Mo Cheuk Yin (3035199111)
Zhong Shun (3035118844)

Virtual Music Tutor

Project background

Music tutoring through machines overview

Research in artificial intelligence has impacted various fields in our life ranging from medicine, finance, robotics, machinery, to everyday consumer retailing, advertisement, urban planning, etc. In such fields, problems tend to be well formalized where a clear-cut metric can be devised to prove/disprove various approaches of AI and allow researchers to improve the algorithms. Music which is traditionally deemed a very subjective domain-does not enjoy such advantages and therefore lacks objective metrics to quantify how good/bad one's playing is. Using machines to gauge a person's playing and provide constructive feedback can be done only to a limited extent. That is why music education heavily relies on tutors to assess the playing of students with their subjective perceptions.

People have resorted to technologies when it comes to music education, albeit to a very limited extent. Metrics such as volume, tempo, rhythm, etc. have been traditionally used as a rough guide to analyse a piece of music and there are existing music tutoring products that incorporate these approaches to assess a student's playing. Branching teaching programs (e.g GUIDO ear-training system) (Holland, 2000) were developed to present pre-recorded materials to students, receive responses from them and compare literally the response with predefined answers, after which a certain branch will be selected to present the next materials. This is a very rigid process that lacks the essential interactive elements in music education that is required to tailor-made teachings according to a student's performance and address the area for improvement. These programs are also limited to the field of music theory, for example ear trainings specifically because it more often than not requires only objective standardised answers. *Match My Sound* is a company that offers web API that matches students' playing with the pre-set score or MIDI track in the system and provides automatic assessment. Simple note per note assessment and chord detection are implemented using signal processing to retrieve music information. However, the choice of using score or MIDI as the standard base limits the comparison and as a result it is not able to detect more advanced but common techniques in instruments, for example, the bending and sliding in guitar. *Yousician* is another company that provides similar services but on the mobile platforms. It gamifies guitar learning using very interactive UI but again, the assessment is limited to simple rhythm and chord recognition based on score or MIDI.

The above shortcomings in current music learning products in the market challenge our team to come up with a truly intelligent "Virtual Music Tutor" which detects not only simple tempo/chord correctness in a student's playing but also more nuanced subtle differences in the student's playing compared to that of a tutor.

Problem statement

This project aims to focus on how to automate the process of assessing the quality of student's music performance. It is also our goal to help music tutors communicate more clearly to their students by showing them what is lacking in their playing in a straightforward way by highlighting the differences of their playing with the standard rendition by the tutor, using our app. Ultimately students will receive scores in several aspects of their playing, e.g. rhythm/pulse, note accuracy, timbre and the total score of their similarity to their tutors' performance.

For beginners, learning a musical instrument is to follow their teachers' way of playing. A musical instrument tutor often plays the role of assessing student's performance accurately on aspects such as rhythm, pitch, techniques or even the "feel". However, from the Baroque period ~~up~~ now the 21st century where music can even be composed by A.I, music tutoring is still being conducted in the old-fashioned way - students meeting tutors once per week for an hour, after which follows practicing on their own without much guidance and reporting back to tutors thereafter. Considering that most of the practice actually happens at home, issues they have with regard to the right note/chord or techniques cannot be addressed until the next time they report to the tutor. This is a very inefficient way to receive education without much self-learning means. There is a very large room for technologies to come in to fill in the gap in students' everyday practice.

With the advancement of machine learning and audio signal processing (ASP), it becomes possible to use modern approaches in Music Information Retrieval (MIR) to extract data on the quantifiable aspects of a student's playing (e.g volume, tempo, timbre) and compare more accurately with the "standard rendition" of a musical piece. From there, interactive ways to highlight the differences and guidance can be provided instantaneously, which means students' mistakes will not be perpetuated until the next time they report to their tutors. Students will therefore make more improvement during their solo practice.

Project objectives

This project is to develop a system which can judge how similar a student's performance is to his tutor's. The preliminary target musical instrument we will compare is guitar and we will first focus on beginner level (roughly Rockschool G1 - G3) guitar exercises and the ultimate goal is to assess **intermediate** level (roughly Rockschool G5) guitar exercises. We will try to model a typical guitar tutor's comments on his student's performance in our system. The scope of the assessment will be on 5 main areas: techniques, rhythm/pulse steadiness, note accuracy, chord accuracy and dynamics.

Figure 1 is a table which is a brief assessment criteria for our system which takes inspiration from the assessment criteria of guitar exam set by the authoritative pop music exam board Rockschool. Here, "Grade" means the level of excellence that a student achieves if it could complete the corresponding items. In simple terms, if the student can replicate completely his teacher's playing, he is eligible to get an "Excellent".

Grade	Description
Qualified for assessment	• Guitar is used • Guitar is tuned properly (to standard tuning EADGBE) • Guitar is played loud enough and background noise is minimal • Guitar effect is used correctly (distortion / clean guitar)
Pass	• Some secure techniques • Generally steady pulse / rhythm • Some note accuracy • Some chord accuracy • Some proper dynamics
Merit	• Secure techniques • Overall steady pulse / rhythm • Secure note accuracy • Secure chord accuracy • Overall proper dynamics
Excellent	• Consistent and secure techniques • Consistently steady pulse / rhythm • Consistent note accuracy • Consistent chord accuracy • Consistent proper dynamics

Figure 1. Assessment criteria of “Virtual Music Tutor”.

Through ASP, high-level features can be derived to identify e.g. the existence of an instrument, key, chords, melody, use of techniques of a song. These will be used to assess the level of similarity and the details of the methods to assess each aspect listed in *Figure 1* will be illustrated in the next section.

In short, we hope to develop a system that is capable of assessing the

- steadiness of pulse and rhythm
- accuracy of note
- accuracy of chord (multiple notes)
- dynamics
- techniques (e.g. bend, hammer-on, pull-off)

- miscellaneous features that may block more detailed assessment (row with Grade equals "Qualified for assessment" in *Figure 1*)

based on the comparison between a student's playing with a standard rendition from the tutor in each of the aspects.

Project methodology

Music has three main representations in computing, which are score (formats e.g. MusicXML, Finale), symbolic (e.g. MIDI) and audio representation (MP3 or WAV encoding). Regarding the choice of music representation to compare a student's audio performance to his tutor's, we propose to use the audio signal of a tutor's reference performance, based on the following reasons:

1. Recording audio takes much less time than notating on a score

   A fairly practical reason, many guitar tutors to our knowledge dislike composing on a score notation e.g. MuseScore or Guitar Pro. On the other hand, recording on a mobile phone takes them much less time and it is the easiest way to demonstrate to their students how to play a piece, especially if they are doing it virtually over the internet.

2. Audio signal captures the subtlest musical information, e.g. techniques, minor changes in dynamics

   Given that the actual music playing comprises far more elements beyond simple chord recognition and tempo calibration, for instance, the nuances of pedalling (in piano), sliding (in guitar), etc, which are lost in representations like a MIDI file or score. For example, consider a guitar technique called "bend" which happens when a guitarist stretches a string with left hand to increase the pitch of a fretted note. Notations can tell a guitarist that he should bend a note in a nebulous sense: either instantly, very quickly, gradually or slowly, but how he/she should actually do it is left to his/her own interpretation. The difference cannot be pointed out to a student if the score or MIDI is used which does not contain such rich musical information. A more appropriate way to develop music tutoring applications is thus to use the actual audio file as the base standard, which will then contain much rich music information to guide students how exactly they should handle the dynamics in music.

At the centre of our project lies the question: how similar is the student's playing to his tutor's standard rendition? As Knees and Schedl (2016) mentioned, the notion of music similarity is subjective as music is listened through human's perception process, which differs from person to person. In order to computationally model tutor's music perception, 4 main areas of it must be considered – music content, music context, user properties and user context. Imagine a situation in which a guitar tutor assigns an assignment to his students to play a short exercise of E major scale, utilising the model suggested in the book, the following figure can be constructed to examine the factors of our problem.

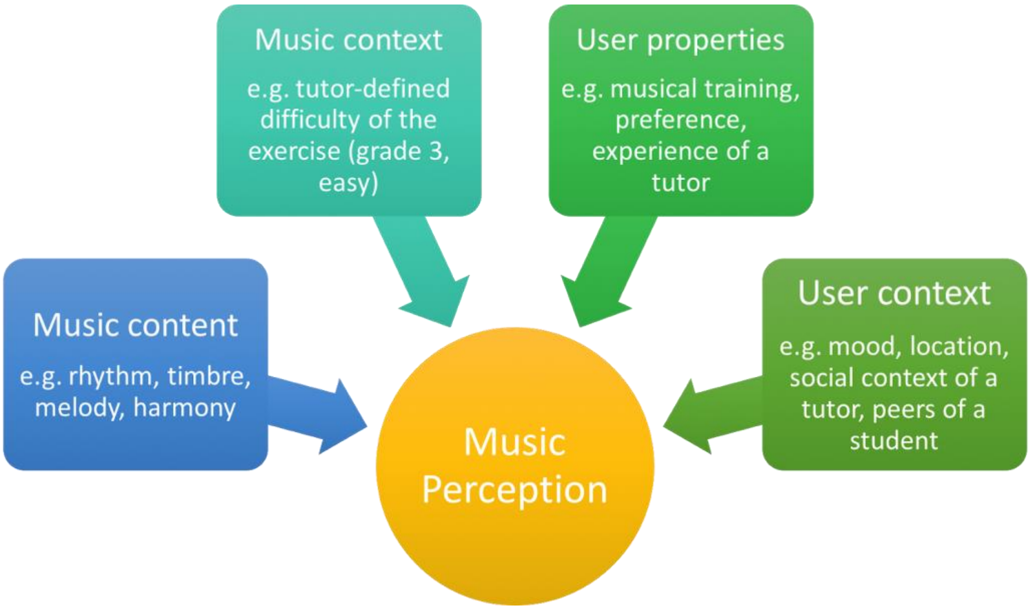
Music context
e.g. tutor-defined difficulty of the exercise (grade 3, easy)
User properties
e.g. musical training, preference, experience of a tutor
Music content
e.g. rhythm, timbre, melody, harmony
User context
e.g. mood, location, social context of a tutor, peers of a student
Music Perception

Figure 2. Factors of music perception. Adapted from Music Similarity and Retrieval (p.14), by Knees, Peter; Schedl, Markus, 2016, Springer Berlin Heidelberg. Copyright 2016 by Springer.

However, as we mentioned earlier that we would assess a student's performance based on objective measures in *Figure 1*. In this project, the more subjective elements (user properties and user context) in this system will be neglected. Now we shall give a more technical overview on how this project is divided to.

Taking into account the music content and music-context in *Figure 2*, this project will consist of two modules, which we will name roughly the frontend component and backend component for the time being.

Frontend component

The frontend component is responsible for handling user interactions. For students, it will be a simple portal to record or upload their instrument playing and receive grades in dimensions of rhythm, melody, timbre and expressiveness. For teachers, it will be a simple portal to record their reference performance and set optional parameters such as suggested leniency(strictness) in comparing the performances.

Backend component

The backend component is responsible for audio signal processing (ASP), feature extractions and machine learning of music.

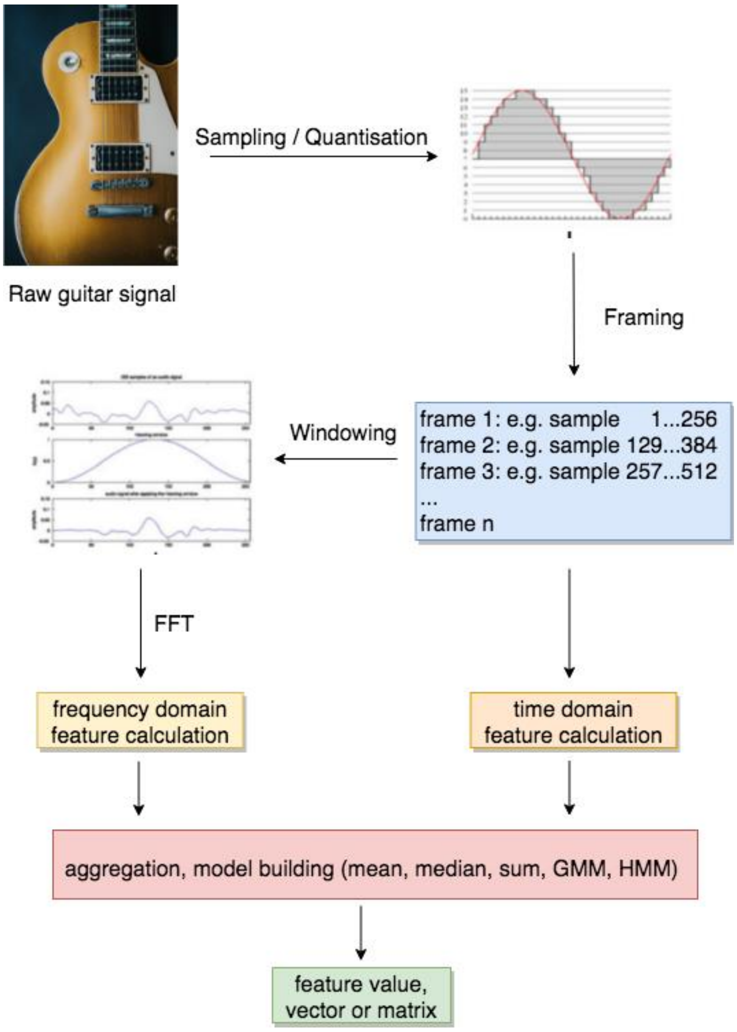
Raw guitar signal
Sampling / Quantisation
Framing
frame 1: e.g. sample 1...256
frame 2: e.g. sample 129...384
frame 3: e.g. sample 257...512
...
frame n
Windowing
FFT
frequency domain feature calculation
time domain feature calculation
aggregation, model building (mean, median, sum, GMM, HMM)
feature value, vector or matrix

Figure 2. Standard workflow in feature extraction. Adapted from Music Similarity and Retrieval (p.14), by Knees, Peter; Schedl, Markus, 2016, Springer Berlin Heidelberg. Copyright 2016 by Springer.

The standard workflow in MIR involves sampling the audio data, framing and windowing it and performing various kinds of FFTs (Knees & Schedl, 2016). These in turn return time-domain and frequency-domain data in which features can be calculated upon. Finally, aggregation and model building of the data provide features such as HMM or coefficients which could be used to compute the similarity of student's performance to his tutor's performance.

Apart from this standard approach, recent researches show that Deep Neural Networks (DNNs) have the possibility to outperform the above model built by standard hand-crafted features (Pati, Gururani & Lerch, 2018). However, this approach requires a lot of data for its unsupervised training for features, our team would consider that in around November if the classic workflow failed to bootstrap our project ideas.

We will first focus on offline audio processing and gradually add support to real-time assessment once we have solid knowledge on implementing and tuning our feature detectors.

Feature extraction

Assuming we have obtained the audio signal of a tutor's demonstration, running it through Short-Time Fourier Transform (STFT) will give us both time-domain and frequency-domain frames of audio data, through which various levels of features could be extracted to help us understand the properties of the track. Müller (2015) provided an overview of the common features that are essential to the tasks of e.g. note detection, chord recognition or beat tracking which are crucial parts of the assessment as illustrated in *Figure 1*. His textbook on music processing, together with recent research papers on guitar-specific music processing, will be used as references to aid our feature extraction process.

Let us recall that there are 5 main areas of assessment (pulse/rhythm, note accuracy, chord accuracy, dynamics and techniques) plus 4 other items (whether a guitar is used, whether guitar is tuned properly, whether the guitar signal is loud enough and whether guitar effect is applied correctly) that we would like to detect.

Pulse and rhythm

Müller (2015) presented in his book 2 main approaches to track pulse and beat. Predominant local pulse (PLP) shows the most prominent pulse in a neighbourhood of each time position, with tempo variations and local changes in pulse level be captured to a certain degree. For music with strong and steady beat, beat tracking by dynamic programming can be applied to extract tempo that is more or less constant throughout.

Melody extraction

Melody extraction refers to the process of deriving the fundamental frequency (F0) sequences of a particular musical instrument. Salamon & Gómez (2012) presented an efficient and effective algorithm on melody extraction by characterisation of pitch contour. Chen & Yang (2015) has used it to extract the melody in guitar solos in their guitar techniques detection algorithm.

Chord recognition

Müller (2015) presented template matching and hidden Markov models (HMMs) as classic approaches to recognise chords. A common step before doing further analysis in chord recognition is to turn frequency-domain signal into chromagrams, which are feature vectors that aggregate the spectral information in each pitch class (in total 12 pitch classes, {C, C#, D, …, B}). The main idea of template matching is to compare chromagrams against a known distribution of notes in a chord. The main idea of HMM is to give a transitional model between states (chords) that expresses the probability of passing from one chord to another. Over the years various advances have been made, but more or less are based on HMM (McVicar, Santos-Rodriguez, Ni & De Bie, 2014). These include models extending 1st order HMM, e.g. duration-explicit HMMs, key-chord HMMs and Dynamic Bayesian Networks. These approaches require different combinations of expert knowledge and labelled data to support the algorithm.

Dynamics

In music, dynamics refers to loudness of a sound and also the symbols that indicate that, for instance, *mf* means mezzo-forte, half-loud. It turns out that the intensity of sound measured in decibel (dB) is not useful in modelling human perception as loudness varies with sensitivity of ear (Knees & Schedl, 2016). Instead we usually use phon to represent the perceived loudness of sound.

As dynamics played on a piece is relative to an overall loudness of a pitch, we propose that by first computing the phons of each notes / chord played from the tutor's demo and the average phon of it, we can get the distances of each note's phon to the average. We can then assess whether the loudness of each note played by the student can resemble that.

Techniques

Chen & Yang (2015) proposed an approach to detect electric guitar techniques which include bend, vibrato, hammer-on, pull-off and slide with F-scores for ranging from 57.7% to 87.7% depending on the playing techniques. This is a good start for our project and we will continue to explore other approaches to detect guitar techniques.

Instrument recognition

Fuhrman (2012) presented an instrument recognition approach that recognises instrument from polyphonic music audio signals, which is useful for our system to guard any non-guitar playing from students that may involve backing track or background noise.

Guitar tuning

Guitar tutors often remind students to tune their guitars before asking them for a demonstration. However, chances are low-priced guitars owned by most beginners cannot always be tuned properly. Hence, we propose to set a tolerance level for the distance between the frequencies of an intended note and the out-of-tune note played by a student.

Guitar effects

Stein (2010) presented an approach which uses 541 spectral, cepstral and harmonic features to detect audio effects in guitar and its classification accuracy reached 99.2%. Schmitt & Schuller (2017) presented an electric guitar effect classifier which extracts audio features such as RMS energy, spectral features to classify 10 different types of guitar effects and concluded that spectral features and delta coefficients are the most useful features for recognising guitar effects.

Scrum as software development methodology

In order to attain the most flexible milestone targets and schedule for our team, the project will be developed under the scrum framework. Scrum is particularly suitable for us as it enables teams to self-organise and collaborate closely, through processes such as bi-weekly face-to-face sprint planning, daily scrum meeting and bi-weekly sprint-end reviews. This is crucial as our team members have different schedules at school and will depend mostly on online communication. We also would ensure the team can deliver a minimum viable product (MVP) at each sprint end from both frontend and backend components so as to maintain a healthy incremental progress week by week. Although both of our team members are developers, the core responsibilities will be different. Jacky will work mainly on the backend component and Shaun will work mainly on frontend component. As our team has limited experience in ASP and MIR, we expect to pick up the skills together and will foster knowledge sharing among ourselves.

Programming tools and datasets

For the frontend component, in light of the abundance of ready-to-go open source modules in the language, the JavaScript library React will be used. React is supported by a large community of developers and it is easy to use and learn. It is also proven to be scalable, which helps our team consolidate an underground for potential future development of the frontend component after the period of FYP. Choosing the web as the platform also allows easy cross-platform access.

With the ubiquity of ASP and machine learning libraries in Python, we would use Python as our primary language for the backend component. Although knowledge in low-level implementation of ASP is helpful to our problem, it would be much easier if there are handy libraries like librosa and scikit-learn which alleviate the pain of handling all the low-level operations.

As mentioned in the project background, the target instrument that this project mainly compares is guitar. Xi, Bittner, Pauwels, Ewert and Bello (2018) published a dataset called Guitar-Set which includes guitar performances from both electric and acoustic guitar with annotations including string and fret positions, downbeats and playing style. With its detailed annotations, the dataset will be useful as being reference tracks of a music tutor. Several other datasets, e.g. IDMT-SMT-Audio-Effects, IDMT-SMT-Guitar datasets (Kehling, Abeßer, Dittmar & Schuller, 2014) and Guitar playing techniques dataset (Su, Yu & Yang, 2014) are useful for guitar effects and techniques detection.

For student audio data, our initial plan is that we would spend time to play the excerpts in the above annotated dataset which act as the student's rendition of the same piece of music. We would consider asking for external help to make data collection more effective.

Project schedule and milestones

Milestones

30 Sep: Deliverables of Phase 1

- Detailed project plan
- Project web page

18 Nov: A working prototype of "Virtual Music Tutor" which can

- Give simple statistics on similarity in aspects of 2 of the assessment criteria listed in *Figure 1*
- Critical point to decide whether to go for classic hand-crafted features modelling approach or leveraging the use of DNNs as described in Project Methodology

20 Jan: Deliverables of Phase 2

- Enhanced version of "Virtual Music Tutor" which took into account full aspects of assessment criteria listed in *Figure 1*
- It should be able to give detailed scores in dimensions of techniques, note accuracy, chord accuracy and dynamics on a granular level (e.g. per track)
- Detailed interim report

17 Mar: Enhanced version of "Virtual Music Tutor" which can

- Give a detailed similarity score in dimensions of techniques, note accuracy, chord accuracy and dynamics on a finer level (e.g. per 1 bar)
- Give real-time feature detection and assessment

14 Apr: Deliverables of Phase 3

- Finalised tested version of "Virtual Music Tutor"
- Final report

29 Apr: Project exhibition

- Exhibition materials (foam boards, decorations, etc.)

***Official FYP deadlines bolded**

Schedule

17 Sep 2018 – 18 Nov 2018

- Team commitment level: Mid (Start of 1st semester and settlement of schoolwork)
- Working prototype that can demonstrate our system is capable of giving simple similarity stats in 2 of the following areas
    - pulse/rhythm
    - note accuracy
    - chord accuracy
    - dynamics and techniques
- Decide whether we should go for DNNs approach if these hand-crafted features cannot serve our purpose (see Project Methodology)
- Preliminary machine learning model exploration
- Frontend and backend modules deployment #1

19 Nov 2018 – 20 Jan 2018

- Team commitment level: High (long winter vacation)
- Explore all the possible low-level or mid-level audio features that could be utilised in computing the similarity scores in dimensions of pulse/rhythm, note accuracy, chord accuracy, dynamics and techniques.
- Enhance the system to give a detailed similarity score in dimensions of pulse/rhythm, note accuracy, chord accuracy, dynamics and techniques on a very granular level (e.g. per whole track comparison)
- Polish the user interface so as to facilitate better demonstration in the first presentation
- Near the end of this phase, organise a user focus group which consists of instrument tutors, beginner and experienced students and educators to collect user feedback
- Finalise the target features of the system to be implemented and adjust the roadmap of the project according to feedback collected
- Preparation of first presentation and interim report
- Frontend and backend modules deployment #2

21 Jan 2019 – 17 Mar 2019

- Team commitment level: High (Start of 2nd Semester)
- Enhance the system to give a detailed similarity score in dimensions of pulse/rhythm, note accuracy, chord accuracy, dynamics and techniques on a finer level (e.g. per phrase or per 1 bar comparison)
- Enhance the system to support real-time assessment
- Near the end of this phase, organise another user focus group to collect the second batch of feedback
- Enhance the target features of the system to be implemented
- Frontend and backend modules deployment #3

18 Mar 2019 – 14 April 2019

- Team commitment level: Mid (Towards the end of 2nd Semester)
- Thorough testing of the final product
- Buffer for feature enhancement and bug fixes
- Preparation for final presentation and final report

15 Apr 2019 – 28 April 2019

- Team commitment level: Mid (The end of 2nd Semester)
- Final adjustment to the look and feel of the user interface to facilitate a good project exhibition
- Preparation of exhibition materials

References

Chen, Y. P., Su, L., & Yang, Y. H. (2015, October). Electric Guitar Playing Technique Detection in Real-World Recording Based on F0 Sequence Pattern Recognition. In ISMIR (pp. 708-714).

Fuhrmann, F. (2012). Automatic musical instrument recognition from polyphonic music audio signals (Doctoral dissertation, Universitat Pompeu Fabra).

Holland, S. (2000). Artificial Intelligence in Music Education: {A} critical review. Readings in Music and Artificial Intelligence, 20(January 2000), 239–274.

Kehling, C., Abeßer, J., Dittmar, C., & Schuller, G. (2014, September). Automatic Tablature Transcription of Electric Guitar Recordings by Estimation of Score-and Instrument-Related Parameters. In DAFx (pp. 219-226).

Knees, P., & Schedl, M. (2016). Music similarity and retrieval: an introduction to audio-and web-based strategies (Vol. 36). Springer.

McVicar, M., Santos-Rodríguez, R., Ni, Y., & De Bie, T. (2014). Automatic chord estimation from audio: A review of the state of the art. IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), 22(2), 556-575.

Müller, M. (2015). Fundamentals of music processing: Audio, analysis, algorithms, applications. Springer.

Pati, K. A., Gururani, S., & Lerch, A. (2018). Assessment of Student Music Performances Using Deep Neural Networks. Applied Sciences, 8(4), 507.

Salamon, J., & Gómez, E. (2012). Melody extraction from polyphonic music signals using pitch contour characteristics. IEEE Transactions on Audio, Speech, and Language Processing, 20(6), 1759-1770.

Schmitt, M., & Schuller, B. (2017). Recognising Guitar Effects-Which Acoustic Features Really Matter?. INFORMATIK 2017.

Stein, M. (2010). Automatic detection of multiple, cascaded audio effects in guitar recordings. In Proceedings of the 13th International Conference on Digital Audio Effects (DAFx) (pp. 4-7).

Su, L., Yu, L. F., & Yang, Y. H. (2014, October). Sparse Cepstral, Phase Codes for Guitar Playing Technique Classification. In ISMIR (pp. 9-14).

Xi, Q., Bittner, R. M., Pauwels, J., Ewert, S., & Bello, J. P. (2018). Guitarset: A Dataset for Guitar Transcription. ISMIR.