sensors

MDPI

Review

Time Signature Detection: A Survey

Jeremiah Abimbola, Daniel Kostrzewa * and Pawel Kasprowski

Department of Applied Informatics, Silesian University of Technology, 44-100 Gliwice, Poland; jeremiah.oluwagbemi.abimbola@polsl.pl (J.A.); pawel.kasprowski@polsl.pl (P.K.)
* Correspondence: daniel.kostrzewa@polsl.pl

Abstract: This paper presents a thorough review of methods used in various research articles published in the field of time signature estimation and detection from 2003 to the present. The purpose of this review is to investigate the effectiveness of these methods and how they perform on different types of input signals (audio and MIDI). The results of the research have been divided into two categories: classical and deep learning techniques, and are summarized in order to make suggestions for future study. More than 110 publications from top journals and conferences written in English were reviewed, and each of the research selected was fully examined to demonstrate the feasibility of the approach used, the dataset, and accuracy obtained. Results of the studies analyzed show that, in general, the process of time signature estimation is a difficult one. However, the success of this research area could be an added advantage in a broader area of music genre classification using deep learning techniques. Suggestions for improved estimates and future research projects are also discussed.

Keywords: time signature; meter; metre; measure signature; music information retrieval; signal processing; deep learning

Citation: Abimbola, J.; Kostrzewa, D.; Kasprowski, P. Time Signature Detection: A Survey. *Sensors* **2021**, *21*, 6494. https://doi.org/10.3390/s21196494

Academic Editor: Stefania Perri

Received: 27 August 2021
Accepted: 25 September 2021
Published: 29 September 2021

1. Introduction

The majority of popular music scores are composed in a particular style known as lead sheet format. It summarizes a song by representing the notes of the main theme, the chord series, and other cues such as style, tempo, and time signature. Symbols are used in standard staff music notation to denote note duration (onset and offset times). Onset notes refers to the beginning of a notation or another sound and all musical notes have an onset, but do not always contain the first transient [1]. Offset is about the duration of the part from the beginning of the piece. It is the sum of the previous duration only when there is no rest and there are no place where two notes play together [1,2]. In addition, the staff contains details about the tempo, the beginning, the end of the bars, and the time signature. The time signature (sometimes referred to as a meter signature or metre signature) is a symbol for western music which specifies the number of beats (pulses) in each measure (bar) [3]. It is defined as a ratio of two integer numbers, where the numerator indicates the number of beats in a bar and the denominator specifies the note relation [4]. There are simple and compound time signatures that are relatively easy to estimate from the lead sheet or audio files. Examples include $\frac{2}{2}$, $\frac{3}{4}$, $\frac{4}{4}$, or $\frac{6}{8}$ which means 2 minim beats, 3 crotchet beats, 4 crochets beats, and 6 quaver beats in a bar, respectively. The compound signatures are a multiples of the simple time signatures in terms of the number of beats [5]. Examples include $\frac{6}{8}$, $\frac{9}{8}$, $\frac{12}{8}$. There are also irregular time signatures that are much more difficult to estimate [6]. Examples include $\frac{5}{8}$, $\frac{7}{8}$, and $\frac{11}{8}$. Time signature estimation and detection cannot be possible without understanding the concept of upbeat, downbeat, and anacrusis. Upbeats and downbeats represent the simplest manner of associating downward motions with melodic movements to metrically stable points [7]. "Down" beats are times of stronger metric stability in the field of meter [8]. Anacrusis was defined by Lerdahl and Jackendoff [9,10] in two ways: "from the start of the group to the most powerful beat inside

a group" and "from an upbeat to its related downbeat". With anacrusis in mind, estimating the time signature becomes tougher when the first note of the piece is not the strongest note. Additionally, the idea of strong beats must be considered in this process because strong beat spectrum peaks will occur during repetition moments in highly organized or repeated music. This shows both pace and relative intensity of certain beats, so that different types of rhythms may be distinguished at the same timing [11–13].

The music industry is fast-growing and with songs being produced every day, curators like Apple Music, Spotify, Audio Mack, etc. need a genre classification system to accurately curate playlist for their users. This involves grouping musical data together based on defined similarities such as rhythm—time signature and tempo—or harmonic content. In this domain, many attempts have been made to classify various genres of songs with notable successes [14–17]. However, with extracted features, such as time signature, the overall accuracy could get even better but this area is comparatively unexplored by reason of the estimation being difficult.

Estimating time signature is a challenging task because all the beat times after the downbeat (strong beat) before the next downbeat do not always correspond to the total number of beats in a bar, especially for audio music. The main reason for this is because the tempo of any music track affects the time signature significantly. Beat times here refers to the time in seconds for a beat to sound relative to the entire duration of the track. For example, a track of 80 bpm with beat times as 1.02, 2.03, 3.03, 4.02 could estimate as $\frac{4}{4}$; 1.02 being the downbeat time, whereas the same track played with 120 bpm could have beat times as 1.02, 1.33, 1.82, 2.13 which absolutely cannot be estimated as a $\frac{4}{4}$ if it is assumed that a second beat time corresponds to a beat. Therefore, a couple of factors need to be put into consideration to accurately estimate the time signature, namely upbeat, downbeat, anacrusis, onset note, and tempo. However challenging, the automatic detection of time signature could help to reduce computational time for other temporal processes such as beat tracking, tempo estimation, and deep learning techniques. Moreover, it can be a preprocessing step to other tasks, such as gathering knowledge about music, automatic tagging of songs, improving genre classification, and recommendation systems.

Understanding time signature and its function in feature extraction, estimation, and music genre classification would open the door to new possibilities for the music information retrieval domain [18,19]. Feature extraction for audio signal analysis is one of the most important steps for further studies related to time signature detection [20]. YAAFE [21], an audio feature extraction software, was developed in 2010 which has features including, but not limited to, speech/music discrimination, music genre, or mood recognition, and as a result, there has been improvement over the years. Garima Sharma et al. [22] also highlights the trends of various methods that have been used to extract audio signal features.

A significant amount of study has been conducted on the automated retrieval of meta data from musical audio signals. Pitch detection [23,24], onset detection [25,26], key signature estimate [27], and tempo extraction [28,29] are some of the meta data obtained by various algorithms. The aim is two-fold: to equip computers with the capabilities of a human music listener in order to interpret a piece of music and derive explanations of specific musical properties. This enables a variety of applications, including automated transcription, playlist generation, and Music Information Retrieval (MIR) systems as is always discussed at every International Symposium on Music Information Retrieval (ISMIR) [30].

The algorithms that have been employed so far can be divided into two major approaches: the classical and the deep learning approach. The classical or manual approach involves using methods in the digital signal processing domain as evident in [31] by Meinard et al. in a study that showed how a piece of music can be analyzed by signal processing, by using comb filters [32] proposed by Klapuri just to mention a few. In an attempt to estimate time signature, one must have a sound knowledge about concepts such as frequency, tones, notes, duration, timbre, audio spectrum, beats, tempo, and timing.

The deep learning approach, on the other hand, makes use of deep learning models. There are ideas that are common to both methods such as the use of Fourier transforms and analysis and the conversion of audio signals to log spectrograms—a more scientifically usable form. It is important to note that the majority of these approaches were implemented using MATLAB [33–35] and C++ for collaborations, testing and method validation up until around 2015 and beyond that Python became the go-to. This interest was sparked by several reasons, such as ease of understanding the language and the availability of high-quality machine study libraries, like scikit-learn [36] and librosa [37], just to name a few.

This paper summarizes and reviews numerous research in this field, taking into account similar works, datasets, and a possible road map. To the best of our knowledge, no paper has ever conducted a survey on time signature detection or estimation. As a result, it is important that this survey be conducted in order to identify potential paths for creative ideas. Additionally, in this study, in the course of exploration, a deeper knowledge of frequencies in the time domain is obtained which may be useful in other domain areas like medicine and psychology, which have referred to beats as the pulse of the heart.

The study is organized as follows. Section 2 discusses the music input signals and their impact on the methodologies. In Section 3, datasets utilized in this domain are highlighted in depth. In Section 4, state-of-the-art classical approaches are described, while in Section 5, deep learning approaches are examined in depth. Section 6 concludes the paper with a review of the results achieved and the proposed future course for this domain.

2. Musical Input Signals

Time signature estimation can be carried out by using two types of input data: music audio samples or Musical Instrument Digital Interface (MIDI) signals. The music audio samples basically refer to compressed sample files like mp3, uncompressed files like wav, or any other audio format usable with a range of roughly 20 to 20,000 Hz, which corresponds to the lower and upper limits of human hearing. For example, the audio signal on a compact disc is limited to a maximum frequency of 20 kHz, sampled at 44.1 kHz and encoded, with 16 bits per sample [38] and nearly perfect audio signals are obtained with 64 kb/s [38]. Sampling in music refers to the use of a part (or sample) of a sound file of another recording. Samples can be layered, equalized, sped up or slowed down, re-pitched, looped, or otherwise manipulated and can include elements such as rhythm, harmony, voice, vibrations, or whole bars of music [31].

On the other hand, the MIDI is a standard digital interface for communication with a musical instrument and other associated audio devices for performing, editing, and recording music [39]. A MIDI music piece's sound quality is determined by the synthesizer (sound card), and has other restrictions, such as the inability to save voice, which takes up far less space, making it much easier to store, share, adjust, and manipulate as well as being universally accepted and allowing for greater comparison between music works played on various instruments [40]. This is why some researchers prefer this format. A summary of their differences is shown in Table 1. As a result, this section has been divided into two subsections in order to properly understand how these input signals have impacted previous studies.

2.1. Audio Samples as Data

An audio signal can be analyzed at three levels in a time scale as discovered by Klapuri et al. [41]: at the temporally atomic tatum pulse level, the tactus pulse level that corresponds to a piece's tempo, and the harmonic measure level as shown in Figure 1. Christian Uhle et al. [42] as the pioneers of this research area in 2003, were very much interested in estimation and detection of three basic rhythm features: tempo, micro-time, and time signature in which musical pieces can be partly characterized by. The estimation of these three features was combined and individually separated by the integer ratios between them. The process involved the decomposition of four-second audio signal samples into

frequency bands, a high-pass filter was applied—as the human ear cannot perceive sounds below 20 Hz [43], half-wave rectified amplitude envelopes were used to track onsets notes, and the filtered signal envelopes of each band were removed.

Table 1. Differences between the input signals.

Criteria	MIDI	Digital Audio
Definition	A MIDI file is a computer software that provides music info.	A digital audio refers to digital sound reproduction and transmission.
Pros	Files of small size fit on a disk easily. The files are perfect at all times.	The exact sound files are reproduced. It replicates superior quality.
Cons	There is variation from the original sound.	They take more disk space with more minutes of sound, files can get corrupted with a little manipulation.
Format Type	Compressed.	Compressed.
Information Data	Does not contain any audio information.	Contains recorded audio information.

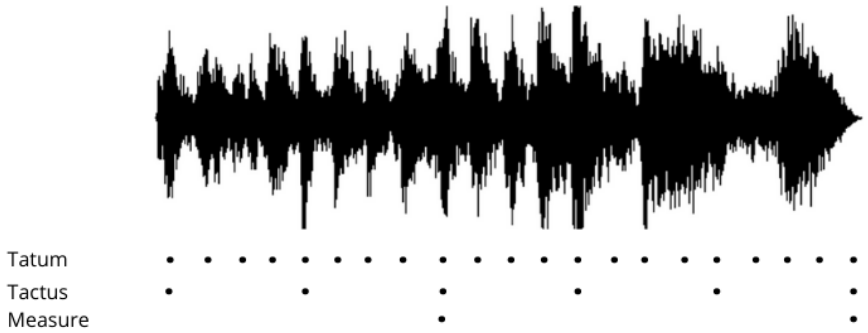

Figure 1. An audio signal with three metrical levels illustrated: tatum, tactus, and measure levels.

The inter-onset intervals (IOIs) are then determined from the note onset times, and the tatum duration is measured using an IOI histogram. Using an auto-correlation system, periodicities in the temporal progression of the amplitude envelopes are observed in the subsequent processing. The auto-correlation function peaks refer to the time lags at which the signal is most close to itself. The envelopes of two segments are accumulated in advance, allowing for the measurement of a bar duration of up to four seconds. This estimate is inspired by the assumptions that self-similarity exists at the tatum, beat, and bar levels [44]. The algorithm's [42] output was evaluated using 117 samples 8-second-long each of percussive music. Music from different backgrounds and cultures, such as West African, Brazilian, and Japanese folkloristic music, and solo drum-set performance, were included in the test results. The presented research technique calculated tempo, micro time, and time signature from percussive music. A total of 84.6 percent of the tempo values, 83.8 percent of the micro duration, and 73.5 percent of the time signatures were accurately measured from 117 quotations of eight seconds length. However, this approach does not explicitly gives the estimation in terms of the numerator and denominator of the time signature which is our main focus.

2.2. MIDI Signals as Data

Nowadays, the MIDI signals are not really used anymore for this task because technology has provided better options, however, they were famously used back then because they are easier to work with owing to the precise signal patterns. For instance, the detection of

onset notes can be obtained more precisely [11,45] because of the patterns that exist among the notes and a lot of researchers have exploited this advantage. Although it would be outstanding if a DAW like Logic Pro X can automatically determine the time signature by dragging a MIDI file into it, today, this is not common practice as MIDI data can adapt to any tempo and time signature specified. Grohganz et al. in [46] showed that the musical beat and tempo information is often defined in the MIDI files at a preset value that is not associated with the actual music content, so they introduced the method for determining musical beat grids in the provided MIDI file. They also showed, as a major addition, how the global time signature estimate may be utilized to fix local mistakes in the Pulse Grid estimate. Unlike the digital audio signal, when the notes are not perfectly on the grid, they could be quantized first before any process of time estimation is done.

The assumption that the MIDI track is repetitive almost throughout the song was also used by Roig et al. in [47], and similar to the ASM, the Rhythm Self Similarity Matrix (RSSM) was employed for this study. In order to construct the RSSM using the tactus as a measuring unit, the rhythmic elements will be divided into the number of tactus corresponding to their duration. As a result, the inter onset interval (IOI) of each note is separated into tactus intervals.

3. Datasets

In every classification, estimation, or detection project, the dataset selection is critical. Sometimes, there are a range of potentially viable datasets available, each with their own set of advantages and drawbacks, and the decision to choose one dataset over another may have a huge impact on the project's outcome [48]. The journey of obtaining robust and well-balanced datasets has seen a shift from a very simple set to attempts at providing larger and more diverse datasets as shown in Table 2.

Table 2. Datasets and their statistics.

Dataset Name	Year Created	Number of Samples	Data Samples
RWC [49]	2002	365	Audio
CAL500 [50]	2008	502	MIDI
GZTAN [51]	2002	1000	Audio
USPOP [52]	2002	8752	MIDI
Swat10K [53]	2010	10,870	MIDI
MagnaTagATune [54]	2009	25,863	Audio
FMA [55]	2016	106,574	Audio
MusicCLEF [56]	2012	200,000	Audio
MSD [57]	2011	1,000,000	CSV

The RWC dataset [49] was one of the first set of datasets that was put together solely for academic purposes. Shared libraries that made important contributions to scientific advancements were popular in other fields of scholarly study. It includes six original collections: the Popular Music Database (100 songs), the Royalty-Free Music Database (15 songs), the Classical Music Database (50 pieces), the Jazz Music Database (50 pieces), the Music Genre Database (100 pieces), and the Musical Instrument Sound Database (50 instruments). The data files of this dataset consist of audio signals, corresponding regular MIDI archives, and text files with lyrics all totaling 365 musical pieces performed and recorded. It also takes account of individual sounds at half-tone intervals with a variety of playing techniques, dynamics, instrument makers, and musicians. This collection served as a baseline to which researchers tested and analyzed different structures and methods. Unfortunately, this dataset is very small and unbalanced.

Six years later, Ju-Chiang Wang et al. created another dataset, CAL500 [50], for music auto-tagging as an improvement of the RWC datasets with about 502 songs but the audio files are not provided in the dataset. The tag labels are annotated in the segment level

instead of the track level. Unfortunately, 502 songs is inadequate to get better and accurate results for auto-tagging.

The evolution of datasets in the music domain or music information retrieval space cannot be discussed without mentioning the GTZAN dataset [51] collected by G. Tzanetakis and P. Cook. It is by far the most popular dataset out there containing 1000 song excerpts of 30 s, sampling rate 22,050 Hz at 16 bit collected from various sources including personal CDs, radio, microphone recordings, and so on. Its songs are distributed evenly into 10 different genres: Blues, Classical, Country, Disco, Hip Hop, Jazz, Metal, Pop, Reggae, and Rock. Since its publication in 2002, the GTZAN has been widely used in music genre classification analysis [58–62]. It was selected mostly because it was well-organized and widely quoted in previous studies. This precedent lends authority while also providing a frame of reference for results. However, there are a few disadvantages to using this dataset. Its relatively small size is the most limiting factor.

Mandel and Ellis created USPOP [52], centered only on popular artists with over 8752 audio songs without the raw file provided. Obviously, this is not a good dataset as its skewing can be questioned. Skewed datasets usually have a very high impact on the solutions they are used for as highlighted in these studies [63–65].

Chris Hartes, in 2010, created the Beatles datatset [66] which contains 180 songs and was well annotated by the musicologist Alan W. Pollack. Each music recording contains on average 10 sections from 5 unique section-types. It was one of the datasets used to generate the Million Song Dataset.

Another notable dataset is SWAT10K [67]. This dataset was obtained from the Echo Nest API in conjunction with Pandora, having 10,870 audio songs that are weakly labeled using a tag vocabulary of 475 acoustic tags and 153 genre tags with the files also not provided. For developers and media firms, the Echo Nest is a music intelligence and data platform located in Somerville, MA bought by Spotify in 2014. The Echo Nest originated as an MIT Media Lab spin-off to investigate the auditory and textual content of recorded music. Its designer's intentions for the APIs are for music recognition, recommendation, playlist construction, audio fingerprinting, and analysis for consumers and developers [68]. Pandora is a subscription-based music streaming service headquartered in Oakland, California. It focuses on suggestions based on the "Music Genome Project", a method of categorizing individual songs based on musical characteristics. Like the SWAT10K, MagnaTagATune [54] which has 25,863 audio files provided as csv was also created based on the Echo Nest API. Another dataset for popular music is the MusicCLEF [56] with 200,000 audio songs provided for research purpose.

The Free Music Archive (FMA) by Defferrard et al. [55] contains over 100,000 tracks, each with its own genre label. There are many variations of the dataset available, ranging from the *small* version (8000 30-s samples) to the *full* version (all 106,574 songs in their entirety). The size of this dataset makes it suitable for labeling, and the fact that the audio files are available for download ensures that features can be derived directly from the audio.

The Million Song Dataset (MSD) [57] is a set of audio features and metadata for a million contemporary songs (as the name implies) that is publicly accessible. Release year, artist, terms of the artist, related artists, danceability, energy, length, beats, tempo, loudness, and time signature are among the metadata and derived features included in the dataset although audio files with proper tag annotations (top-50 tags) are only available for about 240,000 previews of 30 s [69].

A very recent dataset, Augmented Maps (A-MAPS) [70] was created in 2018 with no precise number of MIDI files specified. However, it is the most common dataset used for automatic transcription of music. Adrien Ycart et al. updated the previous version of the original MIDI files, containing onset, offsets, and additional annotations. The annotations include duration of notes in fraction relative to a $\frac{1}{4}$th note (a crotchet), tempo curve, time signature, key signature (annotated as a relative major key), separate left and right-hand staff, and text annotations from the score (tempo indications, coda). However, due to MIDI format constraints, they do not contain all of the details required for staff-notation

music transcription. It is difficult to say how this dataset was obtained because the original dataset MAPS is not readily available at the time of writing this paper.

Among all these datasets, having seen their advantages and drawbacks, the two that seem very useful to this review in terms of time signature extraction are the FMA and the Million Song Dataset which are both extracted from the Echo Nest API. However, the metadata from the MSD have been pre-processed which makes it difficult to know how it was carried out, although there is a confidence level for the data we are most interested in (time signature).

4. Classical Methods

The methods discussed in this section consists of digital signal processing of audio samples tasks such as window framing in Figure 2, filtering and Fourier analysis [71]. Audio tracks are usually divided into perceivable audio chunks known as frames where 1 sample at 44.1 KHz is 0.0227 ms. This time is far shorter than what the human ear can meaningfully resolve—10 ms. Therefore in order to avoid spectral leakage, a windowing function is applied which eliminates samples at both ends of the frame hence the importance of the frame overlap to have a continuous signal again. Some of these processes will be explained in detail and a brief summary can be found in Table 3.

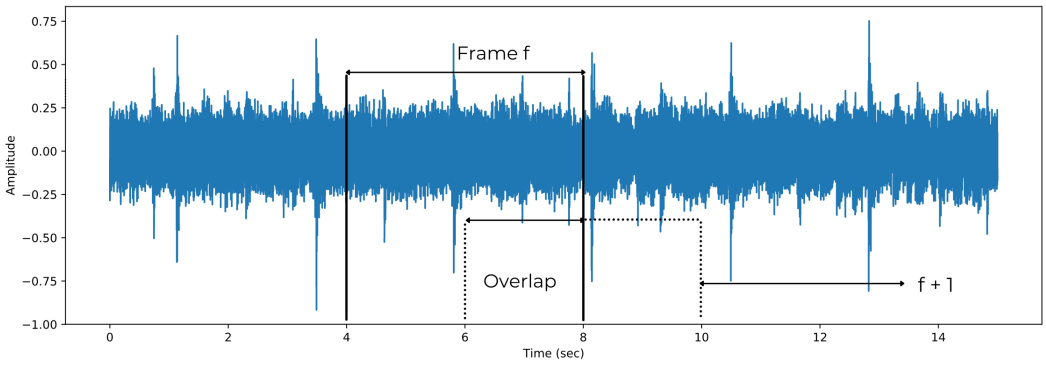

Figure 2. One of the most common methods of audio preprocessing—splitting the whole signal into frames with overlapping.

Table 3. Summary of classical estimation methods.

Year	Method	Dataset	Data	Accuracy (%)
2003	SVM [72]	Self generated	Audio	83
2003	ACF [42]	Excerpts of percussive music	Audio	73.5
2004	SSM [73]	Greek music samples	Audio	95.5
2007	ASM [74]	Commercial CD recordings	Audio	75
2009	ACF, OSS [75]	Usul	MIDI	77.8
2009	BSSM, ASM [76]	Generated samples	Audio	95
2011	Comb Filter [77]	Indian Music DB	Audio	88.7
2013	SVM [78]	Generated Samples	Audio	90
2014	RSSM [47]	MIDI keyboard scores	MIDI	93
2020	Annotation Workflow [79]	ACMUS-MIR	Audio	75.06

Aggelos Pikrakis et al. in [73] presented an extraction method for time signature which was referred to as meter. This method was also based on the assumption that the music meter was constant throughout the audio signals. Their assumption was valid given the type of music they used for the estimation—300 raw audio samples of Greek traditional dance music whose tempo ranges from 40 bpm to 330 bpm. It is important to note that there is a huge relationship between the speed of any music track (in bpm) and the time signature, as pointed out by Lee in [80]. By considering a similar approach as the ASM, a self-similarity matrix was used for this experiment which showed that periodicities corresponding to music meter and beat are revealed at the diagonals of the matrix of the audio spectrogram.

Consequently, by examining these periodicities, it is possible to estimate both meter and beat simultaneously. In the first step, each raw audio recording was divided into non-overlapping long-term segments with a length of 10 s each. The meter and tempo of the music were removed segment by segment. A short-term moving window, in particular, produces a series of function vectors for each long-term fragment. The approximate values for the short-term window duration and overlap duration between successive windows are 100 ms and 97 ms, implying a 3 ms moving window phase. The overall result accounted for the successful extraction of the rhythmic features while most mistaken results were produced for meters such as 2/4 with 4/4 or 5/4; 7/8 with 3/4 or 4/4.

Since the ASM method proved to be effective, Gainza in [76] combined it with a Beat Similarity Matrix to estimate the meter of audio recordings. To begin, a spectrogram (a pictorial representation of the power of a signal or "loudness" of a signal over time at different frequencies of a specific waveform [81]) of the audio signal was generated using windowed frames with a length of $L = 1024$ samples and a hop size of $H = 512$ samples, which is half the frame length. Then, individual audio similarity matrices were calculated by comparing the spectrogram frames of the piece of music every two beats. Following that, a beat similarity matrix was constructed by combining similarity measures obtained from the individual audio similarity matrices. Finally, by processing the diagonals of the beat similarity matrix, the presence of identical patterns of beats was studied. The equation to obtain the matrix diagonals is defined as

$$X(m,k) = abs\left[\sum_{n=0}^{L-1} x(n+mH)w(n)^* e^{-j(2/\tau/N)k.n}\right] \quad (1)$$

$w(n)$ is a windowing function which in this case is the Hanning window that selects a L length block from the input signal $x(n)$, and m, N, H, and k are the frame index, fast Fourier transform (FFT) length, hop size, and bin number respectively; $k \in \{1 : N/2\}$. The choice of the window type function was based on previous studies [82,83]. The findings obtained demonstrate the robustness of the presented approach, with 361 songs from a database of quadruple meters, a database of truple meters, and another of complex meters yielding a 95% accuracy.

Furthermore, Gouyon and Herrera in [72] proposed a method to determine the meter of music audio signals by seeking recurrences in the beat segment. Several approaches were considered with the aim of testing the hypothesis that acoustic evidence for downbeats can be calculated on signal low-level characteristics, with an emphasis on their temporal recurrences. One approach is to determine which of the low-level audio features corresponding to specific meters were relevant for meter detection. This approach is limited because it was simplified to two-groupings only (duple and triple group meters) while not considering the cases for irregular meters. With a frame size of 20 ms and a hop size of 10 ms, features such as energy, spectral flatness, and energy in the upper half of the first bark band were extracted from each signal frame. Beat segmentation was also carried out as a different approach based on these features already extracted. For the study, a database of 70 sounds (44,100 Hz, 16 bit, mono) was used. Each extract is 20 s long. Bars for beginnings and endings were set at random, and music from Hip-hop, Pop, Opera, Classical, Jazz, Flamenco, Latin, Hard-rock, and other genres were included.

As a more advanced technique to this problem, they also considered the classification methods to assign the value for the meter: from a non-parametric model (Kernel Density estimation) to a parametric one (Discriminant Analysis), including rule induction, neural networks, 1-Nearest Neighbor (1-NN), or Support Vector Machines (SVMs). For this, on a frame-by-frame basis, the following features were computed: energy, zero-crossing rate, spectral centroid, spectral kurtosis, spectral skewness, two measures of spectral flatness (one is the ratio geometric mean/arithmetic mean and the other is the ratio harmonic mean/arithmetic mean), 13 Mel-Frequency Cepstrum Coefficients (MFCCs), and energy in 26 non-overlapping spectral bands. The evaluation showed that, when 27 features were

used, error rates for all cases were found to be less than 17.2% (the best technique, Naive Bayes, yielded just 5.8%, whereas a rule induction technique yielded 17.2%).

Meter detection was also studied from the aspect of breaking down the metrical structure of a single bar by Andrew and Mark in [84] using some excerpts from Bach which eventually gave a 80.50% F-measure. They started by using the hierarchical tree structure of notes as seen in Figure 3. This gave insight for evaluation on each of the three levels (sub-beat, beat, and bar) of the guessed metrical tree. If it matched exactly a level of the metrical tree, it was counted as a true positive and otherwise, a clash was counted as a false positive. In another study [85], they pushed this model furthermore to accurately detect the meter. The suggested model was based on two musicological theories: a reasonably steady rate of the tatum without great discontinuities and notes that are relatively similar to those tatums. Each state in the model represents a single bar, with a list of tatums from that bar and a metrical hierarchy defining which tatums are beats and sub-beats. The tatum list and the downbeat of the next bar are obtained. The tatums are listed in ascending chronological order. The metrical hierarchy of a state has a certain number of tatums per sub-beat, sub-beats per beat, and beats per bar, as well as an anacrusis duration, which is determined by the number of tatums that fall before the first downbeat of a given piece. The first downbeat position probability was also considered by De Haas et al. [86] with a model—Inner Metric Analysis (IMA). The number of tatum per sub-beat was restricted to 4. Although, in principle, this could be any number. The set of possible sub-beat per beat and beat per bar pairs (i.e., time signatures) are taken all of those found in our training set ($\frac{2}{X}$, $\frac{3}{X}$, $\frac{4}{X}$, $\frac{6}{X}$, $\frac{9}{X}$, and $\frac{12}{X}$), where X could be any value ranging from 1 to 12.

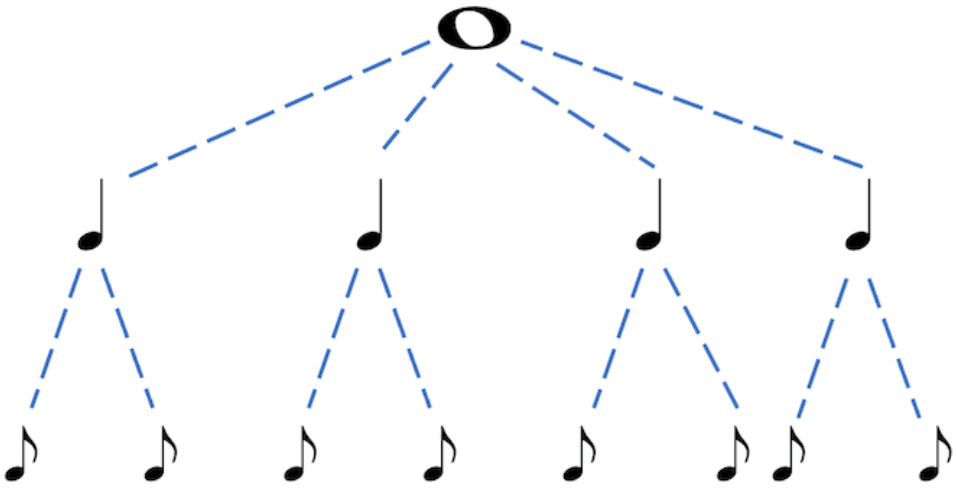

Figure 3. The hierarchical tree structure of notes—the metrical structure of a $\frac{4}{4}$ bar (1 whole note = 4 quarter notes = 8 eighth notes).

Gulati et al. then took on this very difficult challenge to estimate the meter of irregular time signature using the case study of Indian classical music in their study with meters of 7/8 [77]. The incoming audio stream is transformed to a mono channel after being downsampled to 16 kHz. The data is divided into 32 ms frames with a 5 ms hop size and a frame rate of 200 Hz. Each frame is subjected to a Hamming window, and a 512-point FFT is calculated. With 12 overlapping triangle filters that are equally spaced on the Mel-frequency scale, the frequency bins are reduced to 12 non-linear frequency bands. The time history of the amplitudes of each of these 12 bands is represented by a band envelope with a sampling frequency of 200 Hz (frame rate). The band envelope is then transformed to log scale (dB) and low pass filtered using a half-wave raised cosine filter. The meter vector $\vec{m}$ is obtained when narrow comb filter banks are set up around integer multiples of tatum duration retrieved from the differential signal. The number of comb filters implemented per filter bank is equal to twice the integer multiple of the tatum duration plus one to account for the tatum duration's round-off factor. For each filter bank, the filter with the

maximum output energy (i.e., with a certain delay value) is chosen, and the total energy of this filter over all Mel bands is calculated. The salience value for each feasible meter is calculated in Equations (2)–(4) i.e., for double, triple, and septuple. A simple rule-based technique is used to calculate the final meter value from $\vec{m}$.

$$S_2 = [\vec{m}(4) + \vec{m}(8) + \vec{m}(16)] \cdot \frac{1}{3} \tag{2}$$

$$S_3 = [\vec{m}(3) + \vec{m}(6) + \vec{m}(9) + \vec{m}(18)] \cdot \frac{1}{4} \tag{3}$$

$$S_7 = [\vec{m}(7) + \vec{m}(14)] \cdot \frac{1}{2} \tag{4}$$

A salience value for each conceivable meter is constructed, i.e., double, triple, and septuple, as shown in Equations (3)–(5), respectively. The ultimate meter of the song is determined by the sum of S2, S3, and S7.

Holzapfel and Stylianou in [75] set out to estimate the rhythmic similarities in Turkish traditional music and on this path, the time signature was estimated with a data set consisting of 288 songs distributed along the six classes of different rhythmic schemes (9/8, 10/8, 8/8, 3/4, 4/4, 5/8). Although this was not the aim of this research, he proposed a method for estimating the time signature because the overall study was compared to a start-of-the-art estimation technique which Like Uhle proposed in [42]. The onset periods are read from the MIDI files, and each onset is allocated a weight. After evaluating several strategies for assigning weights, the most popular scheme was adopted: the weight of an onset may be compared to the note length, to melody characteristics, or all onsets are assigned the same weight. To evaluate a piece's time signature, all pairwise dissimilarities between songs were computed using either the scale-free auto correlation function (ACF) or the STM vectors, and a cosine distance; a similar method was used in [87]. The same method used by Brown in [88] since it is a count of the number of events that occur during an occurrence at time zero if events are clustered from measure to measure, with a higher occurrence of an event happening with the measure's time isolation, therefore peaks in the auto-correlation function should show the periods when measurements begin [89]. A single melody line was extracted from the music score for analysis. This produced dissimilarity matrices with values close to zero when two parts were discovered to be alike in terms of rhythmic information. The accuracy of an updated k-Nearest Neighbor (kNN) classification was calculated in order to calculate the consistency of the proposed rhythmic similarity metric [90–93]. The power of a similarity matrix in this sphere lies with the distance between the notes in comparison. That is, the higher the distance, the lesser the similarity and vice versa. Hence the need to evaluate the impact of the value of K on the nearest neighbor. Each individual song was then used as a query for classification into one of the available groups. The dissimilarity matrix was classified using the modified kNN. The melodic line $x[n]$ was subjected to a short time auto-correlation calculation defined as

$$A[m] = \sum_{n=0}^{N-1} x[n]x[n+m] \tag{5}$$

where the average is taken over N samples and m is the auto-correlation time in samples.

Coyle and Gainza in [74] proposed a method to detect the time signature from any given musical piece by using an Audio Similarity Matrix (ASM). The ASM compared longer audio segments (bars) from the combination of shorter segments (fraction of a note). This was based on an assumption that musical pieces have repetitive bars at different parts. A spectrogram with a frame length equal to a fraction of the duration of the song's beat was generated using prior knowledge of the song's tempo; a technique asserted by Kris West in [94]. Following that, the song's first note was obtained. The reference ASM was then produced by taking the Euclidian distance between the frames beginning with the first note and this enables the parallels between minor musical incidents such as short notes to be

captured. Then, a multi-resolution ASM technique is used to create other audio similarity matrices representing different bar lengths. After computing all of the ASMs within a certain range, the ASM with the greatest resemblance between its components would conform to the bar duration and a technique for detecting the song's anacrusis—an anticipatory note or notes occurring before the first bar of a piece, is added. Finally, the time signature is estimated, as well as a more precise tempo measurement.

The music meter estimation problem can also be considered as a classification task as demonstrated by Varewyck et al. in [78]. Having considered the previous methods in this field that worked, they used the Support Vector Machine (SVM) for this purpose. Prior to the periodicity analysis, an external beat tracker was used to perform beat-level analysis, alongside, spectral envelope and pitch analysis were also carried out. Furthermore, a similarity analysis of the interval between two successive beats which they called Inter-Beat-Interval (IBI) already shown by Gouyon and Herrera (2003) [72] was performed. Hereafter, a hypothesis for the meter generated was developed and the meter was obtained. The similarity of the IBI was calculated using cosine similarity as shown in the equation below

$$CS(b) = \frac{\langle \vec{z}(b-1), \vec{z}(b) \rangle}{\|\vec{z}(b-1)\| \|\vec{z}(b)\|} \tag{6}$$

where b is the beat, $\vec{z}(b-1)$ and $\vec{z}(b)$ are low dimensional vectors grouped by related features. Eventually, they created an automatic meter classification method with the best combination of features that made an error of around 10% in duple/triple meter classification and around 28% in meter 3, 4, and 6 with a balanced set of 30 song samples.

Meter estimation for traditional folk songs is especially more challenging as much research is usually carried out on Western music. However, Estefan et al. in [79] made some attempt to estimate the meter and beat of Colombian dance music known as the bambuco. The bambuco has a superposition of $\frac{3}{4}$ and $\frac{6}{8}$ m but due to the caudal syncopation and the accentuation of the third beat, the case of downbeat does not hold for this type of music. With the ACMUS-MIR dataset (V1.1), a collection of annotated music from the Andes region in Colombia, they were able to perform beat tracking and meter estimation. For the study, 10 candidates were asked to tap to the rhythm in order to choose 10 bambuco packs with Sonic Visualiser's on the computer keyboard. There were two sets of annotations: (1) beats were taped while the audio was playing (without any visual information) and participants were not granted permission to make any adjustments. (2) Participants were permitted to change the Sonic Visualiser's first beat annotations using both audio and audio waveform visuals. Three musicologists from Colombia evaluated the beat annotations from these 10 participants in order to establish the underlying meters of each track. Each annotation was mapped directly to a certain meter, either $\frac{3}{4}$, $\frac{6}{8}$, or a combination; even though the participants were asked to naturally tap to the beats. They also performed beat tracking using two methods; madmon and multiBT while evaluating the F1 score for each perceived meter. For $\frac{3}{4}$, madmon had 76.05% while multiBT had 42.79% and for $\frac{6}{8}$, madmon had 41.13% while multiBT had 45.15%. In conclusion, in the annotations made by the research participants, five metric alternatives were discovered.

5. Deep Learning Techniques

Things are a little different with deep learning, because more knowledge is gathered. With deep learning, it is basically a neural network with three or more layers. Although a single-layer neural network may still generate approximate predictions, more hidden layers can assist optimize and tune for accuracy. In resolving several complicated learning issues, such as sentiment analysis, extraction of functions, genre classification, and prediction, Convolutional Neural Networks (CNNs) have been used extensively [95–97]. For tempo data such as audio signals and words sequencing, a hybrid model of CNNs and Recurrent Neural Networks (RNNs) was recently used [98]. Audio data is represented by frameworks and the sequential character of audio is entirely overlooked in the traditional RNN approach for temporal classification, hence the need for a well-modeled sequential network; the long-term recurrent neural network (LSTM) which has recorded successes

for a number of sequence labeling and sequence prediction tasks [99,100]. Convolutional-Recurrent Neural Networks (CRNNs) are complicated neural networks constructed by the combination of CNN and RNN. As an adapted CNN model, the RNN architecture is placed on CNN structure with the aim of obtaining local features using CNN layers and temporal summation by RNN networks. The main components for a CNN network are: input type, rate of learning, batches and architectural activation features, and the ideal type of input for music information collection is the mel-spectrogram [97]. Mel spectrograms are comprised of broad functionality for latent feature learning and onset and offset detection since the Mel scale has been shown to be similar to the human auditory system [81,101]. In order to obtain a mel-spectrogram signal, the pre-processing phase is necessary for STFT (Fourier short transform) and the log amplitude spectrogram. The methods in this section discussed and summarized in Table 4 consist of neural networks that extract the time signature as a feature that can be used as input for further calculation or classification problems in the MIR domain rather than estimating it exactly. Handcrafted features like Mel Frequency Cepstral Coefficients (MFCC), Statistical Spectrum Descriptors (SSD), and Robust Local Binary Patterns (RLBP) [102], used in deep learning are extracted based on human's domain knowledge [101]. However, these features have not been totally proven to be correlated to meter or time signature detection and their effectiveness and validity are not very clear.

Table 4. Summary of deep learning estimation methods.

Year	Method	Dataset	Accuracy (%)
2011	CNN [103]	MSD	Not stated
2016	CNN [104]	Multiple datasets	90
2017	CNN [69]	MSD, MagTagATune	88
2019	TCN [105]	Annotated dataset	93
2019	CNN [106]	MSD	89
2019	CRNN [107]	Beatles	72
2019	GMM-DNN [108]	Poetry corpus	86

Rajan et al. in [108] proposed a meter classification scheme using musical texture features (MTF) with a deep neural network and a hybrid Gaussian mixture model-deep neural network (GMM-DNN) framework. The proposed system's performance was assessed using a freshly produced poetry corpus in Malayalam, one of India's most widely spoken languages, and compared to the performance of a support vector machine (SVM) classifier. A total of 13 dim MFCCs were extracted using frame-size of 40 ms and frame-shift of 10 ms alongside seven other features; spectral centroid, spectral roll-off, spectral flux, zero crossing, low energy, RMS, and spectrum energy. Rectified linear units (ReLUs) were chosen as the activation function for hidden layers, while the softmax function was used for the output layer. These methods produce an accuracy of 86.66 percent in the hybrid GMM-DNN framework. The overall accuracies for DNN and GMM-DNN were 85.83 percent and 86.66 percent, respectively.

5.1. Convolutional Neural Netorks

In a study conducted by Sander Dieleman et al. in [103] where unsupervised pre-training was performed using the Million Song Dataset, the learnt parameters were transferred to a convolutional network with 24 input features. Timbre properties from the dataset were presented to the network as shown in Figure 4. Two input layers composed of chroma and timbre characteristics were stacked with separate convolution layers and the output of these layers was then maxpooled. The performance of the max-pooling layer was invariant to all displacements of less than one bar (up to 3 beats). The accuracy in terms of time signature was not stated since this was not the main objective of the research.

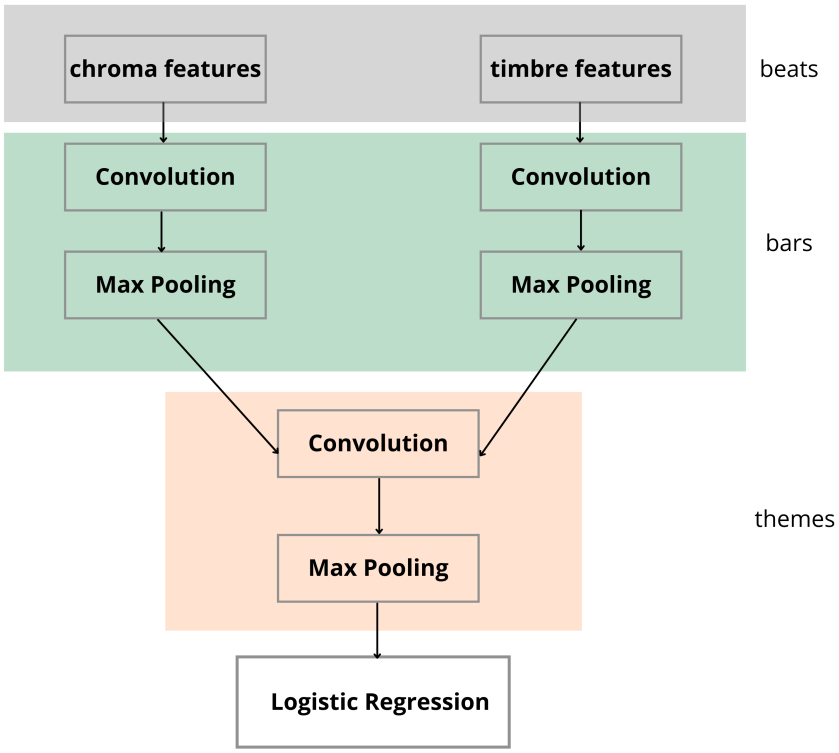

Figure 4. The convolutional neural network architecture block diagram with two kinds of input features (chroma and timbre).

Sebastian Böck et al. [105] showed that tempo estimation can be achieved by learning from a beat tracking process in a multi-task learning algorithm since they are highly interconnected; a method that has been used in other research areas for optimization [109,110]. This approach proved effective in that mutual information of both tasks was brought forward by one improving the other. The multi-task approach extends a beat tracking system built around temporal convolutional networks (TCNs) and feeds the result into a tempo classification layer. Instead of using raw audio as data input, dilated convolutions are applied to a heavily sub-sampled low-dimensional attribute representation. This 16-dimensional function vector is generated by adding several convolution and max pooling operations to the input audio signal's log magnitude spectrogram. The log magnitude spectrum is obtained because this is what the human ear can perceive [111,112]. The spectrogram is produced using a window and FFT size of 2048 samples, as well as a hop size of 441 samples. The convolutional layers each have 16 filters, with kernel sizes of 3×3 for the first two layers and 1×8 for the final layer. The method was tested on a variety of existing beat- and tempo-annotated datasets, and its success was compared to reference systems in both tasks. Findings show that the multi-task formulation produces cutting-edge efficiency in both tempo estimation and beat recording. The most noticeable improvement in output occurs on a dataset where the network was trained on tempo labels but where the beat annotations are mostly ignored by the network. The underlying beat tracking system is inspired by two well-known deep learning methods: the WaveNet model [38] and the latest state-of-the-art in musical audio beat tracking, which employs a bi-directional long short-term memory (BLSTM) recurrent architecture. To train the system, annotated beat training data as impulse were represented at the same temporal resolution as of the input feature (i.e., 100 frames per second) and different datasets were used for this training and eventual evaluation, unlike other approaches where one single dataset is divided into training and test sets.

Tracking meter at a higher metrical level is a task pursued under the title of downbeat detection [113]. Therefore we can also consider downbeat detection with deep learning features. Durand and Essid in [104] suggested a random field method conditioning an audio

signal's downbeat. In the first instance the signal generated four additional characteristics pertaining to harmony, rhythm, melody, and bass, and the tatum level was separated. Adapted convolutional neural networks (CNN) were then used for feature learning based on each feature's characteristics. Finally, a feature representation concatenated from the networks' final and/or penultimate layers was used to describe observation feature functions and fed into a Markovian model of Conditional Random Field (CRF) that produced the downbeat series. The model was evaluated using a Friedman's test and a Tukey's honestly significant criterion (HSD) and was found to have a F-measure improvement of +0.9% using features from the last layer and 95% confidence interval.

With a deep learning approach, music domain assumptions are relevant when not enough training data are available as suggested by Pons et al. in a study done recently in 2017 [69]. They were able to automatically categorize audio samples using waveforms as input and a very small convolutional filter on a convolutional neural network—a common architecture for music genre classification as shown in Figure 5 and thus indirectly calculated various attributes, one of which was the meter. The CNN architecture was divided into input, front-end, back-end, and output for easy implementation. The front-end that takes in the spectrogram is a single-layer CNN with multiple filter shapes divided into two branches: top branch—timbral features, and lower branch—temporal features. The shared backend is made up of three convolutional layers (each with 512 filters and two residual connections), two pooling layers, and a dense layer. With two models combined where one implemented classical audio features extraction with minimal assumption and the other dealt with spectrograms, and a design that heavily relies on musical domain knowledge, meter tags were obtained.

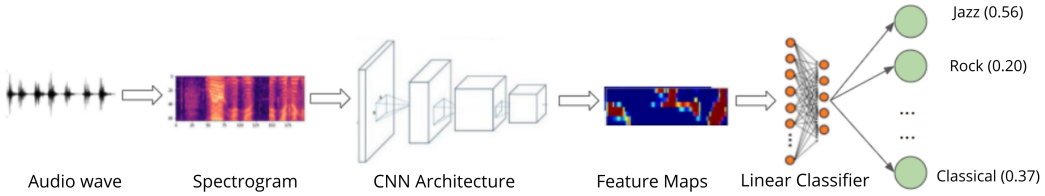

Figure 5. A typical convolutional neural network architecture used to time signature detection—audio signal processed into spectrogram which is an input to convolutional layers, and then an outcome is an input to classical artificial neural network.

This kind of pipeline was also suggested by Humphrey et al. [114] where it was advocated to move beyond feature design to automatic feature learning. A fraction of the Million Song Dataset alongside the MagnaTagATune (25 k songs) which have been mentioned in the dataset section, and a private dataset of 1.2 M songs were combined together to validate the two distinct music auto-tagging design approaches considered. The result of this study brought about an approach to learn timbral and temporal features with 88% accuracy.

Purwins et al. in [106] showed how deep learning techniques could be very useful in audio signal processing in the area of beat tracking, meter identification, downbeat tracking, key detection, melody extraction, chord estimation, and tempo estimation by processing speech, music, and environmental sounds. Whereas in traditional signal processing, MFCCs are the dominant features; in deep learning the log-mel spectrograms (see Section 1) are the pre-dominant features. As confirmed by Purwmins, the convolutional neural networks have a fixed flexible field, which limits the temporal context taken into account for a prediction while also making it very simple to expand or narrow the context used. While it was not explicitly stated which of the three popular methods of deep learning performs the best, the data used sometimes determines the method to be used. For this analysis, the Million Song Dataset was chosen to reduce a 29 s log-mel spectrogram to an 1×1 feature map and categorized using 3×3 convolutions interposed with max-pooling which yielded a good result of 0.894 AUC.

5.2. Convo-Recurrent Neural Networks

Fuentes et al. in [107] combined a non-machine learning approach as well as deep learning to estimate downbeat and in the process extract the time signature. The deep learning approach was a combination of a convolutional and recurrent network which they called CRNN proposed in their previous work [115]. By using the Beatles dataset because of its peculiarity in annotated features such as beats and downbeats, they considered a set of labels Y which represents the beat position inside a bar, then took bar lengths of 3 and 4 beats, corresponding to 3/4 and 4/4 m. The output labels y are a function of two variables: the beat position $b \in \mathcal{B} = \{1, \ldots, b_{max}(r)\}$ and the number of beats per bar $r \in \mathcal{R} = \{r_1, \ldots, r_n\}$, which relates to the time signature of the piece. The model experienced a level of success but it was incapable of identifying rare music variations in order to fit the global time signature consistently. For example, it estimated more 4/4 pieces than 3/4. Consequently, this model improves the downbeat tracking performance of the mean F-measure from 0.35 to 0.72.

6. Conclusions and Future Pathways

In this paper, we presented a summary of different methods for estimating time signature in music, considering both state-of-the-art classical and deep learning methods with a focus on the dataset used and the accuracy obtained as shown on the dendrogram in Figure 6. Since there has not been a study like this, there is a need for this analysis. The history of datasets has also been explored in terms of their production processes and purposes. The experiments that have been conducted so far have produced promising findings, indicating that time signature may be a significant feature of music genre classification.

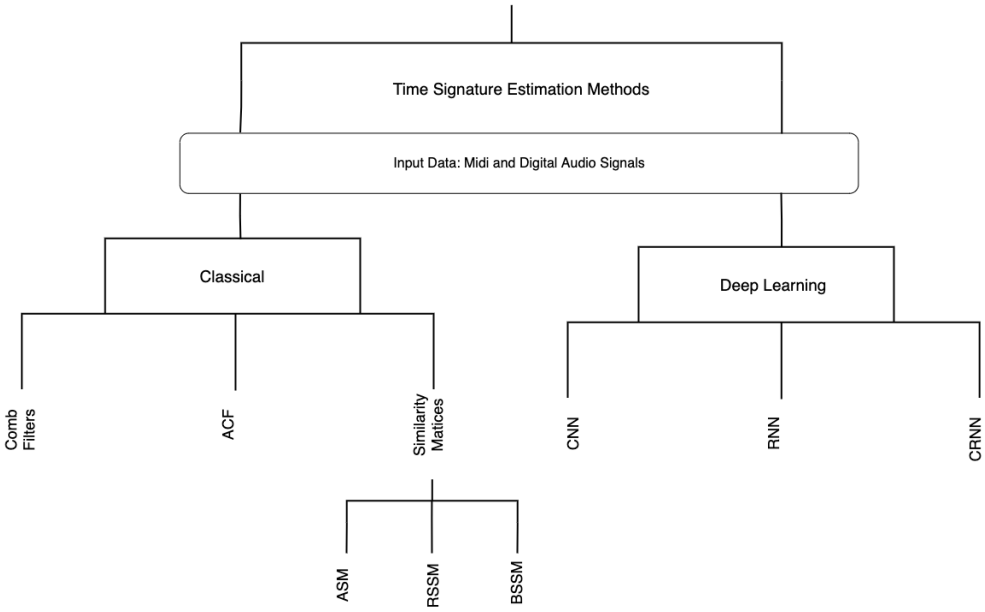

Figure 6. A dendrogram of both classical and deep learning techniques discussed in this paper.

This survey has shown that in order to estimate the time signature using digital signal processing analysis, the most promising approach has come from generating some similarity matrices of the temporal features of audio or MIDI files when music signal is converted into an appropriate feature sequence. Based on a similarity measure, a self-similarity matrix is generated from the feature sequence. The SSM generates blocks and pathways with a high overall score. Each block or path specifies a pair of segments that are comparable. Using a clustering step, whole groups of mutually comparable segments are generated from the pairwise relations. A more detailed research into similarity matrices of MFCCs between 4 and 20 could yield better results. It is important to note that the ASM, RSSM, BSSM, and ACF work better on MIDI files than on digital audio files, however,

MIDI files are not popularly used anymore. With audio samples, time signature estimation becomes relative to the tempo of the track which these other methods did not take seriously. In terms of using any deep learning approach, network architectures such as RNN has shown some level of success but cannot retain audio information for too long, however, the CNN architecture is definitely the way forward in this kind of task because it gives more accuracy for a wide range of both regular and irregular time signatures but it also takes more computational time and power to perform this task. A combination of two architectures like CNN and RNN where features are extracted in the convoluted layer and later transferred to recurrent layer has also proven to be effective in time-based series of audio signals. This implies that transfer learning—an approach that has not been fully explored in this research area could also be given more attention.

More than 70% of the studies considered in this review assumed that music pieces had repeated bars at various points in the piece, which is not always the case. Estimating musical parts with an irregular signature or beat is challenging. As a result, additional research may be conducted in this field. The aim of this analysis is to chart a course for future study in feature extraction of machine learning algorithms used in music genre classification, time signature estimation and identification, and beat and tempo estimation in the Music Information Retrieval domain. Using a better approach as a pre-processor to retrieve the time signature as an additional feature in a neural network pipeline could drastically increase the accuracy of the model eventually.

Author Contributions: Conceptualization, D.K. and J.A.; writing—original draft preparation, J.A. and D.K.; writing—review and editing, J.A., D.K. and P.K.; supervision, D.K. and P.K.; funding acquisition, D.K. and P.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by Statutory Research funds of the Department of Applied Informatics, Silesian University of Technology, Gliwice, Poland (02/100/BKM21/0011—D.K., 02/100/BK_21/0008—P.K.).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Benetos, E.; Dixon, S. Polyphonic music transcription using note onset and offset detection. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 37–40.
2. Benetos, E.; Dixon, S.; Duan, Z.; Ewert, S. Automatic music transcription: An overview. *IEEE Signal Process. Mag.* **2018**, *36*, 20–30. [CrossRef]
3. Tuncer, D. In Music Education, in the Context of Measuring Beats, Anacrusic Examples Prepared with Simple Time Signature. *Procedia-Soc. Behav. Sci.* **2015**, *197*, 2403–2406. [CrossRef]
4. Smith, S.M.; Williams, G.N. A visualization of music. In Proceedings of the Visualization'97 (Cat. No. 97CB36155), Phoenix, AZ, USA, 19–24 October 1997; pp. 499–503.
5. Kaplan, R. *Rhythmic Training for Dancers*; ERIC: Champaign, IL, USA, 2002.
6. Kan, Z.J.; Sourin, A. Generation of Irregular Music Patterns With Deep Learning. In Proceedings of the 2020 International Conference on Cyberworlds (CW), Caen, France, 29 September–1 October 2020; pp. 188–195.
7. Bottiroli, S.; Rosi, A.; Russo, R.; Vecchi, T.; Cavallini, E. The cognitive effects of listening to background music on older adults: Processing speed improves with upbeat music, while memory seems to benefit from both upbeat and downbeat music. *Front. Aging Neurosci.* **2014**, *6*, 284. [CrossRef]
8. Still, J. How down is a downbeat? Feeling meter and gravity in music and dance. *Empir. Musicol. Rev.* **2015**, *10*, 121–134. [CrossRef]
9. Temperley, D. *The Cognition of Basic Musical Structures*; MIT Press: Cambridge, MA, USA, 2004.
10. Attas, R.E.S. Meter as Process in Groove-Based Popular Music. Ph.D. Thesis, University of British Columbia, Vancouver, BC, Canada, 2011.
11. Goto, M.; Muraoka, Y. A beat tracking system for acoustic signals of music. In Proceedings of the Second ACM International Conference on Multimedia, San Francisco, CA, USA, 15–20 October 1994; pp. 365–372.

12. Foote, J.; Uchihashi, S. The beat spectrum: A new approach to rhythm analysis. In Proceedings of the IEEE International Conference on Multimedia and Expo, IEEE Computer Society, Tokyo, Japan, 22–25 August 2001; pp. 224–224.
13. Burger, B.; Thompson, M.R.; Luck, G.; Saarikallio, S.; Toiviainen, P. Music moves us: Beat-related musical features influence regularity of music-induced movement. In Proceedings of the 12th International Conference in Music Perception and Cognition and the 8th Triennial Conference of the European Society for the Cognitive Sciences for Music, Thessaloniki, Greece, 23–28 July 2012; pp. 183–187.
14. Bahuleyan, H. Music genre classification using machine learning techniques. *arXiv* **2018**, arXiv:1804.01149.
15. Oramas, S.; Barbieri, F.; Nieto Caballero, O.; Serra, X. Multimodal deep learning for music genre classification. *Trans. Int. Soc. Music Inf. Retr.* **2018**, *1*, 4–21. [CrossRef]
16. Feng, T. Deep Learning for Music Genre Classification. Private Document. 2014. Available online: https://courses.engr.illinois.edu/ece544na/fa2014/Tao_Feng.pdf (accessed on 24 September 2021).
17. Kostrzewa, D.; Kaminski, P.; Brzeski, R. Music Genre Classification: Looking for the Perfect Network. In *International Conference on Computational Science*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 55–67.
18. Kameoka, H.; Nishimoto, T.; Sagayama, S. Harmonic-temporal structured clustering via deterministic annealing EM algorithm for audio feature extraction. In Proceedings of the ISMIR 2005, 6th International Conference on Music Information Retrieval, London, UK, 11–15 September 2005; pp. 115–122.
19. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep feature extraction and classification of hyperspectral images based on convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6232–6251. [CrossRef]
20. Liu, Z.; Wang, Y.; Chen, T. Audio feature extraction and analysis for scene segmentation and classification. *J. VLSI Signal Process. Syst. Signal Image Video Technol.* **1998**, *20*, 61–79. [CrossRef]
21. Mathieu, B.; Essid, S.; Fillon, T.; Prado, J.; Richard, G. YAAFE, an Easy to Use and Efficient Audio Feature Extraction Software. In Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010, Utrecht, The Netherlands, 9–13 August 2010; pp. 441–446.
22. Sharma, G.; Umapathy, K.; Krishnan, S. Trends in audio signal feature extraction methods. *Appl. Acoust.* **2020**, *158*, 107020. [CrossRef]
23. Hsu, C.; Wang, D.; Jang, J.R. A trend estimation algorithm for singing pitch detection in musical recordings. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 393–396. [CrossRef]
24. Nakamura, E.; Benetos, E.; Yoshii, K.; Dixon, S. Towards Complete Polyphonic Music Transcription: Integrating Multi-Pitch Detection and Rhythm Quantization. In Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 15–20 April 2018; pp. 101–105. [CrossRef]
25. Degara, N.; Pena, A.; Davies, M.E.P.; Plumbley, M.D. Note onset detection using rhythmic structure. In Proceedings of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, Dallas, TX, USA, 14–19 March 2010; pp. 5526–5529. [CrossRef]
26. Gui, W.; Xi, S. Onset detection using leared dictionary by K-SVD. In Proceedings of the 2014 IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA), Ottawa, ON, Canada, 29–30 September 2014; pp. 406–409. [CrossRef]
27. Mounir, M.; Karsmakers, P.; Waterschoot, T.V. Annotations Time Shift: A Key Parameter in Evaluating Musical Note Onset Detection Algorithms. In Proceedings of the 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), New Paltz, NY, USA, 20–23 October 2019; pp. 21–25. [CrossRef]
28. Alonso, M.; Richard, G.; David, B. Extracting note onsets from musical recordings. In Proceedings of the 2005 IEEE International Conference on Multimedia and Expo, Amsterdam, The Netherlands, 6 July 2005; p. 4. [CrossRef]
29. Wu, F.H.F.; Jang, J.S.R. A supervised learning method for tempo estimation of musical audio. In Proceedings of the 22nd Mediterranean Conference on Control and Automation, Palermo, Italy, 16–19 June 2014; pp. 599–604.
30. Downie, J.S.; Byrd, D.; Crawford, T. Ten Years of ISMIR: Reflections on Challenges and Opportunities. In Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009, Kobe, Japan, 26–30 October 2009; pp. 13–18.
31. Muller, M.; Ellis, D.P.; Klapuri, A.; Richard, G. Signal processing for music analysis. *IEEE J. Sel. Top. Signal Process.* **2011**, *5*, 1088–1110. [CrossRef]
32. Klapuri, A. Musical Meter Estimation and Music Transcription. Cambridge Music Processing Colloquium. Citeseer. 2003; pp. 40–45. Available online: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.77.8559&rep=rep1&type=pdf (accessed on 24 Septmber 2021).
33. Lartillot, O.; Toiviainen, P. A Matlab toolbox for musical feature extraction from audio. In Proceedings of the International Conference on Digital Audio Effects, Bordeaux, France, 10–15 September 2007; Volume 237, p. 244.
34. Villanueva-Luna, A.E.; Jaramillo-Nuñez, A.; Sanchez-Lucero, D.; Ortiz-Lima, C.M.; Aguilar-Soto, J.G.; Flores-Gil, A.; May-Alarcon, M. *De-Noising Audio Signals Using Matlab Wavelets Toolbox*; IntechOpen: Rijeka, Croatia, 2011.
35. Giannakopoulos, T.; Pikrakis, A. *Introduction to Audio Analysis: A MATLAB® Approach*; Academic Press: Cambridge, MA, USA, 2014.
36. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

37. McFee, B.; Raffel, C.; Liang, D.; Ellis, D.P.; McVicar, M.; Battenberg, E.; Nieto, O. librosa: Audio and music signal analysis in python. In Proceedings of the 14th Python in Science Conference, Austin, TX, USA, 6–12 July 2015; Volume 8, pp. 18–25.
38. Mallat, S. *A Wavelet Tour of Signal Processing*; Elsevier: Amsterdam, The Netherlands, 1999.
39. Cataltepe, Z.; Yaslan, Y.; Sonmez, A. Music genre classification using MIDI and audio features. *EURASIP J. Adv. Signal Process.* **2007**, *2007*, 1–8. [CrossRef]
40. Ozcan, G.; Isikhan, C.; Alpkocak, A. Melody extraction on MIDI music files. In Proceedings of the Seventh IEEE International Symposium on Multimedia (ISM'05), Irvine, CA, USA, 14 December 2005; p. 8.
41. Klapuri, A.P.; Eronen, A.J.; Astola, J.T. Analysis of the meter of acoustic musical signals. *IEEE Trans. Audio Speech Lang. Process.* **2005**, *14*, 342–355. [CrossRef]
42. Uhle, C.; Herre, J. Estimation of tempo, micro time and time signature from percussive music. In Proceedings of the Internatioanl Conference on Digital Audio Effects (DAFx), London, UK, 8–11 September 2003.
43. Jiang, J. Audio processing with channel filtering using DSP techniques. In Proceedings of the 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 8–10 January 2018; pp. 545–550.
44. Foote, J. Visualizing music and audio using self-similarity. In Proceedings of the Seventh ACM International Conference on Multimedia (Part 1), Orlando, FL, USA, 30 October–5 November 1999; pp. 77–80.
45. Saito, S.; Kameoka, H.; Takahashi, K.; Nishimoto, T.; Sagayama, S. Specmurt analysis of polyphonic music signals. *IEEE Trans. Audio Speech Lang. Process.* **2008**, *16*, 639–650. [CrossRef]
46. Grohganz, H.; Clausen, M.; Müller, M. Estimating Musical Time Information from Performed MIDI Files. In Proceedings of the International Conference on Music Information Retrieval (ISMIR), Taipei, Taiwan, 27–31 October 2014; pp. 35–40.
47. Roig, C.; Tardón, L.J.; Barbancho, I.; Barbancho, A.M. Automatic melody composition based on a probabilistic model of music style and harmonic rules. *Knowl.-Based Syst.* **2014**, *71*, 419–434. [CrossRef]
48. Akujuobi, U.; Zhang, X. Delve: A dataset-driven scholarly search and analysis system. *ACM SIGKDD Explor. Newsl.* **2017**, *19*, 36–46. [CrossRef]
49. Goto, M.; Hashiguchi, H.; Nishimura, T.; Oka, R. RWC Music Database: Popular, Classical and Jazz Music Databases. *ISMIR* **2002**, *2*, 287–288. Available online: https://staff.aist.go.jp/m.goto/RWC-MDB/ (accessed on 24 September 2021).
50. Turnbull, D.; Barrington, L.; Torres, D.; Lanckriet, G. Semantic annotation and retrieval of music and sound effects. *IEEE Trans. Audio Speech Lang. Process.* **2008**, *16*, 467–476. Available online: http://slam.iis.sinica.edu.tw/demo/CAL500exp (accessed on 24 September 2021). [CrossRef]
51. Tzanetakis, G.; Cook, P. Musical genre classification of audio signals. *IEEE Trans. Speech Audio Process.* **2002**, *10*, 293–302. Available online: https://www.kaggle.com/andradaolteanu/gtzan-dataset-music-genre-classification (accessed on 24 September 2021). [CrossRef]
52. Turnbull, D.; Barrington, L.; Torres, D.; Lanckriet, G. Exploring the Semantic Annotation and Retrieval of Sound. CAL Technical Report CAL-2007-01. San Diego, CA, USA, 2007. Available online: https://www.ee.columbia.edu/~dpwe/research/musicsim/uspop2002.html (accessed on 24 September 2021).
53. Tingle, D.; Kim, Y.E.; Turnbull, D. Exploring automatic music annotation with "acoustically-objective" tags. In Proceedings of the International Conference on Multimedia Information Retrieval, Philadelphia, PA, USA, 29–31 March 2010; pp. 55–62. Available online: http://calab1.ucsd.edu/~datasets/ (accessed on 24 September 2021).
54. Law, E.; West, K.; Mandel, M.I.; Bay, M.; Downie, J.S. Evaluation of algorithms using games: The case of music tagging. In Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009, Kobe, Japan, 26–30 October 2009; pp. 387–392. Available online: https://mirg.city.ac.uk/codeapps/the-magnatagatune-dataset (accessed on 24 September 2021).
55. Defferrard, M.; Benzi, K.; Vandergheynst, P.; Bresson, X. Fma: A dataset for music analysis. *arXiv* **2016**, arXiv:1612.01840.
56. Schedl, M.; Orio, N.; Liem, C.C.; Peeters, G. A professionally annotated and enriched multimodal data set on popular music. In Proceedings of the 4th ACM Multimedia Systems Conference, Oslo, Norway, 28 February–1 March 2013; pp. 78–83. Available online: http://www.cp.jku.at/datasets/musiclef/index.html (accessed on 24 September 2021).
57. Bertin-Mahieux, T.; Ellis, D.P.; Whitman, B.; Lamere, P. The Million Song Dataset. 2011. Available online: http://millionsongdataset.com/ (accessed on 24 September 2021).
58. Panagakis, I.; Benetos, E.; Kotropoulos, C. Music genre classification: A multilinear approach. In Proceedings of the International Symposium Music Information Retrieval, Philadelphia, PA, USA, 14–18 September 2008; pp. 583–588.
59. Benetos, E.; Kotropoulos, C. A tensor-based approach for automatic music genre classification. In Proceedings of the 2008 16th European Signal Processing Conference, Lausanne, Switzerland, 25–29 August 2008; pp. 1–4.
60. Chang, K.K.; Jang, J.S.R.; Iliopoulos, C.S. Music Genre Classification via Compressive Sampling. In Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010, Utrecht, The Netherlands, 9–13 August 2010; pp. 387–392.
61. Chathuranga, Y.; Jayaratne, K. Automatic music genre classification of audio signals with machine learning approaches. *GSTF J. Comput. (JoC)* **2014**, *3*, 14. [CrossRef]
62. Zhang, W.; Lei, W.; Xu, X.; Xing, X. Improved Music Genre Classification with Convolutional Neural Networks. In Proceedings of the Interspeech, San Francisco, CA, USA, 8–12 September 2016; pp. 3304–3308.

63. Kotsiantis, S.; Kanellopoulos, D.; Pintelas, P. Handling imbalanced datasets: A review. *GESTS Int. Trans. Comput. Sci. Eng.* **2006**, *30*, 25–36.
64. Seiffert, C.; Khoshgoftaar, T.M.; Van Hulse, J.; Napolitano, A. RUSBoost: Improving classification performance when training data is skewed. In Proceedings of the 2008 19th International Conference on Pattern Recognition, Tampa, FL, USA, 8–11 December 2008; pp. 1–4.
65. López, V.; Fernández, A.; Herrera, F. On the importance of the validation technique for classification with imbalanced datasets: Addressing covariate shift when data is skewed. *Inf. Sci.* **2014**, *257*, 1–13. [CrossRef]
66. Harte, C. Towards Automatic Extraction of Harmony Information from Music Signals. Ph.D. Thesis, Queen Mary University of London, London, UK, 2010.
67. Ellis, K.; Coviello, E.; Lanckriet, G.R. Semantic Annotation and Retrieval of Music using a Bag of Systems Representation. In Proceedings of the 12th International Society for Music Information Retrieval Conference, ISMIR 2011, Miami, FL, USA, 24–28 October 2011; pp. 723–728.
68. Andersen, J.S. Using the Echo Nest's automatically extracted music features for a musicological purpose. In Proceedings of the 2014 4th International Workshop on Cognitive Information Processing (CIP), Copenhagen, Denmark, 26–28 May 2014; pp. 1–6.
69. Pons, J.; Nieto, O.; Prockup, M.; Schmidt, E.; Ehmann, A.; Serra, X. End-to-end learning for music audio tagging at scale. *arXiv* **2017**, arXiv:1711.02520.
70. Ycart, A.; Benetos, E. A-MAPS: Augmented MAPS dataset with rhythm and key annotations. In Proceedings of the 19th International Society for Music Information Retrieval Conference Late-Breaking Demos Session, Electronic Engineering and Computer Science, Paris, France, 23–27 September 2018.
71. Lenssen, N. *Applications of Fourier Analysis to Audio Signal Processing: An Investigation of Chord Detection Algorithms*; CMC Senior Theses, Paper 704; Claremont McKenna College: Claremont, CA, USA, 2013. Available online: https://scholarship.claremont.edu/cmc_theses/704/ (accessed on 24 September 2021).
72. Gouyon, F.; Herrera, P. Determination of the Meter of Musical Audio Signals: Seeking Recurrences in Beat Segment Descriptors. Audio Engineering Society Convention 114. Audio Engineering Society. 2003. Available online: https://www.aes.org/e-lib/online/browse.cfm?elib=12583 (accessed on 24 September 2021).
73. Pikrakis, A.; Antonopoulos, I.; Theodoridis, S. Music meter and tempo tracking from raw polyphonic audio. In Proceedings of the ISMIR 2004, 5th International Conference on Music Information Retrieval, Barcelona, Spain, 10–14 October 2004.
74. Coyle, E.; Gainza, M. Time Signature Detection by Using a Multi-Resolution Audio Similarity Matrix. Audio Engineering Society Convention 122. Audio Engineering Society. 2007. Available online: https://www.aes.org/e-lib/online/browse.cfm?elib=14139 (accessed on 24 September 2021).
75. Holzapfel, A.; Stylianou, Y. Rhythmic Similarity in Traditional Turkish Music. In Proceedings of the ISMIR—International Conference on Music Information Retrieval, Kobe, Japan, 26–30 October 2009; pp. 99–104.
76. Gainza, M. Automatic musical meter detection. In Proceedings of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing, Taipei, Taiwan, 19–24 April 2009; pp. 329–332.
77. Gulati, S.; Rao, V.; Rao, P. Meter detection from audio for Indian music. In *Speech, Sound and Music Processing: Embracing Research in India*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 34–43.
78. Varewyck, M.; Martens, J.P.; Leman, M. Musical meter classification with beat synchronous acoustic features, DFT-based metrical features and support vector machines. *J. New Music Res.* **2013**, *42*, 267–282. [CrossRef]
79. Cano, E.; Mora-Ángel, F.; Gil, G.A.L.; Zapata, J.R.; Escamilla, A.; Alzate, J.F.; Betancur, M. Sesquialtera in the colombian bambuco: Perception and estimation of beat and meter. *Proc. Int. Soc. Music Inf. Retr. Conf.* **2020**, *2020*, 409–415.
80. Lee, K. *The Role of the 12/8 Time Signature in JS Bach's Sacred Vocal Music*; University of Pittsburgh: Pittsburgh, PA, USA, 2005.
81. Panwar, S.; Das, A.; Roopaei, M.; Rad, P. A deep learning approach for mapping music genres. In Proceedings of the 2017 12th System of Systems Engineering Conference (SoSE), Waikoloa, HI, USA, 18–21 June 2017; pp. 1–5.
82. Schoukens, J.; Pintelon, R.; Van Hamme, H. The interpolated fast Fourier transform: A comparative study. *IEEE Trans. Instrum. Meas.* **1992**, *41*, 226–232. [CrossRef]
83. Chen, K.F.; Mei, S.L. Composite interpolated fast Fourier transform with the Hanning window. *IEEE Trans. Instrum. Meas.* **2010**, *59*, 1571–1579. [CrossRef]
84. McLeod, A.; Steedman, M. Meter Detection and Alignment of MIDI Performance. ISMIR. 2018; pp. 113–119. Available online: http://ismir2018.ircam.fr/doc/pdfs/136_Paper.pdf (accessed on 24 September 2021).
85. McLeod, A.; Steedman, M. Meter Detection From Music Data. In *DMRN+ 11: Digital Music Research Network One-Day Workshop 2016*; Utkal University: Bhubaneswar, India, 2016.
86. De Haas, W.B.; Volk, A. Meter detection in symbolic music using inner metric analysis. In Proceedings of the International Society for Music Information Retrieval Conference, New York, NY, USA, 7–11 August 2016; p. 441.
87. Liu, J.; Sun, S.; Liu, W. One-step persymmetric GLRT for subspace signals. *IEEE Trans. Signal Process.* **2019**, *67*, 3639–3648. [CrossRef]
88. Brown, J.C. Determination of the meter of musical scores by autocorrelation. *J. Acoust. Soc. Am.* **1993**, *94*, 1953–1957. [CrossRef]
89. Hua, X.; Ono, Y.; Peng, L.; Cheng, Y.; Wang, H. Target detection within nonhomogeneous clutter via total bregman divergence-based matrix information geometry detectors. *IEEE Trans. Signal Process.* **2021**, *69*, 4326–4340. [CrossRef]
90. Wu, Y.; Ianakiev, K.; Govindaraju, V. Improved k-nearest neighbor classification. *Pattern Recognit.* **2002**, *35*, 2311–2318. [CrossRef]

91. Lai, J.Z.; Huang, T.J. An agglomerative clustering algorithm using a dynamic k-nearest-neighbor list. *Inf. Sci.* **2011**, *181*, 1722–1734. [CrossRef]
92. Dong, W.; Moses, C.; Li, K. Efficient k-nearest neighbor graph construction for generic similarity measures. In Proceedings of the 20th International Conference on World Wide Web, Hyderabad, India, 28 March–1 April 2011; pp. 577–586.
93. Zhu, W.; Sun, W.; Romagnoli, J. Adaptive k-nearest-neighbor method for process monitoring. *Ind. Eng. Chem. Res.* **2018**, *57*, 2574–2586. [CrossRef]
94. West, K.; Cox, S. Finding An Optimal Segmentation for Audio Genre Classification. In Proceedings of the ISMIR 2005, 6th International Conference on Music Information Retrieval, London, UK, 11–15 September 2005; pp. 680–685.
95. Roopaei, M.; Rad, P.; Jamshidi, M. Deep learning control for complex and large scale cloud systems. *Intell. Autom. Soft Comput.* **2017**, *23*, 389–391. [CrossRef]
96. Li, T.L.; Chan, A.B.; Chun, A. Automatic musical pattern feature extraction using convolutional neural network. In Proceedings of the International MultiConference of Engineers and Computer Scientists 2010 (IMECS 2010), Hong Kong, China, 17–19 September 2010; pp. 546–550.
97. Polishetty, R.; Roopaei, M.; Rad, P. A next-generation secure cloud-based deep learning license plate recognition for smart cities. In Proceedings of the 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), Anaheim, CA, USA, 18–20 December 2016; pp. 286–293.
98. Lai, S.; Xu, L.; Liu, K.; Zhao, J. Recurrent convolutional neural networks for text classification. In Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, TX, USA, 25–30 January 2015.
99. Dai, J.; Liang, S.; Xue, W.; Ni, C.; Liu, W. Long short-term memory recurrent neural network based segment features for music genre classification. In Proceedings of the 2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP), Tianjin, China, 17–20 October 2016; pp. 1–5.
100. Feng, L.; Liu, S.; Yao, J. Music genre classification with paralleling recurrent convolutional neural network. *arXiv* **2017**, arXiv:1712.08370.
101. Jia, B.; Lv, J.; Liu, D. Deep learning-based automatic downbeat tracking: A brief review. *Multimed. Syst.* **2019**, *25*, 617–638. [CrossRef]
102. Pereira, R.M.; Costa, Y.M.; Aguiar, R.L.; Britto, A.S.; Oliveira, L.E.; Silla, C.N. Representation learning vs. Handcrafted features for music genre classification. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8.
103. Dieleman, S.; Brakel, P.; Schrauwen, B. Audio-based music classification with a pretrained convolutional network. In Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR-2011), Miami, FL, USA, 24–28 October 2011; pp. 669–674.
104. Durand, S.; Essid, S. Downbeat Detection with Conditional Random Fields and Deep Learned Features. In Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR, New York, NY, USA, 7–11 August 2016; pp. 386–392.
105. Böck, S.; Davies, M.E.; Knees, P. Multi-Task Learning of Tempo and Beat: Learning One to Improve the Other. In Proceedings of the 20th ISMIR Conference, Delft, The Netherlands, 4–8 November 2019; pp. 486–493.
106. Purwins, H.; Li, B.; Virtanen, T.; Schlüter, J.; Chang, S.Y.; Sainath, T. Deep learning for audio signal processing. *IEEE J. Sel. Top. Signal Process.* **2019**, *13*, 206–219. [CrossRef]
107. Fuentes, M.; Mcfee, B.; Crayencour, H.C.; Essid, S.; Bello, J.P. A music structure informed downbeat tracking system using skip-chain conditional random fields and deep learning. In Proceedings of the ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, UK, 12–17 May 2019; pp. 481–485.
108. Rajan, R.; Raju, A.A. Deep neural network based poetic meter classification using musical texture feature fusion. In Proceedings of the 2019 27th European Signal Processing Conference (EUSIPCO), A Coruna, Spain, 2–6 September 2019; pp. 1–5.
109. Zhang, Y.; Yang, Q. A survey on multi-task learning. *arXiv* **2017**, arXiv:1707.08114.
110. Sener, O.; Koltun, V. Multi-task learning as multi-objective optimization. *arXiv* **2018**, arXiv:1810.04650.
111. Burges, C.J.; Platt, J.C.; Jana, S. Extracting noise-robust features from audio data. In Proceedings of the 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing, Orlando, FL, USA, 13–17 May 2002; Volume 1, pp. 1–1021.
112. Das, S.; Bäckström, T. Postfiltering Using Log-Magnitude Spectrum for Speech and Audio Coding. In Proceedings of the Interspeech, Hyderabad, India, 2–6 September 2018; pp. 3543–3547.
113. Srinivasamurthy, A.; Holzapfel, A.; Cemgil, A.T.; Serra, X. Particle filters for efficient meter tracking with dynamic bayesian networks. In *ISMIR 2015, 16th International Society for Music Information Retrieval Conference, Málaga, Spain, 26–30 October 2015*; Müller, M., Wiering, F., Eds.; International Society for Music Information Retrieval (ISMIR): Canada, 2015. Available online: https://repositori.upf.edu/handle/10230/34998 (accessed on 24 September 2021).
114. Humphrey, E.J.; Bello, J.P.; LeCun, Y. Moving beyond feature design: Deep architectures and automatic feature learning in music informatics. In Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Porto, Portugal, 8–12 October 2012; pp. 403–408.
115. Fuentes, M.; McFee, B.; Crayencour, H.; Essid, S.; Bello, J. Analysis of common design choices in deep learning systems for downbeat tracking. In Proceedings of the 19th International Society for Music Information Retrieval Conference, Paris, France, 23–27 September 2018.