An Audio-Visual Framework for Transcription and Fingering Optimization in Bass Guitar Performance

Hyunwoo Bae
Chung-Ang University
Seoul, Republic of Korea
tlth1224@cau.ac.kr

Taegyun Kwon
KAIST
Daejeon, Republic of Korea
ilcobo2@kaist.ac.kr

Eunji Park*
Chung-Ang University
Seoul, Republic of Korea
eunjipark@cau.ac.kr

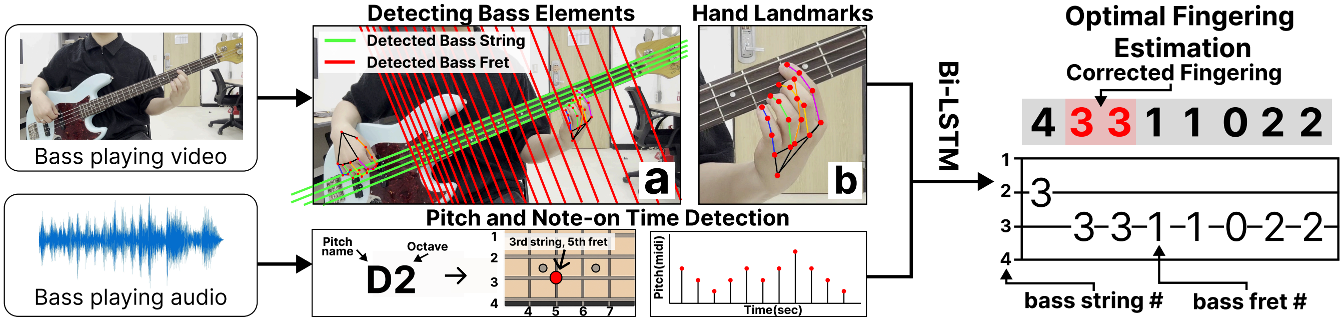

Figure 1: Framework overview

Abstract

Fingering plays a crucial role in musical performance, yet most computational approaches focus on piano, overlooking the spatial complexity of string instruments like the bass. This paper presents a framework for estimating optimal bass fingering by analyzing video input, detecting strings, frets, finger positions, and played notes. Using a Bi-LSTM model, the system estimates optimal fingering sequences. The framework supports both user-recorded and online videos, offering a scalable solution for bass performance analysis.

CCS Concepts

• **Applied computing** → **Sound and music computing**.

Keywords

Transcription, Multimodal music processing, Bass guitar fingering

ACM Reference Format:
Hyunwoo Bae, Taegyun Kwon, and Eunji Park. 2025. An Audio-Visual Framework for Transcription and Fingering Optimization in Bass Guitar Performance. In *The 38th Annual ACM Symposium on User Interface Software and Technology (UIST Adjunct '25), September 28–October 01, 2025, Busan, Republic of Korea.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3746058.3758448

1 Introduction

Fingering refers to assigning specific fingers to specific keys or strings during playing a musical instrument, such as a bass guitar or piano. As efficient and well-practiced fingering ability enables high-level play performance [2, 10], it has been considered a crucial skill not only for beginners but also for experienced players [2, 14]. Previous studies proposed various methods for estimating optimal fingering: cost-based methods [1, 7, 8, 15–17], statistical methods [11–13, 20], and more recently, deep learning-based methods [5, 6, 13, 18]. However, existing studies have primarily focused on the piano, while optimal fingering for string instruments such as the bass guitar or violin remains underexplored.

The key difference between the piano and string instruments lies in the dimensionality of their playing surfaces. While piano fingering is constrained to a one-dimensional (1D) linear layout of keys, bass guitar fingering requires two-dimensional (2D) movement across both strings and frets. This spatial complexity results in fundamentally different motion patterns: piano playing involves primarily horizontal movements, whereas bass guitar playing demands coordinated motion in both horizontal and vertical directions. Additionally, unlike the piano, a single note on the bass guitar can often be played in multiple positions, offering several fingering options for the same pitch. Thus, bass guitar fingering analysis must account for multi-directional motion and spatial complexity.

Accordingly, this paper proposes a novel framework that provides an optimal fingering method for the bass guitar. The framework detects bass-related elements (e.g., strings and frets), the player's finger position, and the notes being played. The input videos can be either user-recorded or crawled from online sources such as YouTube. We used Bi-LSTM model to estimate the optimal fingering sequence.

*Corresponding author

UIST Adjunct '25, Busan, Republic of Korea
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2036-9/25/09
https://doi.org/10.1145/3746058.3758448

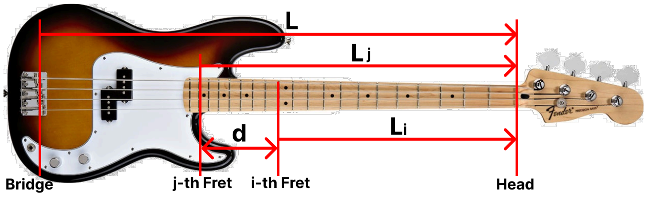

Figure 2: bass fret calculation

2 Implementation

2.1 Detecting Bass Elements

2.1.1 ***Bass string.*** To detect bass string, we used `OpenCV`'s `houghlines` function. The function performs the Hough Transform [4] and returns the distance r from the top-left corner of the image to the detected line and its angle θ. To remove irrelevant lines (e.g., from desks or windows), we first excluded outliers using the 3-sigma rule on θ values. Then, we applied k-means clustering (k=4) to group lines corresponding to each bass string. As a result, each bass string is represented by the r and θ of the cluster center (See Figure 1 (a)).

2.1.2 ***Bass fret.*** Detecting bass frets using line detection is challenging, as frets are short and occluded by strings, causing them to appear as fragmented lines. To address this, we estimated bass fret positions by detecting regularly spaced circular position marks aligned with the frets. Position marks are detected using `OpenCV`'s `houghcircles` function, which returns the x, y coordinates and the radius of circles. After detection, we removed circles outside the range of the first and fourth bass strings, then applied k-means clustering (k=9) to merge detected circles corresponding to each position mark. Clusters are labeled in descending order of their center x-coordinates and assigned the corresponding position mark labels (i.e., 3, 5, 7, 9, 12, 15, 17, and 19).

Leveraging the fact that the fret spacing follows the 12-tone equal temperament (12-TET) system, each bass fret position can be calculated mathematically based on the coordinates of the position marks as follows:

$$L = L_{i-1} \cdot R^{1-i}, \;\; L_{i-1} = \frac{2d}{1+R-R^{j-i}-R^{j-i+1}}, \;\; R = 2^{-1/12} \qquad (1)$$

where i and j are position mark numbers, d is the distance between i-th and j-th position marks, L is the distance from the bass bridge to the bass head, and L_{i-1} is the distance from the bass bridge to the i-th fret. One octave doubles the frequency with 12 notes, so R is defined as $2^{-1/12}$ based on the inverse relationship between distance and frequency (See Figure 2).

2.2 Detecting Finger Position

In bass guitar, where the same pitch can be played in multiple positions, identifying finger positions is essential to determine where each note was played. To achieve this, we used `Google MediaPipe` Hand Landmarks Detection [21] (See Figure 1 (c)). However, since bassists tend to place fingers near the frets [3], raw hand landmarks can reduce recognition accuracy. Therefore, we added 10% of the fret spacing as padding to both sides of the left-hand landmarks.

2.3 Recognizing Pitch and Note-on timing

The positions of each finger on the bass guitar can be identified, but determining which finger played a note requires pitch and note-on timing. To address this, Crepe and Crepe Note are used to recognize the pitch and note-on timing of the played notes. Crepe estimates the pitch and its confidence from the audio over time [9]. Based on this, Crepe Note detects the pitch, note-on times of each note [19].

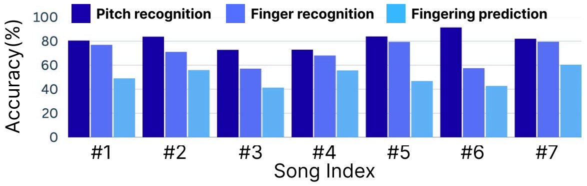

Figure 3: Data analysis result

3 Data acquisiton

We collected a total of 21 YouTube videos, featuring 10 different players. All bass guitars used by the players had 20 frets and circular position markers. Additionally, the videos were recorded from a fixed front-facing angle without any change in camera composition.

We also recruited 8 participants (7 male, 1 female) from local universities and musician recruitment platforms. Their playing experience ranged from 2 to 30 years (Mean = 8.5, SD = 9.20). Participants used their own 4-string, 20-fret bass guitar when available. Otherwise, a Squier Classic Vibe '60s Jazz Bass was provided. They listened through Marshall Major 5 headphones and monitored sound via a Cort CM15G amplifier. Performances were recorded in 4K@60fps using an iPhone 14 Pro on a tripod. Participants selected 5 out of 8 provided songs and had one week to practice.

4 Modeling and Evaluation

4.1 Modeling

Bidirectional LSTM (Bi-LSTM) has shown strong performance in fingering estimation for monophonic piano melodies in a single hand [5, 6]. We trained the Bi-LSTM model by providing input features relevant to fingering decisions on the bass guitar, including the finger number used to press the string and the spatial distance between the previous and current notes, represented as 2D coordinates across strings and frets.

4.2 Evaluation Metrics

To evaluate the transcription framework and the optimal fingering estimation model, we manually annotated the fingering sequence of performers by reviewing seven performance videos. We also converted the sheet music into data to obtain ground truth pitch information for the performed music.

4.2.1 ***Pitch Recognition Module.*** We compared the recognized pitches from audio with ground truth notes from the sheet music.

4.2.2 ***Finger Recognition Module.*** We compared the finger numbers identified from the video with the manually annotated fingering sequences. Recognition was considered correct when both the finger number and its corresponding string/fret position matched the annotation.

4.2.3 ***Optimized Fingering Estimation Module.*** We compared the predicted fingering sequences from the Bi-LSTM model with the recognized finger positions obtained from the video.

4.3 Results

The note recognition accuracy was 0.81 on average (SD=0.06), the position recognition accuracy was 0.77 (SD = 0.12), and the fingering prediction accuracy was 0.50 (SD = 0.07) as in the Figure 3. There were some notes missed by the framework in pitch recognition, which directly affect both finger recognition and fingering prediction accuracy. Therefore, improving the recognition performance for the missed notes in the framework is expected to enhance the overall model performance.

5 Conclusion

This paper presents the first deep learning-based approach for bass guitar fingering estimation, along with a framework for extracting fingering data. Results show that the framework achieves high accuracy and produces playable fingering predictions, supporting future research on string instrument fingering.

References

[1] Matteo Balliauw, Dorien Herremans, Daniel Palhazi Cuervo, and Kenneth Sörensen. 2017. A variable neighborhood search algorithm to generate piano fingerings for polyphonic sheet music. *International Transactions in Operational Research* 24, 3 (2017), 509–535.

[2] Eric Clarke, Richard Parncutt, Matti Raekallio, and John Sloboda. 1997. Talking fingers: An interview study of pianists' views on fingering. *Musicae Scientiae* 1, 1 (1997), 87–107.

[3] Gianpaolo Evangelista. 2011. Physical model of the string-fret interaction. In *DAFx'11, Paris, 2011*. 345–351.

[4] Robert Fisher, S Perkins, A Walker, and E Wolfart. 2003. Hough transform. *Hypermedia Image Processing Reference* (2003).

[5] Hongzhao Guan, Zhao Yan, and Timothy Hsu. 2021. Automatic Piano Fingering Estimation Using Recurrent Neural Networks. (2021).

[6] Xin Guan, Haoyue Zhao, and Qiang Li. 2022. Estimation of playable piano fingering by pitch-difference fingering match model. *EURASIP Journal on Audio, Speech, and Music Processing* 2022, 1 (2022), 7.

[7] Melanie Hart, Robert Bosch, and Elbert Tsai. 2000. Finding optimal piano fingerings. *The UMAP Journal* 21, 2 (2000), 167–177.

[8] Al Kasimi. 2007. A simple algorithm for automatic generation of polyphonic piano fingerings. *(No Title)* (2007), 355.

[9] Jong Wook Kim, Justin Salamon, Peter Li, and Juan Pablo Bello. 2018. Crepe: A convolutional representation for pitch estimation. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 161–165.

[10] Leopold Mozart. 1985. *A treatise on the fundamental principles of violin playing*. Vol. 6. Early Music.

[11] Wakana Nagata, Shinji Sako, and Tadashi Kitamura. 2014. Violin fingering estimation according to skill level based on hidden markov model. In *ICMC*.

[12] Eita Nakamura, Nobutaka Ono, and Shigeki Sagayama. 2014. Merged-Output HMM for Piano Fingering of Both Hands.. In *ISMIR*. 531–536.

[13] Eita Nakamura, Yasuyuki Saito, and Kazuyoshi Yoshii. 2020. Statistical learning and estimation of piano fingering. *Information Sciences* 517 (2020), 68–85.

[14] Caroline Palmer. 1997. Music performance. *Annual review of psychology* 48, 1 (1997), 115–138.

[15] Richard Parncutt, John A Sloboda, Eric F Clarke, Matti Raekallio, and Peter Desain. 1997. An ergonomic model of keyboard fingering for melodic fragments. *Music Perception* 14, 4 (1997), 341–382.

[16] Toky Hajatiana Raboanary, Fanaja Harianja Randriamahenintsoa, Heriniaina Andry Raboanary, Tantely Mahefatiana Raboanary, and Julien Amédée Raboanary. 2017. Finding optimal bass guitar fingerings. In *2017 IEEE AFRICON*. IEEE, 65–71.

[17] Daniele Radicioni, Luca Anselma, Vincenzo Lombardo, et al. 2004. A segmentation-based prototype to compute string instruments fingering. In *Proceedings of the Conference on Interdisciplinary Musicology*, Vol. 17. 97.

[18] Pedro Ramoneda, Dasaem Jeong, Eita Nakamura, Xavier Serra, and Marius Miron. 2022. Automatic piano fingering from partially annotated scores using autoregressive neural networks. In *Proceedings of the 30th ACM International Conference on Multimedia*. 6502–6510.

[19] Xavier Riley and Simon Dixon. 2023. CREPE Notes: A new method for segmenting pitch contours into discrete notes. *arXiv preprint arXiv:2311.08884* (2023).

[20] Yuichiro Yonebayashi, Hirokazu Kameoka, and Shigeki Sagayama. 2007. Automatic Decision of Piano Fingering Based on a Hidden Markov Models.. In *IJCAI*, Vol. 7. 2915–2921.

[21] Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. 2020. Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214* (2020).