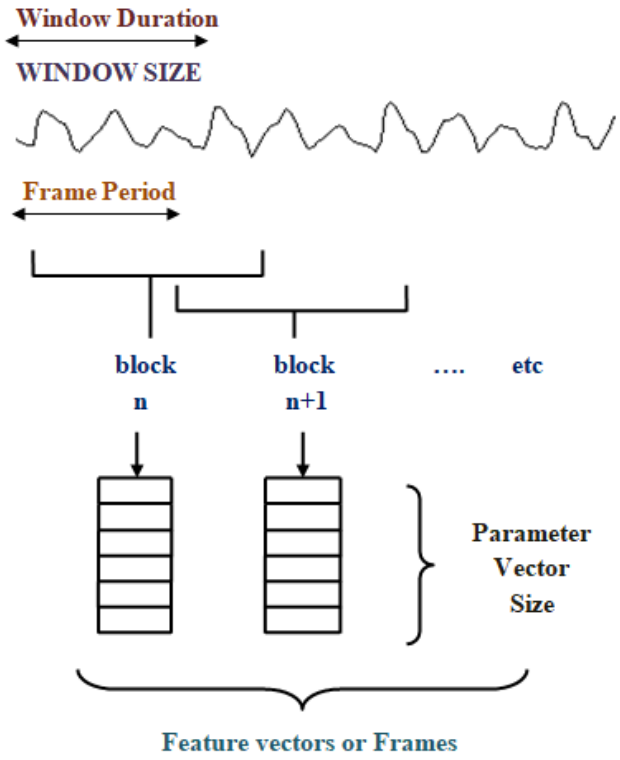

Figure 4: Frame Blocking

IV FEATURE EXTRACTION

4.1. Sonogram

Once the audio signal gets preprocessed, MFCC, Sonogram and MFCC-Sonogram combined features are extracted. Preemphasis takes place for the musical signal subsequent to frame blocking and windowing. The music segment is converted by the use of FFT into spectrogram representation. Bark scale is employed and frequency bands are clustered into 24 critical bands. Spectral masking effect is attained utilizing spreading function. The spectrum energy values are converted into decibel scale. Equal loudness contour is integrated for the calculation of the loudness level. The loudness sensation per critical band is determined and SIFT is calculated for every part of the pre-processed music. A frame size of 20 ms is organized with 50% overlap among the frames. The sampling frequency of 1s time period is 16 kHz. The general structure of sonogram based feature extraction is depicted in Figure 4.

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547

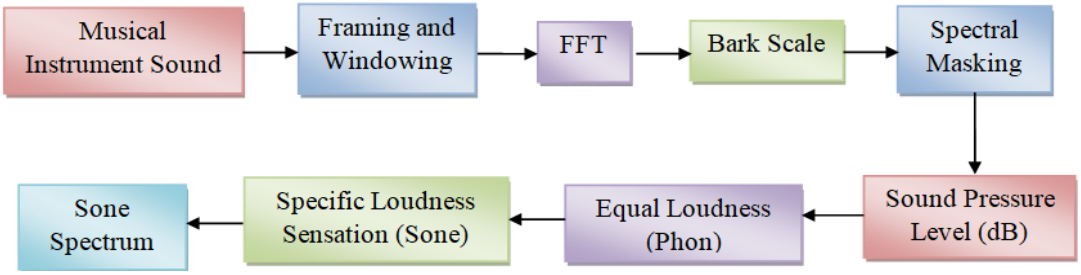

Figure 4: Sonogram based Feature Extraction

V CLASSIFIERS

5.1. Support Vector Machine

The SVM is a useful statistic machine learning technique that has been successfully applied in the pattern recognition area. If the data are linearly non separable but nonlinearly separable, the nonlinear support vector classifier will be applied [8]. The fundamental notion is to transform input vectors into a high-dimensional feature space using a nonlinear transformation (∅), and then to do a linear division in feature space as shown in Figure 5.

To construct a nonlinear support vector classifier, the inner product (x, y) is replaced by a kernel function K (x, y)

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{l} \alpha_i y_i K(x_i, x) + b\right) \qquad (9)$$

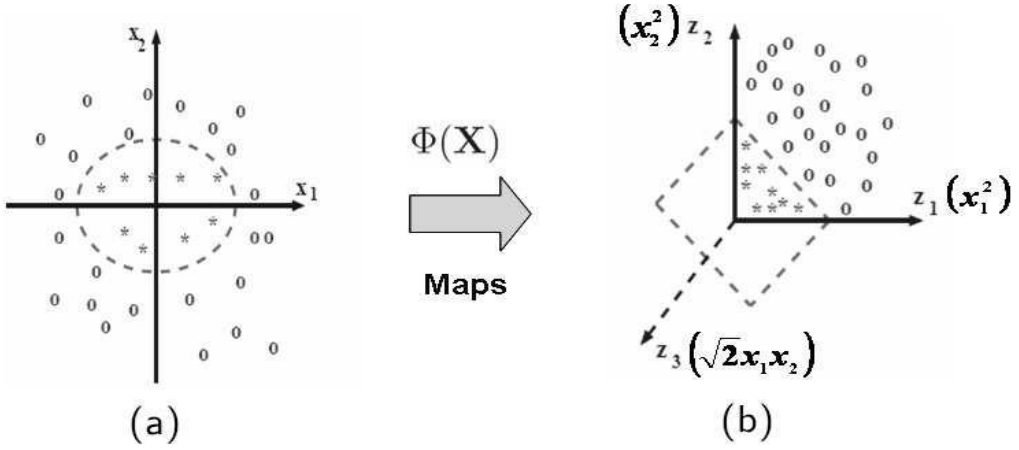

Figure 5: An example for SVM kernel function ∅ (x) maps two-dimensional input space to higher three-dimensional feature space.

(a) Nonlinear problem (b) Linear problem

Linear SVM

For linearly discrete problems, the dichotomous issue is creating a classification hyperplane in such a way that positive as well as negative instances are finally divided. As illustrated in Figure4.4, a solid sample points on the left stand for positive samples and hollow sample points on the right stand for negative instances. It is many classifier planes among H_1 and H_2, each of

that are capable to finally divide the positive as well as negative samples. When one of the classification faces could not only finally divide the positive as well as negative instances, it maximizes the geometric spacing, after that these classification lines are known as the better classification hyperplane. The supposed geometric spacing is the distance among H_1 and H_2. H is the classification plane, and H_1 and H_2 are straight lines equivalent to H and at the same time passing with the 2 kinds of samples nearest to the distance H. The sample point that occurs to fall on H_1 and H_2 is the support vectors is talking about. It can be support vectors which jointly create the better classification hyperplane. Let the linear discriminant function is $g(x) = wx + b$. With normalization, $\{x_1 \cdots x_n\}$ satisfies $g(x) \geq 1$, and now, the classification interval is defined by $2l||w||$.

$$y_i[wx_i + b] - 1 \geq 0, i = 1, \dots, n \qquad (10)$$

Noticeably, the better classification hyperplane must both assure Eq. (10) and minimize$||w||$. The SVM is an instance of the procedure (11). In brief, an effective classification hyperplane corresponds to the following restraint optimized issue:

$$min \ \ ||w||^2/2 \qquad (11)$$
$$s.t. y_i[wx_i + b] - 1 \geq 0, i = 1, \dots, n$$

During these methods, the results of SVM is completely changed into QP issues, thus theoretically the solution of SVM offers a globally single better result. Initially, a Lagrangian function is provided below:

$$M \arg \ in = 2l||w||$$
$$L(w, a, b) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} a_i \, y_i(x_i \cdot w + b) \ + \sum_{i=1}^{n} a_i \, , a_i \geq 0, i = 1,2, \dots, n \quad (12)$$

During the formula, a_i is the Lagrangian factor, next, correspondingly, differentiate the w and b in the above formula and create them equivalent to0, and obtain $w = \sum_{\mathrm{i}} a_i \, y_i x_i$ and $\sum_{\mathrm{i}} a_i \, y_i = 0$ for converting the original optimization problem into a dual problem:

$$\max W(a) = \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{ij=1}^{n} y_i \, y_j a_i a_j (x_i \cdot x_j) \qquad (13)$$
$$s.t. \sum_{i=1}^{n} y_i \, a_i = 0 a_i \geq 0, i = 1,2,3, \dots, n$$

Only the samples equivalent to a_i which are not 0 are support vectors. Generally only a tiny part of the instances have a_i not 0. The last classification function discriminant is given as follows:

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2026-547**

$$f(x) = sgn\left[\sum_{i=1}^{n} a_i^* \, y_i(x_i \cdot x) + b\right] \quad (14)$$

$$b = \frac{1}{2}\left[\sum_{i=1}^{n} a_i \, y_i x_i \cdot x_r + \sum_{i=1}^{n} a_i \, y_i x_i \cdot x_s\right] \quad (15)$$

The b can be computed by the above formula is the skew amount. If a_i^* in the formula is not $0, x_r$ and x_s signify some pair of support vectors in the 2 kinds of instances. Actually, it can be frequently control the noise in which the classification samples could not be divided linearly, and so an uncorrected classification hyperplane could not attained. It can be noticeably a sample of the negative class. These strange samples make the linearly separate problem linear and inseparable. Generally, this type of problem is known as "Approximate linear separability". To this type of problem, the common technique is the sample point in which originally the user is coincidentally mislabeled the sample that is interference, noise, and must be ignored. Now further fault tolerance and permit a hard threshold to be additional to a hard variable that permits many instance points to reduce in the geometric interval the term becomes the follows structure:

$$y_i[wx_i + b] \geq 1 - \xi_i \qquad \xi_i \geq 0, i = 1, \ldots, n \quad (16)$$

A slack variable is non-negative; i.e., the end outcome is which the sample interval is permitted to be smaller than 1. If the interval among the sample points is computed to be lesser than 1, it implies that the classifier quit the correct classification of these singular points. It also permits that the classified hyperplane to be stimulated to this sample points without being influenced by some sample points, resultant to superior geometric spacing.

It is significant that $||w||^2$is an objective function, with possible value, thus the loss might be an amount that creates $||w||^2$ superior. It is generally 2 methods for measuring loss; the initial is a 2nd-order soft-interval classifier:

$$\sum_{i=1}^{n} \xi_i^2 \quad (17)$$

The other is a 1st-order soft-interval classifier:

$$\sum_{i=1}^{n} \xi_i \quad (18)$$

In addition, a loss of objective function needs a penalty factor, thus the original optimization issues are expressed as:

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2026-547**

$$min\frac{1}{2}||w||^2 + C\sum_{i=1}^{n}\xi_i \quad (19)$$

$$s.t.\, y_i[wx_i + b] \geq 1 - \xi_i, \xi \geq 0 \;\; i = 1,2,3,\ldots,n \quad (20)$$

Nonlinear SVM

A fundamental rule of the SVM is resolving approximate linear separability problem. Practically, the minimum-dimensional sample space, the sample is highly inseparable. There is no issue exists to determine the classification hyperplane, there is always several singular points which does not meet the constraints. Currently, it can be essential for mapping the linearly inseparable sample data in the minimum-dimensional space to the maximum-dimensional space. While the mapping is not entirely linearly separable behind the mapping, it can be at minimum "approximate linear separable." Next with, the slack variable is utilized for managing a tiny count of singular points and it obtains extremely optimal outcomes. Mapping a sample from a minimum-dimensional space to a maximum-dimensional space requires to be executed by implying a kernel function, in order that the kernel function is:

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \quad 21)$$

A kernel function ensures the Mercer situation. Its essential function is to input the vector in 2 minimum-dimensional spaces and after that compute the vector inner product value of a changed maximum-dimensional space.

$$maxW(a) = \sum_{i=1}^{n} a_i - \frac{1}{2}\sum_{ij=1}^{n} y_i\, y_j a_i a_j K(x_i \cdot x_j) \quad (22)$$

$$s.t.\sum_{i=1}^{n} y_i\, a_i = 00 \leq a_i \leq C, i = 1,2,3,\ldots,n$$

A discriminate function becomes

$$f(x) = sgn\left[\sum_{i=1}^{n} a_i^*\, y_i K(x_i \cdot x) + b\right] \quad (23)$$

$$b = \frac{1}{2}\left[\sum_{i=1}^{n} a_i\, y_i K(x_i \cdot x_r) + \sum_{i=1}^{n} a_i\, y_i K(x_i \cdot x_s)\right] \quad (24)$$

Kernel function

A kernel function creates that the SVM execute well if managing nonlinear separable problems. The nonlinear classifiers created by several kernel functions are also by its variants. Some of the generally utilized kernel functions are given as follows:

(a) Linear kernel function:

ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2026-547**

$$K(x, x_i) = (x_i \cdot x) \quad (25)$$

(b) Polynomial kernel function:

$$K(x, x_i) = [p(x_i \cdot x) + s]^q \quad (26)$$

(c) Sigmoid kernel function:

$$K(x, x_i) = \tan h(\mu(x_i \cdot x) + c) \quad (27)$$

(d) Radial kernel function:

$$K(x, x_i) = \exp(-\gamma|x - x_i|^2) \quad (28)$$

The most extremely utilized of the above kernel functions is the radial basis kernel function that has a broad convergence filed and is fitting to several conditions namely minimum dimensional, maximum dimensional, tiny sample and massive sample.

SVM Multi-classification Method

Recently, a SVM multi-class classifier has been presented by developers and classified into two types: One is to develop the fundamental two kinds of SVM into multi-class classifier SVM, and these kinds of technique solve the optimized problem. The other is to regularly change the multi-class classifier problem into two kinds of classifier problems, i.e., to structure a multi-class classifier with multiple two-class classification SVM. Presently, such techniques are extremely utilized, and there are two generally utilized classification approaches: One Against One approach and One Against All approach.

Multi-SVM

As we have discussed about SVM, the aim of multi SVM is to allocate labels to occurrence by using SVM in which the labels are taken from a finite set of numerous elements [9]. Each training point belongs to one of N different classes. As the music classification involves multiclass classification and SVM is generally two-class based pattern classification model, diverse binary SVMs have to be carried out to produce a MSVM. The MSVM comprises of integrating $(k - 1)$ binary classifiers to a multiclass classifier.

5.2 k-Nearest Neighbor Classification Model

The k-Nearest Neighbor (KNN) model is a fundamental and significant classification technique in machine learning. It comes under the supervised learning and determines intensive function in pattern analysis, DM and intrusion prediction. It is highly not used in actual situations as it is non-parametric, which refers that, it does not create basic considerations on data distribution (as opposed to alternate techniques like GMM that considers a Gaussian distribution of provided data).

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547

The KNN technique considers the affinity among novel case and accessible cases which fix the novel case into the type which is most identical to the accessible types. It is utilized for Regression and Classification but the maximum attention is given to Classification problems. It is also known as lazy learner technique as it does not understand from training set adversely rather it saves the dataset and during classification, an action has been performed in the dataset. Assume that there are 2 classes, namely, Category A and Category B, with a novel data point x1, hence the data point comes under these classes. In order to perform this, KNN model has been applied proficiently. Under the application of KNN, the class of specific dataset can be identified easily. Figure 6 shows the process involved in KNN classifier.

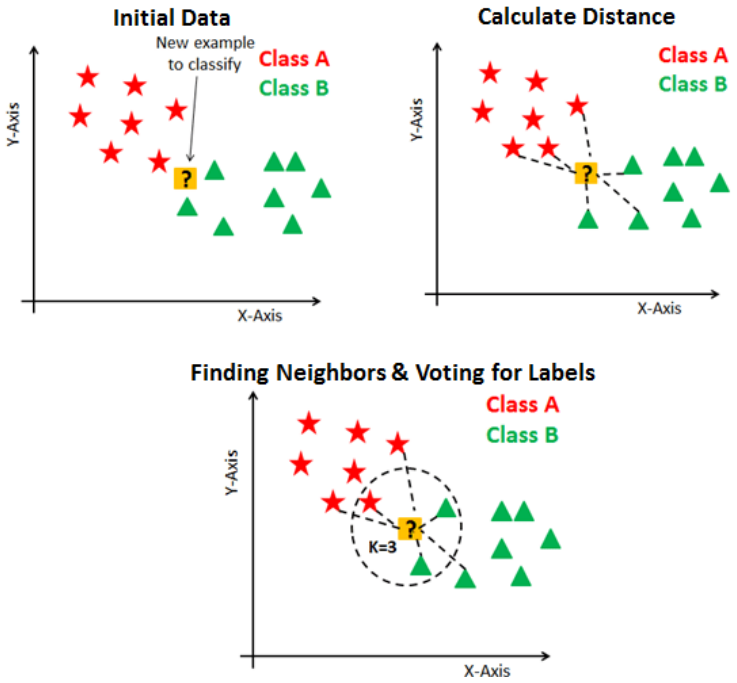

Figure 6: Process involved in KNN Classifier

The performance of KNN has been defined based on the following steps:

- Choose the value K of neighbors
- Compute the Euclidean distance of K number of neighbors
- Get the KNN according to the computed Euclidean distance.
- Between this k neighbors, calculate the number of the data points in all categories.
- Allocate a novel data point in which category of the neighbor is highest.

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547

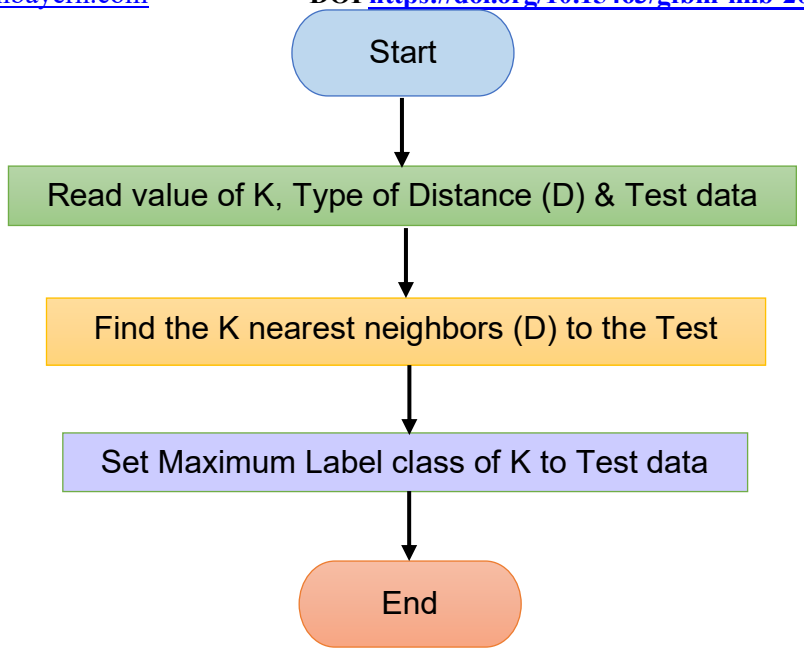

Figure 7: Flowchart of KNN model

VI EXPERIMENTAL RESULTS

6.1. Dataset

The dataset are collected from Instrument Recognition in Musical Audio Signals (IRMAS) online musical database. IRMAS consists of audio recordings of isolated musical instrument sounds recorded in realistic musical contexts. The dataset was created to support research on recognizing which musical instrument is present in an audio signal, particularly focusing on timbre-based classification. It includes recordings of single instruments playing different musical excerpts, making it suitable for monophonic instrument recognition tasks.

The dataset contains 12 instrument classes, covering both string, woodwind, brass, and keyboard instruments, such as cello, clarinet, flute, guitar, organ, piano, saxophone, trumpet, trombone, violin, and voice. Each audio file is labeled with its corresponding instrument, enabling supervised machine learning and deep learning experiments. The recordings vary in pitch, dynamics, and playing style, which helps models learn robust and generalizable audio features.

Total 1200 musical audio clips were took for this work such as Banjo, Bass Clarinet, Bassoon, Cello, Clarinet, Flute, Frenchhorn, Guitar, Trombone, Trumpet, Tuba and Violin, each musical instrument has 100 clips respectively. Each clip consists of music data ranging from 1 second to 5 seconds duration and the music is sampled at 16 kHz and encoded by 16-bit. 1000 musical data samples were used for training and 200 for testing. The music clips are preprocessed via pre-emphasis, segmentation and windowing for further implementation.

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547

6.2. Acoustic Feature Extraction

The training data are segmented into fixed-length and overlapping frames (in this experiment 20 ms frames with 10 ms overlapping). When neighboring frames are overlapped, the temporal characteristics of music content can be taken into consideration in the training process. Since 16 kHz sampling rate is deployed, 20 ms frames consists of 160 values. These 160 values are converted into 22 Sonogram coefficients which are for one frame. So, there are 100 such frames for 1 second music data. Experiments are conducted for the acoustic feature Sonogram and the performance of SVM and KNN is studied.

6.3. Performance Measures

In ML the major challenging factor is concerned to be statistical classification, a confusion matrix, named as error matrix, and it is a Table 1 layout which activates visualization of a method, and generally a supervised learning. Every row of matrix shows the instances in detected class whereas a column depicts the samples in actual class. The name is evolved from the fact which makes a simple process when a system is confusing 2 classes. It is a specific type of contingency table, with 2D ("actual" and "predicted"), and similar sets of "classes" in these integrations of dimension and class is defined as a variable in contingency table. The confusion matrix displays the ways for classifying the confused state while making predictions. It is composed of 2 feasible predicted classes: "Positive" and "Negative".

Table 1 Confusion Matrix

Experts	Machine Learning Techniques	
	Predicted Positive	Predicted Negative
True Positive	TP	FP
True Negative	FN	TN

Recall: It determines the ratios of positive samples are classified accurately.

$$\text{Recall} = \frac{\sum \text{TP}}{\sum \text{TP} + \sum \text{FN}} \qquad (29)$$

Accuracy: It determines the ratio of exactly classified samples as positives and negatives over the total samples.

$$\text{Accuracy} = \frac{\sum \text{TP} + \sum \text{TN}}{\sum \text{TP} + \sum \text{TN} + \sum \text{FP} + \sum \text{FN}} \qquad (30)$$

Precision: It estimates the count of TPs classified by the number of TPs and the count of FPs

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547

$$Precision = \frac{\sum TP}{\sum TP + \sum FP} \quad (31)$$

6.4. Evaluation using SVM

A linear support vector classifier is used to discriminate the various categories. The N class classification problem can be solved using N SVMs. Each SVM separates a single class from all the remaining classes (one-vs-rest approach). SVM is trained to distinguish acoustic features of a category from all other categories. Twelve SVMs are created for each acoustic feature for each category.

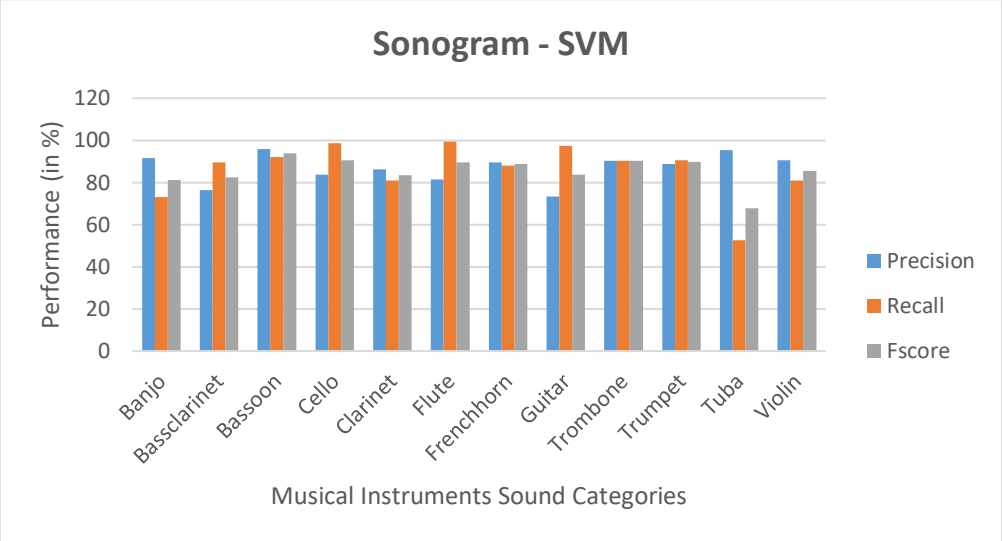

Figure 8: Classification results of SVM with Sonogram

For training, 1000 feature vectors are extracted for 1 second duration each. Hence, this results in 1000 feature vectors each of 22 dimensions. The training process analyzes audio training data to find an optimal way to classify audio frames into their respective classes. The derived support vectors are used to classify music data.

For testing, 200 acoustic feature vectors (1 sec of a music file) are given as input to SVM model and the distance between each of the feature vectors and the SVM hyperplane is obtained. The average distance is calculated for each model. The average distance gives better performance than using distance for each feature vector. The category of the audio is decided based on the maximum distance. The same process is repeated for different features. The classification results for the different features are shown in Figure 8.

6.5. Evaluation using KNN

The parameter K is assigned to indicate the number of nearest neighbors, the distance between the query-instance and all the training samples are calculated by Euclidean distance method. Here the value of K is 5. The distance is sorted for all the training data and the nearest neighbor

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com DOI https://doi.org/10.15463/gfbm-mib-2026-547

found based on the K^{th} minimum distance. All the categories of the training data are received for the sorted value which falls in K. Then the majority of the nearest neighbors used as the prediction value. KNN is trained to identify MISC features. For each value of k, the test has iterated k times with diverse training and testing sets. For training 1000 feature vectors, each of 22 dimensional Sonogram are extracted from the musical audio signal.

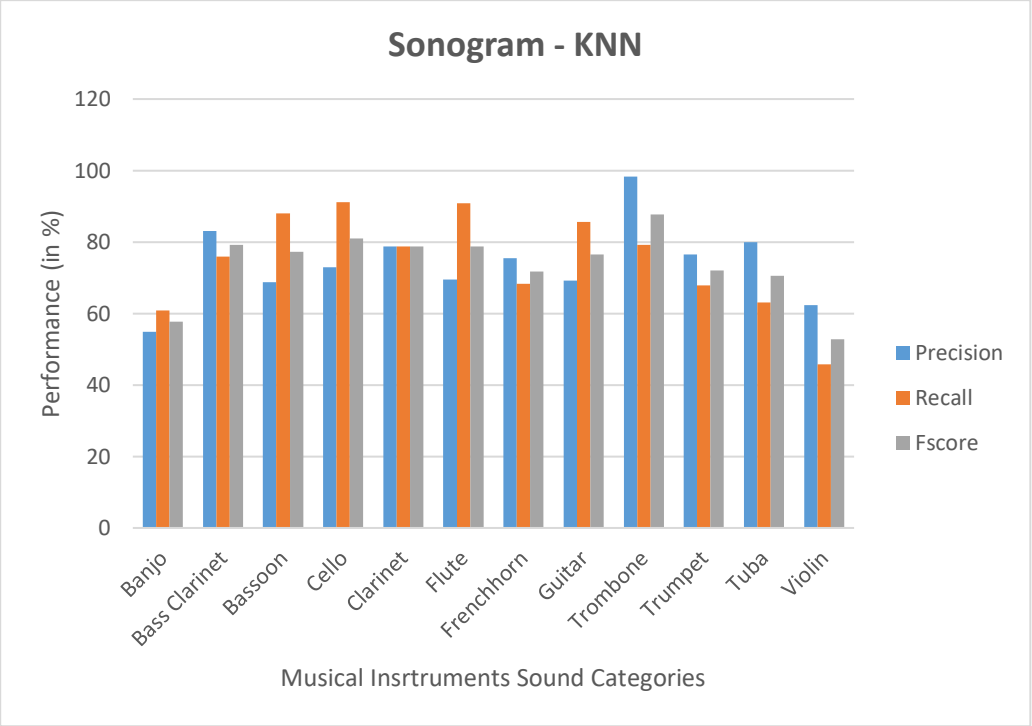

Figure 9: Classification results of KNN using Sonogram

For testing the 200 feature vectors, each of 22 dimensions are given as input to the KNN model and the distance between each of the feature vector and the majority nearest neighbor found among the data in voting procedure. The average distance is calculated for each class. The average distance gives a better performance than using distance for each feature vector. Figure 9 shows the classification performance of musical instruments sound classification using KNN.

6.6. Comparative Results Analysis

The overall performance of the Sonogram with the machine learning models SVM and KNN classifiers are analyzed in this work. Sonogram features with SVM gives the highest accuracy of 97.98 % when compared to the classifier KNN. The performance of SVM and KNN with Sonogram features and the comparison results are shown in Figure 10 and Table. 6.

Musik in bayern
ISSN: 0937-583x Volume 91, Issue 4 (April -2026)
https://musikinbayern.com **DOI https://doi.org/10.15463/gfbm-mib-2026-547**

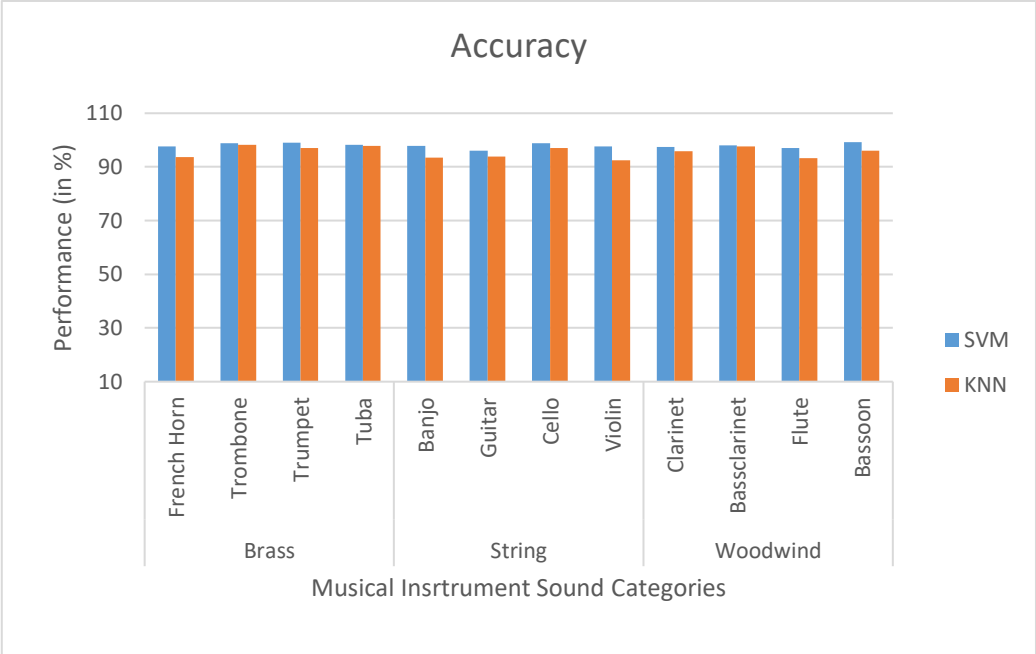

Figure 10: Comparison of SVM and KNN with Sonogram

Table 2 Overall Comparison of SVM and KNN with Sonogram

Features	Classifiers	Accuracy (in %)
Sonogram	SVM	97.98
	KNN	95.54

VII CONCLUSION

In this paper, the proposed method classifies the sound of the musical instruments. 22 dimensional Sonogram features are used for classification of musical instruments sound. Then, the machine learning classifiers SVM and KNN are applied to identify the class label of the musical instrument for the corresponding audio signal. The proposed work signified that the Sonogram-SVM and Sonogram-KNN models have achieved the accuracy values of 97.98% and 95.54% respectively.

VIII REFERENCES

[1] Ian McLoughlin. (2009). Applied Speech and Audio Processing: With MATLAB *Examples, Cambridge University Press.*

[2] Aucouturier J and Pachet F. (2002). Representing Musical Genre: A State of Art, *Journal of New Music Research,* 83-93.

[3] Tzanetakis, G. (2011). Audio feature extraction. *Music Data Mining*, pp.44-69.

[4] Shelar, V.S. and Bhalke, D.G. (2013). Musical Instrument Recognition and Transcription using Neural Network. In *Proceedings on Emerging Trends in Electronics and Telecommunication Engineering (NCET 2013)*. 31-36.

[5] Mazarakis, G., Tzevelekos, P. and Kouroupetroglou, G. (2006). Musical Instrument Recognition and Classification using Time Encoded Signal Processing and Fast Artificial Neural Networks. *In Hellenic Conference on Artificial Intelligence*, 246-255.

[6] P. Aurchana and P. Dhanalakshmi, "SVM Based Classification of Epithelial Dysplasia Using SURF and SIFT Features", *International Journal of Pure and Applied Mathematics,* Volume 117, No15,pp 1163-1175, 2017.

[7] S. Prabavathy, V. Rathikarani, P. Dhanalakshmi, (2019) An Enhanced Musical Instrument Classification using Deep Convolutional Neural Network, *International Journal of Recent Technology and Engineering (IJRTE)*, Volume-8 Issue-4, November 2019.

[8] Chang, Y.W., Hsieh, C.J., Chang, K.W., Ringgaard, M. and Lin, C.J. (2010), Training and testing low-degree polynomial data mappings via linear SVM. *Journal of Machine Learning Research*, Vol. 11, 1471-1490.

[9] Hsu, C. W., and Lin, C. J. (2002). A Comparison of Methods for Multiclass Support Vector Machines. *IEEE transactions on Neural Networks*, Vol. *13*(2), 415-425.

[10] Thiruvengatanadhan, R. (2016). Speech/Music Change Point Detection using Sonogram and AANN. *International Journal of Information & Computation Technology. Volume 6, Number 1,* 45-49.

[11] Ridoen, J.A., Sarno, R., Sunaryo, D. and Wijaya, D.R. (2017). Music Mood Classification using Audio Power and Audio Harmonicity Based on MPEG-7 Audio Features and Support Vector Machine. *In 2017 3rd International Conference on Science in Information Technology (ICSITech),* 72-76.

[12] Toman, S.H., Sahib, M.G.A. and Toman, Z.H. (2017). Content-Based Audio Retrieval by using Elitism GA-KNN Approach. *Journal of AL-Qadisiyah for Computer Science and Mathematics, 9(1),* 153-168.

[13] Chandwadkar, D. M., & Sutaone, M. S. (2012). Role of features and classifiers on accuracy of identification of musical instruments. *2nd National Conference on Computational Intelligence and Signal Processing (CISP)*.

[14] Anuz, H., Masum, A.K.M., Abujar, S., Hossain, S.A. (2021). Musical Instrument Classification Based on Machine Learning Algorithm. In: Tavares, J.M.R.S.,

Chakrabarti, S., Bhattacharya, A., Ghatak, S. (eds) Emerging Technologies in Data Mining and Information Security. *Lecture Notes in Networks and Systems*, vol 164.

[15] Chakraborty, S.S., Parekh, R. (2018). Improved Musical Instrument Classification Using Cepstral Coefficients and Neural Networks. In: Mandal, J., Mukhopadhyay, S., Dutta, P., Dasgupta, K. (eds) *Methodologies and Application Issues of Contemporary Computing Framework.*

[16] N. Karunakaran and A. Arya, (2018), "A Scalable Hybrid Classifier for Music Genre Classification using Machine Learning Concepts and Spark," *International Conference on Intelligent Autonomous Systems (ICoIAS)*, pp. 128-135, doi: 10.1109/ICoIAS.2018.8494161.

[17] E. Chaudary, S. Aziz, M. U. Khan and P. Gretschmann, "Music Genre Classification using Support Vector Machine and Empirical Mode Decomposition," 2021 *Mohammad Ali Jinnah University International Conference on Computing (MAJICC),* 2021, pp. 1-5, doi: 10.1109/MAJICC53071.2021.9526251.

[18] Tzanetakis, G., 2011. Audio feature extraction. *Music data mining*, pp.44-69.

[19] S. Vashishtha, R. Narula and P. Chaudhary, (2024), "Classification of Musical Instruments' Sound using kNN and CNN," *11th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, 2024, pp. 1196-1200

[20] Mengmeng Chen, Diying Tang, Yu Xiang, Lei Shi, Turker Tuncer, Fatih Ozyurt, Sengul Dogan (2025), Instrument sound classification using a music-based feature extraction model inspired by Mozart's Turkish March pattern, *Alexandria Engineering Journal*, Volume 118, 354-370, ISSN 1110-0168.

[21] Bhagyalakshmi, R., & Anandaraju, M. B. (2025). Identification of specific Musical instruments using Machine Learning models. *Journal of Integrated Science and Technology*, 13(5), 1108.

[22] Borovčak, K., & Bagić Babac, M. (2025). Instrument Classification in Musical Audio Signals using Deep Learning. *Croatian Regional Development Journal*, 6(1), 84–99. DOI:10.2478/crdj-2025-0006

[23] A Deep-Learning Framework with Multi-Feature Fusion and Attention Mechanism for Classification of Chinese Traditional Instruments. (2025). *Electronics*, 14(14), 2805.