Computers and Electrical Engineering 100 (2022) 107978

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

An intelligent music genre analysis using feature extraction and classification using deep learning techniques

Wang Hongdan [a,*], Siti SalmiJamali [b], Chen Zhengping [c], Shan Qiaojuan [a], Ren Le [d]

[a] College of Art & Sciences, Universiti Utara Malaysia, Kedah DarulAman, Malaysia
[b] School of Creative Industry Management and Performing Arts-College of Arts and Sciences, Universiti Utara Malaysia, 06010, Sintok, Kedah, Malaysia
[c] Department of Music and Dance, College of Chinese&Asean Arts, Chengdu University, SiChuan, China
[d] Information system management engineer, Zigong Fourth People's Hospital,SiChuan,China

ARTICLE INFO

Keywords:
Music files
MIR
Music genre
Accuracy
Classification
Deep learning
BiLSTM
VGG-16 Net

ABSTRACT

Music genre designations are useful for grouping songs, albums, and performers with comparable musical characteristics into larger categories. The goal of our study and research is to develop a deep learning method that can predict and classify song genres better than existing algorithms. Here the dataset of music genre information has been collected and processed for predicting genre of songs. We present a new approach including feature extraction and classification that takes into account the disparities in spectrums. The dataset namely MSD-I dataset, GTZAN Dataset and ISMIR2004 Genre dataset are utilized for feature extracted using BiLSTM and classification of extracted features has been done using VGG-16 Net. The effect of proposed approach is then evaluated in experiments on single and multi-label genre classification. The results are obtained based on the parameters of accuracy of 97%, precision of 94%, recall of 86.5%, F-1 score of 77.8%, average loss of audio signal of 40% for proposed technique.

1. Introduction

Music genres are a collection of keywords and descriptions that provide high-level information about a particular piece of music. Music genre categorization uses auditory signals to identify and forecast music type. The ability to automate the task of recognising musical tags allows for the generation of engaging content for both users and content providers, such as music discovery and playlist creation [1]. The genre, which is determined by specific aspects of music such as rhythmic structure harmonic content andinstrumentation [2], is one technique to categorise and organise songs. For audio streaming services like Spotify and iTunes, being able to automatically classify and tag songs in a user's library based on genre would be advantageous. This research looks at how machine learning (ML) techniques can be used to identify as well as classify genre of an audio recording. First model presented in this paper employs CNN that are trained end-to-end on audio signal's MEL spectrogram. In second phase of research, we extract features from the audio signal in both the time domain and the frequency domain. These characteristics are then incorporated into traditional machine learning models such as LR, RF, GB and SVM which are trained to categorise the audio file [3].

Music classification is used by businesses to make suggestions to their clients or simply as a product. Identifying music genres is the initial step in performing either of the following two roles. We can use Machine Learning techniques to assist us do this. These machine

* Corresponding author.
E-mail address: wang_hongdan@ahsgs.uum.edu.my (W. Hongdan).

https://doi.org/10.1016/j.compeleceng.2022.107978
Received 14 December 2021; Received in revised form 16 March 2022; Accepted 31 March 2022
Available online 8 April 2022

learning techniques also come in help when it comes to music analysis. It is based on a song's digital signatures for acoustics, tempo, danceability,energy and other elements to find types of music that a person is interested in listening to. Music is differentiated by categorised classifications known as genres. Humans are the ones who invent these genres [4]. A music genre is defined by the features that its members have in common. These features are usually linked to the music's rhythmic structure, instrumentation, and harmonic content. In the subject of MIR, which deals with viewing, organising, and finding vast music collections, categorising music files into their proper genres is a difficult issue. Genre classification may be quite useful in explaining several intriguing challenges, such as developing song references, hunting down related songs, identifying societies that will enjoy that particular music, and it can even be utilised for survey reasons. AMGC can help or even replace people in this process, making it a particularly useful addition to music information retrieval systems [5]. Furthermore, automatic genre classification of music provide a foundation for the generation and estimation of features for any type of content-based musical signal analysis. Because of rapid growth of the digital entertainment sector, the concept of automatic music genre classification has been highly popular in recent years. Although dividing music into genres is arbitrary, there are perceptual characteristics relating to instrumentation, rhythmic structure, and texture of the music that can help to define a genre. Genre classification for digitally downloadable music has been done manually up until now. Automatic genre classification algorithms has useful contribution to development of AIR systems for music [6]. In the existing system of music genre classification, classification accuracy is highly influenced by features selected for classification. In certain classification system, occurrence of training error may influence the performance.

The contributions of this paper are as follows:

- To collect data of music genre based on spectrogram audio for retrieving the feature vector of the audio features
- The signal has been processed and their training and testing is carried out using feature extraction and classification by deep learning architecture
- The feature vector extraction for both training and testing signal is carried out using BiLSTM and classification using VGG-16Net.
- The experimental results shows analysis of various dataset in terms of accuracy, precision, recall, F-1 score, average loss of audio signal

The remainder of the article is organized as follows: the related work is given in Section 2, the proposed approach is given in Section 3, performance investigation is given in Section 4 and the article is concluded in Section 5.

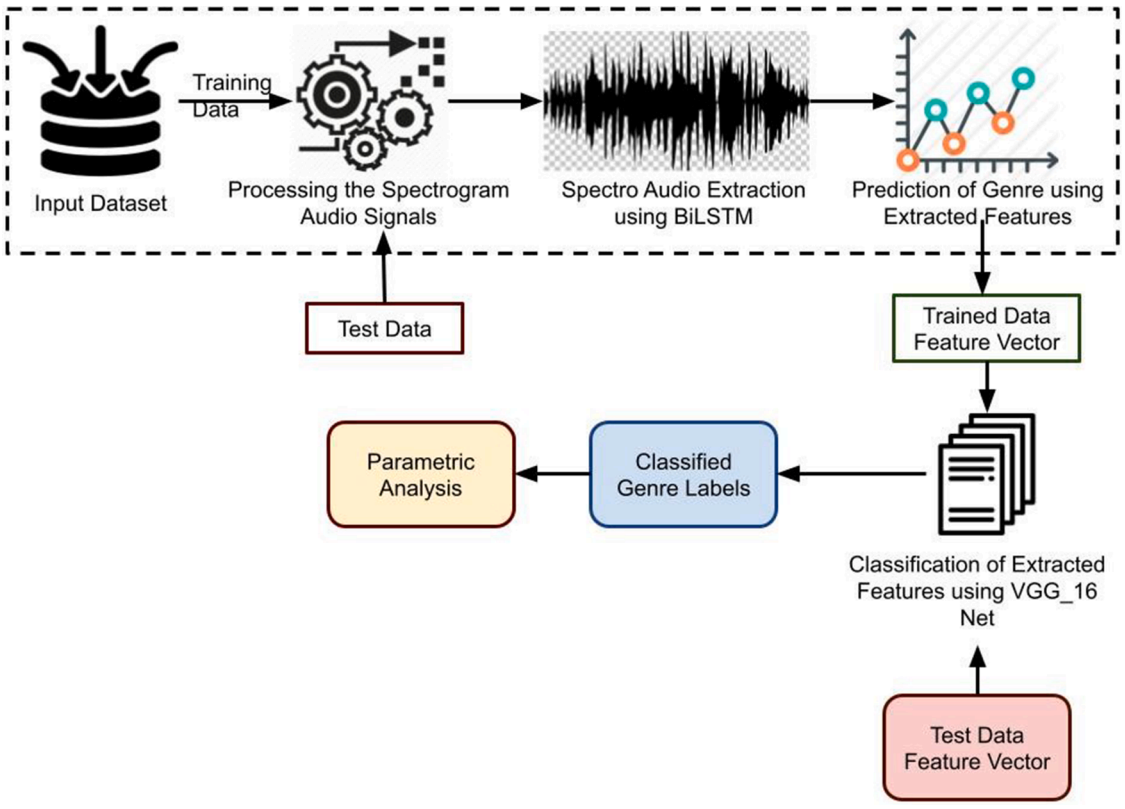


Fig. 1. Overall architecture of proposed design.

2. Related works

The majority of published music genre classification methods rely on auditory sources (for a comprehensive overview of the subject, see). Handcrafted audio characteristics such as MFCCs, are commonly used as input to a machine learning classifier in traditional procedures [7]. Recent deep learning algorithms make use of spectrograms, which are visual representations of audio signals. These visual representations of audio are fed into CNNs, which are trained in the same way that image classification is done. For this task, text-based techniques have also been considered. For example [8], describes one of the early attempts at categorization of music reviews, including studies on multi-class genre classification and star rating prediction. In a similar vein [9], extends these tests with a unique method for predicting music usages via agglomerative clustering, concluding that bigram features are more informative than unigram features. Furthermore [10], uses POS tags in conjunction with pattern mining algorithms to derive descriptive patterns for separating unfavourable from positive evaluations.[11] uses additional textual evidence to train a kNN classifier for predicting song subjects, taking into account lyrics and words pertaining to the meaning of the song. Album reviews are semantically enriched and categorised into 13 genre classifications using an SVM classifier in the work of [12]. There are only a few studies [13] that deal with music genre classification based on images. The majority of multimodal techniques in the literature mix audio and song lyrics [14]. Other modalities have been investigated, such as audio and video [15]. For music classification, the writers of [16] blend cultural, symbolic, and auditory elements. In other disciplines, multi-label classification is a well-studied subject [17]. Tag categorization from audio are examined from a multilabel perspective in the context of MIR, employing classic machine learning algorithms [18–20] and, more recently, deep learning approaches [21]. However, there are just a few methods for multilabel music genre categorization [22], and none of them are based on representation learning or multimodal data.

3. System model

The audio clip is the only input in the proposed system, and it is processed and features retrieved from it. The retrieved features are then sent into the RNN model's LSTM layer, which produces a trained model for music genre prediction. Initially music dataset is taken and feature vectors are generated using BiLSTM. The generated vectors are passed to the VGG16 NET classification framework that classifies diverse genre of music. The user can then interact with the trained algorithm in order to guess the genre. Fig. 1 depicts the overall architecture.

The above figure shows overall proposed architecture for music genre classification and its label prediction. The input dataset has been trained and tested for extracting their feature vectors. The signal has been processed based on the spectrograms of the audio signals. Then this spectro audio has been extracted using BiLSTM and predict the genre label through the extraction of the features. The output of this feature extraction will be trained and tested data feature vector. These feature vector has been compared and classified using VGG-16 Net based classification for genre label classification. The classification output will be classified genre labels.

3.1. BiLSTM based feature extraction

The Bi-LSTM has access to both the preceding and subsequent contextual information, and the information obtained by Bi-LSTM can be regarded as two different textual representations. at each time step t, given the word vector wt, the previous hidden state ht−1 and the previous cell state ct−1, the current state can be calculated as Eqs. (1–6):

$$i_f = \sigma(W_i w_i + U_i h_{t-1} + b_i) \tag{1}$$

$$f_t = \sigma\left(W_f w_t + U_f h_{t-1} + b_f\right) \tag{2}$$

$$o_t = \sigma\left(W_\rho w_t + U_0 h_{f-1} + b_o\right) \tag{3}$$

$$\widetilde{c}_t = \tanh(W_c w_t + U_\epsilon h_{t-1} + b_c) \tag{4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot c_t \tag{5}$$

$$P(w_1, \ldots, w_T) = \Pi_{i=1}^{i-T} P\left(w_j \middle| w_1, \ldots, w_{i-1}\right) \approx \Pi_{i=1}^{i=T} P\left(w_i \middle| w_{i-(n-1)\ldots+q_i w_{i-1})}\right) \tag{6}$$

Based on the output ht-1 of the BiLSTM unit at the previous moment and the output xt of the current time, the input gate decides the information to be reserved and discarded at time t, is given by Eqs. (7–10):

$$i_t = \sigma(W_i \cdot [h_{t-1} \cdot x_t] + b_i) \tag{7}$$

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{8}$$

$$p(w_2|w_1) = \frac{\text{count}(w_1, w_2)}{\text{count}(w_1)} \tag{9}$$

$$p(w_3|w_1,w_2)=\frac{\text{count}(w_1,w_2,w_3)}{\text{count}(w_1,w_2)} \tag{10}$$

Previous n input word vectors as shown in Eqs. (11–13).

$$\widetilde{C}_t=\tanh(W_c\cdot[h_{t-1}\cdot x_t]+b_c) \tag{11}$$

$$y=\text{softmax}\left(W^{(2)}\tanh\left(W^{(1)}x+b^{(1)}\right)+W^{(3)}x+b^{(3)}\right) \tag{12}$$

$$h_t=Wf(h_{t-1})+W^{(hx)}x_t \tag{13}$$

This function as sum over entire vocabulary at time-step t as shown in Eq. (14).

$$J^{(t)}(\theta)=-\Sigma_{j=1}^{j=|V|}y,t\times\log\left(\widehat{y}_{t,j}\right) \tag{14}$$

The cross entropy error over a corpus of size T is given by Eqs. (15–16):

$$J=-\frac{1}{T}\Sigma_{t=1}^{T}J^{(t)}(\theta)=-\frac{1}{T}\Sigma^{j=|V|}_{t=1}y_{t,t}\times\log\left(\widehat{y}_{t,j}\right) \tag{15}$$

$$\frac{\partial E}{\partial W}=\Sigma_{t=1}^{T}\frac{\partial E_t}{\partial W} \tag{16}$$

Applying chain rule differentiation to Eqs. (17) and (18) yields the error for each time step. The relevant differentiation is shown in Eq. (19). The partial derivative of ht with regard to all preceding k time-steps is denoted by dht/dhk.

$$\frac{\partial E_t}{\partial W}=\Sigma_{k=1}^{t}\frac{\partial E_t}{\partial y_t}\frac{\partial y_t}{\partial h_t}\frac{\partial h_t}{\partial h_k}\frac{\partial h_k}{\partial W} \tag{17}$$

$$\frac{\partial h_t}{\partial h_k}=\Pi_{j=k+1}^{t}\frac{\partial h_j}{\partial h_{j-1}}=\Pi_{j=k+1}^{t}W^T\times\text{diag}\left[f'\left(j_{j-1}\right)\right] \tag{18}$$

Becauseis each $\frac{\partial h_j}{\partial h_{j-1}}$ the Jacobian matrix for h:

$$\frac{\partial h_j}{\partial h_{j-1}}=\left[\frac{\partial h_j}{\partial h_{j-1,1}}\cdots\frac{\partial h_j}{\partial h_{j-1,D_n}}\right]=\begin{bmatrix}\frac{\partial h_{j,1}}{\partial h_{j-1,1}} & \cdot & \cdot & \frac{\partial h_{j,1}}{\partial h_{j-1,D_n}}\\ \cdot & \cdot & \cdot & \cdot\\ \cdot & & \cdot & \cdot\\ \frac{\partial h_{j,D_n}}{\partial h_{j-1,1}} & \cdot & \cdot & \frac{\partial h_{j,D_n}}{\partial h_{j-1,D_n}}\end{bmatrix} \tag{19}$$

Putting Eq. (20) together, we have the following relationship.

$$\frac{\partial E}{\partial W}=\Sigma_{t=1}^{T}\Sigma_{k=1}^{t}\frac{\partial E_t}{\partial y_t}\frac{\partial y_t}{\partial h_t}\left(\Pi_{j=k+1}^{t}\right)\frac{\partial h_j}{\partial h_{j-1}}\frac{\partial h_k}{\partial W} \tag{20}$$

Norm of partial gradient at every time-step, t is estimated through relationship shown in Eqs. (21–22).

$$\frac{\partial h_j}{\partial h_{j-1}}\le W^T\text{diag}\left[f'\left(h_{j-1}\right)\right]\le\beta_w\beta_k \tag{21}$$

$$\begin{aligned}&\|\frac{\partial h_t}{\partial h_k}\|=\|\Pi_{j=k+1}^{t}\frac{\partial h_j}{\partial h_{j-1}}\|\le\left(\beta_w\beta_{h_h}\right)^{t-k}\\ &\overrightarrow{h}_t=f\left(\overrightarrow{W}_{x_t}+\overrightarrow{V}\overrightarrow{h}_{t-1}+\overrightarrow{b}\right)\\ &\overleftarrow{h}_t=f\left(\overline{W}x_t+\overleftarrow{V}\overleftarrow{h}_{t+1}+\overline{b}\right)\\ &y_t=g(Uh_t+c)=g\left(U\left[\overrightarrow{h}_t;\overleftarrow{h}_t\right]+c\right)\end{aligned} \tag{22}$$

The output, ŷ, at each time-step is result of propagating input parameters through all hidden layers as given by Eq. (23)

$$\begin{aligned}
\overrightarrow{h}_t^{(i)} &= f\left(\overrightarrow{W}^{(i)} h_t^{(i-1)} + \overrightarrow{V}^{(\rho)} \overrightarrow{h}_{t-1}^{(i)} + \overrightarrow{b}^{(i)}\right) \\
\overleftarrow{h}_i^{(i)} &= f\left(\overleftarrow{W}^{(i)} h_i^{(i-1)} + \overleftarrow{V}^{(i)} \overleftarrow{h}_{i+1}^{(i)} + \overleftarrow{b}^{(i)}\right) \\
\widehat{g}_1 &= s(Uh_1 + c) = g\left(U\left[\overrightarrow{h}_i(t); \overleftarrow{h}_t^{(L)}\right] + c\right) \\
h_t &= \phi(h_{t-1}, x_t) = f\left(W^{(hh)} h_{t-1} + W^{(hx)} x_t\right) \\
h_t &= \phi(h_{t-1}) = f\left(\mathrm{W}^{(ht)} h_{t-1}\right) \\
y_t &= \mathrm{softmax}\left(W^{(S)} h_t\right)
\end{aligned} \tag{23}$$

The current time memory unit's state Ct is decided by the input gate and the forgetting gate, as well as the current candidate unit Ct and the self-state Ct-1 is given by Eqs. (24–26)

$$C_t = f_t * C_{t-1} + i_t * C_t \tag{24}$$

$$\max_\theta \frac{1}{N} \Sigma_{n=1} \times \log\left(p_\theta\left(y^{(\theta)} \middle| x^{(n)}\right)\right) \tag{25}$$

$$h_t = \phi(h_{t-1}, c, y_{t-1}) \tag{26}$$

Let's look at how a GRU uses h (t1) and x (t) to construct the next hidden state h (t) mathematically. After that, we'll delve into the architecture's intuition as shown by Eq. (27).

$$\begin{aligned}
t^{(t)} &= \sigma\left(W^{(i)} x^{(t)} + U^{(i)} h^{(t-1)}\right) \\
f^{(t)} &= \sigma\left(W^{(f)} x^{(t)} + U^{(f) h^{(t-1)}}\right) \\
o^{(t)} &= \sigma\left(W^{(0)} x^{(t)} + U^{(0)} h^{(t-1)}\right) \\
c^{(t)} &= \tanh\left(W^{(c)} x^{(t)} + U^{(c)} h^{(t-1)}\right) \\
z^{(t)} &= \sigma\left(W^{(z)} x^{(t)} + U^{(z)} h^{(t-1)}\right) \\
L^{(t)} &= \sigma\left(W^{(t)} x^{(t)} + U^{(r)} h^{(t-1)}\right) \\
h^{(t)} &= \tanh\left(r^{(t)} \circ U h^{(t-1)} + W x^{(t)}\right) \\
c^{(t)} &= f^{(t)} \circ e^{(t-1)} + i^{(t)} \circ e^{(t)} \\
h^{(t)} &= \left(1 - z^{(t)}\right) \circ \widetilde{h}^{(t)} + z^{(t)} \circ h^{(t-1)} \\
h^{(t)} &= o^{(t)} \circ \tanh\left(c^{(t)}\right)
\end{aligned} \tag{27}$$

The output gate Ot can be expressed Eq. (28):

$$\begin{aligned}
O_t &= \sigma(W_o \cdot [h_{t-1} \cdot x_t] + b_o) \\
H_t &= O_t * \tanh(C_t) \\
\overrightarrow{h}_t &= \sigma\left(W_{\overrightarrow{hx}} x_t + W_{\overrightarrow{h}} \overrightarrow{h} \overrightarrow{h}_{t-1} + b_{\overrightarrow{h}}\right)
\end{aligned}$$

$$y_t = W_{y\rightarrow} \overrightarrow{h}_t + W_{yh} \overleftarrow{h}_t + b_y \tag{28}$$

Created a layered BiLSTM network in which the lower layer's output yt becomes the top layer's input as shown in Eq. (29)

$$h_t = W_{hh} \overrightarrow{h}_t + W_{ht} \overleftarrow{h}_t + b_{h-} \tag{29}$$

Use a stacked BiLSTM to extract the HQ and HA hidden state matrixes from the questions and replies as given in eq. (30)

$$\begin{aligned}
h_t^4 &= \mathrm{sBiLSTM}\left(h_{t-1}^q, h_{t+1}^4, q_t\right), h_0^4 = 0 \\
h_t^a &= \mathrm{sBiLSTM}\left(h_{t-1}^a, h_{t+1}^a, a_t\right), h_0^{\mathrm{l}} = h_{s'}^{\mathrm{A}} \\
H_{\mathrm{Q}} &= \left[h_1^{\phi}, h_2^{\mathrm{q}}, \ldots, h_n^{\mathrm{q}}\right] \in R^{d*n} \\
H_{\mathrm{A}} &= \left[h_1^{\mathrm{a}}, h_2^{\mathrm{a}}, \ldots, h_m^{\alpha}\right] \in \mathbb{R}^{d \cdot m} \\
A^{\mathrm{A}} &= \mathrm{softmax}\left(L^{\mathrm{T}}\right) \in R^{**m}
\end{aligned} \tag{30}$$

Here, CQ and CA are results of interaction between question and answer vector as Eq. (31):

$$
\begin{aligned}
&C^{Q} = H_{\mathrm{A}} A^{\mathrm{Q}} \in R^{d*n} \\
&C^{Q} = H_{\mathrm{A}} A^{\mathrm{Q}} \in R^{d*n} \\
&O_q = \max_{\mathrm{Octech}} C_t^{Q} \\
&M_{\Delta q}(t) = \tanh\left(W_{aw} C_i^{\lambda} + W_{qp} O_4\right) \\
&S_{aq}(t) \propto \exp\left(w_{mu}^{T} M_{aq}(t)\right) \\
&O_a = \Sigma_{r=1}^{m} C_t^{\lambda} S_{nq}(t) \\
&\mathrm{Score}_{\mathrm{conine}}(O_4, O_a) = \frac{\mathrm{O_a} \cdot \mathrm{O_a}}{|\mathrm{O_a}|\mathrm{O_a}|} \\
&\mathrm{Score}_{\mathrm{Yualidren}}(O_9, O_2) = \frac{1}{1|O_4 - O_a|_2}
\end{aligned}
\tag{31}
$$

Using the hinge loss function, the positive and negative samples can be entered concurrently during training. The training aim is defined as the hinge loss function as Eq. (32):

$$
L = \max\left\{0, M - \mathrm{Score}(O_8, O_{2+}) + S\mathrm{core}\left(O_q, O_{\Delta-}\right)\right\} + \lambda \parallel \theta 1 \tag{32}
$$

3.2. VGG-16 Net based classification

There are 4 types of layers: convolutional, max pooling, fully-connected layers and softmax function. Input for neural network is an image with a size of 224 × 224 × 3. The filters are 3 × 3 matrices and the stride of which is fixed to 1. The padding size is always 1, while max-pooling is carried out over a pixel window of size 2 × 2, with stride of 2.

Initially, Convolutional neural network is used where the input video frame images are passed through a series of layers i.e. pooling, convolutional,flattening and fully connected layers and then output of VGG_16 is generated which classify video frame images. After building VGG_16 Net models from the scratch, then the model is fine tuned by using image augmentation technique. Consequently, one of the pre-trained models VGG_16 Net is employed to classify image and check accuracy for training data and validation data.

For the group$G = \mathbb{R}^r$ the Fourier transform of a function $A \in L^1(\mathbb{R}^r)$ is bounded, continuous function Eq. (33)

$$
\widehat{a}(y) := \int_{\mathbb{R}^r} a(x) exp(-2\pi i x \cdot y) dx, y \in \mathbb{R}^r, \text{ where } x \cdot y := x_1 y_1 + \cdots + x_r y_r \tag{33}
$$

$\widehat{a}$is a common scalar product. If an is absolutely integrable, for example, under sufficient assumptions, the Fourier inversion formula Eq. (34)

$$
a(x) = \int_{\mathbb{R}^r} \widehat{a}(y) \exp(+2\pi i x \cdot y) dy \tag{34}
$$

holds almost everywhere. For fixed y map $x \mapsto \langle x, y\rangle := \exp(-2\pi i x \cdot y)$ is a character on $\mathbb{R}^r$, i.e., a continuous group homomorphism from $\mathbb{R}^r$into circle group $\mathrm{S}^1 := \{z \in \mathbb{C}; |z| = 1\}$let $Gr_{count}(\mathbb{R}^r, \mathrm{S}^1)$ denote multiplicative group of all characters with multiplication of functions given by Eq. (35).

$$
\mathbb{R}^r \cong \mathrm{Gr}_{\mathrm{cont}}\ \left(\mathbb{R}^r, \mathrm{S}^1\right), y \mapsto \langle -, y\rangle \tag{35}
$$

and the Fourier inversion has form shown in Eq. (36)- (39)

$$
\widehat{a}(y) := \int_{\mathbb{R}^r} a(x)\langle x, y\rangle dx \tag{36}
$$

$$
a(x) := \int_{\mathbb{R}^r} \widehat{a}(y)\langle -x, y\rangle dy, \langle -x, y\rangle = \langle x, y\rangle^{-1} = \langle x, y\rangle \tag{37}
$$

$$
\widehat{a}(\widehat{g}) := \int_{G} a(g) g, \widehat{g} dg, a \in \mathrm{L}^1(G), \tag{38}
$$

$$
a(g) := \int_{G} \widehat{a}(\widehat{g})\langle -g, \widehat{g}\rangle d\widehat{g}\langle g, \widehat{g}\rangle = \langle -g, \widehat{g}\rangle^{-1} = \langle -g, \widehat{g}\rangle \tag{39}
$$

where$d\widehat{g}$ are suitably normalized Haar measures on G given by Eqs. (40–41)

$$
: -G \times \widehat{G} \to \mathbb{Z}/\mathbb{Z}d \text{ suchthat}, \widehat{G} \cong \mathrm{Hom}(G, \mathbb{Z}\,/\,\mathbb{Z}d), \widehat{g} \mapsto (-) \cdot \widehat{g}, \text{and} G \cong \mathrm{Hom}(\widehat{G}, \mathbb{Z}\,/\,\mathbb{Z}d), g \mapsto g \cdot (-) \tag{40}
$$

$$
\mathbb{Z}/\mathbb{Z}d \cong \mu := \langle\zeta\rangle = \left\{1, \zeta, \cdots, \zeta^{d-1}\right\} \subseteq \mathrm{S}^1, \overline{k} \mapsto \zeta^{\overline{k}} := \zeta^{k} \tag{41}
$$

The nondegenerate form • thus induces nondegeneratebimultiplicative form Eq. (42)

$$\langle -,-\rangle : G\times\widehat{G}\to\mu,\langle g,\widehat{g}\rangle := \zeta^{g\cdot\widehat{g}}, \text{suchthat}\widehat{G}\cong \mathrm{Gr}(G,\mu),\widehat{g}\mapsto\langle -,\widehat{g}\rangle, \text{and}G\cong \mathrm{Gr}(\widehat{G},\mu), g\mapsto\langle g,-\rangle \tag{42}$$

The multiplicative abelian group μ of homomorphisms from additive abelian group G to multiplicative abelian group is denoted by Gr(G, μ). Canonical group isomorphisms by Eqs. (42–44)

$$\widehat{G}\cong (Hom)(G,\mathbb{Z}/\mathbb{Z}d)\cong \mathrm{Gr}(G,\mu)=\mathrm{Gr}(G,\mathrm{S}^1) \tag{42}$$

$$d>0, G:=\widehat{G}=\mathbb{Z}\Big/\mathbb{Z}d, \overline{k}\cdot\overline{l}=\overline{kl}, \langle\overline{k},\overline{l}\rangle=\exp\left(-2\pi i\frac{kl}{d}\right) \tag{43}$$

$$\mathrm{Four}_G:\mathbb{C}^G\mathbb{C}^{\widehat{G}}, a\mapsto\widehat{a},\widehat{a}(\widehat{g}):=\Sigma_{g\in G}a(g)\langle g,\widehat{g}\rangle, \text{and}\mathrm{Four}_{\widehat{G}}^{\wedge}:\mathbb{C}^{\widehat{G}}\to\mathbb{C}^G,\mapsto\widehat{b},\widehat{b}(g):=\Sigma_{\widehat{g}\in\widehat{G}}b(\widehat{g})\langle g,\widehat{g}\rangle \tag{44}$$

Fourier inversion formula has form Eq. (45)

$$[1]N^{-1}\widehat{\widehat{a}}(-g)=a(g),\ \text{where}\ a\in\mathbb{C}^G, N:=\mathrm{ord}(G) \tag{45}$$

Least common multiple as Eq. (46)

$$\exp(G):=\mathrm{lcm}(d_1,\cdots,d_r)\ \text{with}\ \mathbb{Z}\exp(G)=\{k\in\mathbb{Z};kG=0\} \tag{46}$$

If G and H are additively written abelian groups, Hom(G, H) = HomZ(G, H) is used to designate the group of all additive or Z-linear homomorphisms from G to H. If r > 0 and K is a field, map as shown Eqs. (47–49)

$$K^r\times K^r\to K, x\cdot y:=x_1y_1+\cdots+x_ry_r\ \text{for}\ x=(x_1,\cdots,x_r) \tag{47}$$

Let $g=(\overline{g_1},\cdots,\overline{g_r})=(\overline{g_1'},\cdots,\overline{g_r'})$;hence $g_\varrho'=g_e+k_\rho d_\varrho, k_e\in\mathbb{Z}$, for ϱ = 1, ⋯, *r*But then

$$\Sigma_{\varrho=1}^r g_\varrho' h_\varrho=\Sigma_{\varrho=1}^r g_\varrho h_\varrho\frac{d}{d_e}+\Sigma_{\varrho=1}^r g_\varrho h_\varrho k_e d\in\Sigma_{\varrho=1}^r g_\varrho h_\varrho\frac{d}{d_e}+Zd, \tag{48}$$

$$\Sigma_{\varrho=1}^r g_\varrho' h_\varrho\frac{d}{d_p}=\Sigma_{\varrho=1}^r g_\varrho h_\varrho\frac{d}{d_e}=g\cdot h \tag{49}$$

Assume that $(\ -\ )\bullet h=0$. For ϱ = 1, ⋯, *r* let $\delta_\varrho:=\left(0,\cdots,0,\frac{\varrho}{1},0,\cdots,0\right)$ denote analogue of standard basis such that $(\overline{g_1},\cdots,\overline{g_r})=\Sigma_{\varrho=1}^r g_\varrho\delta_\varrho$ for all $g\in G$. Then from Eqs. (50–51)

$$0=\delta_\varrho\cdot h=\overline{h_\varrho\frac{d}{d_e}}\in\mathbb{Z}\Big/\mathbb{Z}d; hencefor \tag{50}$$

$$d\Big|h_\varrho\frac{d}{d_e}\mathrm{ord}_\varrho\Big|h_\varrho\ \text{and}\overline{h_\varrho}=0\ \text{in}\mathbb{Z}\Big/\mathbb{Z}d_\varrho; i.e., h=0. \tag{51}$$

Let $\varphi: G\to\mathbb{Z}/\mathbb{Z}d$ be any homomorphism. Equation$d_\phi\delta_\varrho=0$, implies $d_\phi\phi(\delta_\varrho)=0$ in $\mathbb{Z}/\mathbb{Z}d$; hence $\varphi(\delta_\varrho)=\overline{h_\varrho\frac{d}{d_\varrho}}=\delta_\varrho\cdot h, h_\varrho\in\mathbb{Z}$ and for $g\in G:\varphi(g)=\varphi(\Sigma_{\varrho=1}^r g_\varrho\delta_e)=\Sigma_{\varrho=1}^r g_\rho\varphi(\delta_e)=\Sigma_{\varrho=1}^r g_\varrho\delta_{\underline{Q}}\cdot h=(\Sigma_{\varrho=1}^r g_\varrho\delta_\varrho)\cdot h=g\cdot h$ and $\phi=(\ -\ )\bullet h$

By taking use of the following, FTare speed up convolutions Eq. (52)

$$x[n]*y[n]=\Sigma_{k=-\infty}^{\infty}x[k]y[n-k]\ \leftrightarrow^{DTFT}X\left(e^{j\omega}\right)Y\left(e^{j\omega}\right) \tag{52}$$

Convolution of two signals is equal to multiplication of their FT, according to the equation above. A convolution becomes a single element-wise multiplication when the input is transformed into frequency space. In other words, the Fourier Transform can be used to convert input to frequencies, multiply it once, and then transform it back to frequencies using IFT. The overhead of translating inputs into Fourier domain and then using IFT to return replies to spatial domain comes at a cost.

However, the speed gain acquired by performing a single multiplication rather than multiplying the kernel with various sections of image is offset by speed gain received by executing a single multiplication instead of multiplying kernel with various sections of image. FT was divided into even and odd indexed sub-sequences by Eq. (53).

$$\begin{cases} n=2r & \text{if even} \\ n=2r+1 & \text{if } odd \end{cases}$$
$$\text{where}\ r=1,2,\ldots,\frac{N}{2}-1 \tag{53}$$

Summation of two terms after doing some algebra. The benefit of this method is that even and odd indexed sub-sequences are-estimated simultaneously as shown by Eq. (54).

$$
\begin{aligned}
x[k] &= \Sigma_{n=0}^{N-1} x[n] e^{\frac{-j2\pi kn}{N}} \\
x[k] &= \Sigma_{r=0}^{\frac{N}{2}-1} x[2r] e^{\frac{-j2\pi kn(2r)}{N}} + x[k] \Sigma_{r=0}^{\frac{N}{2}-1} x[2r+1] e^{\frac{-j2\pi kn(2r+1)}{N}} \\
x[k] &= \Sigma_{r=0}^{\frac{N}{2}-1} x[2r] e^{\frac{-j2\pi kn(2r)}{N}} + x[k] e^{\frac{-j2\pi k}{N}} \Sigma_{r=0}^{\frac{N}{2}-1} x[2r+1] e^{\frac{-j2\pi k(2r)}{N}} \\
x[k] &= \Sigma_{r=0}^{\frac{N}{2}-1} x[2r] e^{\frac{-j2\pi kn(r)}{N/2}} + x[k] e^{\frac{-j2\pi k}{N}} \Sigma_{r=0}^{\frac{N}{2}-1} x[2r+1] e^{\frac{-j2\pi k(r)}{N/2}} \\
x[k] &= x_{\text{even}}[k] + e^{\frac{-j2\pi k}{N}} x_{\text{odd}}[k]
\end{aligned}
\tag{54}
$$

Butterfly diagram to represent flow of data across time, assuming N = 8. DFT is computed for both even and odd terms at the same time. Then, using the method from above, we determine x[k]. Multiply latter by amount of time it took to evaluateDFT on half of original input. It takes N steps to add up FT for a specific k in the final step. For this by adding N to final product given by Eq. (55).

$$
\begin{aligned}
2DFT\downarrow_{\text{DFT on } N/2 \text{ elements}}\ 2x[k] &= x_{\text{even}}\ [k] + e^{\frac{-j2\pi k}{N}} x_{\text{odd}}\ [k] \\
&= 2 \times \frac{N^2}{4} + N \\
&= \frac{N^2}{2} + N \\
O\left(\frac{N^2}{2} + N\right) &\sim O\left(N^2\right)
\end{aligned}
\tag{55}
$$

The input of a VGG 16 is commonly an order 3 tensor, such as an image with H rows, W columns, and 3 channels. Higher order tensor inputs, on the other hand, can be handled in a similar way by VGG 16. After then, the input passes through a series of processing steps in order. A layer is a processing step that are convolution layer, a pooling layer, a normalising layer, a fully connected layerand so on.

Input x^1, make it pass processing of 1st layer, and get x^2 . In turn, x^2 is passed into 2nd layer. Achieve $x^L \in \mathbb{R}^c$, which evaluates x^lposterior probabilities belonging to C categories. CNN output prediction by Eq. (56)

$$
\arg\max_i x_i^L \tag{56}
$$

Already evaluated terms$\frac{\partial z}{\partial w^{i+1}}$ and $\frac{\partial z}{\partial x^{i+1}}$. Compute $\frac{\partial z}{\partial w^i}$ and $\frac{\partial z}{\partial x^i}$, using chain rule in Eq. (57)

$$
\begin{aligned}
\frac{\partial z}{\partial \left(\text{vec}\left(w^i\right)^T\right)} &= \frac{\partial z}{\partial \left(\text{vec}\left(x^{i+1}\right)^T\right)} \frac{\partial \text{vec}\left(x^{i+1}\right)}{\partial \left(\text{vec}\left(w^i\right)^T\right)}, \\
\frac{\partial z}{\partial \left(\text{vec}\left(x^i\right)^T\right)} &= \frac{\partial z}{\partial \left(\text{vec}\left(x^{i+1}\right)^T\right)} \frac{\partial \text{vec}\left(x^{i+1}\right)}{\partial \left(\text{vec}\left(x^i\right)^T\right)}
\end{aligned}
$$

$$
\begin{aligned}
&\forall n \in \left[1, 2, \ldots, n_C^{[l]}\right]: \\
&\text{conv}\left(a^{[l-1]}, K^{(n)}\right)_{x,y} = \psi^{[l]}\left(\Sigma_{i=1}^{n_H^{n-1}} \Sigma_{j=1}^{n_W^{l-1}} \Sigma_{i=1}^{n_C^{l-1}} K_{i,j,k}^{(n)} a_{x+i-1,y+j-1,k}^{[l-1]} + b_n^{[l]}\right) \\
&\dim\left(\text{conv}\left(a^{l-1}, K^{(n)}\right)\right) = \left(n_H^{[l]}, n_W^{[l]}\right)
\end{aligned}
\tag{57}
$$

Thus by Eq. (58):

$$
\begin{aligned}
a^{[l]} = \\
\left[\psi^{[l]}\left(\text{conv}\left(a^{[l-1]}, K^{(1)}\right)\right), \psi^{[l]}\left(\text{conv}\left(a^{[l-1]}, K^{(2)}\right)\right), \ldots, \psi^{[l]}\left(\text{conv}\left(a^{[l-1]}, K^{\left(n_C^{[l]}\right)}\right)\right)\right] \\
\dim\left(a^{[l]}\right) = \left(n_H^{[l]}, n_W^{[l]}, n_C^{[l]}\right)
\end{aligned}
\tag{58}
$$

With Eq. (59):

$$
\begin{aligned}
n_{H/W}^{[l]} &= \left[\frac{n_{H/W}^{(l-1)} + 2p'^l - f^{ll}}{s^{ll}} + 1\right]; s > 0 \\
&= n_{H/W}^{[l-1]} + 2p^{[l]} - f^{[l]}; s = 0 \\
n_C^{[l]} &= \text{ number of filters}
\end{aligned}
\tag{59}
$$

The learned parameters at the l^{th} layer are:

- Filters with $(f^{[l]} \times f^{[l]} \times n_C^{[l-1]}) \times n_C^{[l]}$ parameters

- Bias with $(1 \times 1 \times 1) \times n_C^{[l]}$ parameters (broadcasting)

3.2.1. Pooling layer

As previously stated, at convolutional layer, apply convolutional products to input, this time using several filters, by ψ activation function.

More preciously, at $\mathbf{l}^{\text{th}}$ layer:

- Input: $a^{[l-1]}$ with size $(n_H^{[l-1]}, n_W^{[l-1]}, n_C^{[l-1]})$, $a^{[0]}$ being image in input
- Padding: $p^{[l]}$, stride: $s^{[l]}$
- Number of filters: $n_C^{[l]}$ where every $K^{(n)}$ has dimension: $(f^{[l]}, f^{[l]}, n_C^{[l-1]})$
- Bias of n^{th} convolution: $b_n^{[l]}$
- $\psi^{[l]}$ is activation function
- Output: $a^{[l]}$ with size $(n_H^{[l]}, n_W^{[l]}, n_C^{[l]})$

$$\begin{aligned} a_{x,y,z}^{[l]} &= \text{pool}\left(a^{[l-1]}\right)_{x,y,z} = \phi^{[l]}\left(\left(a_{x+i-1,y+j-1,z}^{[l-1]}\right)_{(i,j)\in\left[1,2,\ldots,f^{[l]}\right]^2}\right) \\ \dim\left(a^{[l]}\right) &= \left(n_H^{[l]}, n_W^{[l_1)}, n_C^{[l]}\right) \end{aligned} \tag{60}$$

With by Eq. (61)

$$\begin{aligned} n_{H/W}^{[l]} &= \left\lfloor \frac{n_{H/W}^{[l-1]} + 2p^{[l]} - f^{[l]}}{s^{[l]}} + 1 \right\rfloor; s > 0 \\ &= n_{H/W}^{[l-1]} + 2p^{[l]} - f^{[l]}; s = 0 \\ n_C^{[l]} &= n_C^{[l-1]} \end{aligned} \tag{61}$$

The pooling layer has no parameters to learn.

3.2.2. Fully connected layer

It is made up of a limited number of neurons that receive a vector as input and output another vector. In general, consider j^{th} node of i^{th} layer indicated by following by Eq. (62):

$$\begin{aligned} z_j^{[i]} &= \Sigma_{l=1}^{n_{i-1}} w_{j,l}^{[i]} a_l^{[i-1]} + b_j^{[i]} \\ \rightarrow a_j^{[i]} &= \psi^{[i]}\left(z_j^{[i]}\right) \end{aligned} \tag{62}$$

To plug into fully connected layer tensor flatten to a 10 vector having dimension: $(n_H^{[i-1]} \times n_W^{[i-1]} \times n_C^{[i-1]}, 1)$, thus by Eq. (63):

$$n_{i-1} = n_H^{[i-1]} \times n_W^{[i-1]} \times n_C^{[i-1]} \tag{63}$$

Learned parameters at l^{th} layer are:

- Bias with n_l parameters
- Weights $w_{j,l}$ with $n_{l-1} \times n_l$ parameters

Loss function $L_i(\theta_i) = E_{(s,a,\ r,\ s')}[(y_i - Q(s,\ a;\ \theta_i))^2]$
With $y_i = r + \gamma \max_{a'} Q(s', a'; \theta_i^-)$

$$V(s_t) \leftarrow V(s_t) + \alpha[r_t + \gamma V(s_{t+1}) - V(s_t)] \tag{64}$$

where α is a learning rate. Similarly, VGG_16 learns action value function with update rule by Eq. (65)

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\left[r + \gamma \underset{a_{t+1}}{max} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\right] \tag{65}$$

If the same acts are taken and the same rewards are received the next time, the total future reward will diverge. As a result, the above prizes are replaced with a discounted future awardas shown in Eqs. (66–67):

$$R_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots + \gamma^{n-t} r_n \tag{66}$$

$$R_t = r_t + \gamma(r_{t+1} + \gamma(r_{t+2} + \cdots)) = r_t + \gamma R_{t+1} = \Sigma_{k=0}^{\infty}\gamma^k r_{t+k} \tag{67}$$

Without loss of generality, the goal of VGG_16 is to estimate an optimal policy π* which satisfies by Eq. (68)

$$J_{\pi^*} = \max_{\pi} J_{\pi} = \max_{\pi} E_{\pi}\left[\Sigma_{t=0}^{\infty}\gamma^t r\right] \tag{68}$$

The state value function for a stationary policy is defined as follows in this paper by Eq. (69)

$$V^{\pi}(s) = E_{\pi}\left|\Sigma_{t=0}^{\infty}\gamma^t r\right| s_0 = s] \tag{69}$$

As a result, the optimal value function is defined as follows by Eq. (70)

$$V^{\pi*}(s) = E_{\pi*}\left[\Sigma_{t=0}^{\infty}\gamma^t r_t\right| s_0 = s] \tag{70}$$

To facilitate policy enhancement, state-action value function$Q^{\pi}(s,a)$ is defined by Eq. (71),

$$Q^{\pi}(s,a) = E_{\pi}\left[\Sigma_{t=0}^{\infty}\gamma^t r_t\right| s_0 = s, a_0 = a] \tag{71}$$

and optimal state-action value function by Eq. (72)

$$Q^{\pi*}(s,a) = E_{\pi*}\left[\Sigma_{t=0}^{\infty}\gamma^t r_t\right| s_0 = s, a_0 = a] \tag{72}$$

According to Bellman optimality from Eqs. (73–75)

$$V^{\pi}(s) = \Sigma\pi(a|s)E[R_{t+1} + \gamma V(s_{t+1})|S_t = s] \tag{73}$$

$$V^{*}(s) = E[R_{t+1} + \gamma\max{}^{\pi}V(s_{t+1})|]S_t = s \tag{74}$$

$$Q^{*}(s,a) = E[R_{t+1} + \gamma\max_{a'}Q(s_{t+1},a')|S_t = s, A_t = a] \tag{75}$$

So, the optimal policy is computed by Eqs. (76–77)

$$\pi^{*}(s) = \mathrm{argmax}_a Q^{*}(s,a) \tag{76}$$

$$\widetilde{\pi}^{*}(s)\mathrm{argmax}_a \widetilde{Q}^{*}(s,a) \tag{77}$$

To estimate the action-valued function, it is typical to utilise a function approximate without any generalisation by Eq. (78)

$$Q(s,a;\theta) \approx Q^{*}(s,a) \tag{78}$$

When loss function is differentiated with respect to parameters, gradient is given by Eq. (79)

$$\nabla_{\theta_i}L(\theta_i) = E_{s,a,r,s'}\left[\left(r + \gamma\max_{a'}Q\right)\left(s',a',\theta_i^{-}\right) - Q(s,a;\theta_i)\nabla_{\theta_i}Q(s,a;\theta_i)\right] \tag{79}$$

Consider case where there are K anchor vectors in N-dimensional unit sphere, given by$\mathbf{a}_k \in R^N, k = 1, \cdots, K$. For given x, its K rectified correlations with ak, k " 1, ···, K, defines a nonlinear transformation from x to an output vector given by Eqs. (80–82).

$$\mathbf{y} = (y_1, \cdots, y_k, \cdots, y_K)^T \tag{80}$$

$$y_k(\mathbf{x}, \mathbf{a}_k) = m\left(0, \mathbf{a}_k^T\mathbf{x}\right) \equiv \mathrm{Rec}\left(\mathbf{a}_k^T\mathbf{x}\right) \tag{81}$$

$$\Phi_W(f) := \bigcup_{n=0}^{\infty}\Phi_W^n(f) \tag{82}$$

where $\Phi_W^0(f) := \{f * \psi_{(-J,0)}\}$, and given by Eq. (83)

$$\Phi_W^n(f) := \left\{\left(U\underbrace{\left[\lambda^{(j)},\ldots,\lambda^{(p)}\right]}_{n\ \text{indices}}f\right) * \psi_{(-J,0)}\right\}_{\lambda^{(j)},\ldots,\lambda^{(p)}\in\Lambda_w\setminus\{(-J,0)\}} \tag{83}$$

for all $n \in \mathbb{N}$, with Eq. (84)

$$U\left[\lambda^{(j)},\ldots,\lambda^{(p)}\right]f := \underbrace{\left|\cdots\left|f * \psi_{\lambda}^{(j)}\right| * \psi_{\lambda^{(k)}}\right|\cdots * \psi_{\lambda^{(p)}}\Big|}_{n-\ \text{fold conrolution followed by modulus}}$$
$$\Psi_{\Lambda_w} := \{T_b I\psi_{\lambda}\}_{b\in\mathbb{R}^d,\lambda\in\Lambda_w} \tag{84}$$

for$L^2(\mathbb{R}^d)$and hence satisfies by Eq. (85)

$$\Sigma_{\lambda\in\Lambda_w}\int_{\mathbb{R}^d}|f,T_bI\psi_\lambda|^2\mathrm{d}b=\Sigma_{\lambda\in\Lambda_w}f*\psi_{\lambda 2}^2=\|f\|_2^2 \tag{85}$$

there exists a constant C > 0 such that for all $f\in H_W$, and $\tau\in C^1(\mathbb{R}^d,\mathbb{R}^d)$ with $\|D\tau\|_\infty\le\frac{1}{2d}$ deformation error has following deformation stability bound given by Eq. (86)

$$|\Phi_W(F_zf)-\Phi_W(f)\|\|C\left(2^{-J}\|\tau\|_\infty+J\|D\tau\|_\infty+D^2\tau_\infty\right)\|f\|_{H_W} \tag{86}$$

At comparison to scattering networks, replace wavelet-modulus operation in the n-th network layerby Eqs. (87) and (88), (89)

$$f\mapsto S_n^{d/2}P_n(f)(S_n\cdot) \tag{87}$$

$$f_\mathrm{d}\in\dot{\ell}^2(\mathbb{Z}):=\left\{f_\mathrm{d}:\mathbb{Z}\to\mathbb{C}|\Sigma_{k\in\mathrm{Z}}|f_\mathrm{d}[k]|^2<\infty\right\} \tag{88}$$

$$f_\mathrm{d}\mapsto h_\mathrm{d}:=f_\mathrm{d}[S\cdot] \tag{89}$$

In formal, network's m$^{\mathrm{th}}$ layer input is represented by $I^{(m)}$. The neural network'sm$^{\mathrm{th}}$ three dimensional convolutional layer is $n_1^{(m-1)}\times n_2^{(m-1)}\times n_3^{(m-1)}$three dimensional object having$n_c^{(m-1)}$ so $I^{(m-1)}\in(\mathbb{R}^{n1^{(m-1)}\times n_2^{(m-1)}\times n_3^{(m-1)}})$ and I (m,`) i,j,k denotes its elements. The three dimensional volume are indexed and the channel is chosen by i, j, and k. The m$^{\mathrm{th}}$ convolutional layer output is described by its dimensions given by $n_1^{(m-1)}\times n_2^{(m-1)}\times n_3^{(m-1)}$ and also the number of channels or filters it obtains $n_c^{(m-1)}$. The m$^{\mathrm{th}}$ layer output is a input's convolution with a filter and it is calculated by Eqs. (90) and (91)

$$I_{i,j,k}^{(m,l)}=f_{tanh}\left(b^{(m,l)}+\Sigma_{i,j,k,l}I_{i,j,k}^{(m-1,l)}W_{i-i,j-j,k-k,l-l}^{(m,l)}\right) \tag{90}$$

$$\widehat{h}_\mathrm{d}(\theta):=\Sigma_{k\in\mathbb{Z}}h_\mathrm{d}[k]e^{-2\pi ik\theta}=\frac{1}{S}\Sigma_{k=0}^{S-1}\widehat{f}_\mathrm{d}\left(\frac{\theta-k}{S}\right) \tag{91}$$

We also consider a variation on the logistic output function by Eqs. (92–95):

$$f=a+(b-a)(1+\exp\left(b^{(o)}+\Sigma_jW_j^{(o)}I_j^{(N)}\right)^{-1} \tag{92}$$

$$U_n[\lambda_n]f_2^2=S_n^d\int_{\mathbb{R}^d}\left|P_n\left(M_n\left(f*g_{\lambda_n}\right)\right)(S_nx)\right|^2\mathrm{d}x=\int_{\mathbb{R}^d}\left|P_n\left(M_n\left(f*g_{\lambda_n}\right)\right)(y)\right|^2\mathrm{d}y \tag{93}$$

$$=P_n\left(M_n\left(f*g_{\lambda_n}\right)\right)_2^2\le R_n^2M_n\left(f*g_{\lambda_n}\right)_2^2f*g_{\lambda_n}{}_2^2\le\Sigma_{\lambda_n'\in\Lambda_n}f*g_{\lambda_n'}{}_2^2\le B_n\|f\|_2^2 \tag{94}$$

$$\le L_n^2R_n^2f*g_{\lambda_n}{}_2^2\le B_nL_n^2R_n^2\|f\|_2^2 \tag{95}$$

Exactly, consider R0 be l × l × n as size, where the feature map dimension is l × l and the number of feature mapsis given by Eqs. (96–97):

$$f_i=\max-pool^{(2-i)}\{R_i\},i=\{0,1,2\} \tag{96}$$

$$\|U[q]f\|_2^2\le\left(\Pi_n^{k=1}B_kL_k^2R_k^2\right)\|f\|_2^2 \tag{97}$$

for$q\in\Lambda_1^n$ and $f\in L^2(\mathbb{R}^d)$the loss of classification is estimated as negative log-likelihood as shown below Eqs. (98–100):

$$L_{cs}=-\frac{1}{N}\Sigma_i\log\left(Y^i\middle|X^i,W_s,W_{cls}\right) \tag{98}$$

$$T_k^i=\frac{2\left(G_k^i-P_k^i\right)}{S_k}k\in\{x,y,z\} \tag{99}$$

$$T_d^i=\log\left(\frac{G_d^i}{\sqrt{S_2^2+S_y^2+S_z^2}}\right) \tag{100}$$

4. Performance analysis

The dataset and experimental methodology for single-label genre categorization from audio are described in this part. More specifically, we conducted an experiment to classify track genres using different data modalities: only audio, only album cover artwork, and both. The experimental results shows analysis of various dataset in terms of accuracy, precision, recall, F-1 score, average

loss of audio signal. The process of class labelling is done based on the diverse genre. The performance of the genre classification is investigated with the assistance of performance evaluation and the acquired outcome shows that the proposed technique is highly effective that is elaborated in this technique.

4.1. Dataset description

4.1.1. MSD-I dataset

Million Song Dataset contains metadata and pre-computed audio characteristics for 1 million songs. A dataset with annotations of 15 top-level genres with a single label per song was released alongside this dataset (Schreiber, 2015). We use information in the MSD/Echo Nest mapping archive to merge the CD2c version of this genre dataset4 with a collection of album cover photos taken from 7digital.com. The final collection contains 30,713 MSD recordings and their corresponding album cover photos, each labelled with a distinct genre classification from one of 15 classes. Based on a preliminary examination of the photos, we discovered that this collection of music is linked to 16,753 albums, with an average of 1.8 songs per album.

4.1.2. GTZAN Dataset

A compilation of ten genres, each with 100 audio recordings, each lasting 30 seconds. The GTZAN dataset is the most often used public dataset for music genre recognition evaluation in machine learning research (MGR). In order to represent a diversity of recording settings, the files were acquired in 2000-2001 from a number of sources, including personal CDs, radio, and microphone recordings.

4.1.3. ISMIR2004 Genre dataset

The audio tracks in the dataset come from the following eight genres: classical, electronic, jazz & blues, metal, punk, rock, pop, and world music. The data was divided into six categories for the genre recognition competition: classical, electronic, jazz-blues, metal-punk, rock-pop, and world, with two genres blended into a single category in some situations. Note that these 6 classes are used in ground-truth files, but the data is sometimes organised by original genre.

The above Table 1 shows comparative analysis in music genre classification for various datasets. The comparison has been made based on the music genre classification for identifying type of music genre from input dataset. Among all the techniques compared above, the proposed BiLSTM-VGG-16 Netbased classification has obtained the enhanced and précised output in analysing the input music genre dataset.

The above Fig. 2 shows the comparison of parameters in terms of accuracy, precision, recall, F-1 score, average loss of audio signal. The above graph has been plotted for MSD-I dataset for accuracy which acquires accuracy of 97%, precision of 94%, recall of 86.5%, F-1 score of 77.8%, average loss of audio signal of 40% for proposed technique. Here from this comparison the proposed technique has obtained optimal results for the parameters for MSD-I dataset.

The above Fig. 3 shows the comparison of parameters in terms of accuracy, precision, recall, F-1 score, average loss of audio signal. The above graph has been plotted for GTZAN Dataset for accuracy which acquires accuracy of 97.8%, precision of 93.8%, recall of 87.9%, F-1 score of 77.8%, average loss of audio signal of 36% for proposed technique which is enhanced output in music genre classification than existing technique. As the accuracy increases, occurrence of loss is minimized and the huge loss will decrease the accuracy.

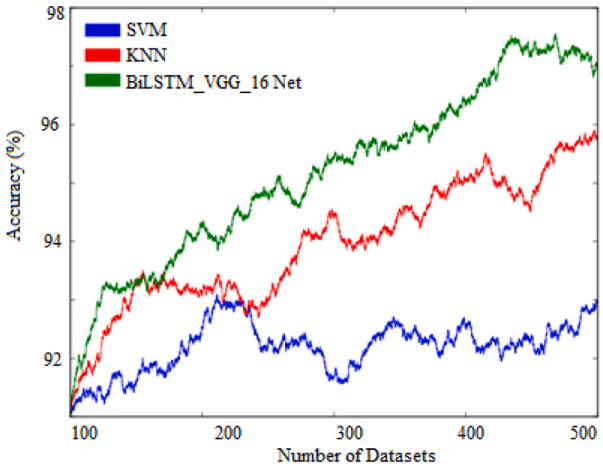
The above Fig. 4 shows the comparison of parameters in terms of accuracy, precision, recall, F-1 score, average loss of audio signal. The graph has been plotted between the % of parameters with number of epochs. For ISMIR2004 Genre datasetfor accuracy of 96.5%, precision of 94%, recall of 87.8%, F-1 score of 77%, average loss of audio signal of 35% for proposed technique which is the enhanced results of ISMIR2004 Genre datasetusing proposed technique (Table 2).

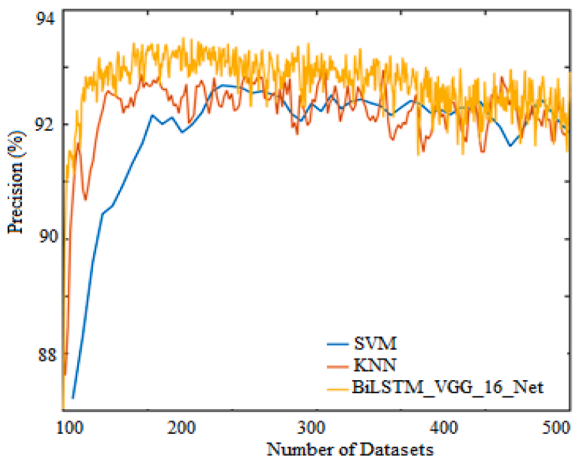
5. Conclusion

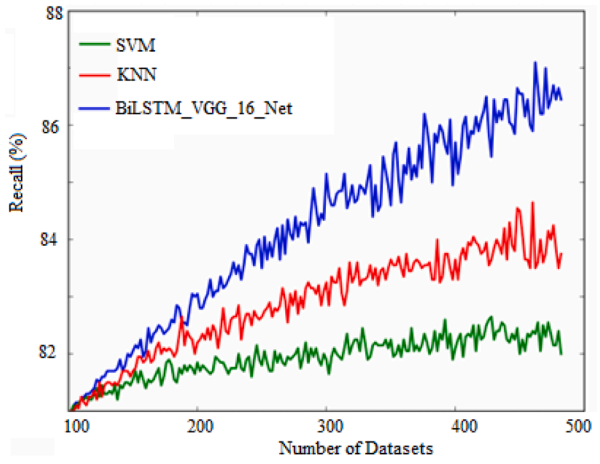
The aim of this paper is to propose novel technique in predicts and classify the genre of songs. Initially to collect data of music genre based on spectrogram audio for retrieving the feature vector of the audio features. In the field of MIR, categorising music files according to their genre is a difficult issue. Communities are identified by type of music they write or listen to. Taking advantage of deep neural networks' superior performance in computer vision, some researchers use them to classify music genres using audio spectrograms as input, which are analogous to RGB photographs. These strategies are based on the implicit premise that spectrums with varying temporal steps are equally important. However, it contradicts both psychology's processing bottleneck theory and our findings from audio spectrograms. The signal has been processed and their training and testing is carried out using feature extraction and classification by deep learning architecture. The feature vector extraction for both training and testing signal is carried out using

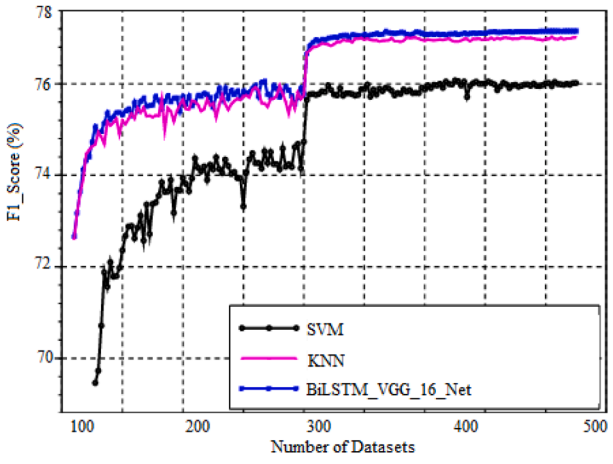
Table 1
Comparative analysis in music genre classification for various datasets.

Dataset	Music genre input	Processed input data	Extracted feature vector using BiLSTM	VGG-16 Net based classification
MSD-I dataset				
GTZAN Dataset				
ISMIR2004 Genre dataset				









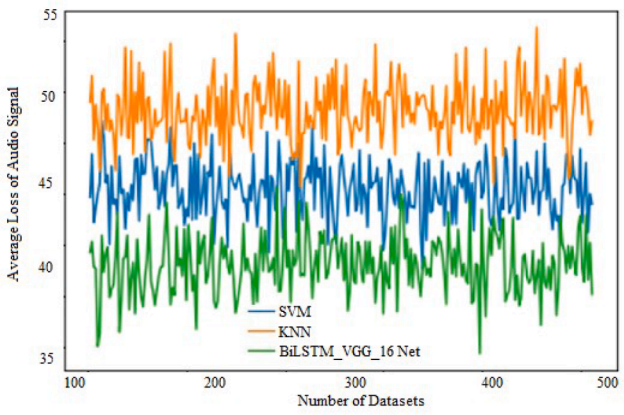
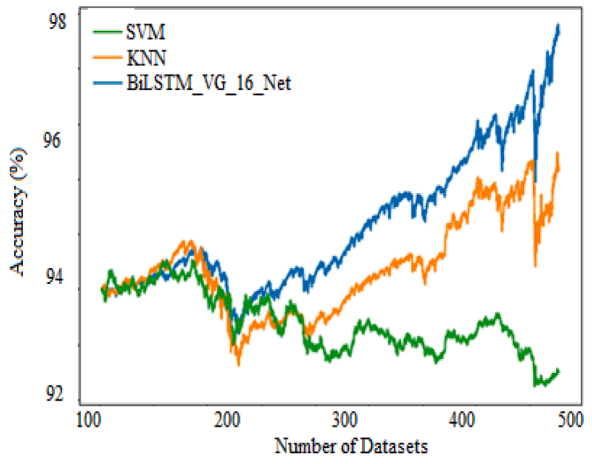
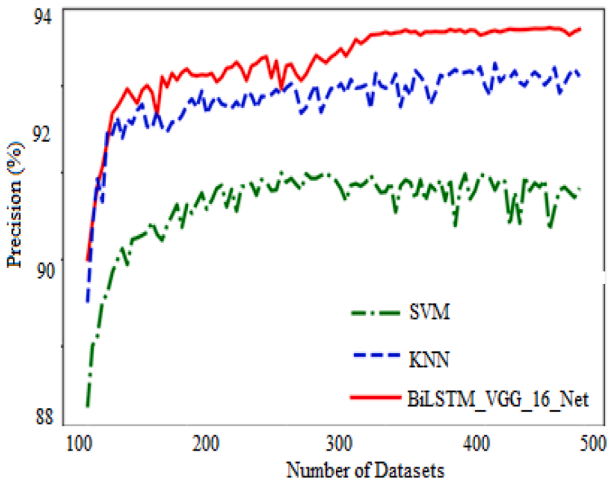
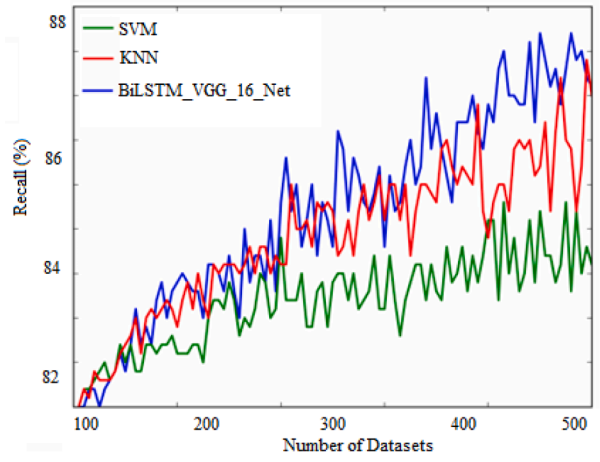


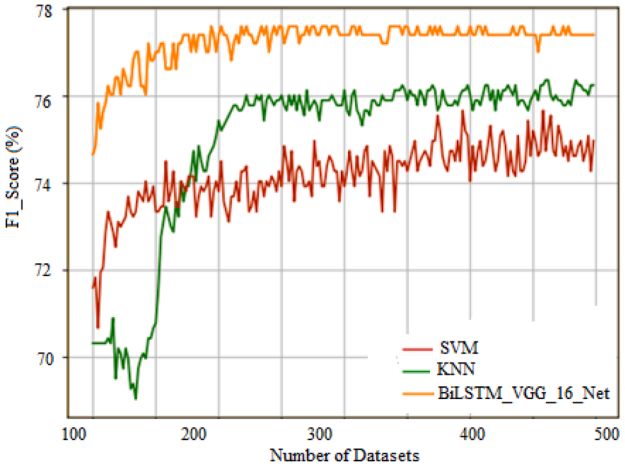
Fig. 2. Comparison of parameters for MSD-I dataset (a) Accuracy, (b) Precision, (c) Recall, (d) F1 score. (e) average loss of audio signal.

BiLSTM and classification using VGG-16Net. . The signal has been processed based on the spectrograms of the audio signals. Then this spectro audio has been extracted using BiLSTM and predict the genre label through the extraction of the features. The output of this feature extraction will be trained and tested data feature vector. These feature vector has been compared and classified using VGG-16 Net based classification for genre label classification. The classification output will be classified genre labels. The experimental results shows analysis of various dataset in terms of accuracy, precision, recall, F-1 score, average loss of audio signal. In future, approach can









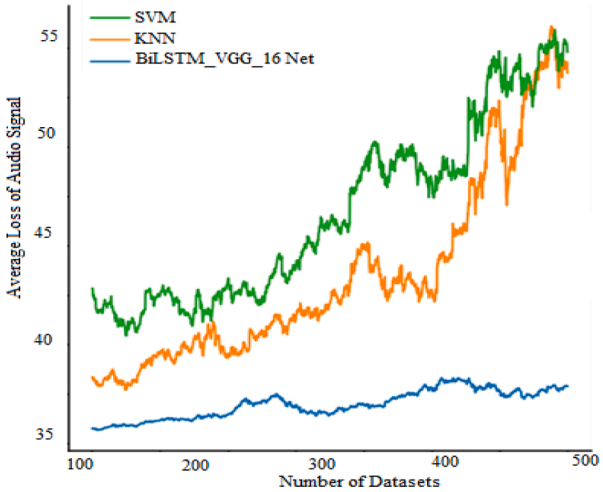


Fig. 3. Comparison of parameters for GTZAN **Dataset** (a) Accuracy, (b) Precision, (c) Recall, (d) F1 score. (e) average loss of audio signal.

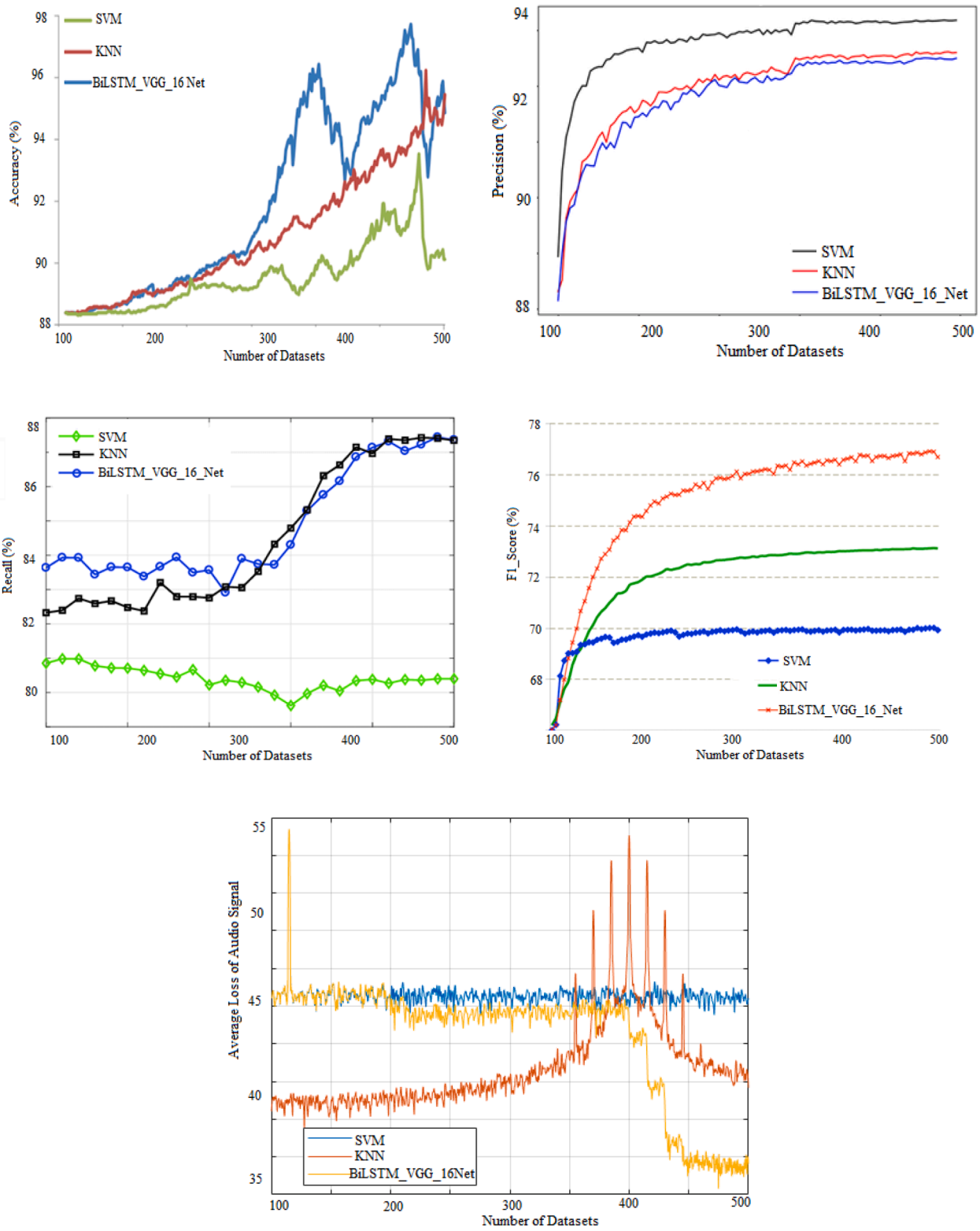


Fig. 4. Comparison of parameters for **ISMIR2004 Genre dataset**(a) Accuracy, (b) Precision, (c) Recall, (d) F1 score. (e) average loss of audio signal.

Table 2
Comparative analysis of Proposed technique with existing techniques.

Datasets	Techniques	Accuracy	Recall	Precision	F-1 score	average loss of audio signal
MSD-I dataset	SVM	93	92	85.2	76	50
	KNN	96	93.5	86	77.2	45
	BiLSTM-VGG-16 Net	97	94	86.5	77.8	40
GTZAN Dataset	SVM	93	91.2	84	75	54
	KNN	96	92.5	87.3	76	54.5
	BiLSTM-VGG-16 Net	97.8	93.8	87.9	77.8	36
ISMIR2004 Genre dataset	SVM	90	93.1	81.5	70	45
	KNN	95.4	93.5	87.2	73	40
	BiLSTM-VGG-16 Net	96.5	94	87.8	77	35

be enhanced with optimal feature vector selection with the assistance of optimization technique that can also reflect in the performance of the classification.

Declaration

Ethics Approval and Consent to Participate:
No participation of humans takes place in this implementation process
Human and Animal Rights:
No violation of Human and Animal Rights is involved.

Funding

No funding is involved in this work.

CRediT authorship contribution statement

There is no authorship contribution.

Declaration of Competing Interest

Conflict of Interest is not applicable in this work.

Acknowledgment

We are grateful for financial support from the Philosophy and Social Sciences of Sichuan Province of China (No. YWHY21-09).

References

[1] Nanni L, Costa YM, Aguiar RL, Silla Jr CN, Brahnam S. Ensemble of deep learning, visual and acoustic features for music genre classification. J New Music Res 2018;47(4):383–97.
[2] Oramas S, Barbieri F, Nieto Caballero O, Serra X. Multimodal deep learning for music genre classification. Trans Int Soc Music Inf Retrieval. 2018 2018;1(1):4–21.
[3] Tang CP, Chui KL, Yu YK, Zeng Z, Wong KH. Music genre classification using a hierarchical long short term memory (LSTM) model. Third International Workshop on Pattern Recognition, 10828. International Society for Optics and Photonics; July 2018, 108281B.
[4] Liu J, Wang C, &Zha L. A Middle-Level Learning Feature Interaction Method with Deep Learning for Multi-Feature Music Genre Classification. Electronics 2021;10(18):2206.
[5] Costa YM, Oliveira LS, Koerich AL, Gouyon F, Martins JG. Music genre classification using LBP textural features. Signal Process 2012;92(11):2723–37.
[6] Nanni L, Costa YM, Lumini A, Kim MY, &Baek SR. Combining visual and acoustic features for music genre classification. Expert Syst Appl 2016;45:108–17.
[7] Yu Y, Luo S, Liu S, Qiao H, Liu Y, Feng L. Deep attention based music genre classification. Neurocomputing 2020;372:84–91.
[8] Liu J, Wang C, & Zha L. A Middle-Level Learning Feature Interaction Method with Deep Learning for Multi-Feature Music Genre Classification. Electronics 2021;10(18):2206.
[9] Senac C, Pellegrini T, Mouret F, &Pinquier J. Music feature maps with convolutional neural networks for music genre classification. In: Proceedings of the 15th international workshop on content-based multimedia indexing; June, 2017. p. 1–5.
[10] Liu C, Feng L, Liu G, Wang H, Liu S. Bottom-up broadcast neural network for music genre classification. Multim Tools Appl 2021;80(5):7313–31.
[11] Nanni L, Costa YM, Aguiar RL, Silla Jr CN, Brahnam S. Ensemble of deep learning, visual and acoustic features for music genre classification. J New Music Res 2018;47(4):383–97.
[12] Rhanoui M, Mikram M, Yousfi S, &Barzali S. A CNN-BiLSTM model for document-level sentiment analysis. Mach Learn Knowl Extract 2019;1(3):832–47.
[13] Jang B, Kim M, Harerimana G, Kang SU, Kim JW. Bi-LSTM model to increase accuracy in text classification: combining Word2vec CNN and attention mechanism. Appl Sci 2020;10(17):5841.
[14] Zhu Y, Gao X, Zhang W, Liu S, Zhang Y. A bi-directional LSTM-CNN model with attention for aspect-level text classification. Fut Internet 2018;10(12):116.
[15] Cai L, Zhou S, Yan X, Yuan R. A stacked BiLSTM neural network based on coattention mechanism for question answering. Comput Intell Neurosci 2019;2019.
[16] Liu J, Wang C, &Zha L. A Middle-Level Learning Feature Interaction Method with Deep Learning for Multi-Feature Music Genre Classification. Electronics 2021;10(18):2206.

[17] Shreyash A, Dhanure SP, Rathod PP, Ashay C, Pritesh G. Identification of Music Genre using Convolutional Neural Network. New Arch-Int J Contemp Architecture 2021;8(2):2103–9.
[18] Puppala LK, Muvva SSR, Chinige SR, &Rajendran PS. A Novel Music Genre Classification Using Convolutional Neural Network. In: 2021 6th International Conference on Communication and Electronics Systems (ICCES). IEEE; July, 2021. p. 1246–9.
[19] Kumaraswamy B, &Poonacha PG. Deep Convolutional Neural Network for musical genre classification via new Self Adaptive Sea Lion Optimization. Appl Soft Comput 2021;108:107446.
[20] KM, A. (2021). Deep Learning Based Music Genre Classification Using Spectrogram.
[21] Qiu L, Li S, Sung Y. DBTMPE: Deep Bidirectional Transformers-Based Masked Predictive Encoder Approach for Music Genre Classification. Mathematics 2021;9(5):530.
[22] Raval M, Dave P, &Dattani R. Music genre classification using neural networks. Int J Adv Res Comput Sci 2021;12(5).

Wang Hongdan entered the College of Art & Sciences, Universiti Utara Malaysia, Kedah Darul Aman, Malaysia to study for a phd degree from May 2021. She has participated in the project research on Chinese language and culture and has published 1 article. Expertise: vocal singing, vocal music teaching

DR. SITI SALMI JAMALI working in School of Creative Industry Management and Performing Arts-College of Arts and Sciences,Universiti Utara Malaysia,She has published a total of 9 articles, the main research direction is Educational Technology, Multimedia, Augmented Reality. Expertise: Educational Technology, Multimedia, Augmented Reality

Chen Zhengping working in Departmentof Music and Dance, College of Chinese & Asean Arts, Chengdu University, SiChuan, China. She has published a total of 2 articles. She has participated in the project research on Chinese language and culture,the main research direction isChinese National Vocal Music Research, Performance and Teaching. Expertise: Chinese National Vocal Music Research, Performance and Teaching

Shan qiaojuan entered the College of Art & Sciences, Universiti Utara Malaysia, Kedah Darul Aman, Malaysia to study for a phd degree from May 2021. Expertise: animation, animation technology, animated computer education Biographies: Ren Le working in Zigong Fourth People's Hospital,SiChuan,China,he is Information system management engineer. Expertise: Information system management engineer