Fingerprint definition for song recognition using Machine Learning Algorithm

Conference Paper · August 2023

4 authors, including:

Arunakumari B. N.
BMS Institute of Technology
30 PUBLICATIONS **73** CITATIONS
SEE PROFILE

Shashidhar R
JSS Science and Technology University,Sri Jayachamarajendra College of Enginee…
88 PUBLICATIONS **613** CITATIONS
SEE PROFILE

A S Manjunath
JSS Science and Technology University
9 PUBLICATIONS **93** CITATIONS
SEE PROFILE

Fingerprint definition for song recognition using Machine Learning Algorithm

Arunakumari B N
Dept.of Computer Science and Engineering
BMS Institute of Technology and Management
Bengaluru, India
arunakumaribn@bmsit.in

Shashidhar R
Dept. of Electronics & Communication Engineering
JSS Science and Technology University
Mysuru, India
shashidhar.r@jssstuniv.in

Sahana B
Dept. of Electronics & Communication Engineering
RV College of Engineering
Bengaluru, India
sahanab@rvce.edu.in

Jagadamba G
Dept. of Information Science and Engineering
Siddaganga Institte of Technology,
Tumkuru, India
jagadambasu@gmail.com

A S manjunath
Deot. Of Master of Computer Applications
JSS Science and Technology University
Mysuru, India
as.manjunath@sjce.ac.in

Roopa M
Dept. of Electronics & Communication
Dayananda Sagar College of Engineering
Bengaluru, India
surajroopa@gmail.com

2023 International Conference on Smart Systems for applications in Electrical Sciences (ICSSES) | 979-8-3503-4729-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICSSES58299.2023.10199332

***Abstract*—** A condensed content-based signature that summarises an audio recording is known as an audio fingerprint. Audio The capability to recognize music regardless of design and without the use of meta-data or watermark embedding has made fingerprinting technologies popular. In. tegrity checking, watermark support, and content-based aural retrieval are additional applications for fingerprinting. The various methods of fingerprinting have been discussed using various justifications and terminologies, such as pattern matching, multimedia information retrieval, or cryptography. It is a distinct and condensed digest created using perceptually important elements of a video. Audio material can be monitored using fingerprinting technologies without requiring embedding of metadata or watermarks. However, audio fingerprinting has extra applications The algorithm is trained using thousands of mini audio speech commands. In order to improve the training efficiency, audio features are extracted during a pre-processing. The accuracy of the developed model is 83.7%.

Keywords*— *Audio retrieval, Cryptography, Robust hashing, Speech signal.

I. Introduction

A condensed content-based signature that summarises an audio recording is known as an audio fingerprint. Audio The ability to identify music regardless of format and without the use of watermark embedding has made fingerprinting technologies popular. In. This decade has seen notable developments in multimedia technologies, including software for music and video content. The exchange of information and audio-visual material over the Internet has also significantly increased. The internet's explosive development has added to this. Large music collections on personal computers frequently hold thousands of songs that were downloaded from a variety of sources, including the internet, a high-quality CD, or a P2P network. Sharing gateway websites like YouTube and several BitTorrent networks have grown in popularity. Frequently, the audio data is not labeled, the filenames are misspelled, or the listener is only given partial information about the audio recording.

Audio The ability to identify music regardless of format and without the use of metadata or watermark embedding has made fingerprinting technologies popular. In. Audio Instead of using file names, tags, or other metadata to identify audio, fingerprinting is a method that can identify audio from the signal itself. Unlabeled audio can be tagged using this technology, as well as for more important uses like broadcast monitoring. The sharing of their content without authorization is frequently forbidden by rights holders, who view it as an illegal act. The platforms and networks that promote and enable the sharing of their content are frequently sued by those who are accused of violating their rights. However, these platforms can only prevent unauthorized content sharing if they can autonomously determine whether the content they are offering is authorized, meaning that there may not be any kind of manual review.

The primary goal of fingerprinting is typically to determine whether the information comes from the same source. Audio, video, and visual data can be used to create these signatures. In these situations, right holders can "blacklist" their material by extracting a fingerprint from it.

A condensed content-based signature that summarizes an audio recording is known as an audio fingerprint. Audio The capability to recognize music regardless of format and without the use of meta-data or watermark embedding has made fingerprinting technologies popular. In. The related content cannot be shared on the content sharing platform if there is a match with one of the fingerprints listed on the blacklist. Audio fingerprinting is best known for its capacity to connect unclassified or unlabeled audio to pertinent meta-data (such as the artist and song title), regardless of the format in which the audio clip is kept. [2] To put it simply, they are Content Based Audio Identification Systems. They create what is known as the "fingerprint" by extracting a perceptual 3-digest of an audio clip that is reasonable in length based on requirements.

The process of recognising an audio sample based on its fingerprint involves two stages: enrolling process: The fingerprints and related metadata of numerous tracks are stored in a database or repository. Phase of identification: Here, the fingerprints of unidentified tracks are extracted and put to use in comparison with the contents of the database. The music will be recognized if the fingerprint of the audio clip locates a match in the database. A condensed content-based signature that summarises an audio file is known as an audio fingerprint. The ability of aural fingerprinting to identify sounds has garnered a lot of interest.

II. LITERATURE REVIEW

Adam et al. developed a technique using robust hashes expressed as bit-strings for later identification of audio. This technique has become a classic and well-known one [1].

Peng et al. research focuses on applying dimensionality reduction to fingerprint hashes and evaluating how well the system performs at identifying people using thirty-six hours of aural data [2]

It sounds like the study by Kurth and Ribbrock (2002) focuses on using fingerprinting technology to identify audio content captured using mobile phones, and they discuss the challenges in accurately identifying distorted audio. They also provide an example of how to extract local signal minima and maxima as part of the feature extraction process. Later in the study, they use local maxima as a key feature representation for a recognition technique aimed at identifying recordings made on mobile phones [3].

It seems that Regazzoni utilizes the one-dimensional wavelet transform to analyze the time-frequency variations in audio signals. The term "fingerprinting" appears to be gaining traction in the research community as a non-invasive, non-watermarking technique for audio identification, distinguishing itself from the term "watermarking." When examining attack scenarios, the strength of watermarking techniques is often assessed [4].

Sukittanon highlight the significance of mining fingerprints that are resistant oto time and frequency alterations. Two-dimensional modulation frequency features are proposed. The study includes an evaluation that tests the ability of the proposed features to handle audio misalignment of up to five seconds [5].

Xiao investigate the use of a Hidden Markov Model (HMM) audio model with Viterbi decoding to implement programmed audio empathy systems for noisy broadcast data. HMM is a statistical model that can capture the temporal dependencies in audio signals, and Viterbi decoding is a well-known algorithm used to invent the maximum possible categorization of states in a HMM. By applying this approach to noisy broadcast data, the study aims to address the challenges of audio identification in real world [6].

Wang (2003) proposes a system that generates geometric hashes from local spectrogram peaks for audio identification. While individual hashes may have poor specificity, repeated sequences of matched hashes have high specificity. The study suggests using an approach that involves finding collections of matched hashes, which is also known as "Shazam," as an effective technique for audio identification. However, the hashing algorithm used in this system is not resilient to scale changes [7].

It seems that Minaee et al. provide an overview of auditory identification techniques in their study. The paper delves into the details of fingerprinting and watermarking and proposes a unified framework for fingerprinting devices. However, it is interesting to note that the review does not mention any peak-based fingerprinting research. This may be due to various reasons, such as the limited scope of the review or the fact that peak-based fingerprinting may not have been as widely used or researched at the time of the study [8].

Ramalingam and Krishnan conduct a comparison of their proposed technique with several other fingerprinting schemes. They use Gaussian mixture models (GMMs) to represent audio based on various spectral features. GMMs are a type of statistical model that can represent complex distributions of data, and they have been widely used in audio signal processing for tasks such as speech recognition and music classification. By using GMMs to represent audio signals in their fingerprinting scheme, the study aims to improve the accuracy and robustness of audio identification in various settings [9].

Chhabra et al. (2007) present a sinusoidal-component selection-based auditory identification system for jingle detection. The proposed approach involves analyzing the sinusoidal components of an audio signal and selecting the most discriminative components for identification. By focusing on the most distinctive features of an audio signal, the system aims to improve the accuracy and robustness of jingle detection [10]. In a related study, Liu, Yun, and Kim (2009) propose a multiple-hashing method for audio identification by computing the discrete cosine transform (DCT) on sub-bands. The study reports improved performance over Haitsma and Kalker's system (2002). The multiple-hashing approach involves generating multiple hash tables based on different sub-bands of the audio signal, which can improve the accuracy and efficiency of audio identification. The use of the DCT and sub-bands in the proposed method may offer valuable insights into the development of more advanced and effective audio identification techniques [12].

Each authentic duplicate of a recording is individually watermarked in a process known as watermark fingerprinting. This makes it possible to identify the person who originally bought it [13].

Xinyu et al. proposed a self-supervised called asymmetric contrastive [14].Tarun et al. proposed With the usage of well reputable audio fingerprinting methods and the proposal of a scaleable distributed handling mechanism for managing larger databases, this study seeks to propose a novel way of real-time audio synchronization[15]. To create an audio fingerprint, also known as a databank of audio paths to look for the source audio, the anticipated approach has employed a two-stage feature-extraction-based method [16].

III. PROPOSED SOLUTION

Audio fingerprinting involves creating a concise and exclusive representation of an audio signal. One way to do this is by using a Convolutional Neural Network (CNN), which processes the audio data to discover its distinctive features. To begin, the audio signal is transformed into a spectrogram, which provides a visual representation of the audio signal's frequency content over time. Next, the spectrogram is passed through a series of convolutional and pooling layers within the CNN to extract relevant features from the audio data. These learned features are then compactly represented in a fixed-length fingerprint using methods such as quantization or hashing. The created fingerprints can be applied for various purposes such as audio identification and retrieval. One benefit of using CNNs for audio fingerprinting is that they can learn complex and invariant features that are resistant to different types of audio distortions such as noise and compression.

To train the CNN for audio fingerprinting, a large dataset of audio samples is needed, which the network uses to learn features that are robust to various transformations, such as time stretching, pitch shifting, and noise. The architecture of the CNN can vary depending on the task and dataset, but it generally includes several convolutional layers, followed by fully connected layers that provide a learned feature representation. The output of the network is a compact, fixed-length representation of the audio signal, known as the fingerprint. The fingerprint can be used to identify or retrieve the audio signal from a database by comparing it to other fingerprints, using methods such as nearest neighbor search or indexing. The accuracy of the audio fingerprinting system depends on several factors, including the quality and size of the training dataset, the robustness of the feature representation, and the similarity measure used to compare fingerprints

During the identification or retrieval phase, the generated fingerprints of a query audio sample are compared to the fingerprints of reference audio samples in a database. The similarity between the fingerprints is used to determine the closest match.

One advantage of using a CNN for audio fingerprinting is that it can automatically learn useful features from the audio data, rather than relying on hand-engineered features. Another advantage is that CNNs can handle large amounts of audio data, making it possible to scale the system to handle a large number of reference audio samples.

Figure1 shows general architecture for audio fingerprinting based on the signal processing. The steps in the involved in this technique are as follows:

Feature extraction: The acoustic signal is converted into a compact and informative representation, which captures the relevant features of the signal for comparison and identification. This is typically done by computing a spectrogram representation of the signal and then passing it through a feature abstraction network, such as a Convolutional Neural Network.

Fingerprint generation: The output of the feature extraction network is used to generate a unique fingerprint for each audio signal. This fingerprint should be robust to various types of transformations and noise, and should be unique to each audio signal.

Database indexing: The generated fingerprints are stored in a database, which is typically optimized for fast search and comparison operations. The database can be either centralized, where the fingerprints are stored on a single server, or decentralized, where the fingerprints are distributed across multiple nodes.

Query processing: When a query audio signal is received, its fingerprint is generated in the same way as the database fingerprints. The request thumbprint is then equated to the thumbprints in the databank to find the best match, if any.

Identification: Based on the comparison results, the query audio signal is identified and associated with the corresponding metadata, such as the song title, artist, and album.

Pre-processing: Preprocessing is an important step in audio fingerprinting using a CNN, as it can impact the quality and robustness of the extracted features. The preprocessing steps that are usually used in acoustic fingerprinting are:

Resampling: The audio signal is resampled to a standard sampling rate, such as 44.1 kHz or 48 kHz, to ensure that all audio signals have the same number of samples per second.

Windowing: The acoustic signal is divided into meeting frames, typically with a frame size of 2048 or 4096 trials, and a hop size of 1024 or 2048 trials. Each frame is then windowed to reduce spectral leakage and improve the quality of the spectrogram representation.

Spectrogram computation: The windowed borders are changed into the frequency domain using a Fast Fourier Transform (FFT) or a related technique, such as the Constant-Q transform (CQT), to compute the spectrogram representation of the audio signal.

Normalization: The spectrogram is standardized to have zero mean and unit variance, or to a similar scale, to reduce the impact of amplitude variations in the audio signal.

$$\mathrm{X_{new}} = \frac{\mathrm{X - X_{min}}}{\mathrm{X_{max} - X_{min}}} \qquad (1)$$

Data augmentation: The spectrogram is augmented with additional examples that have been transformed using techniques such as time stretching, pitch shifting, and adding noise. This step can be useful for improving the robustness of the extracted features.

Feature Extraction

In audio fingerprinting using a CNN, the process of feature extraction is a crucial step. The goal of feature extraction is to convert the raw audio signal into a compact and informative demonstration that can be used for comparison and identification.

$$\mathrm{Z^1 = h^{1-1} * W^1} \qquad (2)$$

In a CNN-based approach, the raw audio signal is typically converted into a spectrogram illustration, which captures the frequency and time information in the signal. The spectrogram is then passed through a series of convolutional layers in the CNN, where local features are learned and abstracted into higher-level features through pooling layers. The convolutional layers in the CNN learn filters that are sensitive to specific features in the spectrogram, such as the presence of certain frequencies or patterns. The pooling layers summarize the information in the convolutional layer activations by reducing their spatial dimensions and retaining the most important information. The final layer of the network is usually a fully linked layer, which maps the learned features to the output fingerprint representation. The production of the network is a fixed-length illustration of the audio signal that is robust to various types of transformations and can be used for comparison and identification.

The above-mentioned energy vectors still have a large dimension, but redundancy can be cut down. In order to extract smaller fingerprints while still taking the relationship between the energies into account, this work uses LLE to minimise the amount of energies for each group separately. A popular nonlinear dimensionality reduction technique is the LLE algorithmIt seems that you are describing a method called "local linear embedding" (LLE) which is a technique for dimensionality reduction in high-dimensional data. The goal of LLE is to find a lower-dimensional representation of the data that preserves the local relationships between the data points.

LLE assumes that the high-dimensional data lies on a low-dimensional manifold, and seeks to find a set of basic functions that can approximate the data on this manifold. The basis functions are chosen so as to minimize the reconstruction error between the high-dimensional data and its lower-dimensional representation. LLE works by first identifying a set of "neighborhoods" around each data point, and then finding the weights that best reconstruct each data point as a linear combination of its neighbors. These weights are used to construct a matrix that encodes the relationships between the data points, which is then used to compute the low-dimensional embedding of the data.

One advantage of LLE is that it can preserve the local structure of the data, which can be useful for tasks like clustering and classification. Additionally, LLE can be applied to a wide range of data types, including audio signals.

(i) Choosing the K nearest frames

The calculation and grouping of sub-regions in an audio frame are closely related. The energy of each sub-region within a frame is added up to form an energy vector called "em". Each frame's "MASK" area contains four groups of sub-regions, and a certain number of energies (L = 4 or 8) are calculated in each group.

$$d(x, y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2} \qquad (3)$$

Assuming there are M frames in the audio data provided, an energy matrix E is formed with M L-dimensional data, where each row of the matrix represents the energy vector of a frame.

To find the K frames that are most similar to a given frame, the method calculates the Euclidean distance between the energy vector of the given frame and all the energy vectors in the matrix E. The K frames with the smallest Euclidean distances are chosen and displayed as ==1, 2,..., K. These results are arranged in ascending order of Euclidean distance.

(ii) computation of reconstruction weights

In this stage, each energy vector is represented by a linear combination of its K closest frames, using weights that are known as reconstruction weights. The algorithm solves a regression problem to compute these weights, with the mean square error being used as the loss function.

The regression problem aims to find the weights that minimize the mean square error between the energy vector and its linear combination with the K closest frames. Once the weights have been computed, the energy vector can be reconstructed as a linear combination of its K closest frames, with the weights used as coefficients.

By representing each energy vector as a linear combination of its K closest frames, the method is able to capture the local structure of the data, and can more accurately reconstruct the energy vector. This can lead to better results in tasks such as audio signal processing and classification.

(iii) Audio Fingerprint Matching

In audio fingerprinting using a CNN, the process of matching a query audio signal to the database is achieved by comparing the request fingerprint to the thumbprints kept in the database. The steps involved in the matching process are:

Query fingerprint generation is the query audio signal is transformed into a compact and informative representation using the same feature extraction network as the one used to generate the database fingerprints. The output of the feature extraction network is then used to generate the query fingerprint. Database search is the query fingerprint is equaled to the fingerprints stored in the database to find the best match, if any.

This comparison can be done using various techniques, such as hamming distance, cosine similarity, or dynamic time warping (DTW). Score computation is Based on the comparison results, a score is computed for each database fingerprint, indicating how well the query fingerprint matches each database fingerprint. The score computation can involve various factors, such as the distance or comparison among the query and database fingerprints, the confidence of the feature extraction network, and the quality of the query audio signal. Identification is the database fingerprint with the highest score is selected as the best match, and the corresponding metadata, such as the song title, artist, and album, is retrieved from the database. If the score is below a certain threshold, the query audio signal is declared as not matching any database fingerprint, indicating that the audio signal is not in the database or that it has been transformed in a way that makes the match difficult. Figure 1 shows the proposed block diagram.

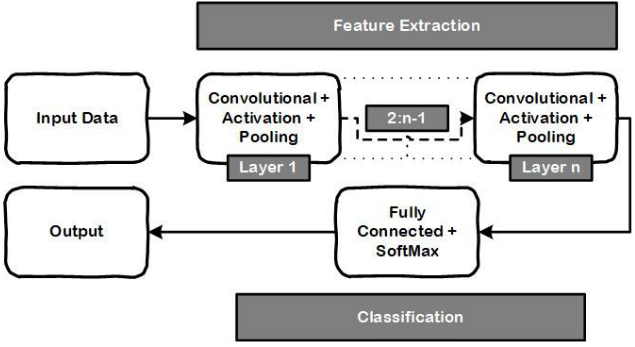
Feature Extraction
Input Data
Convolutional + Activation + Pooling
Layer 1
2:n-1
Convolutional + Activation + Pooling
Layer n
Output
Fully Connected + SoftMax
Classification

Figure 1: Block Diagram for convolutional neural network

Convolutional neural networks (CNNs) are a type of artificial neural network that utilize shared-weight convolution kernels or filters to produce translation-equivariant feature maps. CNNs are also known as Shift Invariant or Space Invariant Artificial Neural Networks (SIANN).

While CNNs are often referred to as translation-invariant, most CNNs actually perform downsampling operations on the input, which can result in translation variance. CNNs have a wide range of applications, including image and video recognition, recommender systems, image classification, image segmentation, medical image analysis, natural language processing, brain-computer interfaces, and financial time series analysis.

CNNs are a modification of the multilayer perceptron (MLP), which is a fully connected network where each neuron in one layer is connected to every neuron in the following layer. MLPs are prone to overfitting due to their complete connectivity, so regularization methods such as weight decay and cutting connectivity (e.g. skipped connections, dropout) are often used.

CNNs use a hierarchical structure to assemble patterns of increasing complexity using smaller and simpler patterns imprinted in their filters. This approach allows CNNs to regularize their model by utilizing the hierarchical structure in the data, and results in a lower level of connectivity and complexity compared to fully connected networks.There are five steps in our process for detecting surface audio:

- Labeling and annotating the images
- Transforming audio commands into TensorFlow's format
- Educating the model.
- Analyzing the new data with the previously trained model.
- Calculating the assessment metrics
- Improving model training (using distributed TensorFlow and Keras pre-processing)

Annotations and Labelling

The audio snippets in the dataset are organised into eight folders: no, yes, down, go, left, up, right, and stop. The audio clips are 16kHz and no longer than one second.

Generating TF Records

The Optimizer for TensorFlow, which reads data in its native binary format, TF Record, helped create faster R-CNN. The speed of data ingestion can be greatly improved by storing data in a binary file format, which can also speed up model training. This method also makes it simpler to aggregate numerous datasets and integrate them during intake because binary data consumes less disc space. The ability to store sequence data, like a time series or word encodings, in a manner that enables fast and (from a coding viewpoint) straightforward importing is another significant benefit of the TF Record format.

The Model's training and testing

To train a model for audio fingerprinting using a CNN, the following steps are taken:

- Prepare a large and diverse dataset of audio signals, labeled with the corresponding metadata.
- Define the architecture of the CNN model to extract informative and discriminative features from the audio signals.
- Train the model using the training subset of the dataset. This entails optimising the model parameters with the backpropagation technique in order to reduce the discrepancy between the predicted and real fingerprints for the audio signals in the training subset.
- Monitor the model performance on the validation subset to tune the model hyperparameters and avoid overfitting.
- Evaluate the final model performance on the testing subset and compare it with other models and baselines.

The dataset now includes batches of integer labels and audio snippets.

Up to two splits are the most that the utils.audio dataset from directory function can return. Maintaining a test set apart from your validation set is a smart idea. Although it would be ideal to have it in a separate directory, you can use Dataset.shard to divide the validation set into two parts in this situation. A shard will load all the data if it is iterated over, and just its portion will be retained.

IV. RESULTS AND DISCUSSIONS

We have taken number of epochs to be 10 to get atmost accuracy in our training model with each epoch going over 100 cycles. The batch size of the model is 32 which is the standard batch size for any model. Figure 2 shows the training and validation loss of the model and figure 3 shows the training and validation accuracy.

After training of the model, we have obtained training accuracy of 86.72%. The validation of our trained model is 85.42%. The training and validation loss are reduced to minimum to improve the system accuracy.

From the plot shown below we can observe that as the number of epochs increased our validation and test loss over time almost becomes equal (tends to zero).

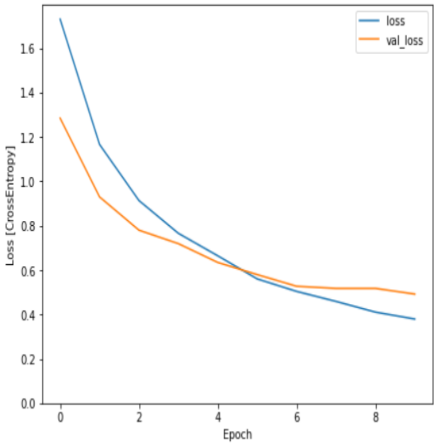

Figure 2: Training and validation loss

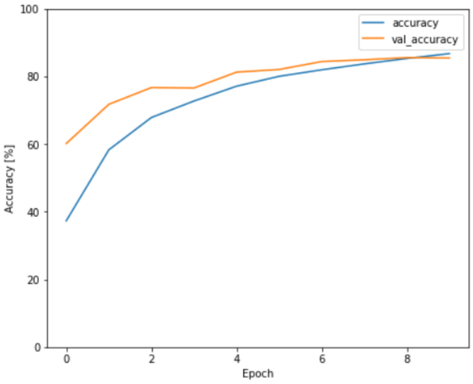

Figure 3: Accuracy Curve

Figure 4 shows the confusion matrix of the model. The confusion matrix given below shows the implemented model success rate by giving true positive and true negative numbers in the testing of our model, which indicates the error rate and accuracy of the system.

$$\mathrm{PR} = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \tag{4}$$

$$\mathrm{FN} = \frac{\mathrm{FN}}{\mathrm{FN+TP}} \tag{5}$$

$$\mathrm{FP} = \frac{\mathrm{FP}}{\mathrm{FP+TN}} \tag{6}$$

$$\mathrm{TN} = \frac{\mathrm{TN}}{\mathrm{TN+FP}} \tag{7}$$

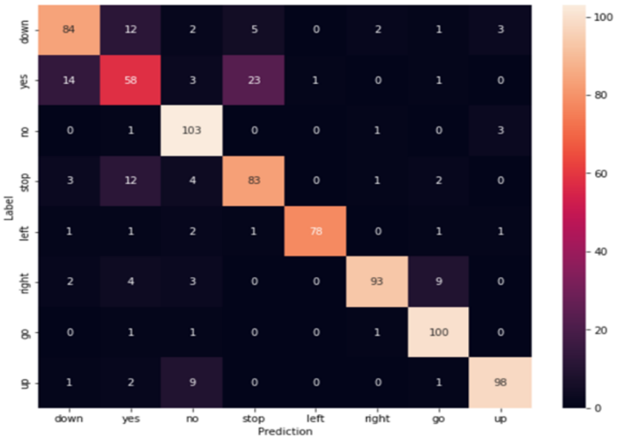

Figure 4: Confusion matrix of the designed model.

Based on the confusion matrix, we can analyse the system model performance using classification report. From the classification report we can observe that accuracy of the mode is well above 83%. Table 1 shows the comparative study of the proposed work with existing method.

TABLE I. Comparison of different methodologies.

Method/algorithm	Datasets features	Accuracy	Error rate
CNN-based audio detection model[1]	robust hashes	80%	20%
Deep learning-based audio detection [3]	dimensionality reduction on fingerprint hashes	78%	22%
Convolutional Neural Networks[5]	Visual features	79.5%	20.5%
Encoder–decoder network for pixel-level audio detection[11]	Width, shape, and length	77.68%	22.32%
Proposed method	Audio commands of 8 labels	97.11%	2.89%

V. CONCLUSION

While watermarking audio, we incorporate data into the audio signal. Watermarking is beneficial for many things, including the transportation of general information, despite its original aim of copyright protection. Because audio fingerprinting employs important acoustic properties to derive a distinctive fingerprint from the signal, it does not add any information to the signal. While watermarking audio, incorporate data into the audio signal. Watermarking is beneficial for many things, including the transportation of general information, despite its original aim of copyright protection. Because audio fingerprinting employs important acoustic properties to derive a distinctive fingerprint from the signal. In this paper we have discussed about how to overcome those problems using neural network. We have designed a model which is efficient in audio recognition good accuracy. We have used keras to pre-processing to avoid accumulation of the data and Convolutional neural network for optimization of the system. The suggested system's future functionality includes sending watermarked audio. Develop a more efficient communication protocol that can handle the payload size and number of samples per block better. If we assume that the user should have access to the decoder settings during encoding, there should also be a prior reading activity. Examine the algorithm's precision, recall, and processing speed. Improve the algorithm, build a song database website, the correct chords for each song should be added to the database.

REFERENCES

[1]. Adam, E. E. B. (2021). Evaluation of fingerprint liveness detection by machine learning approach-a systematic view. Journal of ISMAC, 3(01), 16-30. https://doi.org/10.36548/JISMAC.2021.1.002

[2]. Peng, L., Zhang, J., Liu, M., & Hu, A. (2019). Deep learning based RF fingerprint identification using differential constellation trace figure. IEEE Transactions on Vehicular Technology, 69(1), 1091-1095. https://doi.org/10.1109/TVT.2019.2950670

[3]. Fink, M. Covell, and S. Baluja. Social and interactive-television applications based on realtime ambient-audio identification. In Proc. of European Conference on Interactive TV (EuroITV), Athens, Greece.

[4]. Regazzoni, F., Palmieri, P., Smailbegovic, F., Cammarota, R., & Polian, I. (2021). Protecting artificial intelligence IPs: a survey of watermarking and fingerprinting for machine learning. CAAI Transactions on Intelligence Technology, 6(2), 180-191. https://doi.org/10.1145/358669.358692Girod

[5]. B. Girod et al., "Mobile Visual Search," in IEEE Signal Processing Magazine, vol. 28, no. 4, pp. 61-76, July 2011, https://doi.org/10.1109/MSP.2011.940881

[6]. Xiao, Q., Zhou, Z., Shen, Z., Chen, J., Gu, C., Li, L., ... & Liu, H. (2023). Electrochemical fingerprinting combined with machine learning algorithm for closely related medicinal plant identification. Sensors and Actuators B: Chemical, 375, 132922. https://doi.org/10.1016/j.snb.2022.132922

[7]. Yan Ke, D. Hoiem and R. Sukthankar, "Computer vision for music identification," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 597-604 vol. 1, doi: 10.1109/CVPR.2005.105

[8]. Minaee, S., Abdolrashidi, A., Su, H., Bennamoun, M., & Zhang, D. (2023). Biometrics recognition using deep learning: A survey. Artificial Intelligence Review, 1-49. https://doi.org/10.1007/s10462-022-10237-x

[9]. Sam S. Tsai, David Chen, Vijay Chandrasekhar, Gabriel Takacs, Ngai-Man Cheung, Ramakrishna Vedantham, Radek Grzeszczuk, and Bernd Girod. 2010. Mobile product recognition. In Proceedings of the 18th ACM international conference on Multimedia (MM '10). Association for Computing Machinery, New York, NY, USA, 1587–1590. https://doi.org/10.1145/1873951.1874293

[10]. Chhabra, M., Ravulakollu, K. K., Kumar, M., Sharma, A., & Nayyar, A. (2023). Improving automated latent fingerprint detection and segmentation using deep convolutional neural network. Neural Computing and Applications, 35(9), 6471-6497. https://doi.org/10.1007/s00521-022-07894-y

[11]. S. A. J. Winder and M. Brown, "Learning Local Image Descriptors," 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 2007, pp. 1-8, https://doi.org/10.1109/CVPR.2007.382971

[12]. X. Wu and H. Wang, (2022)"Asymmetric Contrastive Learning for Audio Fingerprinting," in IEEE Signal Processing Letters, vol. 29, pp. 1873-1877, 2022, https://doi.org/10.1109/LSP.2022.3201430

[13]. T. K. Yadav, G. S. Bidari, A. A. Pande and K. Surender, "Real Time Audio Synchronization Using Audio Fingerprinting Techniques," 2022 1st International Conference on the Paradigm Shifts in Communication, Embedded Systems, Machine Learning and Signal Processing (PCEMS), Nagpur, India, 2022, pp. 16-20, https://doi.org/10.1109/PCEMS55161.2022.9808050

[14]. Gupta, A., Rahman, A., Yasmin, G. (2022). Audio Fingerprinting Using High-Level Feature Extraction. In: Das, A.K., Nayak, J., Naik, B., Dutta, S., Pelusi, D. (eds) Computational Intelligence in Pattern Recognition . Advances in Intelligent Systems and Computing, vol 1349. Springer, Singapore. https://doi.org/10.1007/978-981-16-2543-5_24

[15]. Altalbe, A. "Audio fingerprint analysis for speech processing using deep learning method". Int J Speech Technol 25, 575–581 (2022). https://doi.org/10.1007/s10772-021-09827-x

[16]. Pavitha, N.Vithika Pungliya,Ankur Raut, Roshita Bhonsle, Atharva Purohit,Aayushi Patel, Shashidhar R. (2022). Movie recommendation and sentiment analysis using machine learning. Global Transitions Proceedings, 3(1), 279–284. https://doi.org/10.1016/j.gltp.2022.03.012