National College of Ireland

Visual Harmonies: Investigating the Impact of Album Cover Image Features on Music Genre Image Classification for Top Spotify Artists in the USA

MSc Research Project
Artificial Intelligence

Cian McGrane
Student ID: 19181931

School of Computing
National College of Ireland

Supervisor: Sheresh Zahoor

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Cian McGrane
Student ID:	19181931
Programme:	MSc. in Artificial Intelligence **Year:** 2024
Module:	Research Practicum/Internship Part 2
Supervisor:	Sheresh Zahoor
Submission Due Date:	16th August 2024
Project Title:	Visual Harmonies: Investigating the Impact of Album Cover Images Features on Music Genre Classification for Top Spotify Artists in the USA
Word Count:	7, 201 **Page Count**: 20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: 

Date: 11th August 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Visual Harmonies: Investigating the Impact of Album Cover Images Features on Music Genre Classification for Top Spotify Artist in the USA

Cian McGrane
19181931

Abstract

This research explores the potential of image classification models to identify music genres from album cover features, using data from the most popular Spotify artists in the United States. This research compares the efficacy of the k-nearest neighbour (KNN) and support vector machines (SVM) models in classifying genres based on two types of features: most dominant colour (MDC) and pixel intensity histograms (PIH). Data was sourced from a Kaggle dataset containing artist information and the Spotify API for gathering album cover images. The MDC features were extracted using k-means clustering, while PIH features were derived from pixel intensity distributions. Both features were transformed into 3x3 RGB images to standardise the input for the models. The KNN model achieved an overall accuracy of 63% with MDC features and 62% with PIH features, while the SVM ensemble model, combining MDC and PIH features, demonstrated superior performance with 75% accuracy, Key findings reveal that image features can effectively distinguish music genres, offering a novel approach to music classification. Future work will focus on refining feature extraction techniques and exploring the potential for commercial applications in music recommendation systems.

Keywords: Image classification, feature extraction, music, album artwork, KNN, SVM

1 Introduction

The convergence of music and visual art, particularly through the medium of album covers, has long served as a compelling and influential aspect in the music industry. Album covers not only serve as visual representations of an artist's work but also convey subtle cues about the music's style and genre. As the music streaming landscape evolves, the need for accurate and efficient methods of categorising music genres becomes increasingly essential to enhancing users experience on music streaming platforms.

This research explores the fusion of music and visual features by investigating the impact of album cover image characteristics on the accuracy of music genre classification models. By focusing on the most popular Spotify artists in the United States, this research aims to provide valuable insights into the interconnectedness between visual elements used on album covers and musical genres. This research will contribute to the broader understanding of the symbiotic relationship between visual aesthetics and music on contemporary digital platforms. Furthermore, this project addresses a pertinent gap in the research but also holds practical implications for refining the categorisation of music genres in streaming services, thereby

offering an innovative perspective on the role of visual content in enhancing the user experience.

There has been an extensive amount of research conducted in the area of music classification. Various models and types of data have been trialled and tested over the past number of years. (Siddiquee et al. 2023), (Jang 2023), (Masruroh et al. 2023), (Sai & Kalaiarasi 2023), and (Wang et al. 2023) have all used the GTZAN dataset for music genre classification research. This dataset consists of audio files and was created in the early 2000's and has been used for several research projects in 2023. The research carried out by (Datta et al., 2021) and (Yuwono et al., 2023) used the musical metrics from the Spotify API in their research. This research will continue to using the Spotify API but will solely focus on the image data available from the API.

This research proposes to break away and reconsider current music classification trends and introduce novel concepts to this field of research. Below is a list of the new contributions this research is proposing.

1. Create a new modern dataset containing the most dominant colour (MDC) and pixel intensity histogram (PIH) features extracted from the album cover images.
2. Use features that are not widely used in music genre classification problems.
3. Enhance existing methodology to enable classification with MDC and PIH features.
4. Further improve the model performance by combining MDC and PIH features.
5. Explore new approaches for analysing and extrapolating meaning from visual representations of music, in addition to encouraging innovation and creativity in both research and artistic practices.

The above research problem prompts the following research questions: How can image classification models harness album cover features such as MDC, and PIH to identify music genres from the most popular artists on Spotify in the United States of America? How does the accuracy of each image classification model, support vector machines (SVM), and k-nearest neighbour (KNN) compared to one another for each feature? Can combining these features increase the overall model performance?

This document is structured into five sections. The first section, Introduction, outlines the motivation for the proposed research and briefly highlights some of the previous work in this area. The second section, Literature Review, analyses previous research in this field under several headings. The third section, Research Methodology, provides a detailed account of the research methods employed. It also offers a description of the model design. Furthermore, it presents an overview of the outputs produced, the models utilised, and the technology employed. The fourth section, Evaluation, assesses the results of the models. The fifth and final section, Conclusion and Future Work, outlines potential future research directions and provides concluding remarks.

As album cover images serve as a crucial yet underexplored facet of music genre classification, this research departs from traditional methods that rely heavily on audio data and introduces novel visual features—MDC and PIH—as significant classifiers. In line with this approach, it is essential to review the existing literature on music classification, particularly the integration of visual data. The following section examines prior research efforts, focusing on three key areas: image classification, dataset selection, and classification models. This will position the current research within the broader context and highlight how it addresses gaps in the field.1

2 Related Work

There have been many studies published regarding music classification This section will outline the different datasets, models, and methods used for music classification will be outlined. It will also critically analyse the work completed by others and highlight how this research will add to this work. Examination will be undertaken in three areas; Image Classification, Data Selection, and Classification Models

2.1 Image Classification

(Marcellus et al., 2021), (Prathyushaa et al., 2021), (Rao & Kulkarni, 2017), (Koenig, 2019), and (Dammann & Haugh, 2017) all conducted research in the area of image classification.

(Marcellus et al., 2021) uses data consisting of movie posters to train and evaluate a convolutional neural network (CNN) model. The research classifies movie genres into five categories: action, comedy, fantasy, horror, and romance. Colours from the movie posters were extracted in red, green, blue (RGB) format. A specific algorithm, such as k-means clustering that can be used to extract colours from images, is not identified. The colour values were adjusted to account for the light in the cinema lobby. This combined with multi-labelled posters could impact the model's performance. The constant used to neutralise lighting conditions might not be optimal for all scenarios.

(Prathyushaa et al., 2021) discusses developing and evaluating models for converting western music notation to its Carnatic music equivalent. The research uses image processing models with transfer learning and classification techniques to achieve this. Images of sheet music are used to train the three different CNN models, ResNet50, VGG16, and Inception V3. A simple CNN is used as a baseline for performance metrics. (Prathyushaa et al., 2021) models offer practical solutions for musicians and researchers in the field of music notation image conversion. The research lacks details on specific configurations and hyperparameters used for training the model. Information on batch size, learning rate, optimiser settings and epochs will provide a clearer understanding of the training processes involved.

(Rao & Kulkarni, 2017) provides an overview of the challenges and existing techniques in leaf plant image classification. It emphasises the effectiveness of image enhancement, feature extraction, and deep neural network classification. (Rao & Kulkarni, 2017) enhance the leaf images by applying an entropy maximisation technique and auto-regressive modelling. The entropy maximisation technique can make images more distinguishable and informative. Auto-

regressive modelling correlates the relationship between pixels in an image, this can improve low-quality images. (Rao & Kulkarni, 2017) identify specific gaps in existing methods, such as the inability of morphological and shape features to handle non-convex problems and the challenges posed by low-quality images. The choice of entropy maximization for image enhancement and the use of specific feature extraction techniques is not fully justified. The model achieved an accuracy score of 96%.

(Koenig, 2019) presents a method for music genre classification using album cover images, focusing on both single-label and multi-label tasks. (Koenig, 2019) uses of a large dataset of 18,584 images for the single-label task. This demonstrates a robust approach to model training and evaluation. The research uses CNN models for tasks, this is an appropriate model to use given its effectiveness in image classification. However, the model's underperformance would suggest that this architecture is not suitable for this task and that a simpler model such as a KNN might have provided better model performance. The research evaluates the performance of the models using performance metrics, including area under the receiver operating characteristic (AUROC), area under the precision recall graph (AUPRC), precision, and recall. The research compares its results to (Oramas et al., 2017), the lack of precision and recall metrics from the baseline limits the depth of this comparison. The research discusses the limitations of the AUROC score due to dataset imbalance. The research notes that this metric is impacted by the number of true negatives in the result's set. It also highlights how a model can be developed with a strong AUROC score but have a poor overall performance. Finally, the research mentions the potential benefits of principle component analysis (PCA) and linear discriminant analysis (LDA) for preprocessing but rejects these methods to maintain interpretability of the learned features. This trade-off may have compromised performance. (Dammann & Haugh, 2017) utilised this methodology and achieved very strong model performance when classifying album artwork.

(Dammann & Haugh, 2017) integrates multiple types of data, lyrics, audio, and images. This is innovative as most research mentioned above focus on a single type of data. The results from the KNN model are very strong with accuracy just over 91%. While the accuracy score is very strong, the research does not evaluate the performance of the model using other metrics such as precision, recall, and F1-score. The final model combining the three models does not significantly improve over the KNN alone. This suggests that the combination approach might need refinement.

2.2 Data Selection

Selecting the appropriate data for a research project is crucial as it determines what information we may learn and how accurate our conclusions will be. The next section will look at the different datasets that appeared in some image classification projects.

2.2.1 GTZAN dataset

In examining previous work in music genres classification research a pattern emerges regarding the dataset that was used. The GTZAN dataset appears in several projects (Alam Siddiquee et

al., 2023), (Jang, 2023), (Masruroh et al., 2023), and (Wang et al., 2023). This dataset consists of audio files from ten different genres, with one hundred songs for each genre. The dataset also includes a visual representation of each file. This dataset was collected between 2000 and 2001, would now be considered outdated. This research project will use a different data source, Spofity API and the most popular Spotify Artists in the US are employed for the purpose of this research.

(Alam Siddiquee et al., 2023) performed a spectrogram analysis on the audio included in the GTZAN dataset. The spectrogram analysis produced images that showed patterns in the audio for the different songs in each genre. The study focused on three genres: classical, pop, and rock. Features are extracted from the images and are converted into statistical data that was used to train the KNN model. (Wang et al., 2023) applied a similar methodology in their research. This research also used the GTZAN dataset. This created Mel Spectrogram images using the decibel from each audio sample. (Jang, 2023) and (Masruroh et al., 2023) also apply the same methodology of creating a Mel Spectrogram images and extracting features that would be used to train models.

The research detailed above all used the GTZAN dataset with the aim of classifying music into genres. Most of the research focused on converting music audio into images and then extracting features from these images. This research applies a similar methodology but will focus on the album artwork images as opposed to creating images using audio. This research deviates from the GTZAN dataset and in lieu uses data sourced from the Spotify API. The following section will focus on the work that used the Spotify API as a data source.

2.2.2 Spotify API

(Datta et al., 2021) and (Yuwono et al., 2023) used data taken from the Spotify API for their respective research projects. Both projects focused on the metrics available for each song in the Spotify API. These metrics include items such as beats per minute, loudness, tempo, and popularity. The research used a correlation matrix to identify which features would be best suited to train their respective models. With the limited research using the Spotify API this project aims to inspire more work to be carried out. The next section will look at the models that are used in this research project and their previous utilisation for music genre classification problems.

2.3 Classification Models

This research proposes to train and assess the performance of the KNN and SVM models and to assess the performance of each model at classifying image features extracted from album artwork into music genres.

2.3.1 K-Nearest Neighbour

(Alam Siddiquee et al., 2023), (Sai & Kalaiarasi, 2023), and (Wu & Liu, 2020) used the KNN model in their research projects. (Alam Siddiquee et al., 2023) achieved an accuracy score of 90% when classifying Mel Spectrogram images into three separate music genres, classical, pop, and rock. (Sai & Kalaiarasi, 2023) trained the model using the soundwave and frequency

metrics from the GTZAN dataset. They achieved a lower accuracy score of 56%. (Sai & Kalaiarasi, 2023) tried to classify more genres than (Alam Siddiquee et al., 2023) which may account for the lower accuracy score. (Wu & Liu, 2020) enhanced the KNN algorithm by creating a double weighted KNN algorithm. With the enhancements to the algorithm, they were able to achieve an average accuracy score of 82%.

2.3.2 Support Vector Machines

(Costa et al., 2012) and (Yuwono et al., 2023) both used the SVM model for their research. (Costa et al., 2012) extracted two features from the spectrogram images that were created. They compared the performance of two different features: grey level co-occurrence matrix (GLCM) and local binary pattern (LBP). The model performed well for both features. They achieved an accuracy score of 70% for GLCM and 80% for LBP. Their study focused on one genre of music, Latin and the subgenres within this. The performance of their model could fall if there were more genres introduced in the training phase. This study focuses on multiple genres which gives a better picture of how the SVM algorithm performs for music genre classification.

(Yuwono et al., 2023) use a correlation matrix to identify which musical features would be best to use for training their SVM algorithm. The musical features they focused on are taken from the Spotify API. Beats per minute, liveness, and energy all had the highest correlation however all of them were less than 0.3. The model's performance was evaluated using four different kernels: linear, RBF, sigmoid, and polynomial. All kernels achieved good accuracy scores ranging from 67% to 78%. This project emulates the work done by (Yuwono et al., 2023) by using the SVM model and two features from album artwork images.

In conclusion, the research papers reviewed indicate advancements in music genre classification through various methodologies and datasets. However, significant gaps remain, particularly when integrating visual features from album artwork. While existing research, such as the work by (Koenig, 2019), has explored the potential of album covers, the underperformance of the CNN models and the limited use of the KNN model indicate the need for more nuanced approaches. Additionally, studies predominantly rely on outdated datasets like GTZAN, which do not reflect current music trends. Furthermore, the potential of data from modern sources like the Spotify API have been under explored. This research aims to fill these gaps by leveraging album cover features such as MDC and PIH sourced from the most popular artists on Spotify in the United States. Comparing the performance of KNNs and SVMs and exploring the potential benefits of combining the MDC and PIH features. This research seeks to enhance the accuracy and effectiveness of image based music genre classification.

3 Research Methodology

This research employs an integrated approach using KNN, SVM, PCA, LDA, feature engineering, and ensemble methods to classify music genres based on image features. The primary techniques include data normalisation, dimensionality reduction, clustering, and hyperparameter tuning. The implementation leverages the scikit-learn library for robust and scalable machine learning operations. Fig. 1 gives a visual representation of the steps taken

from data sourcing to data evaluation. Python was the programming language used for this research. Libraries such as NumPy and Pandas were used for data sourcing and preprocessing. Scikit-learn and Seaborn were utilised for data modelling and data evaluation. The methodology for this project is develops on the research conducted by (Dammann & Haugh, 2017). The implementation of the proposed solution involved several stages data sourcing, data preprocessing, data modelling, and data evaluation. The following section will outline each step in detail and will highlight the additional to the methodology proposed by Dammann & Haugh.

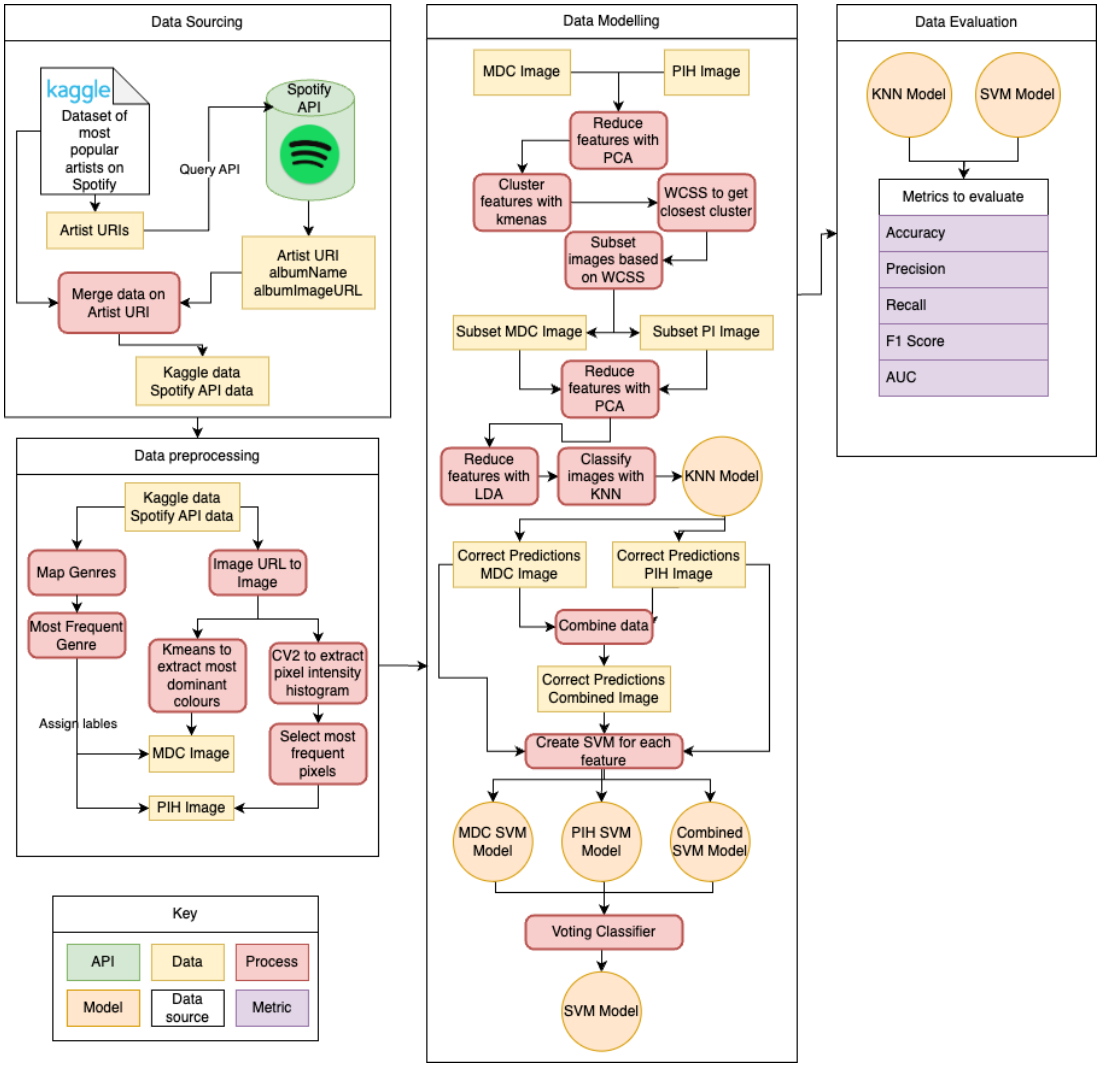

Fig. 1 Process Diagram

3.1 Data Sourcing

The data for this research project has been sourced from two separate locations. A dataset containing a list of the most popular Spotify artists in the United States of America was sourced from Kaggle (Spoorthi U K, 2024). This dataset contains information about the artists such as their name, Spotify artist uniform resource identifier (URI), age, gender, and any genres that are associated with their music. Using the Python package, Tekore (Hildén, 2023), to access the Spotify API a second dataset was created that contains the images of all the albums available for each artist on the Spotify platform. The Spotify artist URI from the first dataset is

used to query the API and gather the relevant information. The two datasets are then merged to create one data source as seen in Fig. 1.

Fig. 2 Resized Album Image Cover

3.2 Data Preprocessing

The Spotify API provides a uniform resource locator (URL) to the album cover image. The first step of the preprocessing involves converting these URL's to images. CV2 is used to read the URLs and save the images to a data frame. The images are resized to 64x64 pixels to ensure consistency within the data. Fig 2 is an example of the album cover after the images are resized. The preprocessing steps for extracting the image features vary. The MDC are the most dominate colours that are present in an image. The k-means clustering algorithm is used to extract the MDC features from each album cover as an RGB value. The algorithm partitions n data points in to k clusters by minimising the within-cluster sum of squares (WCSS). The objective function is:

$$J = \sum_{i=1}^{k} \sum_{x_j \in C_i} \left\| x_j - \mu_i \right\|^2$$

Where: x_j is the data point, μ_i is the centroid of cluster i and C_i represent the set of ponts in cluster i. The centroid are updated as follows:

$$\mu_i = \frac{1}{|C_i|} \sum_{x_j \in C_i} x_j$$

The images are loaded into a matrix where each pixel is represented by its RGB values. For example, the images in this project are a 64x64 image. This image is converted into 4,096x3 matrix, where each row corresponds to a pixel and the three columns correspond to the R, G, and B values. The 3D image matrix (height x width x 3) is reshaped into a 2D matrix (number of pixels x 3). This transformation flattens the images and prepares it for the algorithm. The number of clusters (k) corresponds to the number of dominant colours to extract from the image. The algorithm randomly selects the initial centroids, these are a guess at the most dominant colours. The algorithm then calculates the distance from each pixel to each centroid. The centroids are then recalculated by taking the mean of all the data points assigned to each cluster. This is repeated until the centroids no longer change significantly i.e. the algorithm has converged. These centroids are the mean RGB values of the pixels in each cluster.Fig.3 shows the most dominate colours from Fig. 2. These colours are also visualised in Fig.4.

Fig. 3 Most dominate colours

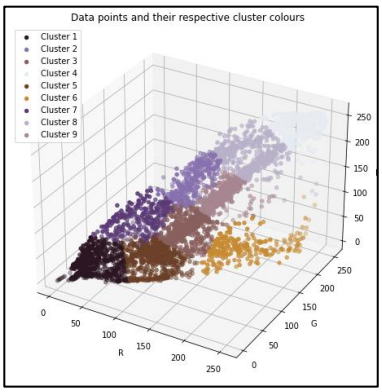

Fig. 4 K-mean cluster for MDC

OpenCV is used to extract the PIH feature for each colour channel; Red, Green, Blue. A PIH is a graphical representation of the distribution of pixel values in an image. It shows how frequently each intensity level occurs in the image as seen in Fig. 5. For the PIH features a data frame is created with the counts for each colour channel with the pixel value also included i.e. 0 to 255. Each row is sorted in descending order, this is the get the pixel values with the highest counts. Fig.6 is an example of an image is created using the 9 most intense pixels from each colour histogram.

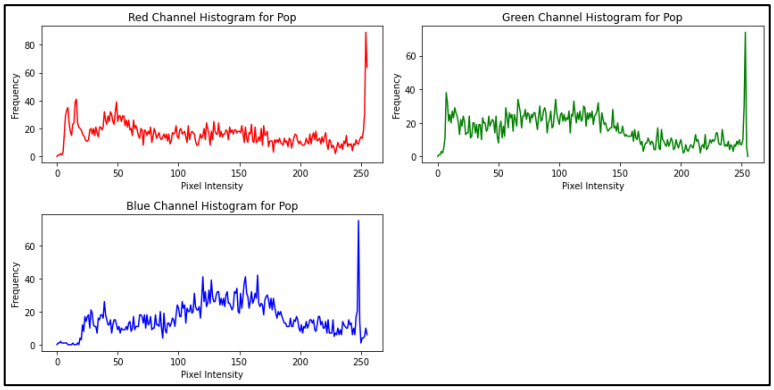

Fig. 5 PIH for RGB channels

Fig. 6 Most frequent PIH

The Spotify API contains lots many sub-genres. Using a mapper these genres are mapped into high level genres such as rock, dance, and hip hop. In the case where there are multiple genres for one album an algorithm was deployed to select the most frequent genre for that album.

3.3 Data Modelling

The MDC features and PIH features undergo a clustering process before they are fed into the KNN model. The clustering process uses the k-means algorithm to identify the images with similar features. The WCSS is used to identify the data points that are the closest to each other. This cluster is selected to be used to train the KNN model. Doing this reduces the noise within the data, increases the efficiency of the training process, and helps highlight key patterns within the data. Each genres is clustered individually before they are combined into a new dataset.

The classes are balanced to ensure there is no bias within the model. Fig.7 and Fig. 8 give a visual representation of how the images in each cluster are cluster based on their feature similarities.

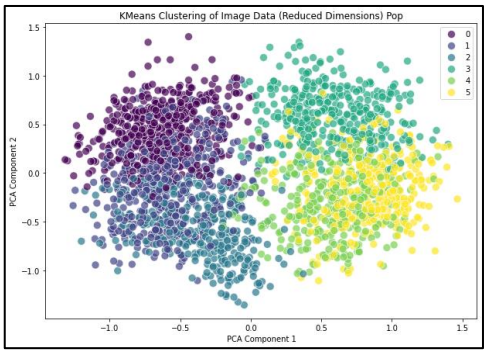

Fig. 7 MDC Cluster

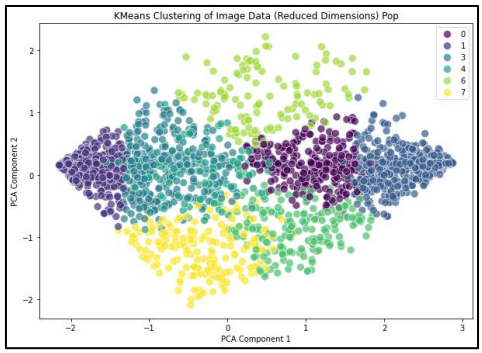

Fig. 8 PIH Cluster

Before the data is fed into the KNN model a PCA and a LDA are performed on the image data. Image data often contains high-dimensional data, especially those with multiple colour channels. PCA can reduce the dimensionality of the image data while retaining most of its variance. This reduction in dimensionality leads to more efficient computations and memory usage during model training. PCA can help in removing noise or irrelevant features from the image data. PCA and LDA can extract meaningful features from the image data. PCA identifies the directions along which the data varies the most, while LDA focuses on finding the directions that maximise class separability. There are four steps to compute PCA: Standardization (Z_{ij}), Covariance Matrix (C), Eigen Decomposition (Ce_i) and Principal Components (PC).

$$Z_{ij} = \frac{X_{ij}-\mu_j}{\sigma_j} \qquad C = \frac{1}{n-1}Z^T Z \qquad Ce_i = \lambda_i e_i \qquad PC = ZE$$

Where: X_{ij} is the ι-th observation of the j-th variable. μ_j is the mean of the j-th variable. σ_j is the standard deviation of the j-th variable. Z^T is the transpose of the standardised data matrix. n is the number of observations. λ_i are the eigenvalues. e_i are the eigenvectors. PC is the matrix of principle components. E is the matrix of eigenvectors.

The LDA comprises of four steps. These are: calculate the mean vectors of each class (μ_k), compute the within-class scatter matrix (S_W), compute the between-class scatter matrix (S_B) and solve the generalised eigenvalue problem for the matrix ($S_W^{-1}S_B w$).

$$\mu_k = \frac{1}{n_k}\sum_{i=1}^{n_k} \mathrm{X}_i \qquad S_W = \sum_{k=1}^{K}\sum_{i=1}^{n_k}(\mathrm{X}_i - \mu_k)\,(\mathrm{X}_i - \mu_k)^T$$

$$S_B = \sum_{k=1}^{K} n_k\,(\mu_k - \mu)(\mu_k - \mu)^T \qquad S_W^{-1}S_B w = \lambda w$$

Where: μ_k is the mean vector for class k. n_k is the number of samples in class k. $\mathrm{X}_i$are the feature vectors. $K$is the number of classes. μ is the overall mean vector of the dataset. w are the eigenvectors (linear discriminants). λ are the eigenvalues.

KNN is a non-parametric algorithm, meaning it does not make any assumptions about the underlying distribution of the data. This flexibility allows KNN to handle complex data distributions, making it suitable for image data, which can be highly variable and non-linear. KNN does not require a training phase. Instead, it memorises the entire training dataset. This characteristic makes it easy to implement and deploy for image classification. KNN makes predictions based on the majority class among the k-nearest neighbours of a given query instance. In image classification, similar images are likely to belong to the same class. Therefore, by considering the labels of neighbouring images KNN can make localised and intuitive decisions. KNN is robust to noisy data because it relies on local information rather than assuming a global decision boundary. Outliers or noisy instances are less likely to significantly impact the classification decision as long as the majority of the nearest neighbours belong to the correct class. Since KNN does not make any assumptions about the underlying data distribution, it can handle complex patterns and relationships in image data without the need for feature engineering or transformation. The formula for calculating the distance between points in KNN takes the following form:

$$d(X, X_i) = \sqrt{\sum_{j=1}^{n} \left(x_j - x_{ij}\right)^2}$$

Where: $d(X, X_i)$ is the distance between test point X and each point X_i. x_i and x_{ij} are the j-th features of the test point and the training point. n is the number of features.

The correct predictions from the KNN model are then used to train a SVM model to see if the accuracy of classification can be improved. The SVM was selected as it robust to overfitting. SVMs aim to maximise the margin between classes, which helps prevent overfitting, especially when dealing with high-dimensional data like images. By maximising the margin SVMs promote generalisation to unseen data leading to better performance. SVMs can efficiently handle non-linear decision boundaries through the use of kernel functions. By mapping the input data into a higher-dimensional space, SVMs can find linear decision boundaries in the transformed space effectively capturing complex relationships in the data. SVMs aims to find the decision boundary that maximises the margin while minimising classification errors. This global optimisation objective leads to robust and stable solutions, ensuring that the resulting classifier is less sensitive to local optima and noise in the data. The optimisation problem for an SVM can be expressed as:

$$\min_{w,b} \frac{1}{2} \|w\|^2 \; subject\ to\ y_i(w.x_i + b) \geq 1, \forall i$$

Where: w is the weight vector. b is the bias term. x_i are training samples. $y_i \in \{-1,1\}$.

SVMs typically use only a subset of the training instances, known as support vectors, to define the decision boundary. SVMs perform well with small to medium-sized datasets. Before the data is passed into the SVM model the data is augmented using the synthetic minority over-

sampling technique (SMOTE) to address class imbalances by generating synthetic samples. The formula for generating a synthetic sample is:

$$x_{new} = x_i + \lambda(x_j - x_i)$$

Where: x_i is a minority class sample. x_j is one of the k-nearest neighbours of x_i. λ is a random number in the range $[0,1]$.

Recursive feature elimination (RFE) is used to select the most significant features. It uses a logistic regression estimator to select the top ten features. The RFE process involves repeatedly fitting the model, ranking the features based on their importance, and eliminating the least important feature until the desired number of features is obtained. The formula for each step is as follows:

$$M\ is\ fitted\ on\ X = \{x_1, x_2 \dots x_p\} \qquad w_i = |\beta_i|\ for\ each\ feature\ in\ x_i$$

$$x_k = arg\ \min_i x_i \qquad X' = X \backslash \{x_k\}$$

Where: M is the model (SVM). p is the total number of features. w_i is the importance score of feature x_i. β_i is the coefficient of feature x_i. x_k is the feature with the lowest importance score.X' is the updated set of features after remove the least important feature.

SVMs can handle datasets with fewer samples by effectively maximising the margin and generalising well to unseen data, leading to reliable performance even with limited training examples. An ensemble model is constructed using a voting classifier, combining individual SVMs trained on different feature sets, combined, MDC, and PIH. The 'soft' voting mechanism aggregates the predictions of each SVM, enhancing classification accuracy and robustness. The formula for the ensemble model is as follows:

$$\hat{y}_{ensemble}(x) = arg\ \max_k \left(\frac{1}{N} \sum_{i=1}^{N} \hat{p}_{i,k}(x) \right)$$

Where: $\hat{p}_{i,k}(x)$ is the predicted probability for class k from the i-ith model. N is the total number of models in the ensemble.

Additional techniques for improving model performance are used during this process. Hyperparameter tuning involves searching for the optimal combination of model hyperparameters, such as neighbour, weights, and leaf size for KNN model. C, kernel, and gamma are assessed for the SVM model. Grid search or random search are used to help fine-tune the model to achieve better performance on the validation set.

3.4 Data Evaluation

The features listed above are utilised to two classification models, with genre labels assigned based on the artist depicted in the album artwork. The performance of each model is assessed. Metrics such as their accuracy, precision, recall, and F1-score are examined. The evaluation

aims to provide insights into the effectiveness of different image features in enhancing the accuracy of music genre classification models for the most popular Spotify artists in the United States of America.

Upon establishing the methodology, the next step involves applying the proposed models to the dataset and measuring their performance. The evaluation process will rely on a set of metrics to assess the effectiveness of the baseline and proposed models. These metrics include accuracy, precision, recall, F1-score, and AUC-ROC, will provide an understanding of the model's ability to classify album cover images across various genres. In the following section will examine these evaluation results and compare the performance of the baseline and proposed models to validate the improvements achieved through the methodology.

4 Evaluation

The section will evaluate the results from the baseline model and compare them to the results of the proposed model. The metrics that are used to evaluate the models and an analysis of the data use are presented below.

4.1 Metrics, Data, and Models

The metrics that were uses to evaluate the performance of the models are as follows:

Accuracy – The proportion of correctly classified instances, composing both true positive and true negative outcomes, relative to the total number of instances evaluated.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Precision – This is the proportion of true positive predictions to the total predicted positives instances, reflecting the exactness of the positive predictions.

$$Precision = \frac{TP}{TP+FP}$$

Recall – The ratio of true positives predictions to the actual number of positive instances, reflecting the model's ability to identify positive instances.

$$Recall = \frac{TP}{TP+FN}$$

F1-score – The harmonic mean of precision and recall, providing a single metric that balances the trade-off between precision and recall.

$$F1\text{-}score = 2\ x\ (\frac{Precision\ x\ Recall}{Precision+Recall})$$

AUC-ROC – This is the area under the receiver operating characteristics curve, this indicates the model's ability to distinguish between positive and negative classes.

$$AUC\text{-}ROC = \int_0^1 TPR\left(FRP^{-1}(x)\right)dx$$

Where: TP = True Positives, TN = True Negatives, FP = False Positives, FN = False Negatives, TPR = True Positive Rate, FPR = False Positive Rate

4.1.1 Exploratory Data Analysis

The data used to train the models consisted of album cover images sourced from the Spotify API, with artist information from a Kaggle (Spoorthi U K, 2024) dataset of popular Spotify artists in the United States. After the images are gathered from the Spotify API there are a total of 86,451 images in the dataset. The distribution of the images for each genre can be seen in Table 2.

Table 2: Distribution of Albums by Genre

Genre	*Number of Albums*
Pop	32,389
Hip hop	28,712
Dance	6,337
Rock	6,134
International	5,905
Latin America	4,525
Reggae	2,458

As the focus of the research project is based on the most popular music on Spotify. The popularity score from Spotify is mainly based on the number of streams an album has and how recent those streams are. Table 3 contains the average popularity score for each genre. The highest popularity score is also included in Table 3.

Table 3: Average and Highest Popularity Score per Genre

Genre	*Highest*	*Average*
Pop	100	53
Hip hop	95	54
Reggae	95	64
Latin America	88	54
Dance	86	57
Rock	85	61
International	81	54

For feature extraction, two main techniques were employed. The MDC features were derived using the k-means clustering algorithm to extract the nine most dominate colours from each album cover, represented as RGB values. Similarly, PIH features were generated by creating pixel intensity histograms for each colour channel, highlighting the distribution of pixel values in the images. Both sets of features were then covered into 3x3 images with three channels corresponding to the RGB values, transforming the extracted data into a format suitable for input into the KNN and SVM models. This transformation allowed for consistent, structured representation of the image data, facilitating efficient training and robust classification performance across various genres. With the goal in mind of seeing if it is possible to classify album images using the MDC and PIH features. Fig. 9 to Fig. 22 below shows that there is difference in MDC and PIH from genre to genre.

Fig. 9 MDC Dance

Fig. 10 MDC Hip hop

Fig. 11 MDC International

Fig. 12 MDC Latin America.

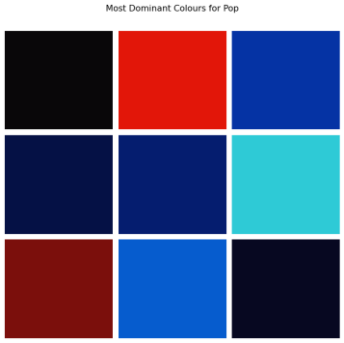

Fig. 13 MDC Pop

Fig. 14 MDC Rock

Fig. 15 MDC Reggae

The PIH graph also show difference from genre to genre. These graphs plot the number of occurrences of one pixel value within an image.

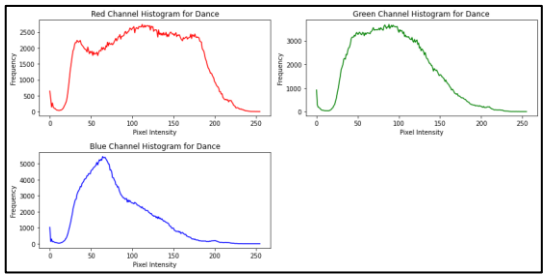

Fig. 16 PIH Dance

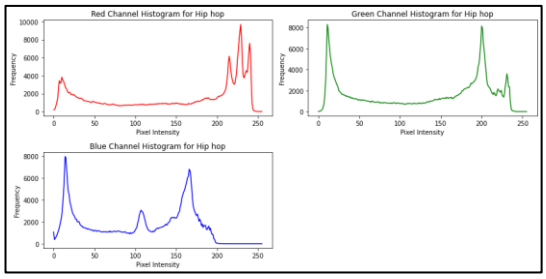

Fig. 17 PIH Hip hop

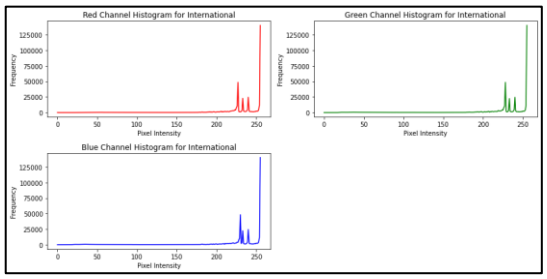

Fig. 18 PIH Dance

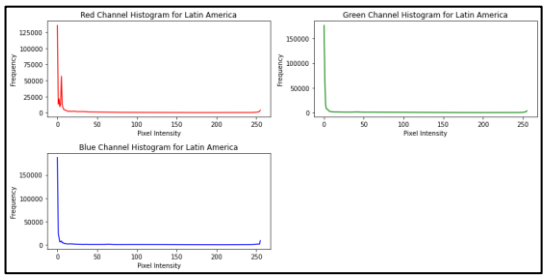

Fig. 19 PIH Hip hop

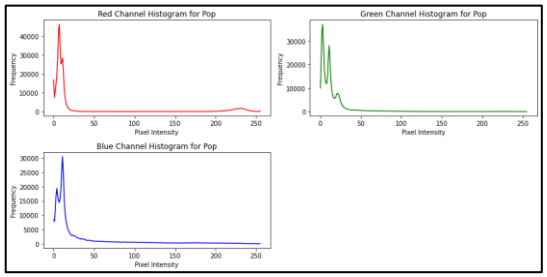

Fig. 20 PIH Pop

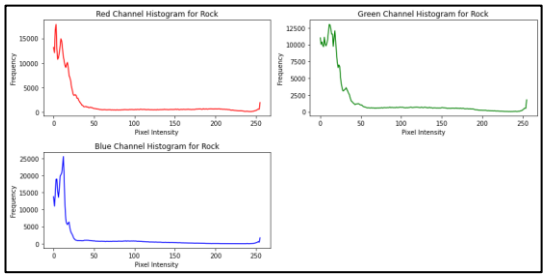

Fig. 21 PIH Rock

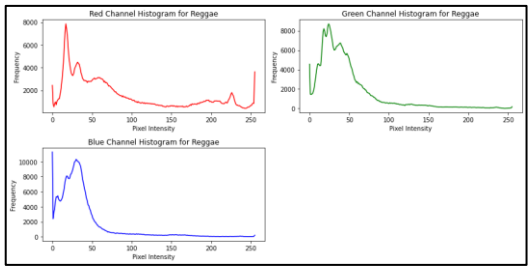

Fig. 22 PIH Reggae

4.1.2 Baseline Performance

Table 4: Results for MDC for KNN model

Genre	*Precision*	*Recall*	*F1-Score*
Hip hop	0.13	0.31	0.19
Latin America	0.14	0.23	0.17
Pop	0.18	0.18	0.18
Reggae	0.12	0.09	0.10
Rock	0.15	0.07	0.10
Dance	0.13	0.05	0.08
International	0.14	0.06	0.08
AUC			0.49
Accuracy			0.14
Macro Avg.	0.14	0.14	0.13
Weighted Avg.	0.14	0.14	0.13

Table 5: Results for PIH for KNN model

Genre	*Precision*	*Recall*	*F1-Score*
Hip hop	0.21	0.20	0.21
Latin America	0.18	0.22	0.20
Pop	0.21	0.16	0.18

Reggae	0.21	0.17	0.19
Rock	0.18	0.19	0.18
Dance	0.18	0.19	0.19
International	0.20	0.24	0.21
AUC			0.55
Accuracy			0.20
Macro Avg.	0.20	0.20	0.20
Weighted Avg.	0.20	0.20	0.19

4.1.3 Proposed Model

The proposed model uses two different classification algorithms, KNN and an ensemble SVM. The two image features, MDC and PIH are passed into each model separately.

Table 6: Results for MDC for KNN model

Genre	*Precision*	*Recall*	*F1-Score*
Hip hop	0.57	0.53	0.55
Latin America	0.47	0.69	0.56
Pop	0.86	0.70	0.77
Reggae	0.57	0.50	0.53
Rock	0.81	0.88	0.84
Dance	0.41	0.38	0.40
International	0.83	0.77	0.80
AUC			0.93
Accuracy			0.63
Macro Avg.	0.65	0.64	0.64
Weighted Avg.	0.64	0.63	0.63

Table 7: Results for PIH for KNN model

Genre	*Precision*	*Recall*	*F1-Score*
Hip hop	0.45	0.39	0.42
Latin America	0.96	0.79	0.87
Pop	0.76	0.73	0.75
Reggae	0.75	0.95	0.84
Rock	0.75	0.78	0.76
Dance	0.43	0.36	0.39
International	0.23	0.32	0.27
AUC			0.91
Accuracy			0.62
Macro Avg.	0.62	0.62	0.61
Weighted Avg.	0.64	0.62	0.63

Table 8: Results for MDC & PIH for SVM model

SVM	MDC	PIH	Combined
Accuracy	74%	71%	62%
Precision	0.82	0.67	0.62
Recall	0.75	0.71	0.62
F1-Score	0.77	0.67	0.62

Table 9: Results for MDC & PIH for SVM ensemble model

Genre	*Precision*	*Recall*	*F1-Score*
Hip hop	0.89	0.80	0.84
Latin America	0.85	0.85	0.85
Pop	0.71	1.00	0.83
Reggae	0.58	0.64	0.61
Rock	0.71	0.50	0.59
Dance	0.50	0.25	0.33
International	0.88	1.00	0.93
AUC			0.94
Accuracy			0.75
Macro Avg.	0.73	0.72	0.71
Weighted Avg.	0.75	0.75	0.74

4.2 Discussion

The results in Table 4 are for the MCD features for the baseline model. These results show relatively low precision, recall, and f1-scores across all genres, with values ranging between 0.12 and 0.31. The overall accuracy of the model was 14%, and the AUC is 0.49, indicating that the model's ability to distinguish between genres is not sufficient. The macro and weighted averages for precision, recall, and F1-score are range from 0.13 to 0.14, further highlighting the model's poor performance.

Table 5 contains the results for the PIH features for the baseline model. These results are slightly improved compared to the results for MDC. Precision, recall, and F1-score are marginally higher with international achieving the best performance across all metrics. The overall accuracy is 20%, and the AUC is 0.55, indicating a small improvement in the model's performance. The macro average for precision, recall, and F1-score is approximately 0.24, while the weighted average is slightly lower at 0.23

Comparing the results of the two feature sets, the PIH features outperform the MDC features. The PIH model achieves higher precision, recall, and F1-score across most genres as well as slightly better overall accuracy and AUC. However, the improvements are incremental and both models still perform poorly overall. Accuracy does not exceed 20% and AUC values are hovering around the 0.5 mark, this indicates of a model with limited practical utility.

Table 6 contains the results after the methodology changes are applied to the model. The results show a good performance across most genres. Precision, recall, and F1-score are high for some genres; international, rock, and pop. The model achieves an overall accuracy of 63% and an

AUC of 0.93, indicating the model's ability to distinguish between positive and negative classes. The macro and weighted averages for precision, recall and the F1-score are all around 0.64, reflecting the model's robustness across different genres.

The PIH features result from Table 7 are mirror the good performance of the MDC model. Precision, recall, and F1-scores are high, particularly for genres like Reggae and Latin America. The model also attains an overall accuracy of 62% and an AUC of 0.91. The macro and weighted averages are similar those of the MDC features.

Table 8 contains the results for the three SVM model. The result show some improvement when compared to the result for the KNN model. The SVM ensemble model whose results are in Table 9 this model consists of three SVM models one for each feature and one where the features are combined. The performance is increase again when compared to the KNN model. Precision, recall, and F1-scores are higher across all genres compared to the individual feature sets. The overall accuracy improves to 75%, and the AUC is 0.94, indicating a strong robust model. The macro and weighted averages for precision, recall, and F1-score are all approximately 0.75, demonstrating the ensemble model's consistency and effectiveness across different genres.

The results above demonstrate the effectiveness of incorporating MDC and PIH features for album cover classification. While the baseline models provided modest performance, the improvements achieved through the proposed methodology highlight the potential for further advancements in this domain. Considering these findings, the next section will summarise the key insights drawn from this research and outline potential avenues for future work, exploring how more sophisticated methods and additional data could further enhance genre classification using visual features.

5 Conclusion and Future Work

This research aimed to explore how image classification models can utilise album colour features, such as MDC and PIH, to identify music genres from the most popular artists on Spotify from the United States.

The methodology involved sourcing data from two main data sources: a Kaggle (Spoorthi U K, 2024) dataset of popular Spotify artists and the Spotify API. The latter provided album cover images, which were processed and resized to 64x64 pixels. MDC features were extracted using k-means clustering, and PIH were generated using OpenCV. These features underwent further processing, including clustering, PCA, and LDA, before being used to train the KNN and ensemble SVM models.

The key findings are:

- The KNN model using MDC and PIH features individually achieved good accuracy values of 63% and 62%, respectively.

- Combining MDC and PIH features in an SVM ensemble model resulted in improved metrics, achieving an overall accuracy of 75% and an AUC of 0.94.
- The ensemble model provided better precision, recall, and f1-scores across various genres, demonstrating its robustness and effectiveness.

These results indicate that album cover features can be effectively used for genre classification, with the ensemble approach offering performance improvements. The use of clustering, PCA and LDA for feature extraction and dimensionality reduction was instrumental in enhancing the models' efficiency and accuracy. This research could be used to create recommendation systems that consider the aesthetic preferences of users, not just their audio preferences. By leveraging visual features in recommendations, music recommendation systems could potentially increase user engagement. Visually appealing album covers might draw users to explore new music they might not have discovered through audio features alone. Understanding the relationship between album visuals and genre could also inform marketing strategies on streaming platforms. However, some limitation includes the increased complexity and computational demands of the ensemble model, and the potential for performance variation across different genres. Another limitation is the size of the dataset that are used to train the models. The genre classes are balanced to ensure there is no bias within the models. This in turn decreased the overall size of the dataset as the class size is equal to the number of albums in the genre with the least items.

Future research can extend these findings in several meaningful ways:

1. Exploring additional image features such as texture analysis, shape descriptors, and temporal features for sequential data could capture different aspects of album covers, potentially improving classification accuracy.
2. Investigating more sophisticated ensemble methods such as, stacking, boosting, or bagging could be better diverse feature sets, further enhancing model performance.
3. Implementing advanced deep learning models such as, CNNs and Recurrent Neural Networks (RNNs), could capture more complex patterns in album cover images, offering significant improvements over traditional machine learning approaches.
4. Developing a real-time classification system based on the ensemble model could have significant commercial applications, such as in music streaming services for automated playlist generation and genre-specific recommendations. This would involve optimising the model for faster inference and integrating it into a scalable platform.
5. Evaluating the model on different datasets can help assess it generalisability and robustness. Cross-dataset validation would provide insights into how well the model performs on unseen data with different genre distributions.
6. Incorporating user feedback into the classification system can help refine genre classification and improve user satisfaction. An interactive system that learns from user preferences and adapts over time could offer a more personalised experience.
7. Techniques such as entropy maximisation and autoregressive modelling (Rao & Kulkarni, 2017) can be applied to enhance the quality of album cover images, potentially improving classification performance.

8. Further exploration of additional data from the Spotify API, such as metrics related to beats per minute, loudness, tempo, and popularity, could provide valuable features for training the classification models.

By addressing these areas, future research can build on the current research's findings, offering enhanced, robust, and commercially viable solutions for music genres classification based on album cover features.

References

Alam Siddiquee, Md.N., Hossain, Md.A. and Wahida, F. (2023) 'An effective machine learning approach for music genre classification with Mel Spectrograms and KNN', *2023 International Conference on Communication, Circuits, and Systems (IC3S)* [Preprint]. doi:10.1109/ic3s57698.2023.10169397.

Costa, Y.M. *et al.* (2012) 'Comparing textural features for music genre classification', *The 2012 International Joint Conference on Neural Networks (IJCNN)* [Preprint]. doi:10.1109/ijcnn.2012.6252626.

Dammann, T. and Haugh, K. (2017) *Genre Classification of Spotify Songs using Lyrics, Audio Previews, and Album Artwork* [Preprint]. Available at: https://cs229.stanford.edu/proj2017/final-reports/5242682.pdf (Accessed: April 2024).

Datta, A. *et al.* (2021) 'Multi-class classification of different region pop songs using Spotify Database', *2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA)* [Preprint]. doi:10.1109/iceca52323.2021.9675998.

Friconnet, G. (2023) 'A K-means clustering and histogram-based colorimetric analysis of metal album artworks: The colour palette of metal music', *Metal Music Studies*, 9(1), pp. 77–100. doi:10.1386/mms_00095_1.

Hildén, F. (2023). *Tekore.* Available at: https://tekore.readthedocs.io/en/stable/ [Accessed: 3 March 2024].

Hrizi, D. *et al.* (2023) 'Lung cancer detection and nodule type classification using image processing and machine learning', *2023 International Wireless Communications and Mobile Computing (IWCMC)* [Preprint]. doi:10.1109/iwcmc58020.2023.10183237.

Jang, Y. (2023) 'Music genre classification with CNN model evaluation', *2023 14th International Conference on Information and Communication Technology Convergence (ICTC)* [Preprint]. doi:10.1109/ictc58733.2023.10392772.

Koenig, C. (2019) *Classifying Album Genres by Album Artwork* [Preprint]. Available at: https://cs230.stanford.edu/projects_spring_2019/reports/18641024.pdf (Accessed: April 2024).

Marcellus, M., Herwindiati, D.E. and Hendryli, J. (2021) 'Movie poster genre classification with CNN', *2021 IEEE Seventh International Conference on Multimedia Big Data (BigMM)* [Preprint]. doi:10.1109/bigmm52142.2021.00020.

Masruroh, S.U. *et al.* (2023) 'Classification of popular music genre using CNN method with data augmentation', *2023 Eighth International Conference on Informatics and Computing (ICIC)* [Preprint]. doi:10.1109/icic60109.2023.10381995.

Oramas, S. *et al.* (2017) 'Multi-Label Music Genre Classification from Audio, Text, and Images using Deep Features', *In International Society for Music Information Retrieval Conference* [Preprint]. doi:https://arxiv.org/pdf/1707.04916.

Prathyushaa, V.K., Chandrasekar, P.S. and Anuradha, R. (2021) 'A comparative study of image classification models for western notation to carnatic notation : Conversion of western music notation to Carnatic music notation', *2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)* [Preprint]. doi:10.1109/i-smac52330.2021.9641052.

Rao, A. and Kulkarni, S.B. (2017) 'An improved technique of plant leaf classificaion using hybrid feature modelling', *2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)* [Preprint]. doi:10.1109/icimia.2017.7975579.

Sai, M.P. and Kalaiarasi, S. (2023) 'Implementation of music genre classification using support vector clustering algorithm and KNN classifier for improving accuracy', *2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)* [Preprint]. doi:10.1109/iconstem56934.2023.10142741.

Spoorthi U K. (2024) *US Top 10K Artists and Their Popular Songs*. Available at: https://www.kaggle.com/datasets/spoorthiuk/us-top-10k-artists-and-their-popular-songs/data. [Accessed: 3 March 2024]

Wang, Y. *et al.* (2023) 'Novel music genre classification system using transfer learning on a small dataset', *2023 IEEE/ACIS 21st International Conference on Software Engineering Research, Management and Applications (SERA)* [Preprint]. doi:10.1109/sera57763.2023.10197805.

Wu, M. and Liu, X. (2020) 'A double weighted KNN algorithm and its application in the music genre classification', *2019 6th International Conference on Dependable Systems and Their Applications (DSA)* [Preprint]. doi:10.1109/dsa.2019.00051.

Xu, X. (2023) 'Research on multi-labels image classification based on self-supervised model', *2022 International Conference on Image Processing and Computer Vision (IPCV)* [Preprint]. doi:10.1109/ipcv57033.2023.00017.

Yuwono, A. *et al.* (2023) 'Music genre classification using support Vector Machine Techniques', *2023 International Conference on Information Management and Technology (ICIMTech)* [Preprint]. doi:10.1109/icimtech59029.2023.10277842.