Interspeech 2018
2-6 September 2018, Hyderabad

Music Genre Recognition using Deep Neural Networks and Transfer Learning

Deepanway Ghosal, Maheshkumar H. Kolekar

Indian Institute of Technology Patna, India
deepanwayedu@gmail.com, mkolekar@gmail.com

Abstract

Music genre recognition is a very interesting area of research in the broad scope of music information retrieval and audio signal processing. In this work we propose a novel approach for music genre recognition using an ensemble of convolutional long short term memory based neural networks (CNN LSTM) and a transfer learning model. The neural network models are trained on a diverse set of spectral and rhythmic features whereas the transfer learning model was originally trained on the task of music tagging. We compare our system with a number of recently published works and show that our model outperforms them and achieves new state of the art results.

Index Terms: music genre recognition, music information retrieval, deep learning, transfer learning

1. Introduction and Related Work

Music information retrieval (MIR) is an interdisciplinary field dealing with the analysis of musical content by combining aspects from signal processing, machine learning and music theory. MIR enables computer algorithms to understand and process musical data in an intelligent way. Music genre recognition (MGR) is one of the most important subfields of MIR. Music genre is defined as an expressive music style incorporating instrumental or vocal tones in a structured manner belonging to a set of conventions. Automatic music genre recognition is a very interesting problem in the context of MIR because it enables systems to perform content based music recommendation, organizing musical databases and discovering media collections.

The first significant work on musical genre recognition were performed in [1] by Tzanetakis and Cook. Timbral texture, rhythmic content & pitch content based features were proposed and classification was done using Gaussian mixture model (GMM) and K-nearest neighbor (K-NN) algorithms. Musical genre recognition using support vector machines were proposed in [2] by Xu et al. In [3] Costa et al. proposed the approach of musical genre recognition using spectrogram features. In [4, 5] specific musical features were used with feature selection techniques. Musical genre classification using deep learning models has been performed in [6, 7, 8]. Survey works performed in [9, 10] gives a comprehensive account of genre classification of musical content and evaluation techniques. Authors in [11] introduced the Million Song Dataset - a collection of audio features and metadata for a million contemporary popular music tracks. A wide range of musical information retrieval systems can be build using this dataset including genre recognition, automatic music tagging, music recommendation, etc [1].

2. Proposed Methodology

In this work we have focused on genre recognition of the songs in the GTZAN dataset [1], which has been widely studied in the area of MGR. The dataset contains songs of ten different genres - blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae & rock. To recognize the genre of a song, we first train our deep neural network models on a set of extracted spectral and rhythmic features. We also utilize a transfer learning system to extract meaningful features from the songs. A multilayer perceptron network is then trained on this transferred features to predict the genres. Finally the predictions of different models are combined using a majority voting ensemble.

2.1. Feature Extraction

2.1.1. Two Dimensional Spectral and Rhythmic Features:

A diverse set of spectral and rhythmic domain features are first extracted from the raw musical wav signals. In the features listed below, 'Tonnetz' and 'Tempogram' are rhythmic features, the rest are spectral features. The musical data in the GTZAN dataset are sampled at 22050 Hz and are around 30 seconds long resulting in a total of roughly 22020×30 = 661500 samples. We compute the features for each sliding window of 2048 samples with shift of 1024 samples. We pad appropriate number of zeros at the end such that there are a total of $661500/1024 = 646$ windows and thus each song is represented as a $(646, k)$ dimensional feature matrix. The exact choice of k depends on the feature being computed.

• **Mel Spectrogram:** Mel-frequency cepstrum (MFC) representations introduced in [12] are widely used in automatic speaker and speech recognition. The mel spectrogram produces a time-frequency representation of a sound imitating the biological auditory systems of human beings. We compute the magnitude spectrum from the time series musical data and then map it on to the mel scale. We used k = 128.

• **Mel Cepstral, Delta and Double Delta Coefficients:** Mel cepstral coefficients (MFCCs) are the coefficients that collectively make up a mel-frequency cepstrum. We used k = 20 mel cepstral coefficients.

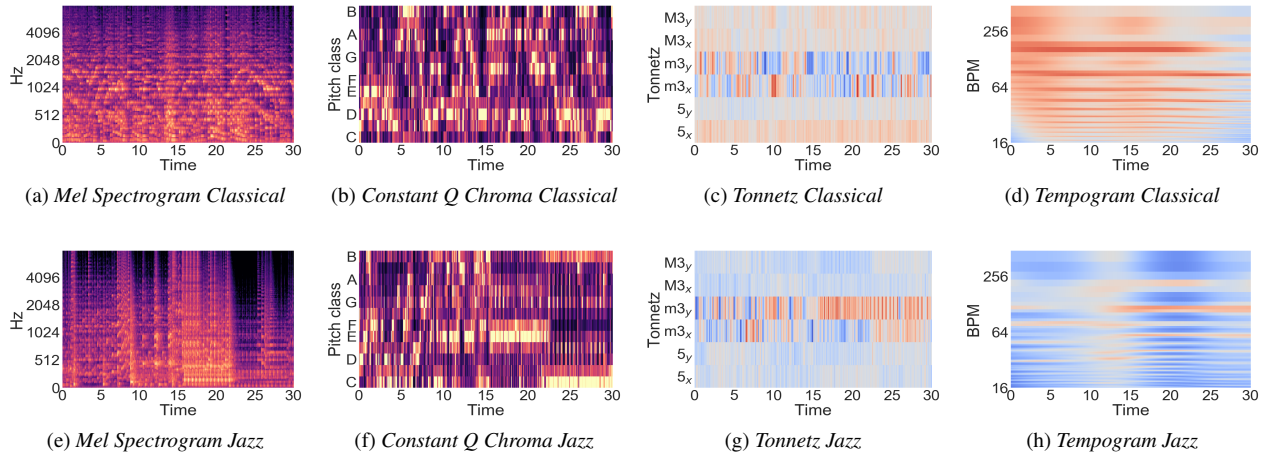
• **Delta Coefficients:** We used k = 20 delta coefficients (derivative of the mel cepstral coefficients).

• **Double Delta Coefficients:** We used k = 20 double delta coefficients (double derivative of the mel cepstral coefficients).

• **Energy Normalized Chromagram:** Chroma audio features are extensively used in musical signal processing. Chroma features are effective in audio matching and retrieval applications [13, 14] as they capture melodic and harmonic characteristics of music and are robust to changes in instrumentation and timbre. In [15] authors introduced Chroma Energy Normalized Statistics (CENS) features by considering short time statistics over energy distributions within the chroma bands. We took k = 12 as it represents 12 distinct semitones of the musical octave.

• **Constant Q Chromagram:** Constant Q transform [16] constitutes of a bank of filters with logarithmically spaced center frequencies $f_n = f_0 \times 2^{\frac{n}{b}}$ where $n = 0, 1, ..$; central frequency of the lowest filter is denoted by f_0 and the number of filters in each octave is denoted by b. An appropriate choice of

10.21437/Interspeech.2018-2045



(a) *Mel Spectrogram Classical* (b) *Constant Q Chroma Classical* (c) *Tonnetz Classical* (d) *Tempogram Classical*

(e) *Mel Spectrogram Jazz* (f) *Constant Q Chroma Jazz* (g) *Tonnetz Jazz* (h) *Tempogram Jazz*

Figure 1: *Distinctive spectral and rhythmic features of two songs belonging to the 'Classical' and 'Jazz' genre. Mel Spectrogram and Constant Q Chroma are spectral domain features, whereas Tonnetz and Tempogram are rhythm domain features. Similar phenomenon is observed for the rest of the features across all the genres.*

f_0 and b directly corresponds to musical notes. This transform also has increasing time resolution towards higher frequencies resembling the human auditory system. k = 12 was taken.

• **Short Time Fourier Transform (STFT) Chromagram:** Chromagram of short-time chroma frames are used with k = 12.

• **Tonnetz:** Tonal centroid features (tonnetz) are computed following works in [17]. Authors show that this features are successful in detecting changes in the harmonic content of musical audio signals, such as chord boundaries in polyphonic audio recordings. We used k = 6 tonnetz features.

• **Tempogram:** The aspects of tempo and rhythm are very important dimensions of music. In [18], the authors introduced a robust mid-level representation that encodes local tempo information by computing local autocorrelation of the onset strength envelope in music signals. This tempogram feature can act as a very important source of information for MGR, specifically where music reveals significant tempo variations. We used k = 128 tempogram features.

2.1.2. One Dimensional Averaged and Transfer Learning Features

We also compute the following one dimensional vectors as a summary statistic of the whole song.

• **Averaged Signal Vector**: This vector is calculated simply by taking the average of all the extracted two dimensional features listed above. After extracting $(646, k_1)$ dim matrix from mel spectrogram, $(646, k_2)$ dim matrix from mel cepstral coefficients, ..., $(646, k_n)$ features from tempogram, the averaging was performed over these 646 windows. Finally vectors of k_1, k_2, ..., k_n dimensions were obtained which were then concatenated to obtain the averaged signal vector. Our particular choices of k_1, ..., k_n led to this vector having dimension of 342.

• **Music Transfer Learning Vector**: Transfer learning is frequently used in computer vision problems. In this kind of systems, generally a deep convolutional net trained on the large scale ImageNet data [19] is used. Although the original network is trained on ImageNet data, it is able to capture a wide variety of visual features which are then used for other recognition tasks. In [7] authors introduce a musical transfer learning system. A deep convolutional neural network is first trained on a large dataset [11] for music tagging. The tags include genre, era, instrumentation, and mood labels. This trained network is then used as a feature extractor for other related tasks. We use the model to extract a 160 dimensional vector for each song.

2.2. Models

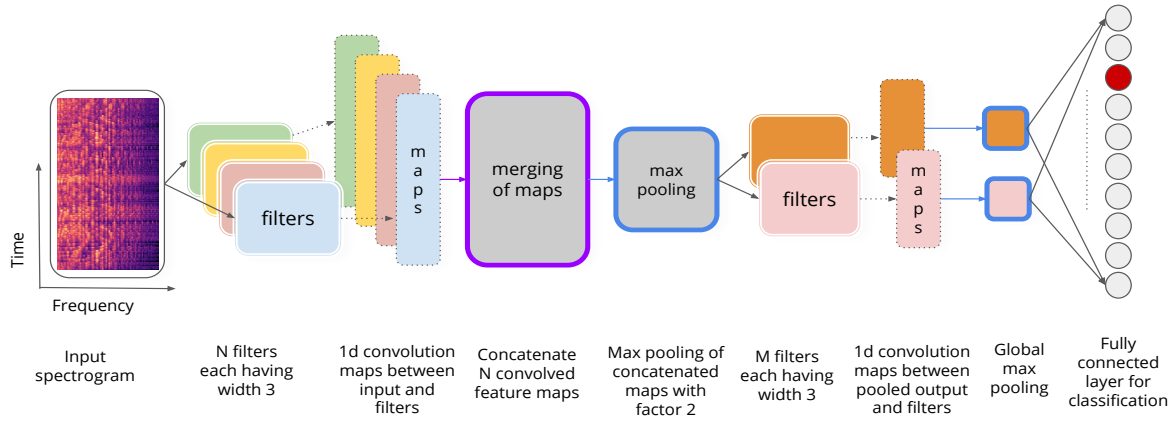
Convolutional neural networks (CNN) are specially designed neural networks for processing data that has a grid-like topology [20]. Introduced in [21] convolutional neural networks have produced excellent results in a wide variety of problems including computer vision [22, 23, 24], speech recognition [25] and natural language processing [26, 27]. Long short term memory (LSTM) networks [28] are also widely used in sequential time series data to capture long term dependencies.

In this work, we apply variants of CNN and CNN-LSTM models for musical genre prediction. Following [26] we use 1D convolution in our models. Here, the extracted features have dimensions of $(646, k)$ (Section 2.1), and our convolutional filters have dimension of $(3, k)$. The 1D convolution operation is performed by sliding the filters over the 646 windowed time-steps. The operation is denoted as 1D convolution because the convolutional filters and the features have same length and hence the sliding of the filters are performed only over the width (time dimension) of the features.

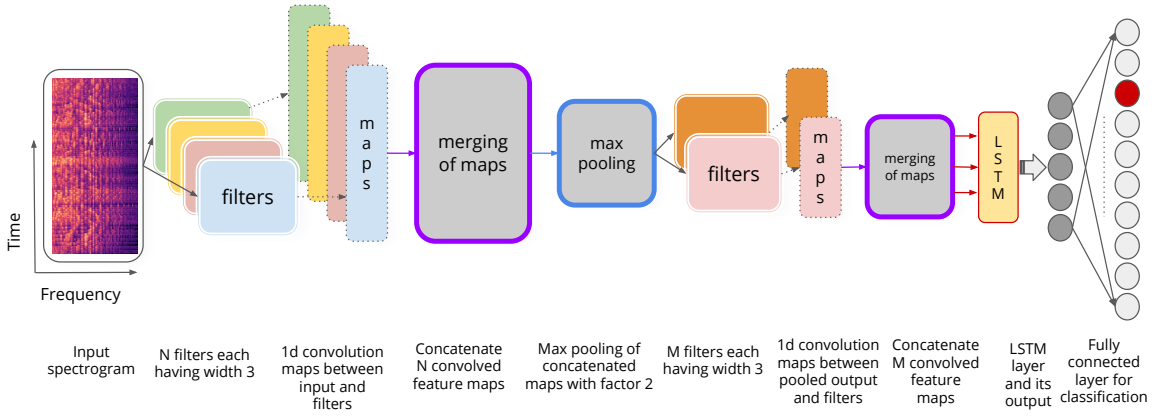
In total, we apply four different CNN and CNN-LSTM models on all the extracted two dimensional features separately to predict the genre of the song. Structure of these models are outlined in Fig. 2. For the two different kinds of one dimensional vectors we use two separate multilayer perceptron (MLP) models for genre prediction. We briefly describe configurations of these models below.

• ***CNN Max Pooling Model***: A two layer deep CNN model is used with 128 and 64 filters (width 3) respectively in the two layers. Between these two layers max pooling is performed with factor two. After the second convolution layer, global max pooling is used to create a representational vector. This vector is then used in a fully connected layer to create the output genre.

• ***CNN Max Pooling LSTM Model***: This model is similar to



(a) *CNN Max Pooling Model*



(b) *CNN Max Pooling LSTM Model*

Figure 2: *a) CNN Max Pooling and b) CNN Max Pooling LSTM models for mel spectrogram features. For CNN Average Pooling and CNN Average Pooling LSTM models, the max pooling and global max pooling functions are replaced with average pooling and global average pooling functions respectively. We use the same model configuration for all the other extracted features.*

CNN Max Pooling Model with the exception of a LSTM layer being used after the second convolutional layer instead of global max pooling. The final hidden state of the LSTM network is used in the fully connected layer for genre prediction.

• ***CNN Average Pooling Model***: The max and global max pooling functions in the *CNN Max Pooling Model* are replaced with average and global average pooling.

• ***CNN Average Pooling LSTM Model***: The max pooling between the convolutional layers in the *CNN Max Pooling LSTM Model* is replaced with average pooling.

• ***Multilayer Perceptron (MLP) Model***: The input to this network is a one-dimensional feature vector. A single hidden layer with 256 nodes is used. The output layer has 10 nodes corresponding to 10 different genres.

For all the models we use ReLU [29] activation in the hidden layers and softmax activation in the output layer. The models are trained with Adam [30] optimizer. 25% dropout [31] is applied in the fully connected layers for regularization.

3. Experiments, Results and Discussion

The GTZAN dataset consists of 1000 audio tracks each being 30 seconds long. All the tracks are 22050Hz mono 16-bit audio files in .wav format. It contains 10 genres of songs - blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae & rock. Each genre is represented by 100 tracks. We evaluate our models in this ten class classification framework. We run our experiments in a 10 fold cross validation setup. We maintain the uniform distribution of musical genres in each fold i.e. there are 80 songs of each genre in the train split and 20 songs of each genre in the validation split for each fold.

The average 10 fold accuracy score of our models are reported in Table. 1. A number of interesting observations can be made from the results. First of all, we observe that the best result is obtained by the multilayer perceptron model when used with music transfer learning features. This result can be expected as the original system was trained on the very large Million Song Dataset [11] containing rich label sets for various aspects of music including *mood*, *era*, *instrumentations* and most importantly *genre*. Also further fine-tuning was performed on our experimental setup leading it to produce the best results. We also observe that the mel spectrogram features produces best results in *CNN Max Pooling* and *CNN Average Pooling* models, whereas mel coefficients produces best results for *CNN Max Pooling LSTM* and *CNN Average Pooling LSTM* models.

The introduction of LSTM resulted in improved perfor-

Features & Models	CNN Max Pooling	CNN Max Pooling LSTM	CNN Average Pooling	CNN Average Pooling LSTM	Multilayer Perceptron
Mel Spectrogram	**83.0**	73.6	**82.5**	75.7	-
Mel Coefficients	80.2	**79.0**	81.6	**80.5**	-
Delta Mel Coefficients	70.4	77.2	74.5	77.0	-
Double Delta Mel Coefficients	72.1	72.9	72.1	76.5	-
Energy Normalized Chromagram	45.7	34.5	43.0	36.2	-
Constant Q Chromagram	60.0	49.4	57.5	45.6	-
STFT Chromagram	62.8	52.5	63.4	53.7	-
Tonnetz Features	50.2	53.5	51.0	55.8	-
Tempogram Features	41.5	42.0	41.6	43.3	-
Averaged Signal Features	-	-	-	-	77.1
Transfer Learning Features	-	-	-	-	**85.5**

Table 1: *Average 10 fold cross validation accuracy scores for different features and models.*

mance for delta coefficients, double delta coefficients, tonnetz features and tempogram features. The performance of max pooling and average pooling is somewhat consistent across all the feature sets. In some cases max pooling performs better, whereas in other cases average pooling performs better.

This heterogeneous and complementary characteristics of the models led us to build an ensemble model which effectively improves the performance by combining the outputs of all the base systems. Our ensemble model is a simple majority voting ensemble of all the deep learning and multilayer perceptron models; e.g. for a particular song, a number of genre predictions will be available from the base models. The genre which is predicted with most frequency will be the final assigned genre. If multiple genres are predicted with highest frequencies then the final decision is made based on the predicted softmax probabilities. By incorporating this simple rule, we were able to get a large improvement in performance as reported in Table 2.

4. Comparative Analysis

Table 2 also shows results of our proposed model compared to other state-of-the-art systems. Grzegorz and Grzywczak [6] reported an accuracy of 78.0%. They extracted features from spectrograms using a deep convolutional network trained for image classification and finally used a SVM for genre prediction. Baniya et al. [32] reported scores of 87.9% using rich statistics and low-level music features. In [7] authors used a transfer learning system trained for music tagging to extract features for genre prediction. They reported scores of 89.8% by taking features from multiple layers of the transfer CNN model. Arabi and Lu [33] reported an accuracy of 90.79 % using a SVM classifier over selected combination of high and low level musical features. In [4] Panagakis et al. used rich, psychophysiologically inspired properties of temporal modulations of music with a sparse representation based classifier to achieve accuracy score of 91.0 %. Mostly pitch, temporal and timbre features were used with non negative matrix factorization as a feature reduction technique. Works by the same authors in [5] further increases the score to 93.7% by the utilization of topology preserving non-negative tensor factorization.

Our ensemble system of *CNN Average Pooling* and *MLP* models achieves an accuracy score of 94.2 %, which is at least 0.5% more than the rest of the comparative systems. One important aspect to note here is the work by Sturm B. L. in [34].

Models	Accuracy Score
Ensemble Models	
CNN Max Pooling & MLP	93.6
CNN Max Pooling LSTM & MLP	91.5
CNN Average Pooling & MLP	**94.2**
CNN Average Pooling LSTM & MLP	91.4
Comparison with state-of-the-art systems	
Grzegorz and Grzywczak [6]	78.0
Baniya et al. [32]	87.9
Choi et al. [7]	89.8
Arabi and Lu [33]	90.8
Panagakis et al. [4]	91.0
Panagakis et al. [5]	93.7
Proposed System	**94.2**

Table 2: *Ensemble models and comparative results with other state-of-the-art systems.*

With rigorous examples and case studies, it is demonstrated that the perfect system in the GTZAN dataset would not be able to surpass the accuracy score of 94.5% due to the inherent noise in the some of the repetitions, mis-labelings and distortions of the songs. Interestingly, our proposed system achieves accuracy of 94.2%, an almost perfect score.

5. Conclusion

In this work we proposed a novel approach for music genre recognition. Firstly variants of CNN and CNN-LSTM based models are trained on a variety of spectral and rhythmic features. Secondly, a MLP network is trained on extracted representational features from a transfer learning system trained for music tagging. Finally, these models are combined in a majority voting ensemble setup. With our experiments we showed that the ensemble model is effective in greatly improving the performance. Our proposed model outperforms the current state-of-the-art systems and achieves a near perfect score for musical genre recognition in the GTZAN dataset.

6. References

[1] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on speech and audio processing*, vol. 10, no. 5, pp. 293–302, 2002.

[2] C. Xu, N. C. Maddage, X. Shao, F. Cao, and Q. Tian, "Musical genre classification using support vector machines," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, vol. 5. IEEE, 2003, pp. V–429.

[3] Y. M. Costa, L. S. Oliveira, A. L. Koericb, and F. Gouyon, "Music genre recognition using spectrograms," in *Systems, Signals and Image Processing (IWSSIP), 2011 18th International Conference on*. IEEE, 2011, pp. 1–4.

[4] Y. Panagakis, C. Kotropoulos, and G. R. Arce, "Music genre classification via sparse representations of auditory temporal modulations," in *Signal Processing Conference, 2009 17th European*. IEEE, 2009, pp. 1–5.

[5] Y. Panagakis and C. Kotropoulos, "Music genre classification via topology preserving non-negative tensor factorization and sparse representations," in *Acoustics speech and signal processing (ICASSP), 2010 IEEE international conference on*. IEEE, 2010, pp. 249–252.

[6] G. Gwardys and D. Grzywczak, "Deep image features in music information retrieval," *International Journal of Electronics and Telecommunications*, vol. 60, no. 4, pp. 321–326.

[7] K. Choi, G. Fazekas, M. B. Sandler, and K. Cho, "Transfer learning for music classification and regression tasks," in *Proceedings of the 18th International Society for Music Information Retrieval Conference, ISMIR 2017, Suzhou, China, October 23-27, 2017*, 2017, pp. 141–149. [Online]. Available: https://ismir2017.smcnus.org/wp-content/uploads/2017/10/12_Paper.pdf

[8] C. Senac, T. Pellegrini, F. Mouret, and J. Pinquier, "Music feature maps with convolutional neural networks for music genre classification," in *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing*. ACM, 2017, p. 19.

[9] N. Scaringella, G. Zoia, and D. Mlynek, "Automatic genre classification of music content: a survey," *IEEE Signal Processing Magazine*, vol. 23, no. 2, pp. 133–141, 2006.

[10] B. L. Sturm, "A survey of evaluation in music genre recognition," in *International Workshop on Adaptive Multimedia Retrieval*. Springer, 2012, pp. 29–66.

[11] T. Bertin-Mahieux and D. P. Ellis, "The million song dataset." in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR)*, 2011.

[12] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," in *Readings in speech recognition*. Elsevier, 1990, pp. 65–74.

[13] M. Müller, F. Kurth, and M. Clausen, "Audio matching via chroma-based statistical features." in *Proceedings of the 6th International Conference on Music Information Retrieval (ISMIR)*, 2005.

[14] F. Kurth and M. Muller, "Efficient index-based audio matching," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 382–395, 2008.

[15] M. Müller and S. Ewert, "Chroma toolbox: Matlab implementations for extracting variants of chroma-based audio features," in *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR), 2011. hal-00727791, version 2-22 Oct 2012*. Citeseer.

[16] J. C. Brown, "Calculation of a constant q spectral transform," *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991.

[17] C. Harte, M. Sandler, and M. Gasser, "Detecting harmonic change in musical audio," in *Proceedings of the 1st ACM workshop on Audio and music computing multimedia*. ACM, 2006, pp. 21–26.

[18] P. Grosche, M. Müller, and F. Kurth, "Cyclic tempogram mid-level tempo representation for musicsignals," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 5522–5525.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[20] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[21] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[23] S. Bhatnagar, D. Ghosal, and M. H. Kolekar, "Classification of fashion article images using convolutional neural networks," in *Image Information Processing (ICIIP), 2017 Fourth International Conference on*. IEEE, 2017, pp. 1–6.

[24] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[25] O. Abdel-Hamid, A.-r. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on audio, speech, and language processing*, vol. 22, no. 10, pp. 1533–1545, 2014.

[26] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[27] M. S. Akhtar, A. Kumar, D. Ghosal, A. Ekbal, and P. Bhattacharyya, "A multilayer perceptron based ensemble technique for fine-grained financial sentiment analysis," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 540–546.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[29] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[32] B. K. Baniya, J. Lee, and Z.-N. Li, "Audio feature reduction and analysis for automatic music genre classification," in *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 457–462.

[33] A. F. Arabi and G. Lu, "Enhanced polyphonic music genre classification using high level features," in *Signal and Image Processing Applications (ICSIPA), 2009 IEEE International Conference on*. IEEE, 2009, pp. 101–106.

[34] B. L. Sturm, "The gtzan dataset: Its contents, its faults, their effects on evaluation, and its future use," *arXiv preprint arXiv:1306.1461*, 2013.