Oktoechos Classification and Generation of Liturgical Music using Deep Learning Frameworks

Rajeev Rajan, Varsha Shiburaj, Amlu Anna Joshy
Audio, Speech and Language Lab
College of Engineering,Trivandrum
APJ Abdul Kalam Technological University, India

Abstract

In the oktoēchos tradition, liturgical hymns are sung in eight modes or eight colours (known as eight 'niram' in Indian liturgy). In this paper, recurrent neural network (RNN) models are used for oktoēchos genre classification with the help of musical texture features (MTF) and i-vectors. The performance of the proposed approaches is evaluated using a newly created corpus of liturgical music in the South Indian language-Malayalam. Long short-term memory (LSTM)-based and gated recurrent unit (GRU)-based experiments report an average classification accuracy of 83.76% and 77.77%, respectively, with a significant margin over the i-vector-DNN framework. The experiments demonstrate the potential of RNN models in learning temporal information through MTF in recognising eight modes of oktoēchos system. Furthermore, since the Greek liturgy and Gregorian chant also share similar musical traits with Syrian tradition, the musicological insights observed can potentially be applied to those traditions. The generation of oktoēchos genre music style is discussed using an encoder-decoder framework. The quality of the generated files is evaluated using a perception test.

Keywords: liturgy, colour, timbral, deep learning

1 Introduction

Liturgical music is specifically designed for worship in a religious rite of the Christian community. Music plays a vital role in the liturgy. Furthermore, the vast diversity of forms, styles, and functions in liturgical music makes it challenging to categorise various sub-genres. The Western Syriac music tradition is one of the most ancient ecclesiastical music systems, unique in its richness of literature and music. It is worth noting that the Greek liturgy and Gregorian chant share some of the musical traits of the Syrian tradition. This music system has been transferred to the Indian Orthodox (Malankara) tradition through its relationship with the orthodox church in Syria (Antiochian liturgy). The Indian Orthodox liturgy, originating from the Syrian liturgy, was transferred to India in the 16th century.

In Syrian liturgical music, melodies are categorised into eight tunes known as eight "colours" (Palackal, 2004). This is the system of singing the exact text in eight different melodies in an eight-week cycle and is referred to as "oktoēchos". The musically composed hymns of the 'oktoēchos' system are traditionally used in various feasts and special occasions. Those colours used in the liturgy for certain special occasions are listed in Table 1. In their musical structure, the colours are very much related to the rāga system[1] of Indian music. They are not at all equal (Vysanethu, 2004). The 'oktoēchos' musical tradition has been transferred to the Indian Orthodox liturgical music over centuries in the form of hymns in Malayalam. [2]

Music genre classification has been addressed extensively over the last two decades. It has intelligent search, information retrieval, playlist recommendation and management applications. The particular category of liturgical music, oktoēchos, is considered in the proposed work for analysis. The task

[1] Rāga is the fundamental melodic framework for both Carnatic and Hindustani traditions.
[2] https://en.wikipedia.org/wiki/Malayalam.

of the oktoechos genre classification is addressed using RNN models in the paper. Different from typical music genre classification approaches, the task of oktoechos classification is challenging due to the segmental similarity and the concept of sharing the same lyrics.

Table 1: Modes for the special occasion

No.	*Festival*	Mode/Colour
1	Yaldo	Colour:1
2	Denho	Colour:2
3	Mayaltho	Colour:3
4	Sooboro	Colour:4
5	Soolokko	Colour:5
6	Feast of Tabernacle	Colour:6
7	Shoonoyo	Colour:7
8	The Feast of the Cross	Colour:8

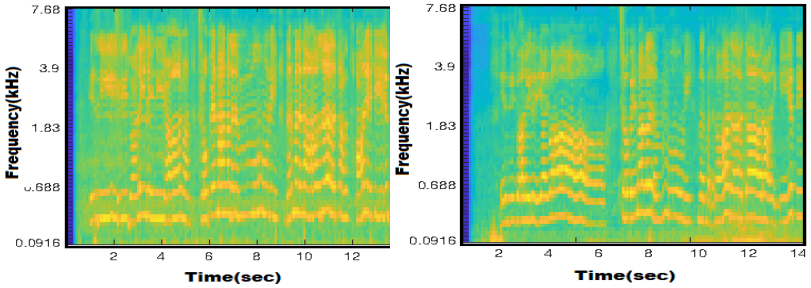

Figure 1: Mel-spectrogram of a song in colour 1 and in colour 2.

1.1 Syrian Liturgical Music

The Western Syriac music tradition is based on the norms prescribed in "Bethgazzo".[3] The vast liturgical music repertory may be divided into two main categories: chants that fall under the system of eight classes of melodies ("oktoechos"), and those that have only a single melody. This study will focus on the chants in the first category (Palackal, 2004). In the first category, the liturgical hymns are sung in eight modes, called "oktoechos". The "oktoechos" system is a group of eight adaptable melody types, known as eight "colours", "modes" or "niram" (Vysanethu, 2004). Different colours embellish the same text differently. Therefore, colours may be considered a vocal music genre (Palackal, 2004). The eight colours come under different emotional moods. The rhythmic aspect of melodies in each colour often depends on the poetic meter of the text. None of the Syriac melodies may cover eight notes in an octave. It may often cover three or four, or five notes. Oktoechos can be compared to rāga in the sense that they are also creating passion or rasa[4] during singing (Vysanethu, 2004). In Indian art music, a hymn in a rāga can be sung or played in another rāga. Similarly, the liturgical chants are sung using eight tunes on many occasions in the oktoechos tradition.

Oktoechos is considered a cyclic system because it is performed in a cycle of eight weeks with two colours in a week. The same verbal text is sung in eight-week cycles within the liturgical year. Each colour begins with the evening prayer of Sunday. If the first colour is used in the evening, the same is continued for the rest of the day. From Monday evening onwards, the fifth colour is used. On Tuesday, it is again switched to the first colour. The following Sunday begins with the second colour. It is continued in the order 1-5; 2-6; 3-7; 4-8; until the fourth Sunday, and on the fifth Sunday onwards, the order becomes 5-1; 6-2; 7-3; 8-4. Mel-spectrograms of songs sung in two different colours, colour one and colour two, are shown in Fig. 1.

The objective of the proposed work is to introduce a deep neural network (DNN)-based architecture for the classification of a given oktoechos music audio file into one of the eight colours of the

[3]Bethgazzo is a Syriac liturgical book that contains a collection of Syriac chants and melodies.

[4]a state of emotional attachment to a literary form or a musical work

music tradition. Sequential processing is an efficient approach to address the challenge of segmental similarity of the oktoecẖos music system. We proposed sequential deep learning architectures for the classification of oktoecẖos music traditions.

1.2 Related Work

Researchers have used generative and discriminative models for music genre classification (Li et al., 2003; Shao et al., 2004). An unsupervised approach for learning rhythmic aspects of genres is explored in (Pesek et al., 2020). In contrast with the standard methods, model-based distances between time series can take into account the structure of the songs by modelling the temporal dynamics of the parameter sequence (Garcia-Garcia et al., 2010). More recent deep learning approaches process spectrograms for the music genre classification task (Choi et al., 2017; Pons et al., 2016). Long short-term memory (LSTM) networks with and without a soft attention mechanism are utilised for genre classification in (Irvin et al., 2016). LSTM without attention gives the best accuracy of 79% in their experiment on the test set. A hierarchical divide-and-conquer strategy to achieve ten genres classification using the mel-frequency cepstral coefficient (MFCC) is attempted in (Wong et al., 2018). An average classification accuracy of 52.975% is reported for the work. A hierarchical attention network (HAN) to exploit the hierarchical layer structure of lyrics for music genre classification can be seen in (Tsaptsinos, 2017). In addition, it learns the importance of the words, lines, and segments for genre classification. Some of the previous attempts of "oktoecẖos" classification can be referred to in (Rajan and Ayasi, 2022; Rajan et al., 2021).

Multi-modal approaches mostly combine audio and lyrics (Laurier et al., 2008). To produce a state-of-the-art classifier, it is evident that the classifier must combine lyrics and audio features(Mayer et al., 2008; Mayer and Rauber, 2011). However, since the same lyrics are used for eight modes in our task, a fusion algorithm's design may not benefit from the textual information for the proposed scheme. The proposed task is similar to music genre classification, but sharing textual content across modes is one of the specific traits of the oktoecẖos genre system. Although there has been significant work in music genre classification, the proposed task of liturgical music genre classification is the first of its kind.

1.3 Contributions

The major contributions of the proposed work are summarized below.

1. A new dedicated corpus of oktoecẖos liturgical music is introduced for future research in this particular music genre category.
2. The efficacy of the sequential deep learning architectures is experimentally proven for the classification of the oktoecẖos music tradition.
3. A pilot study on oktoecẖos music generation from lyrics is carried out, and a subjective evaluation is performed to evaluate the quality of generated songs.

2 System Description

Timbral, rhythmic and i-vector features are computed in the front end. The classification is performed using DNN, convolutional neural network (CNN), LSTM and GRU. Each phase is explained in detail in the following subsections.

2.1 Feature Extraction

2.1.1 Timbral and Rhythmic Features

It has already been proven that the timbral and the rhythmic features are useful in the genre classification task (Baniya et al., 2015). The timbral features, namely MFCC and low-level timbral feature-set (T_{LF}), are computed in the front end. Spectral centroid, spectral roll-off, spectral flux, and spectral entropy (Li et al., 2003) are extracted as low-level timbral feature sets.

Timbral features are described below:

1. MFCC: The efficacy of MFCC as a predictor of perceived similarity of timbre has been proved in numerous speech and music processing tasks (Richard et al., 2013; Seppanan, 2015). 39 dim MFCCs (13 MFCC, delta, and delta-delta features) are computed using a frame size of 40 ms and a frame-shift of 10 ms.
2. Spectral centroid: The center of gravity of the magnitude spectrum is indicated by this measure.
3. Spectral roll-off: Gives the frequency below which 85% of the magnitude distribution is accumulated.
4. Spectral flux: The squared difference of normalized magnitudes of successive spectral distribution is computed using spectral flux.
5. Spectral entropy: The spectral power distribution is measured using spectral entropy in music and speech processing tasks.

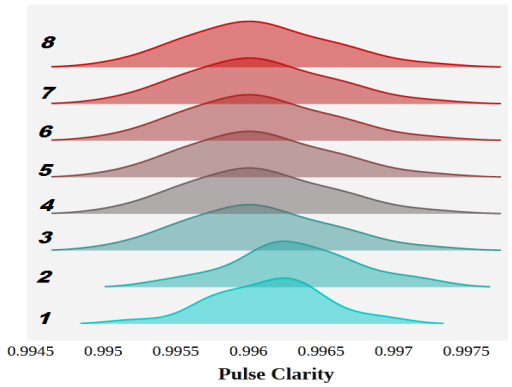

Figure 2: Distribution of pulse clarity for the colours

The salient periodicity in the music signal, analysed using the beat histogram is utilised to access the rhythmic content of the music(Tzanetakis and Cook, 2002). Features, namely, tempo, pulse clarity, and event density (Lartillot et al., 2008) are computed as rhythmic cues (R_F) for the proposed experiment. The rhythmic features are described below:

1. Tempo: Tempo measures the pace of the music piece and is measured in beats per minute (BPM). Tempo is the underlying beat rate of the music.
2. Event density: Event density represents the number of events per unit of time in the music piece (Madison et al., 2011).
3. Pulse clarity: The latent pulsation in music can be analysed by measuring the pulse clarity feature. The distribution of pulse clarity for the corpus is shown in Fig.2. It can be seen that the pulse clarity distribution for niram 1, niram 2 and niram 3 is different from the rest.

Low-level timbral features and rhythmic features are computed using the MIRToolbox. [5]

Mel-spectrogram

Visual representation of audio files such as spectrograms is utilised extensively for music genre classification (Sukhavasi and Adappa, 2019),(Ghosal and Kolekar, 2018). We have also used the mel-spectrogram-CNN framework for the proposed task. Spectrogram with the Mel frequency scale results in a mel-spectrogram visual representation. The Mel scale was developed to try and scale frequency data in a way that more closely resembles how humans perceive sound. Above 500 Hz, the Mels between pitches perceived as "evenly spaced" increase as frequency increases. Mel-spectrogram can be treated as a smoothed spectrogram with high sensitivity in the low-frequency region of the spectrum. It is extracted with a frame size of 40 ms and a hop size of 10 ms with 128 bins.

[5] https://www.jyu.fi/hytk/fi/laitokset/mutku/en/research/materials/

i-vectors

I-vector subspace modelling has been utilised effectively in numerous speech and music processing applications (Eghbal-zadeh et al., 2015; Zhong et al., 2017). A low dimensional space maps the variabilities in the speaker or speech environment(Verma and Das, 2015). "i-vectors" are created by adopting the popular factor analysis technique. Low-dimensional features are computed from the Gaussian mixture model (GMM)-supervectors. The efficacy of i-vector modelling in speech recognition (Eghbal-zadeh et al., 2015; Zhong et al., 2017) motivated us to use the same as a baseline in the proposed task. The Alize tool kit (Bonastre et al., 2005) is employed in the proposed work to compute i-vectors.

In the i-vector system (Dehak et al., 2011), high dimensional GMM supervector space (generated from concatenating the mean values of GMM) is mapped to a low dimensional space called total variability space. The target utterance GMM is adapted from a universal background model (UBM) using eigenvoice adaption. The target GMM supervector can be viewed as a shifted version of UBM. Formally, a target GMM supervector M can be written as:

$$M = m + Tw \tag{1}$$

where m represents the UBM supervector, T is a low dimensional rectangular total variability (TV) matrix, and w is termed an i-vector. Using training data, the UBM and TV matrix is modeled by expectation maximization. In the E-step, w is considered as a latent variable with normal prior distribution $N(0, I)$. Eventually, the i-vectors will be estimated as the mean of the posterior distribution of w, that is (2011),

$$w(u) = (I + T^T \Sigma^{-1}.N(u).T)^{-1} T^T \Sigma^{-1} S(u) \tag{2}$$

where for utterance u, the terms $N(u)$ and $S(u)$ represent zeroth and centralized first-order Baum-Welch statistics respectively, and Σ is the covariance matrix of UBM. 100-dimensional i-vectors (i_{MFCC}) are extracted for each song from frame-level computed MFCC using the Alize tool kit (Bonastre et al., 2005).

2.2 Classification Schemes

We have experimented with four classifiers: DNN, CNN, LSTM and GRU. With the rapid development of high-performance and parallel computing, CNN draws more attention from researchers in numerous image-processing classification schemes. The potential of CNN in multiple image processing tasks motivated us to apply the same in this music genre classification task. Recurrent neural networks have been commonly used for language understanding (Geron, 2018) as language is sequential. The promise of RNN in sequential data processing is explored well in numerous applications due to the long-term dependency on analyzing data streams.

DNN

DNN accepts a set of inputs, carries out complex calculations, and provides solutions for real-world problems like classification and regression. The theory of DNN-based processing can be found to in (Awad and Khanna, 2015). The proposed DNN architecture is based on six hidden layered networks, which use 64, 128, 256, 512, 1024 and 2048 nodes in successive layers with a dropout of 0.25. The network is trained with a batch size is 32 for 150 epochs by the AdaMax optimization algorithm. In addition, ReLU and softmax have been chosen for hidden and output layers, respectively.

CNN

CNN can process an input image to various levels to extract significant features that can be used as cues to classify multiple images. A typical CNN comprises numerous convolution layers preceding sub-sampling (pooling) layers, while the ending layers are fully connected. Details related to the architecture and applications of the convolutional neural network can be accessed in (Alzubaidi et al., 2021). CNNs have been widely used in image processing and time series analysis. The proposed CNN has six convolutional layers, followed by max-pooling. We use filters with very small 3×3 receptive fields for a fixed stride of one and increase the number of filters for the layer by a factor of 2 after every layer. Global max-pooling is adopted in the final max-pooling layer, which is then fed to a fully connected layer. The training is done with 100 epochs using an Adam optimizer with a learning rate of 0.001.

LSTM and GRU

LSTMs have the edge over CNN because of their property of selectively remembering patterns for a long duration of time. LSTM uses the concept of gates to implement calculations that are simple and effective The theory of LSTM-based sequential processing can be found in (Staudemeyer and Morris, 2019). LSTM cell corresponds to a node of a recurrent network and has, in addition to the input and output, a forget gate that avoids overfeeding of the vanishing gradient(Gruber and Jockisch, 2020). The LSTM architecture shown in Table 2 is effectively utilised to track the temporal pattern embedded in the modes of the music using MTF.

Table 2: LSTM architecture used for the experiment. (46,64): The input feature set to the layer is of dimension, 46. The output dimension of the feature set is 64

Serial no.	Output Size	Description
1	(46, 64)	LSTM, 64 hidden units
2	(46, 64)	Drop out (0.25)
3	(1024)	LSTM, 1024 hidden units
4	(1024)	Drop-out (0.25)
5	(8)	Dense (8 hidden units)

An LSTM can be formulated mathematically as follows:

$$i_t = \sigma(W_{xi}.x_t + W_{hi}h_{t-1} + b_i), \tag{3}$$
$$f_t = \sigma(W_{xf}.x_t + W_{hf}.h_{t-1} + b_f), \tag{4}$$
$$u_t = tanh(W_{xu}.x_t + W_{hu}.h_{t-1} + b_u), \tag{5}$$
$$o_t = \sigma(W_{xo}.x_t + W_{ho}.h_{t-1} + b_o), \tag{6}$$
$$c_t = i_t u_t + f_t c_{t-1}, \tag{7}$$
$$h_t = tanh(c_t o_t), output_{class} = \sigma(h_t.W_{outpara}) \tag{8}$$

where i_t, f_t, u_t, o_t, c_t, and $output_{class}$ represent equations for input gate, forget gate, update gate, output gate, cell state, and cell output, respectively. W_{xu}, W_{xi}, W_{xf}, W_{xo} and W_{hu}, W_{hi}, W_{hf} , W_{ho}, $W_{outpara}$ are weights, and b_u, b_i, b_f, b_o are biases to be computed during training. h_t is the output of a neuron at time t. $\sigma()$ denotes a sigma function and $tanh()$ represents the tanh function. The input x_t is the feature set at time t. $output_{class}$ is the classification output. The LSTM structure used in the proposed experiment is given in Table 2.

The main difference between GRU and LSTM is that GRU's architecture has two gates, reset and update gates, while LSTM has three gates. The advantage of GRU cells is that they are just as powerful as LSTM cells (Chung et al., 2014) even with small data sets, but they need less computing power. The LSTM is more complex with its three gates than the two-gated GRU cell. The GRU architecture shown in Table 3 is used in the proposed task. The training and validation accuracy of the proposed network during the experimentation is shown in Fig 3.

Table 3: GRU architecture used for the experiment. "None" means the batch dimension is variable. Any batch size will be accepted. gru29 is the label given for the layer.

Serial No.	Layer	Output shape
1	gru29(GRU)	(None, 46, 8)
2	gru29(GRU)	(None, 46, 16)
3	gru29(GRU)	(None,32)
4	dropout9(Drop out)	(None, 32)
5	dense10 (Dense)	(None, 8)

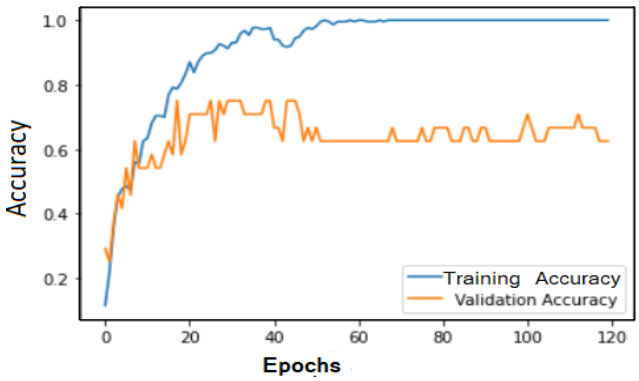

Figure 3: Training and validation accuracy

Table 4: Overall classification accuracy for the experiments

Serial No.	Feature	Method	Accr.(%)
1	MFCC + T_{LF}+ R_F	DNN	48.70
2	i_{MFCC} + T_{LF} + R_F	DNN	50.00
3	Mel-spectrogram	CNN	52.60
4	**MFCC + T_{LF}+ R_F**	**LSTM**	**83.76**
5	**MFCC + T_{LF}+ R_F**	**GRU**	**77.77**

3 Performance Evaluation

3.1 Database

A database is created in a studio environment, consisting of eight nirams (colours), with 384 audio tracks of duration, 25 to 35 sec per file. A total of 15 professional singers in the age group 12 to 50 participated in the data recording, and the whole session was recorded at 44.1kHz. All the singers were familiar with singing modes in "oktoēchos". Malayalam hymns were collected from the liturgical book of the Indian Orthodox church. The recordings were made in successive sessions using a high-quality microphone. A few audio files can be accessed at https://sites.google.com/view/audiosamples-2020/. During experimentation, 60% files of the dataset are used for training, 10% is used for validation and the rest for testing.

3.2 Experimental set-up

MFCCs (39 dim comprising 13 dim MFCC, delta, and delta-delta features), timbral (T_{LF}, four dimensions), rhythmic (R_F, three dimensions) are frame-wise computed with a frame width of 40 ms and frame-shift of 10 ms and fused at feature-level to obtain 46-dimensional MTF. In the i-vector experimental phase, 100-dimensional i-vectors are computed using 128 mixture GMM from MFCC using Alize tool-kit (Bonastre et al., 2005). The UBM model is trained using features derived from an auxiliary database comprising audio files other than the files in the corpus. The auxiliary database comprising 300 audio files (duration of 25 to 35 sec) was prepared in a studio environment. The songs from the training data are used for modelling the total variability matrix T by eigenvoice adaption. In the fusion scheme, track-level aggregated timbral (T_{LF}) and rhythmic (R_F) features are concatenated with track-level computed i-vectors. The precision, recall, and the F1 measure are used as performance metrics.

4 Results and Analysis

The results are tabulated in Table 4. As per the table, the average classification accuracies of 48.70%, 50.00%, 52.60%, 83.76%, and 77.76% are reported for DNN, i-vector framework, Mel-spectrogram-

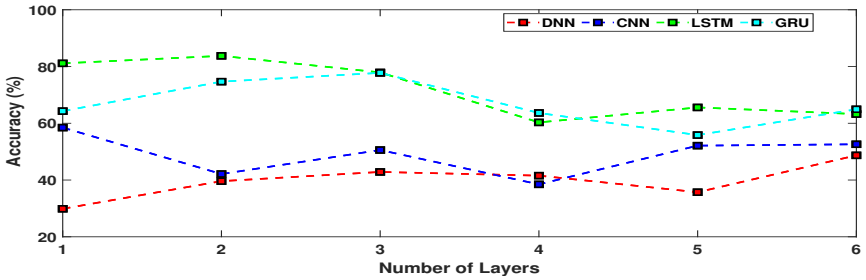

Figure 4: Accuracy with varying number of layers

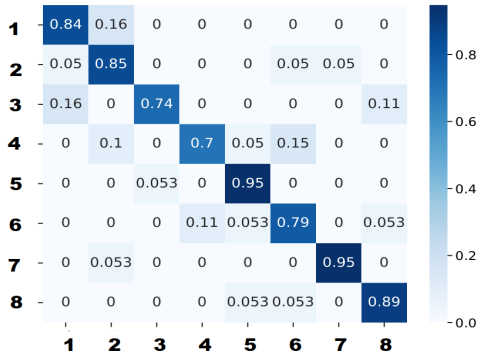

Figure 5: Normalized Confusion Matrix for MTF-LSTM

CNN, LSTM, and GRU, respectively. It is worth noting that the RNN models outperform other approaches by a significant margin. It is reasonable to say that a time pattern capturing scheme is required to recover more relevant information from temporal embedded musical traits (Garcia-Garcia et al., 2010). The experiments show that the RNN approach is promising for the given task, improving on cases where the temporal dynamics are not considered, and a stationary characterization of the sequences is employed. LSTM utilised musical textural features to capture song temporal dynamics effectively to perform oktoēchos classification.

It is important to note that LSTM, with its memory, could capture more information than the GRU. It is well established that the LSTM unit works well on sequence-based tasks with long-term dependencies. The GRU involves fewer parameters than LSTM, and the GRU trains faster than the LSTM.

I-vector subspace modelling reports an accuracy of 50.00% for eight classes in the experiment. It is shown in (Dai et al., 2017) that the relevant music elements can be captured by i-vectors and may potentially benefit the classification of the music signal. A possible cause of the low value of accuracy in the given experimental set-up may potentially be due to the inability to capture the temporal dynamics well with the given UBM framework. Besides this, aggregation of musical texture features at the track level might have deteriorated the performance.

The performance with the varying number of layers of the network is shown in Fig. 4. For the CNN framework, the result is saturated beyond six layers due to overfitting. As the number of layers, n, increases, the model grows in-depth, and the upper layers find efficient feature representations invariant to small perturbations leading to better model generalization. The authors in (Liua et al., 2021) emphasize the need for more training data in the visual representation-based approaches for the genre classification task. As seen in many image processing tasks, CNN needs enormous data to produce reliable results (Kaya and Bilge, 2019). Data augmentation is widely used for creating additional training data during classification experiments.

During the LSTM approach, the maximum accuracy is obtained for two layers, as seen in Fig. 4. The proposed experiment validates the claim that the MTF-LSTM framework has effectively learned temporal information. The performance of sequential data processing can be improved by efficiently designing the temporal architecture (Pons and Serra, 2019).

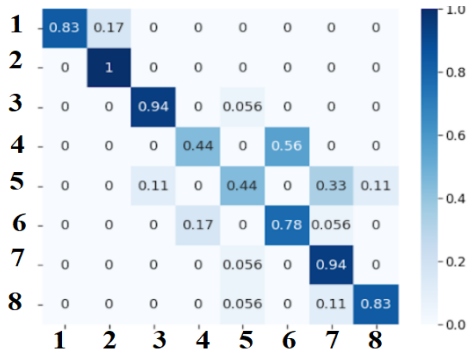

Figure 6: Normalized Confusion Matrix for MTF-GRU

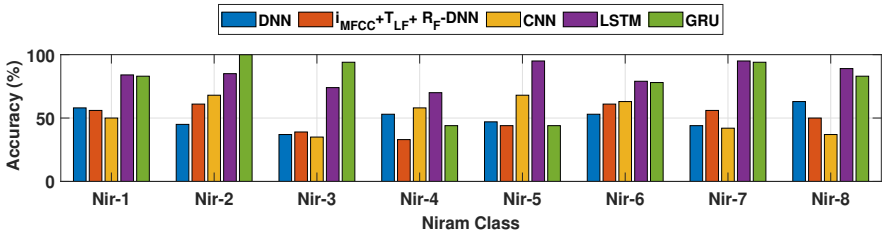

Figure 7: Class-wise performance for all phases of the experiments

The normalized confusion matrices of LSTM and GRU are given in Fig. 5 and 6, respectively. The class-wise classification accuracy of all nirams is greater than 70% for LSTM. Niram 5 and niram 7 report accuracy greater than 90%. Two classes, namely niram 4 and 5, report accuracy less than 50% for GRU, even though, GRU achieves comparable overall accuracy as that of LSTM. The class-wise accuracy can be analysed using Fig. 7. The significant improvement in the class-wise accuracy for nirams 1, 3, 7, and 8 for RNN models over the CNN-based framework can be seen from the plot. However, the performance can be improved using data augmentation and proper choice of architecture. The performance metrics precision, recall, and F1 score for all five approaches are given in Table 5. Average F1 measures of 0.50, 0.50, 0.52, 0.84, and 0.77 are reported for DNN, i-vector-DNN, CNN, LSTM, and GRU, respectively. The high precision, recall, and F1 scores show the significance of RNN models for the proposed task.

t-distributed stochastic neighbor embedding (t-SNE) is used to visualize the high dimensional data in lower dimensions. Fig. 8 and Fig. 9 visualize the output vectors produced by the snippets for the last dense layer of the trained LSTM and GRU networks using t-SNE. Note that there is good clustering (represented with colour) and a general separation of various classes for LSTM. It is important to note the effectiveness of LSTM in the proposed task without using any modelling data or augmentation data as that of i-vector or CNN methodologies.

5 Generation Aspects of Oktoechos Music

A survey of deep learning methods of singing voice synthesis can be found in (Cho et al., 2021). Deep learning-based architectures such as DNN (Nishimura et al., 2016), CNN (Nakamura et al., 2019), a recurrent neural network with LSTM (Kim et al., 2018), and generative adversarial networks (GAN) (Hono et al., 2019) have been successfully utilised for the task. Recent works include transformer-based (Vaswani et al., 2017) XiaoicSing (Lu et al., 2020), HifiSinger (Chen et al., 2020), and DiffSinger (Liu et al., 2021). The main experimental limitation of the recent models is that those models require a large corpus for training.

A pilot study was conducted to generate oktoechos music genres by extending the framework discussed in (Parekh et al., 2020). The system generates music from the lyrics of a song. An encoder-decoder framework (2020) with spectra-to-spectra conversion is utilised for singing voice

Table 5: Precision (P), recall (R), and F1 measure

SL.No	**Colour**	**MFCC+T_{LF}+R_F-DNN**			**i_{MFCC}+T_{LF}+R_F-DNN**			**Mel-spectrogram-CNN**			**MFCC+ T_{LF}+R_F- LSTM**			**MFCC+ T_{LF}+R_F- GRU**		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
1	Niram-1	0.35	0.58	0.44	0.48	0.56	0.51	0.42	0.50	0.45	0.80	0.84	**0.82**	1.00	0.83	**0.91**
2	Niram-2	0.36	0.45	0.40	0.46	0.61	0.52	0.52	0.68	0.59	0.74	0.85	**0.79**	0.86	1.00	**0.92**
3	Niram-3	0.32	0.37	0.34	0.54	0.39	0.45	0.70	0.35	0.47	0.93	0.74	**0.82**	0.89	0.94	**0.92**
4	Niram-4	0.71	0.53	0.61	0.46	0.33	0.39	0.69	0.58	0.63	0.88	0.70	**0.78**	0.73	0.44	**0.55**
5	Niram-5	0.53	0.47	0.50	0.40	0.44	0.42	0.54	0.68	0.60	0.86	0.95	**0.90**	0.73	0.44	**0.55**
6	Niram-6	0.62	0.53	0.57	0.52	0.61	0.56	0.55	0.63	0.59	0.75	0.79	**0.77**	0.58	0.78	**0.67**
7	Niram-7	0.50	0.35	0.41	0.59	0.56	0.57	0.47	0.42	0.44	0.95	0.95	**0.95**	0.65	0.94	**0.77**
8	Niram-8	0.80	0.63	0.71	0.60	0.50	0.55	0.44	0.37	0.40	0.85	0.89	**0.87**	0.88	0.83	**0.86**
	Average	0.52	0.48	**0.50**	0.50	0.50	**0.50**	0.54	0.53	**0.52**	0.85	0.84	**0.84**	0.79	0.78	**0.77**

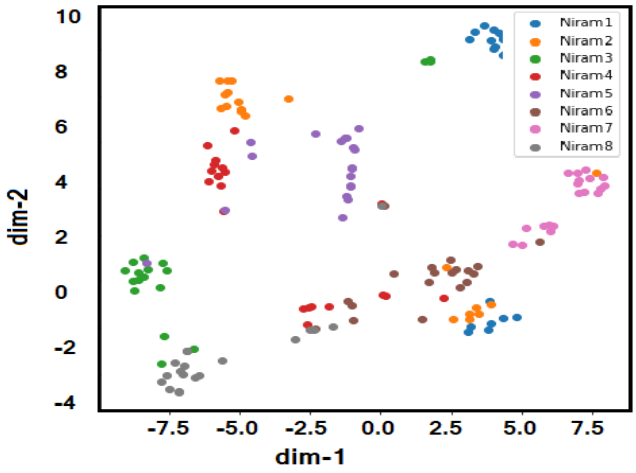

Figure 8: $t - SNE$ plot from LSTM

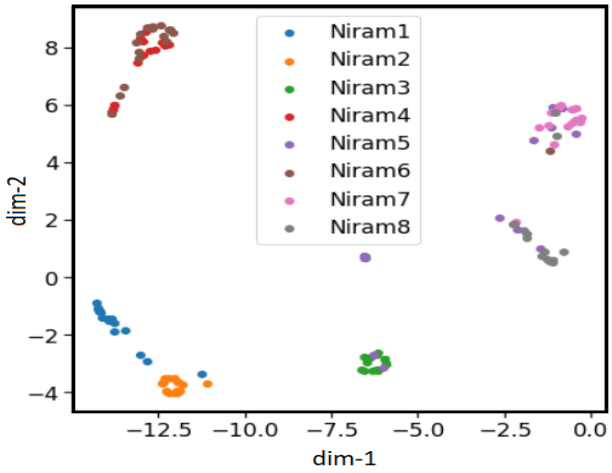

Figure 9: $t - SNE$ plot from GRU

generation as illustrated in Fig 10. A vocal melody extractor (Su, 2018), is used to extract melody contour from the inputted target melody, either humming or reference singing. The melodic pitch contours extracted for some of the modes are shown in Fig. 11. The short-time energy threshold is set at 40 dB below the maximum energy frame to identify silent frames. Any set of three or more consecutive silent frames with 50 ms duration is removed. Later, the speech signal is time-stretched to the same length as its target F0 contour. The log magnitude spectrogram of the speech input is computed using a phase vocoder (McFee et al., 2015).

An encoder-decoder-based deep learning framework (Parekh et al., 2020) produces two encodings, one for speech and another for the target melody obtained in the pre-processing stage. Using these encodings together, a sung version of the speech is produced using a U-net (Ronneberger et al., 2015)-based network architecture. The decoder synthesizes the singing voice by concatenating these two encodings with skip connections from the encoder.

Finally, the GriffinLim algorithm (Griffin and Jae Lim, 1983) is employed to reconstruct the waveform from the log magnitude spectrogram. We use the NUS-48E corpus(Duan et al., 2013), which consists of 48 songs sung by twelve male and female singers and an auxiliary corpus for training the STS module.

A fully convolutional architecture (ID) is utilised to handle the variable length signals with the help of GRU recurrent layers. A down-sampling factor of eight is used on the encoder side and up-sampled

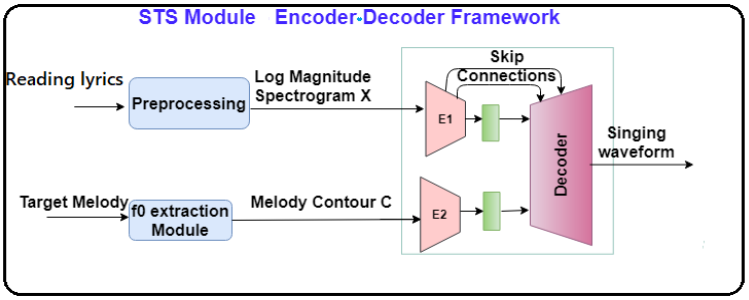

Figure 10: Generation of oktoecȟos music system

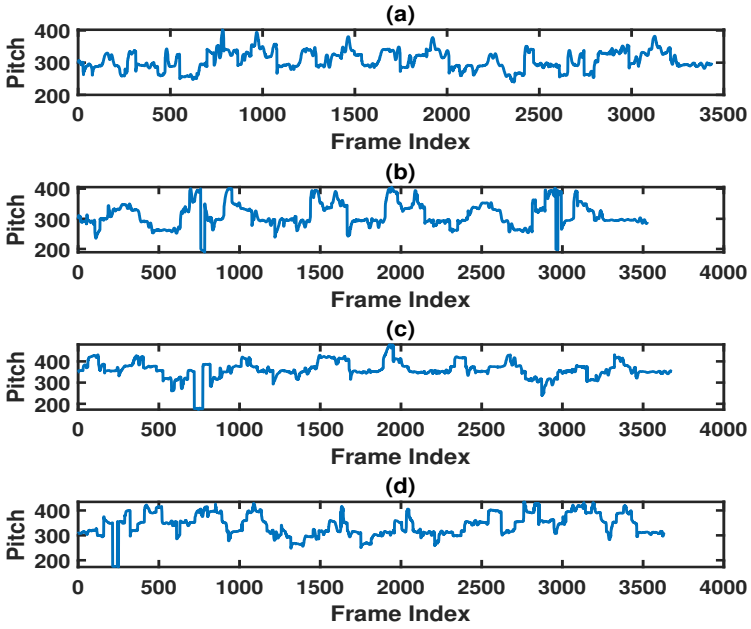

Figure 11: Melodic pitch computed for four nirams; (a) niram-1, (b) niram-2, (c) niram-3, (d) niram-4.

by the same factor on the decoder side. Skip connections between encoder E1 and decoder D are introduced to control the gradient vanishing problem and to train deeper networks (Parekh et al., 2020). The intelligibility is enhanced using a style-transfer-based module (Gatys et al., 2016).

The lyrics to generate a song is shown in Fig. 12 upper pane. The spectrogram of reading out the lyrics without any intonation is shown in Fig 12 (a), and the spectrogram of the music generated from the model is given in Fig. 12 (b). The generated song is of 25 to 35-sec duration. The sample-generated audio file can be accessed at https://sites.google.com/view/audiosamples-2020/home.

We evaluated the subjective quality of the synthesised songs using a mean opinion score (MOS) from 10 listeners. Two perceptual metrics are measured: song adaptation to the target melody and singing quality. Adapting a song to the target melody measures how well the synthesised song adapts to the target melody. Singing quality mainly investigates the quality of the synthesised voice by considering factors such as noise degradation and noticeable breaks. Each metric is measured using five opinion grades, namely excellent (5), very good (4), good (3), fair (2) and poor (1). Listeners were asked to grade the quality and MOS is computed by taking the average of the scores. A score of 3.64 is obtained for the synthesised samples. The performance can be improved by tuning the parameters of the encoder-decoder framework.

കതിരേറിടും കതിരോനെ

ലോകം ദർശിപ്പാനെത്തി

പ്രേക്ഷക ലോകത്തിൽ അക്ഷികൾ മായാവാൻ

നരരൂപത്തിൽ സസ്നേഹം ചെയ്തനുദയം ലോകത്തിൽ

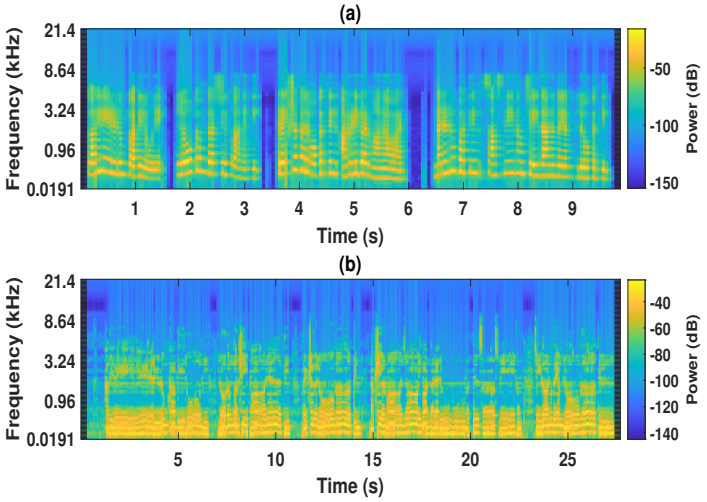

Figure 12: Lyrics of song segment (Upper pane). Lower pane, (a) mel-spectrogram of audio with lyrics reading out. Lower pane, (b) mel-spectrogram of generated music of audio file(niram 1).

6 Conclusion

Oktoēchos classification is addressed in this paper. The performance of the proposed approaches is evaluated using a newly created corpus of liturgical music in Malayalam. The evaluation shows the potential of the MTF-RNN framework in oktoēchos classification with an average classification accuracy of 83.76% and 77.77% for LSTM and GRU, respectively. Since the Greek liturgy and Gregorian chant also share similar musical traits with the Syrian tradition, the musicological insights observed can potentially be applied to those traditions. Some songs are synthesised using the encoder-decoder framework and a perception test is also conducted to analyse the quality of the generated songs.

References

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, *8*(53). https://doi.org/https://doi.org/10.1186/s40537-021-00444-8

Awad, M., & Khanna, R. (2015). Deep neural networks. In *Efficient learning machines: Theories, concepts, and applications for engineers and system designers* (pp. 127–147). Berkeley, CA, Apress. https://doi.org/10.1007/978-1-4302-5990-9_7

Baniya, B. K., Ghimire, D., & Lee, J. (2015). Automatic music genre classification using timbral texture and rhythmic content features. *Proc. of 17th Int. Conference on Advanced Communication Technology*, 434–443. https://doi.org/10.1109/ICACT.2015.7224907.

Bonastre, J.-F., Wils, F., & Meignier, S. (2005). Alize, a free toolkit for speaker recognition. *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, *1*, I/737–I/740 Vol. 1. https://doi.org/10.1109/ICASSP.2005.1415219

Chen, J., Tan, X., Luan, J., Qin, T., & Liu, T.-Y. (2020). Hifisinger: Towards high-fidelity neural singing voice synthesis. *ArXiv. /abs/2009.01776*. https://doi.org/https://doi.org/10.48550/arXiv.2009.01776

Cho, Y.-P., Yang, F.-R., Chang, Y.-C., Cheng, C.-T., Wang, X.-H., & Liu, Y.-W. (2021). A survey on recent deep learning-driven singing voice synthesis systems. *ArXiv. /abs/2110.02511*. https://doi.org/https://doi.org/10.48550/arXiv.2110.02511

Choi, K., Fazekas, G., Sandler, M., & Cho, K. (2017). Convolutional recurrent neural networks for music classification. *in Proc. of IEEE Int. Conference on Acoustics, Speech and Signal Processing*, 2392–2396. https://doi.org/10.1109/ICASSP.2017.7952585

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neuronal networks on sequence modeling, neuronal and evolutionary computing. *arXiv [Preprint]. arXiv:1412.3555.* https://doi.org/https://doi.org/10.48550/arXiv.1412.3555

Dai, J., Xue, W., & Liu, W. (2017). Multilingual i-vector based statistical modeling for music genre classification. *in Proc. of Interspeech*, 459–463. https://doi.org/10.21437/Interspeech.2017-74

Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., & Ouellet, P. (2011). Front-end factor analysis for speaker verification. *IEEE Trans. Audio, Speech, Lang. Process.*, *19*(4), 788–798. https://doi.org/10.1109/TASL.2010.2064307

Duan, Z., Fang, H., Li, B., Sim, K. C., & Wang, Y. (2013). The NUS sung and spoken lyrics corpus: A quantitative comparison of singing and speech. *in Proc. of Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, 1–9. https://doi.org/10.1109/APSIPA.2013.6694316

Eghbal-zadeh, H., Lehner, B., Schedl, M., & Widmer, G. (2015). I-vectors for timbre-based music similarity and music artist classification. *in Proc. of 16th Int. Society for Music Information Retrieval Conference*, 554–560. https://doi.org/10.13140/RG.2.1.1341.1287

Garcia-Garcia, D., Arenas-Garcia, J., Parrado-Hernandez, E., & Diaz-de-Maria, F. (2010). Music genre classification using the temporal structure of songs. *in Proc. of IEEE Int. Workshop on Machine Learning for Signal Processing*. https://doi.org/10.1109/MLSP.2010.5589240

Gatys, L., Ecker, A., & Bethge, M. (2016). Image style transfer using convolutional neural networks. *in Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2414–2423. https://doi.org/doi:10.1109/CVPR.2016.265.

Geron, A. (2018). Hands on machine learning with Scikit-learn and Tensorflow. *New York, O'Reilly*.

Ghosal, D., & Kolekar, M. H. (2018). Music genre recognition using deep neural networks and transfer learning. *in Proc. of Interspeech*, 2087–2091. https://doi.org/10.21437/Interspeech.2018-2045

Griffin, D., & Jae Lim. (1983). Signal estimation from modified short-time fourier transform. *in Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, *8*, 804–807. https://doi.org/10.1109/ICASSP.1983.1172092

Gruber, N., & Jockisch, A. (2020). Are GRU cells more specific and LSTM cells more sensitive in motive classification of text? *Frontiers in Artificial Intelligence*, *3*(40). https://doi.org/10.3389/frai.2020.00040

Hono, Y., Hashimoto, K., Oura, K., Nankaku, Y., & Tokuda, K. (2019). Singing voice synthesis based on generative adversarial networks, 6955–6959. https://doi.org/10.1109/ICASSP.2019.8683154

Irvin, J. A., Chartock, E., & Hollander, N. (2016). Recurrent neural networks with attention for genre classification. *Accessed on line:http://cs229.stanford.edu/proj2016/poster*.

Kaya, M., & Bilge, S. H. (2019). Deep metric learning: A survey. *Symmetry*, *11*(9), 1–26. https://doi.org/https://doi.org/10.3390/sym11091066

Kim, J., Choi, H., Park, J., Kim, S., Kim, J., & Hahn., M. (2018). Korean singing voice synthesis system based on an lstm recurrent neural network. *in Proc. of Interspeech*, 1551–1555. https://doi.org/10.21437/Interspeech.2018-1575

Lartillot, O., Eerola, T., Toiviainen, P., & Fornari, J. (2008). Multi-feature modeling of pulse clarity: Design, validation and optimization. *in Proc. of the 9th Int. Conference on Music Information Retrieval*, 521–526. https://doi.org/10.5072/ZENODO.243404

Laurier, C., Grivolla, J., & Herrera, P. (2008). Multimodal music mood classification using audio and lyrics. *in Proc. of Seventh IEEE Int. Conference on Machine Learning and Applications*, 688–693. https://doi.org/10.1109/ICMLA.2008.96

Li, T., Ogihara, M., & Li, Q. (2003). A comparative study on content-based music genre classification. *in Proc. of 26th Int. ACM Conference on Research and Development in Information Retrieval*, 282–289. https://doi.org/10.1145/860435.860487

Liu, J., Li, C., Ren, Y., Chen, F., & Zhao, Z. (2021). Diffsinger: Singing voice synthesis via shallow diffusion mechanism. https://doi.org/https://doi.org/10.48550/arXiv.2105.02446

Liua, C., Fengb, L., Liuc, G., Wangd, H., & Liub, S. (2021). Bottom-up broadcast neural network for music genre classification. *Multimed Tools Appl*, *80*, 7313–7331. https://doi.org/https://doi.org/10.1007/s11042-020-09643-6

Lu, P., Wu, J., Luan, J., Tan, X., & Zhou, L. (2020). Xiaoicesing: A high-quality and integrated singing voice synthesis system. https://doi.org/https://doi.org/10.48550/arXiv.2006.06261

Madison, G., Gouyon, F., Ullén, F., & Hörnström, K. (2011). Modeling the tendency for music to induce movement in humans: First correlations with low-level audio descriptors across music genres. *Journal of Experimental Psychology: Human Perception and Performance, 37*(5), 1578–94. https://doi.org/10.1037/a0024323

Mayer, R., Neumayer, R., & Rauber, A. (2008). Combination of audio and lyrics features for genre classification in digital audio collections. *In Proc. of the 16th ACM int. conference on Multimedia*, 159–168. https://doi.org/DOI:10.1145/1459359.1459382

Mayer, R., & Rauber, A. (2011). Musical genre classification by ensembles of audio and lyrics features. *in Proc. of Int. Society for Music Information Retrieval Conference*, 675–680.

McFee, B., Raffel, C., Liang, D., Ellis, D., Mcvicar, M., Battenberg, E., & Nieto, O. (2015). Librosa: Audio and music signal analysis in python. *in Proc. of 14th Python in Science Conference*, 18–24. https://doi.org/DOI:10.25080/Majora-7b98e3ed-003

Nakamura, K., Hashimoto, K., Oura, K., Nankaku, Y., & Tokuda, K. (2019). Singing voice synthesis based on convolutional neural networks. https://doi.org/https://doi.org/10.48550/arXiv.1904.06868

Nishimura, M., Hashimoto, K., Oura, K., Nankaku, Y., & Tokuda, K. (2016). Singing voice synthesis based on deep neural networks, 2478–2482. https://doi.org/10.21437/Interspeech.2016-1027

Palackal, J. (2004). Oktoechos of the syrian orthodox churches in south india. *Ethnomusicology*, *48*, 229–250. https://doi.org/https://www.jstor.org/stable/30046265

Parekh, J., Rao, P., & Yang, Y. H. (2020). Speech-to-singing conversion in an encoder-decoder framework. *in Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 261–265. https://doi.org/doi:10.1109/ICASSP40776.2020.9054473.

Pesek, M., Leonardis, A., & Marolt, M. (2020). An analysis of rhythmic patterns with unsupervised learning. *Applied Science*, *10*(1), 1–22. https://doi.org/https://doi.org/10.3390/app10010178

Pons, J., Lidy, T., & Serra, X. (2016). Experimenting with musically motivated convolutional neural network. *in Proc. of Int. Workshop on Content-Based Multimedia Indexing*, 1–5. https://doi.org/doi:10.1109/CBMI.2016.7500246.

Pons, J., & Serra, X. (2019). Randomly weighted CNNs for (music) audio classification. *in Proc. of IEEE Int. Conference on Acoustics, Speech and Signal Processing*, 336–340. https://doi.org/doi:10.1109/ICASSP.2019.8682912

Rajan, R., & Ayasi, A. (2022). Oktoechos Classification in Liturgical Music Using SBU-LSTM/GRU, In *Proc. interspeech 2022*. https://doi.org/10.21437/Interspeech.2022-136

Rajan, R., Joshy, A. A., & Shiburaj, V. (2021). Oktoechos classification in liturgical music using musical texture features. *Proc. of the 15th International Symposium on CMMR*, 57–66.

Richard, G., Sundaram, S., & Narayanan, S. (2013). An overview on perceptually motivated audio indexing and classification. *in Proc. of the IEEE*, *101*(9), 1939–1954. https://doi.org/doi:10.1109/JPROC.2013.2251591.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation (N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi, Eds.). In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *Medical image computing and computer-assisted intervention – miccai 2015*, Cham, Springer International Publishing. https://doi.org/https://doi.org/10.1007/978-3-319-24574-4_28

Seppanan, J. (2015). *Computational models for musical meter recognition* (Masters Thesis). Tampere University of Technology. Department of Information Technology.

Shao, X., Xu, C., & Kankanhalli., M. S. (2004). Unsupervised classification of music genre using hidden Markov model. *in Proc. of IEEE Int. Confernce on Multimedia and Expo, 3*, 2023–2026. https://doi.org/doi:10.1109/ICME.2004.1394661.

Staudemeyer, R., & Morris, E. (2019). Understanding LSTM – a tutorial into long short-term memory recurrent neural networks. https://doi.org/https://doi.org/10.48550/arXiv.1909.09586

Su, L. (2018). Vocal melody extraction using patch-based CNN. *in Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 371–375. https://doi.org/doi:10.1109/ICASSP.2018.8462420.

Sukhavasi, M., & Adappa, S. (2019). Music theme recognition using CNN and self-attention. *preprint arXiv:1911.07041*. https://doi.org/https://doi.org/10.48550/arXiv.1911.07041

Tsaptsinos, A. (2017). Lyrics-based music genre classification using a hierarchical attention network. *in Proc. of Int. Society for Music Information Retrieval Conference*, 694–701. https://doi.org/https://doi.org/10.48550/arXiv.1707.04678

Tzanetakis, G., & Cook, P. (2002). Musical genre classification of audio signals. *IEEE Tran. on Speech and Audio Proces.*, *10*(5), 293–302. https://doi.org/doi:10.1109/TSA.2002.800560.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *in Proceedings of Computation and Language (cs.CL)*, 5998–6008. https://doi.org/https://doi.org/10.48550/arXiv.1706.03762

Verma, P., & Das, P. (2015). I-vectors in speech processing applications: A survey. *International Journal of Speech Technology*, *18*(4), 529–546. https://doi.org/10.1007/s10772-015-9295-3

Vysanethu, P. (2004). Musicality makes the Malankara liturgy musical (moran etho 2). *St.Ephrem Ecumenical Research Institute, Kottayam, Kerala, India*.

Wong, K.-h., Tang, C., Chui, K., Yu, Y., & Zeng, Z. (2018). Music genre classification using a hierarchical long short term memory model. *in Proc. of Third Int. Workshop on Pattern Recognition*, 7. https://doi.org/10.1117/12.2501763

Zhong, J., Hu, W., Soong, F., & Meng, H. (2017). DNN i-vector speaker verification with short, text-constrained test utterances. *in Proc. of Interspeech*, 1507–1511. https://doi.org/10.21437/Interspeech.2017-1036