Hindawi
Complexity
Volume 2021, Article ID 5590996, 10 pages
https://doi.org/10.1155/2021/5590996

Research Article

Harmonic Classification with Enhancing Music Using Deep Learning Techniques

Wen Tang and Linlin Gu

School of Art and Design, Nanchang University, Nanchang, China

Correspondence should be addressed to Linlin Gu; 5204119006@email.ncu.edu.cn

Received 15 February 2021; Accepted 15 April 2021; Published 29 September 2021

Academic Editor: Dan Selistean

Automatic extraction of features from harmonic information of music audio is considered in this paper. Automatically obtaining of relevant information is necessary not just for analysis but also for the commercial issue such as music program of tutoring and generating of lead sheet. Two aspects of harmony are considered, chord and global key, facing the issue of the extraction problem by the algorithm of machine learning. Contribution here is to recognize chords in the music by the feature extraction method (voiced models) that performd better than manually one. The modelling carried out chord sequence, getting from frame-by-frame basis, which is known in recognition of the chord system. Technique of machine learning such the convolutional neural network (CNN) will systematically extract the chord sequence to achieve the superiority context model. Then, traditional classification is used to create the key classifier which is better than others or manually one. Datasets used to evaluate the proposed model show good achievement results compared with existing one.

1. Introduction

The era of art activities such as a key of the musician is the highest-level harmonic representation of the western tonality of music. The key of the piece defines its harmonic center, gives meaning to harmonic progression, and provides a background for the accumulation and release of harmonic tension. Thus, it plays a central role in understanding the meaning of the piece. As a result, understanding not only drives theoretical analyzes of music but is also suitable for contemporary music creators mixing samples of different pieces that fit well into a new composition [1].

A chord is defined as a harmonic set of two or more musical notes that are heard as if they were simultaneously sounding [2]. These are considered to be one of the best characterizations of music. The expansive production of digital music by many artists has made it very difficult to process the data manually but opened the door to automate information retrieval of music although many research studies and algorithms have been devised and applied to extract information from a musical signal [2].

The extraction of harmonic information from the musical sound is fundamental to the computational understanding of music. It describes when tension is formed and how pieces of music work into meaningful parts. It provides background for content that seems important to the listener, such as melody and vocals [3]. Therefore, if we do not take into account the harmonic content of a piece, our understanding (or computer understanding) of it is only superficial. The computational harmonic analysis facilitates many practical applications. Thus, electronic music producers can find musical samples that match well to their tracks. For musicians, the app can suggest metrics for improvising on the progress of a particular chord, it can automatically help in creating master sheets for the songs they want to play, and it can help students master their instrument. Moreover, keeping in mind the practical importance, this study focuses on the artistic task itself, building arithmetic models that extract harmonic information (strings and key) from the acoustic signals of music [4].

The researcher reported expanding the Bi-directional Long-Range Memory Network (BiLSTM) model to address

these shortcomings. The basic idea is to train the model to predict not only string designations but also chord functions, as shown in Figure 1 [5]. We call the resulting model a deep multitasking model or MTH armonizer because it handles some tasks at the same time. We note that the use of chord functions to coordinate melody has been found useful, using hidden Markov models (HMM). Functional harmony clarifies the relationship between strings and scales and describes how harmonic movement directs musical perception and emotion [6].

While the progression of a chord that is made up of randomly selected strings generally appears aimless, a chord progression that follows the rules of functional harmony establishes or conflicts with harmony. Music theorists assign each scale score to tonal, sub, and dominant functions based on the chord associated with that score on a given scale. This post explains the role on which a particular scale score and its associated chord relative to the scale, plays in musical phrasing, and composition but which are very difficult to learn detection of a machine. While a particular format can be considered technically correct in some cases, it can also be considered unimportant in the modern context [7].

However, extract sequences of chords aligned with time from a given. The acoustic music signal is commonly referred to as automatic string estimation (ACE), and it is a well-studied topic in Music Information Retrieval (MIR). ACE systems consist of some variation in extracting acoustic features followed by a pattern matching step where the acoustic features are attached to the chord labels [8].

Both feature extraction and pattern matching are typically implemented in modern ACE systems using machine learning techniques; in the most recent current ACE systems, it usually has some deep learning flavor. Although ACE's recent performance strength allows it to be used in commercial products (e.g., Chordify and Riffstation2), its performance appears to be diminishing in recent years [9]. However, that the visualization of strings in recorded music can be very subjective, which presents a problem in deriving an annotation for naming a single referential chord the "ground truth." This makes the task one of segmentation and labelling, similar to speech recognition. The key difference is that we are interested in both the labels and the timestamps of the segments, whereas in speech recognition, only the label sequence matters [10].

In this paper, the computational machines that extract high-level information from signals face two key problems: (i) how to extract meaningful information from noisy sources, and (ii) how to process this information into sensible output. For chord recognition, thus, translates to acoustic modelling, how to predict a chord label for each position or frame in the audio and temporal modelling and how to cast this information into meaningful segments of chords.

The goal of this paper is on improving frame-wise predictions of acoustic models, while only a few works have explored improvements in temporal models. Thus, a trend was reinforced through the insight that the capabilities of existing temporal models are limited, and temporal models enforce continuity of individual chords rather than provide

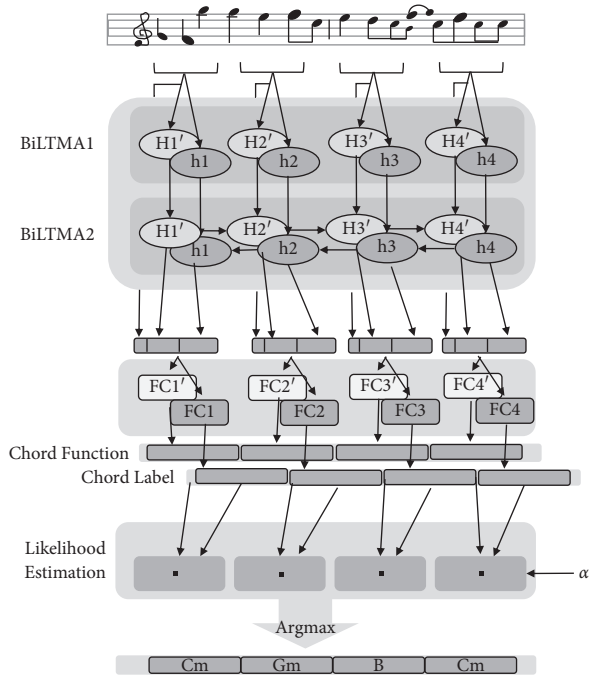

Figure 1: Diagram of the proposed MTHarmonizer, a deep multitask model [7].

information about chord transition, and they mainly model the chord's duration [11]. Also, we can notice that in traditional music, "chord progressions are less predictable than it seems," and thus, knowing chord history does not greatly narrow the possibilities for the next chord [11].

Prediction of future results is needed because of parameters for each model increased and become necessary to be controlled by deep learning such as CNN [12]. The automated behavior of models become worthy in this world so deep learning is important with a chord recognition and key classification models of hormonic musical.

The remainder of this paper is organized as follows. Section 2 reviews related work and background of the harmonic quality of musical and machine learning. Section 3 provides a brief reminder of the chord recognition system. Section 4 describes our key classification of the harmonic field. Section 5 presents the method and discussion including the features, processing, and classification of this study. Section 6 provides the Results and Discussion and the evaluation of this study, and Section 7 concludes the paper.

2. Related Work and Background

The harmonic content of a signal is what gives a sound as Toun and thus makes the tone of strings distinct from a flute or reed pipe. Harmonic distortion introduces additional harmonics to the input signal that are musically related [13].

A novel approach for detecting changes in the harmonic quality of musical audio signals has been suggested. For Equal Tempered Pitch Class Space, this model was used. This model maps 12-bin chroma vectors to the interior space of a 6D polytope; the vertices of this polytope are mapped with pitch groups. The application of adaptive thresholding will

enhance the detection of more severe harmonic shifts. Strong transient signals may trigger the masking of true peaks. This problem can be rectified by adding a Transient/Steady-State distinction to the audio. The outcome tests therefore show that the algorithm can successfully detect harmonic changes in polyphonic audio files, such as chord boundaries [14].

Nevertheless, for feature extraction for singing voice detection, Dieleman and Schrauwen [15] used a unified network for both feature extraction and classification. Logically, better features than current ones should be able to be extracted by using a learnable network for feature extraction. In Dieleman and Schrauwen's study though, simulation results showed that this form of unified. Compared to networks utilising conventional features, the networks did not have greater precision, as one frequently used feature for audio applications is the MFCCC (Mel Frequency Cepstral Coefficient).

However, both the audio and symbolic data have been extensively investigated in the chord recognition problem. Various machine learning methods have been applied to this issue in recent years. RNN-based approaches such as LSTM-based networks have been implemented in audio data processing because of their ability to model the long-term dependence of a time series [16, 17].

Recently study has shown that such models have been applied to the low hierarchical level (directly on audio frames) that prevents learning musical relationships, including expressive models such as recurring neural networks (RNNs).

Nevertheless, temporal models are disengaged into a harmonic language model to be applied to chord sequences and a model of chord length that relates the language model's chord-level predictions to the acoustic model's frame-level predictions. The effect of each model on the chord recognition score is the result of this analysis and shows that the use of harmonic language and length models enhances the results [18].

As the conventional form of audio subjective evaluation involves a large number of people to audition and assess, the subjective sense of hearing variance and sample space data of the tester limited the effect of the experiment's accuracy. In addition, using a deep learning network, the historical audio data has significant distortion issues. In view of the characteristics of audio data repair, an intelligent audio assessment technology is being explored. Therefore, a quality design method is designed to analyze audio data, so the system performance and audio signal quality are tested by extracting the features. The findings of the tests indicate that the device works well; the predictive results and the subjective assessment of the correlation and dispersion metrics are good, up to 0.91 and 0.19 [19].

There are a number of small audio signal characteristics that limit the resolution of musical emotions in different ways. A study of a multifeature fusion music classification algorithm based on deep confidence networks to tackle the limitations of single morphological data in music sentiment classification. Indeed, music signal feature vectors are extracted and fused from multiple angles to form multifeature data. At the same time, by adding fine-tuning nodes to improve the tunability of the model, the traditional deep confidence network is enhanced for music emotion classification. Therefore, in the improved deep confidence network, the training set acquired from the fusion is trained. The test results show that 82.23%, which can be a good aid for music retrieval, is the highest music sentiment classification result [21].

A short-time Fourier transformation transforms into the spectral domain proposed to the windowed signal (STFT). Since the STFT coefficients are complex-valued, before sending them to CNN for processing, we take their modulus. The windowed signal is multiplied by weights comprising sine and cosine coefficients to be realisable in the network structure. As shown in Figure 2, there are 1024 sets of weights, with each one having 2048 coefficients. A spectrogram of size 63-1024 is obtained at the output of the SQRT (square root) layer, where 1024 represents the frequency bins and 63 represents the time instances [14].

In Figure 2, used the square layer to take output squares from sin MYP1D and cos MYP1D, and then, take the added values from the squared roots. The reason for taking the square is to prevent negative values being made. They are not concerned with the signal process, but only with the signal's relative "power" (energy). Thus, the square function is used. In reality, in the experiments, they attempted to eliminate the square and the square root functions, but the accuracy of such an arrangement was much lower [14].

Through previous studies, it is possible to benefit from the previous methods and to develop a method to overcome the defects in the traditional methods.

3. Chord Recognition System

The chord recognition is to segment the audio and label these segments with a chord symbol. This symbol should correspond to the harmonic interpretation of an expert listener. This short description bears the imprint of subjectivity: harmonic interpretations often differ among musical experts. Thus, it complicates the building and evaluation of chord recognition models. The reason for this is that only a subset of all pitches that are perceived to sound simultaneously is deemed relevant for the local harmony. Which subset this is and which pitches are considered to sound simultaneously are subject to interpretation.

Indeed, chord recognition systems more often than not resemble adapted models from speech recognition. The main distinction is that, in chord recognition, start and end times of the labelled segments are vital, while in speech recognition, usually only the sequence of recognized words matters. Chord recognition systems follow the scheme shown in Figure 3. They feature an acoustic model that extracts features from a context of audio and often also predicts a chord label for the center frame of this context. These predictions are then processed by a temporal model, which incorporates more temporal context and outputs homogeneous labelled chord segments. For example, many chord recognition systems are based on chroma features modelled by Gaussian mixtures as an acoustic model, with a hidden Markov model as the temporal model [22].

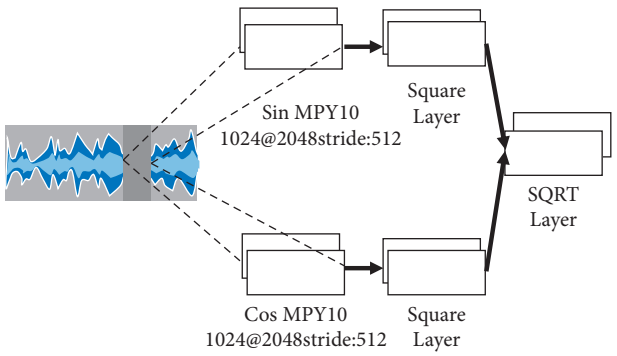

Figure 2: Network structure to compute the spectrogram layer [14].

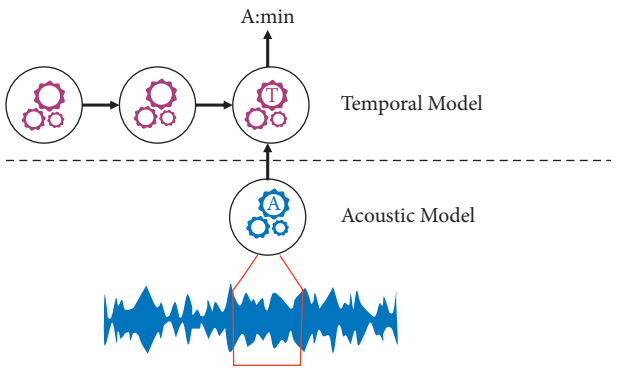

Figure 3: Overview of a chord recognition system [22].

4. Key Classification of Harmonics

The aim of key classification is to locate a piece of musical audio (global key). Thus, as understood by an expert listener, an aggregate harmonic representation over the entire piece should be a global key. This is a subjective undertaking, as in chord recognition, but there are no studies that explore how this subjectivity affects main computational classification models [23].

Key Estimation is referred to by the researchers. Taking into account, however, the same arguments that favors the term Chord Recognition. Classification thus assigns a categorical label to the entire input to more accurately describe the task given a low-level input representation. This is, by definition, a scenario for classification [24].

Hidden Markov models, HMM, are used as the most common method for predicting the chord sequence provided by chroma vectors with involve key estimation [25]. An HMM is a probabilistic model in which it is presumed that the sequence being modelled is a Markov hidden variable loop with a parallel chain of observed variables depending on these hidden variables. When the chords are taken into consideration, the chromatic features (or spectral properties) as in Figure 4(a), are the hidden variables that are discovered by means of the chords. The HMM variables can then be tuned by an expert or calculated from data. In addition, as expert systems, we will refer to the former type of models and to the latter as machine learning models [26].

The approach to machine learning was pioneered in chord estimation. Usually, if a fully annotated training set is available possibly with Laplace correction [27], it estimates the parameters by expectation maximisation or using maximum probability. Recently, a discriminatory parameter estimation approach has also been used, which directly attempts to optimise the performance of the estimation rather than the probability function [28].

Eventually, it was noted that, under different tonal keys, chord change characteristics can be exploited so that the estimation of chords and keys at the same time came naturally. This was done by using more complex HMM topologies, often referred to as Bayesian dynamic networks [27, 28]. These methods use key/chord chains to connect to spread key-to-chord information Figure 3(a). This HMM topology mathematically formalises a probability distribution P(k, c, X/0) for the chroma vectors X and the annotations together, with 0 representing the distribution parameters. Given the optimal parameters 0^*, the key/chord estimation task is equivalent to finding $\{K^*, c^*\}$ that maximizes the joint probability: $\{k^*, c^*\} = \text{argmax}, k, c\ P(k, c, X/0^*)$.

On the contrary, the systems learn parameters 0, for more complex models like these, entirely from a training collection of songs and annotations [29]. The majority of approaches are focused, at least in part, on expert knowledge, where parameters are defined on the basis of developers' music theoretical knowledge [28, 29]. For instance, an expert, often informed by perceptual key-to-key and chord-to-key relationships, can set the key and chord transition parameters [29].

However, although the estimation of the bass note of a chord by using the bassline as an additional sequence was investigated in parallel with research on the inclusion of the key [30], these research lines did not converge until a new system of experts, namely, the Musical Probabilistic (MP) model, was released.

The MP model structure is shown in Figure 3(b). It was hailed as the first device to incorporate most musical features into a single model, allowing main, chord, and bass pitch groups to be inferred simultaneously [30]. This marked a leap forward in study on harmonic analysis, enabling the prediction of complex chords for the first time. The complexity of the structure, nevertheless, has also increased the search space and has led to significant memory usage and processing time problems, limiting its practical use.

5. Method and Discussion

This translates to the acoustic model of how to predict a chord mark for each frame in the audio for chord recognition. Acoustic models therefore derive classifications of frame-wise chords, usually in the form of a distribution across chord labels. These models have been hand-crafted and split into feature extraction and pattern matching in conventional chord recognition systems. Extraction of features transforms audio signals into representations that emphasise harmonic content; typically, this is some sort of pitch-class profiles; matching patterns allocate chord labels

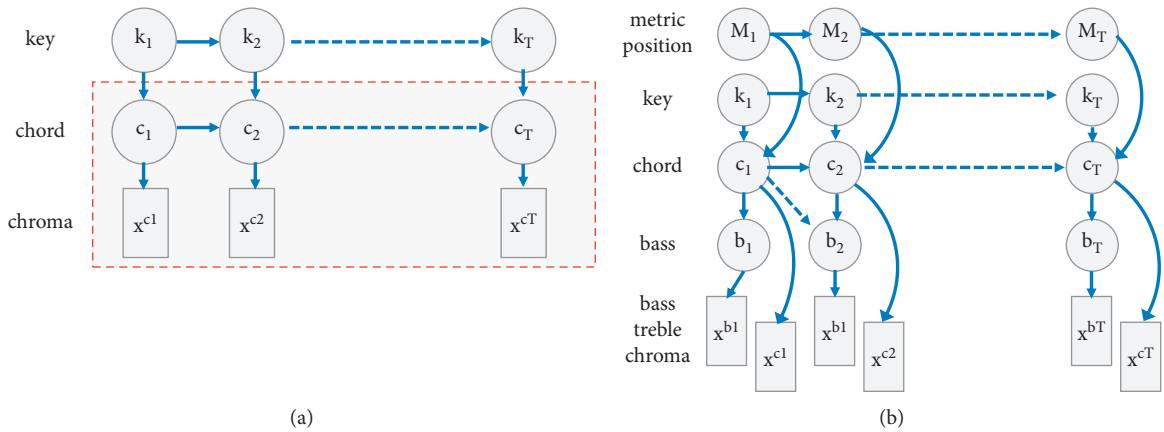

Figure 4: The development of HMM topologies for key/chord estimation systems (a). (b) Image adapted from [28].

to such representations, but only works on single frames or local context [31].

Each chord and global key methods for machine learning are achieved using three main stages, which feature extraction including preprocessing stage and key classification method, which is our main concern. The following will explain it in detail.

5.1. Feature Extraction. Feature extraction is a two-phase process. First, in the preprocessing phase, we transform the signal into a time-frequency representation. Then, we give this description to a convolutional neural network (CNN) and train it to classify chords. We take as a high-level feature extraction the activations of a hidden layer in the network, which we then use to classify the final sequence of chords.

5.1.1. Preprocessing. The first step of our feature extraction pipeline converts the audio input into a time-frequency representation appropriate for a CNN input. CNNs consist of fixed-size filtering that capture the local structure, requiring a similar distribution of spatial relations in each input region. We measure the magnitude spectrogram of the audio to achieve this and apply a filter bank with triangular filters spaced logarithmically.

This time-frequency representation has all areas of the input, and distances between notes (and their harmonics) are equal. Eventually, we compact the value spectrum by logarithmizing the filtered magnitudes. Mathematically, an audio recording's resulting time-frequency representation is defined as

$$Q = \log\left(1 + \underset{\mathbf{B}\log}{\Delta} |S|\right), \tag{1}$$

where S is the short-time Fourier transform (STFT) of the audio and $\underset{\mathbf{B}\log}{\Delta}$ is the logarithmically spaced triangular filter bank. To be concise, we will refer to $\boldsymbol{Q}$ as spectrogram in the remainder of this section.

We feed the network spectrogram frames with context, and the input to the network is not a single column q_i of $\mathbf{Q}$ but a matrix:

$$X_i = [q_{i-c}, \ldots, q_i \ldots, q_{i+c}]. \tag{2}$$

The index of the goal frame is i and the context size is c. For the STFT, we use an 8192 frames size with a hop size of 4410 at a sample rate of 44 100 Hz. Between 65 Hz and 2,100 Hz, the filter bank consists of 24 filters per octave. The background size is $C = 7$, thus each X_i representing 1.50 s of audio for each. Our parameter choice results in an input dimensionality of $X_i \in \mathrm{R}105 \times 15$.

However, we choose the temporal model directly by their capacity to model chord sequences and frame-level chord recognition as explained below.

5.2. Chord Sequences Modeling. We would like to specifically measure the modelling capacity of temporal models. Given the ones already observed, a temporal model predicts the next chord symbol in a sequence. Since we deal with frame-level data and follow a frame rate of 10 fps, there are 10 chord symbols per second in a chord series. More formally, a model M outputs a probability distribution PM $(P_M\ (y_t|y_{1:\,t-1})$ for each yt, given a chord series $y = y1:\mathrm{T}$. We can determine the likelihood of the chord series from this:

$$P_M(y) = P_M(y_1).\prod_{t=2}^{T} P_M\ (y_t|y_1, \ldots, y_{t-1}). \tag{3}$$

We calculate the average log-probability that it assigns to the sequences $y \in \mathcal{Y}$ to measure how well a model M predicts the chord sequences in a dataset $\mathcal{Y}$:

$$\mathcal{Y}(M, \mathcal{Y}) = \frac{1}{N_{\mathcal{Y}}} \sum_{y \in \mathcal{Y}} \log(P_M(y)), \tag{4}$$

where $N_{\mathcal{Y}}$ is the total number of chords' symbols in the dataset.

5.3. Frame-Level Chord Recognition. In the sense of a full chord recognition system, we want to test the temporal models. The task is to predict the correct chord symbol for each audio frame. As in the Chord Sequences model, we

use the same details, the same train/test split, and the same chord vocabulary (major/minor and "no chord").

Table 1 Weighted Chord Symbol Recall of the 24 major and minor chords and the "no-chord" class for the tested temporal models Spectrogram computation, an automatically trained feature extractor and chord predictor, and finally the temporal model are included in our chord recognition pipeline.

We extract a log-filtered and log-scaled spectrogram at 10 frames per second between 65 Hz and 2 100 Hz and feed 1.50 spectral patches into one of three acoustic models: a logistic regression classifier (LogReg) and a deep neural network.

5.4. Key Classification Method. The key classification of musical audio parts with a single global key: in the main classification pipeline, we abandon hand-crafting or tuning elements in contrast to previous works. Our device runs on the spectrogram directly, and from the data, it can estimate all its parameters. However, this study replaces the complete main classification pipeline with a model that can be end-to-end optimized.

The proposed neural network is designed to cover all phases of the classic key classification pipeline, a convolutionary layer preprocessing step, a dense layer that projects the feature maps at the time-frame level into a short description, a global average layer that aggregates this description over time, and a softmax classification layer that predicts a piece's global key.

Figure 5 shows the architecture of our model: five convolutional layers with 8 function maps computed by 5×5 kernels, followed by a dense layer with 46 frame-wise units; this projection is then averaged over time and classified using a softmax layer of 24 way. The exponential-linear activation function is used for all layers (except the SoftMax layer).

In conventional key classification schemes, the convolutional layers constitute the first component of the "feature extraction" equivalent. They are intended to process the input spectrogram, deal with adverse factors such as noise or minor detuning, and compute a short frame-wise definition of harmonic content along with the projection layer. Inputs of arbitrary lengths can be handled via this portion of the network. In the following layers, their production is aggregated.

An average layer lowers the extracted representation to a fixed-length vector before classification. We could use other, more efficient methods (such as recurrent layers), but we found that they struggled to produce better results in preliminary experiments.

Finally, the global key for the audio is predicted by a SoftMax classification sheet. We limit ourselves only to major and minor modes, resulting in an output of 24 possible groups (12 tonics (major and minor)). As most musical pieces are in either major or minor, this is a common limitation since there are no datasets with accurate song-level annotations in other modes.

Table 1: Weighted Chord Symbol Recall for the evaluated temporal versions of the 24 major and minor chords and the "no-chord" class.

Frame	None	MV	HMM	RNN
LogReg	70.1	72.3	73.1	73.5
DNN	73.2	74.8	77.1	76.0
CoveNet	78.1	78.9	79.3	79.0

6. Results and Discussion

The CNN predictions also provide good results in terms of frame-wise precision using the predictions of the pattern matching stage. Chord sequences produced in this manner are always broken, however. Thus, the primary aim of chord sequence decoding is to smooth the sequence recorded. Thus, to add interframe dependencies and find the optimal state sequence using Viterbi decoding [20], we use a linear-chain CRF:

$$P(y_{1:T}|x_{1:T}) = \frac{\exp[E(y_{1:T}, x_{1:T})]}{\sum_{y'1:T} exp[E(y'_{1:T}, x_{1:T})]}, \tag{5}$$

where $y_{1:T}$ is the label vector sequence and $x_{1:T}$ is the feature vector sequence of the same length. We assume each y_T to be the target label in one-hot encoding. The energy function is defined as

$$E(y_{1:T}, x_{1:T}) = \sum_{t=1}^{T} \left[y_{t-1}^T A y_t + y_t^T c + x_t^T W y_t \right] + y_0^T \pi + y_T^T \tau, \tag{6}$$

where A models the interframe potentials, W is the frame-input potentials and label bias, π is the potential of the first label, and τ is the potential of the last label. This form of energy function defines a linear-chain CRF.

From the equations, then 6.1 and 6.2 imply that a CRF can be used as a logistic regression that is generalised. When we set A, 7, and t to 0, they become equal. In addition, logistic regression is analogous to a neural network's softmax output layer. Therefore, we argue that it is possible to view a CRF whose input is computed by a neural network as a generalised SoftMax output layer that allows dependencies between individual predictions. This makes CRFs a natural choice between neural network predictions for integrating dependencies.

However, our model, has 25 states (12 semitones major, minor as illustrated in Table 2, and a class of "no-chord"). Via the weight matrix W, which computes a weighted total of the features for each class, these states are related to observed features. This is in line with what the CNN's global-average-pooling section does. Thus, as input to the CRF, we will use the input to the GAP component, Fi, averaged for each of the 128 function maps. As the operations in between linear convolution and batch normalisation are linear and no dropout is performed at test time, we can pull the average operation from the last layer right after the feature-extraction layer.

According to a Wilcoxon signed-rank test, the findings of NMSD2 are statistically significantly worse than others.

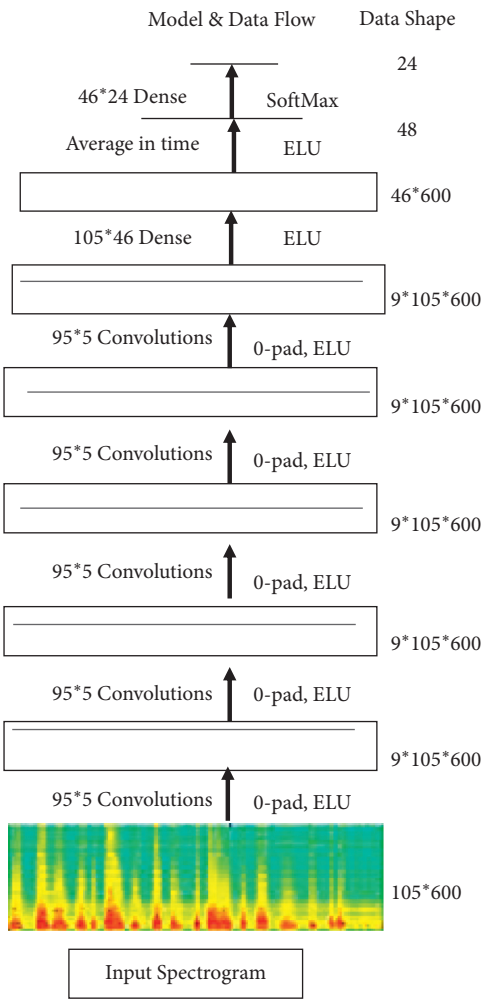

Figure 5: Conventional neural network for musical key classification.

Notice that the CB3, KO1, and NMSD2 train and test data overlap, while the results of our system are calculated by 8-fold cross-validation.

We will formally refer to the input series as $F^{-} \in R^{128 * T}$, where each column f_i^{-} is the average CNN feature output for a given X_i input. Our CRF models $P\left(y_{1:T} | F^{-}\right)$ accordingly.

We train the CRF using Adam, as with CNN, but set a higher learning rate of 0.01. The mini batches consist of 32 sequences of 1024 frames (102.3 sec) in length each. We use the $l1$ regularized negative log-likelihood of all sequences in the dataset as an optimisation criterion:

$$\mathscr{Y} = -\frac{1}{s} \sum_{s=1}^{S} \log\ p\left(y_{1:T}^{(s)} | F^{-(s)}\right) + \lambda |\xi| 1, \tag{7}$$

where S is the number of sequences in the dataset, $\lambda = 10{-}4$ is the $l1$ regularisation factor, and ξ are the CRF parameters. We stop training when validation accuracy does not increase for 5 epochs.

Compared to three state-of-the-art algorithms, Table 3 shows the results of our process. We can see that, while the train set of the reference methods overlaps with the test set, the proposed approach performs marginally better (but not statistically significant).

The dataset contains 69 different chord types. Indeed, these chord forms are unevenly distributed: the four most common types (major, minor, dominant 7, and minor 7) already constitute 85% of all annotations [32]. We just simplify this vocabulary to major/minor chords, where we map chords with a minor 3rd as their first minor interval and all other major chords. We have 24 chord symbols after the mapping (12 root notes (major and minor)) and a "no-chord" symbol, so 25 groups.

Table 3 shows that ℓs (M, Y) and ℓc (M, Y) are stated in addition to ℓ (M, Y). These numbers reflect the average log-probability assigned in the dataset by the model to chord symbols when the current symbol is the same as the previous one and when it has changed. Similarly to ℓ (M, Y), they are computed, but the result in equation (5) catches t only when $y_t = y_{t-1}$ or $y_t \neq y_{t-1}$, respectively. They allow us to think about how well a model can smooth the predictions when the chord is stable and how well chords can be predicted when they shift (this is where "musical knowledge" could come into play).

We can consider its greater modelling ability, and the RNN performs just marginally better than the Markov Chain (MC). This transition is rooted in better predictions as the chord shifts (−5.22 for the RNN vs. −5.42 for the MC). This may mean that the RNN can, after all, model musical knowledge better than the MC. This benefit, however, is minuscule and rarely comes into play: the right chord has an avg. The probability of RNN is 0.0054 vs. MC1 0.0044, and the number of positions where the chord symbol changes is low compared to where it remains the same.

Furthermore, when implemented in a Frame-Level Chord system, we determine whether the marginal improvement given by the RNN translates into better chord recognition precision.

In Table 4, the result showed that the more straightforward first-order HMM does not outperform the complex RNN temporal model. Compared to not using a temporal model at all and a clear majority vote, they boost.

The first observations, however, concentrated on how well a complex temporal model can learn to predict chord sequences compared to a simple first-order one. We saw that the complex model performed only slightly better, despite its significantly higher modelling ability. The second result showed that the RNN temporal model did not outperform the first-order HMM when implemented inside a chord recognition system. The approximate design of the inference algorithm was possibly counteracted by its marginally improved ability to model frame-level chord sequences.

According to the key classification result, a more thorough quantitative analysis was needed than the accuracy scores for computing. In particular, while when designing the method, we consider the task to be a simple 24-way classification problem, and some classes are semantically closer to one another than others. Therefore, Table 4 illustrates that the model proposed has been trained on two datasets (GS and BB).

Table 2: Recall of major and minor chords accomplished by various algorithms including the weighted chord symbol.

Methods	Isophonics	Robbie Williams	RWC
CB3	81.8	—	—
KO1	82.6	—	—
NMSD2	82.2	—	—
Proposed	83.1	82.9	82.6

Table 3: Average log probabilities of the models on the test set.

Symbols	Markov chain	Recurrent NN
ℓ (M, $\mathcal{Y}$)	−0.278	−0.277
ℓc (M, $\mathcal{Y}$)	−6.444	−5.222
ℓs (M, $\mathcal{Y}$)	−0.051	−0.055

Table 4: Results of various training configurations of proposed model systems.

Text sets	Method	Train set	Weighted	Correct	Fifth	Relative	Parallel	Other
GS	CK1	GS^{MTG}	**75.3**	**68.2**	6.8	7.1	4.3	**14.1**
	CK2	BB^{TV}	57.6	47.5	6.7	12.8	16.8	17.7
	CK3		69.5	61.6	6.9	8.7	6.5	16.6
	EDM^{A}		65.9	57.4	7.6	6.8	11.0	17.8
	EDM^{M}	GS^{MTG} and BB^{TV}	70.4	63.5	8.8	2.6	6.5	18.7
	EDM^{T}		44.9	33.9	8.7	15.7	9.7	32.5
	QM		50.8	39.8	12.0	13.5	4.9	31.3
BB^{TE}	CK1	GS^{MTG}	72.9	62.8	7.8	13.4	12.7	**4.4**
	CK2	BB^{TV}	**84.0**	**77.4**	9.2	5.1	4.5	5.0
	CK3		80.0	71.0	9.9	9.3	6.6	4.2
	EDM^{A}		78.9	70.8	11.6	3.0	5.8	9.3
	EDM^{M}	GS^{MTG} and BB^{TV}	30.0	14.8	2.4	16.3	42.2	25.2
	EDM^{T}		75.8	66.9	12.7	6.5	2.9	12.0
	QM		61.0	52.3	11.9	4.4	8.5	23.9

The test results of all training configurations of our proposed model and of the reference systems are shown in Table 4. Using a Wilcoxon-signed rank test, the statistical significance of the results is determined, with the error types reflecting the ranks. Our model clearly outperforms the reference systems if trained on the correct genre: 75.3 vs. 70.4 ($\alpha = 0.010$) for the GiantSteps (GS) dataset and 84.0 vs. 78.9 ($\alpha = 0.014$) for the Billboard (BB) dataset.

We mention that a major decrease in the accuracy of the main classification: a model trained on BBTV (pop/rock) tested on GS (electronic music) achieves a weighted score of just 57.6, compared to 75.3 when trained on GSMTG electronic music. However, the amount of serious errors ("other" category) that our system commits in this configuration is not greater than those of the reference systems. Similar to the reference systems specializing in this genre (17.8% and 187% for EDMA and EDMM, respectively), the model only predicts a completely unrelated key 17.7% and vice versa, and it achieves the lowest rate of serious errors when trained on GSMTG and tested on BBTE (4.5%).

Predicting the most common error that occurs in these cross-genre setups is a wrong mode and predicting the relative minor/major key (resulting in parallel minor/major). This implies that, while certain basic notions of tonality can still be understood by the model, finer features vary too much between parts of various genres.

In the training stage, the proposed model could be trained to provide a good unified key estimator for multiple genres. The resulting CK3 system does not achieve the efficiency of the specialized ones (69.5 vs. 75.3 on GS and 80.0 vs. 84.0 on BBTE), but it performs on GS as well as EDMM, which is manually calibrated to provide good results on electronic music datasets (69.5 vs. 70.4).

For the GiantSteps dataset, the numbers given for the EDM* systems differ from those that were originally published [33]. This is primarily because we have introduced a tougher "fifth" category criterion: we need to align the goal mode with the expected mode, thus ignoring the mode for that category. Also, improvements in the library used in the initial implementation exacerbated the findings compared to the original ones, according to personal correspondence with the author.

We have presented a global key classification system using CNN. Without the need for expert expertise in function design or complex preprocessing steps, feature selection, and frame-level chord, this model can be automatically trained end-to-end compared to the previous work.

We have shown experimentally that, on datasets of electronic music and pop/rock music, the model performs state-of-the-art. In addition, we expect to test more genres or classical music, for the proposed model.

7. Conclusion

We developed two harmonic musical techniques. We first developed powerful acoustic models based on deep neural networks and processed their predictions with a random conditional field, a simple first-order model that smoothed the predictions of the acoustic model primarily. We then researched how data-driven temporal models that go beyond smoothing can be developed. They need, therefore, to work on chord symbol sequences. This leads directly to a range of open problems about models of chord language. The development of hierarchical methods for modelling and assessing chord language models, but also complete chord recognition systems, is important for these points outlined above.

We have considered the main classification in the second part of this paper. We first developed a convolutional layer's neural network inspired by conventional key classification algorithms in its structure.

This paper's contribution is that it can only extract a piece's global key and is unaware of key modulations. While the methods provided can be used to detect keys (using a preprocessing and feature selection), for short audio excerpts, classification accuracy has fallen. We concluded that, to properly track key modulations, future systems need to understand a piece's hierarchical harmonic structure. Future works will expect us to create new network architectures by modelling tonal harmony as a whole in a single neural network. To solve this challenge, we may not be able to rely on standard models.

Data Availability

Two standard datasets were used in the proposed system, each with more than 600 available pieces, to recognize 48 units applied framewise. Two types of the GiantSteps (GS) dataset and the Billboard (BB) dataset were helpful to improving the model and five convolutional layers with 9 feature maps [33].

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1251–1258, Honolulu, HI, USA, July 2017.

[2] R. Shrestha, "Chord classification of an audio signal using artificial neural network," *International Research Journal of Engineering and Technology (IRJET)*, vol. 5, no. 11, 2018.

[3] Z. Rafii, A. Liutkus, F.-R. Stoter, S. I. Mimilakis, D. FitzGerald, and B. Pardo, "An overview of lead and accompaniment separation in music," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 8, pp. 1307–1335, 2018.

[4] S. Basak, A. Bazavov, C. Bernard et al., "Lattice computation of the electromagnetic contributions to kaon and pion masses," *Physical Review D*, vol. 99, no. 3, Article ID 034503, 2019.

[5] F. Korzeniowski and G. Widmer, "On the futility of learning complex frame-level language models for chord recognition," 2017, https://arxiv.org/abs/1702.00178.

[6] N. Takahashi, N. Goswami, and Y. Mitsufuji, "Mmdenselstm: an efficient combination of convolutional and recurrent neural networks for audio source separation," in *Proceedings of the 2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*, pp. 106–110, IEEE, Tokyo, Japan, September 2018.

[7] Y. C. Yeh, W. Y. Hsiao, S. Fukayama et al., "Automatic melody harmonization with triad chords: a comparative study," *Journal of New Music Research*, vol. 50, pp. 1–5, 2021.

[8] F. Simonetta, S. Ntalampiras, and F. Avanzini, "Multimodal music information processing and retrieval: survey and future challenges," in *Proceedings of the 2019 International Workshop on Multilayer Music Representation and Processing (MMRP)*, pp. 10–18, IEEE, Milan, Italy, January 2019.

[9] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.

[10] W. W. Graves, O. Boukrina, E. J. A. Mattheiss, E. J. Alexander, and S. Baillet, "Reversing the standard neural signature of the word-nonword distinction," *Journal of Cognitive Neuroscience*, vol. 29, no. 1, pp. 79–94, 2017.

[11] Y. Wu and W. Li, "Automatic audio chord recognition with MIDI-trained deep feature and BLSTM-CRF sequence decoding model," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 2, pp. 355–366, 2018.

[12] Korzeniowski and G. Widmer, "Genre-agnostic key classification with convolutional neural networks," in *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, September 2018.

[13] G. E. Durán, *Computer System for Harmonic Transcription of Jazz Music*, Pontificia Universidad Católica de Chile, Santiago, Chile, 2020.

[14] S. D. You, C. H. Liu, and W. K. Chen, "Comparative study of singing voice detection based on deep neural networks and ensemble learning," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, pp. 1–8, 2018.

[15] H. M. Huang, W. K. Chen, C. H. Liu, and S. D. You, "Singing voice detection based on convolutional neural networks," in *Proceedings of the 2018 7th International Symposium on Next Generation Electronics*, Taipei, Taiwan, May 2018.

[16] W. W. Graves, O. Boukrina, S. R. Mattheiss, E. J. Alexander, and S. Baillet, "Reversing the standard neural signature of the word–nonword distinction," *Journal of Cognitive Neuroscience*, vol. 29, no. 1, pp. 79–94, 2017.

[17] B. McFee and B. Juan Pablo, "Structured training for large-vocabulary chord recognition," in *Proceedings of the 18th International Society for Music Information Retrieval Conference*, pp. 188–194, Suzhou, China, 2017.

[18] F. Korzeniowski and G. Widmer, "Improved chord recognition by combining duration and harmonic language models," 2018, https://arxiv.org/abs/1808.05335.

[19] H.-W. Dong and Yi-H. Yang, "Convolutional generative adversarial networks with binary neurons for polyphonic music generation," 2018, https://arxiv.org/abs/1804.09399.

[20] C. Jin, W. Zhao, and H. Wang, "Research on objective evaluation of recording audio restoration based on deep learning network," *Advances in Multimedia*, vol. 2018, Article ID 3748141, 13 pages, 2018.

[21] T. Gong, "Deep belief network-based multifeature fusion music classification algorithm and simulation," *Complexity*, vol. 2021, Article ID 8861896, 10 pages, 2021.

[22] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, and T. Sainath, "Deep learning for audio signal processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 206–219, 2019.

[23] E. D. Brown, M. L. Garnett, K. E. Anderson, and J.-P. Laurenceau., "Can the arts get under the skin? Arts and cortisol for economically disadvantaged children," *Child Development*, vol. 88, no. 4, pp. 1368–1381, 2017.

[24] J. Niemeyer, F. Rottensteiner, U. Soergel, and C. Heipke, "Hierarchical higher order crf for the classification of airborne lidar point clouds in urban areas," *ISPRS—International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pp. 655–662, Prague, Czech Republic, July 2016.

[25] L. Turchet, C. Fischione, G. Essl, D. Keller, and M. Barthet, "Internet of musical things: vision and challenges," *IEEE Access*, vol. 6, pp. 61994–62017, 2018.

[26] S. Dang, S. Chaudhury, B. Lall, and P. K. Roy, "Learning effective connectivity from fMRI using autoregressive hidden Markov model with missing data," *Journal of Neuroscience Methods*, vol. 278, pp. 87–100, 2017.

[27] M. J. Baucas and P. Spachos, "Using cloud and fog computing for large scale iot-based urban sound classification," *Simulation Modelling Practice and Theory*, vol. 101, Article ID 102013, 2020.

[28] J. Lee, J. Park, K. Kim, and J. Nam, "Samplecnn: end-to-end deep convolutional neural networks using very small filters for music classification," *Applied Sciences*, vol. 8, no. 1, p. 150, 2018.

[29] F. Radenović, G. Tolias, and O. Chum, "Fine-tuning CNN image retrieval with no human annotation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1655–1668, 2018.

[30] F. Korzeniowski and G. Widmer, "Genre-agnostic key classification with convolutional neural networks," in *Proceedings of the 19th International Society for Music Information Retrieval Conference*, Paris, France, 2018.

[31] T. Cho and J. P. Bello, "On the relative importance of individual components of chord recognition systems," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 2, pp. 477–492, Feb. 2014.

[32] A. Faraldo, S. Jordà, and P. Herrera, "A multi-profile method for key estimation in EDM," in *Proceedings of the AES International Conference on Semantic Audio*, Erlangen, Germany, June 2017.

[33] C. Cannam, M. Mauch, M. E. Davies et al., "MIREX 2016 entry: vamp plugins from the centre for digital music," Technical report, MIREX, Santo Domingo, Dominican Republic, 2016.