Literature review: <https://ismir2001.ismir.net/pdf/tzanetakis.pdf>

It is the goal of this work to investigate techniques for automatic genre classification. In this paper, they propose a set of characteristics for capturing the music surface and rhythmic structure of audio signals. In order to evaluate the performance of this feature set, statistical pattern recognition classifiers are trained on audio collections gathered from compact disks, radio stations, and the internet. Audio signals can be automatically categorised using a genre hierarchy that can be represented as a tree with 15 nodes, which can be represented as a tree with 15 nodes. Two graphical user interfaces for exploring and interacting with huge digital music collections have been built on the basis of this automatic genre classification and the features derived from the music files themselves.

Literature review: <http://cs229.stanford.edu/proj2011/HaggbladeHongKao-MusicGenreClassification.pdf>

Due to the heterogeneity of music genres, it is exceedingly difficult to classify them even with the bear ear. This research was conducted in order to classify four major musical genres: pop, classical, metal, and jazz. They processed the audio using an application called Marsyas, which is an open-source software piece of software. Additionally, they have included a database of songs dubbed "GTZAN Genre Collection" in their references. This database contains a total of 1000 songs, 100 of each genre, with each audio file being 16-bit. They've gathered songs from ten different genres that are 30 seconds long and have the music extension .wav. The files have a sample rate of 22050 Hz. The reason for choosing four major genres is that the success rate decreases as the number of genres increases, as indicated in the article. They used 70% of the 400 songs for training and 30% for testing. However, they have pre-processed the music into a.csv file using Python scripts. The process begins by reading the.csv file into Matlab and extracting the MFCC features for each covariance matrix of the cepstral features, which are then stored as a mel-matrix, effectively modeling the frequency features of each song as a multi-variate Gaussian distribution. Their methodology entails reading the first half of the files as waveforms and then extracting frames at intervals of 20 milliseconds. The Fourier Transform is obtained by multiplying a hamming window and a frame. The following step is to convert frequencies to the mel scale. Humans perceive pitch shifts as linear below 1 kHz and logarithmic above 1 kHz, thus this model simulates this. To arrange the frequency components in a more logical order, they use the Discrete Cosine Transform to approximate the Karhunen-Loeve Transform. Higher frequencies are features that make negligible differences to human perception and offer less information about the music, thus they maintain the top 15 of these 20 frequencies. They used Mel Frequency Cepstral Coefficients (MFCC) to describe their data and apply machine learning algorithms, as earlier research had suggested. By the representation of waveform as a matrix of cepstral features gives them a vector of 15 cepstral frequencies for the number of frames exists in the song. They further compress this matrix representation by saving the mean vector and covariance matrix of the cepstral characteristics across each 20ms frame as a cell matrix. Modeling the frequencies as a multivariate Gaussian distribution further reduced the processing needs for KL Divergence comparisons. The Kullback-Lieber (KL) Divergence is a crucial computation in their k-NN training that is used to determine the distance between two songs, the article mentions. It's worth noting that they obtain four-dimensional standard orthonormal base vectors with each value signifying a genre (classical, jazz, metal, or pop) and each value having a value of either 1 or 0. They preprocess the input data by combining the mean vector and the top half of the covariance matrix into a single feature vector, stemming in 15+ (15 + 1) \* 15/2 features for each song. The proportions of training, validation, and testing data are 70, 15, and 15, respectively. They obtain the highest accuracy of 97 percent with classical and pop music and the lowest accuracy of 67 percent with jazz music using DAG SVM. The greatest score obtained using the k-Means algorithm is 93 percent accuracy for Metal, while the poorest score obtained using the k-Means algorithm is approximately half for Jazz. Other genres fared rather well, with higher than or equal to 88 percent. Jazz is the most difficult genre to detect using k-NN, with an accuracy of nearly two-thirds. Other accuracies above 80. Finally, but certainly not least, NN outperformed all other models. Accuracy for Jazz and Pop was 100 percent, but Metal and Classical were 76 and 88 percent, respectively, making metal the most challenging genre. To further conduct their research, they have transferred images to genres. They have collected images that seemed similar visually, e.g., nature images for the classification of the classical tracks. Features extracted from the images helped to transfer them to Fourier-Mellin 2D (FMT). By using the k-Means clustering with the data obtained from FMT. Each of the generated picture clusters was linked to a genre, such that a song's genre and a random image in the corresponding image cluster could be mapped together. There were some intriguing outcomes created by their music-to-picture mapping tool. Songs like Lady Gaga's Poker Face were appropriately classified as Pop by our system. According to the authors, when they mapped the pop genre to a random picture from its related image cluster, they got a pretty fair match.