

# AR Hero: Generating Interactive Augmented Reality Guitar Tutorials

Lucchas Ribeiro Skreinig\* Ana Stanescu Shohei Mori Frank Heyen Peter Mohr  
Michael Sedlmair Dieter Schmalstieg Denis Kalkofen

Graz University of Technology, University of Stuttgart



Figure 1: We introduce a system for generating interactive Augmented Reality guitar tutorials. The system is capable of generating step-by-step guitar tutorials from video and text description inputs. The tutorial shown in this example has been automatically generated from parsing a common digital guitar tablature (left). Our system shows the finger placement for single steps of the tutorial in augmented reality (right). The fret highlight is augmented directly on the neck of the guitar using a video see-through head-mounted display.

## ABSTRACT

We introduce a system capable of generating interactive Augmented Reality guitar tutorials by parsing common digital guitar tablature and by capturing the performance of an expert using a multi-camera array. Instructions are presented to the user in an Augmented Reality application using either an abstract visualization, a 3D virtual hand, or a 3D video. To support individual users at different skill levels the system provides full control of the playback of a tutorial, including its speed and looping behavior, while delivering live feedback on the user's performance.

**Keywords:** Augmented Reality, Tutorials, Visualization, Authoring, Human computer interaction

**Index Terms:** Human-centered computing [Human computer interaction (HCI)]; Interaction Paradigms—Mixed / Augmented reality; Human-centered computing [Human computer interaction (HCI)]; Interaction Paradigms—Graphical user interfaces

## 1 INTRODUCTION

Learning to play an instrument commonly involves following instructions generated by a teacher. With the success of online knowledge databases, a vast amount of learning materials exist. However, online content, such as text descriptions and video tutorials, often require existing knowledge to be understood, and the passive nature of most teaching material does not provide important feedback on the user's performance. Moreover, learning complexities like hand posture and motion in 3D space can be difficult using two-dimensional teaching material.

\*e-mail: lucchas.ribeiroskreinig@icg.tugraz.at

Thus, newcomers to playing the guitar often have trouble getting started with online teaching material, as a more experienced teacher would customarily teach basic techniques and notation literacy [5]. There are well-established tools for visualizing instructions, such as tablature diagrams (Figure 1) and fretboard charts. However, such visualizations are commonly two-dimensional, abstract and “distributed” [20], which means the visualization is represented separately from the instrument.

Augmented Reality (AR) solutions offer an environment for enhancing traditional learning experiences [7]. Augmentation of teaching material has the potential to make the subject matter more compelling and engaging to a user [11]. In addition, pairing instructions with a user's natural view assists with task localization and efficiency [6]. Chow et al. [2] state that one of the principal difficulties beginner piano learners face is “translating a note from the written score to the physical key on the keyboard”. The authors discuss that an AR-based approach can lessen these barriers and make it easier for new learners to get started with learning to play guitar. AR methods attempt to reduce the need for a “transformational process between external representation [...] and the instrument itself” [8], which should mitigate the barriers to learning an instrument. Rio-Guerra et al. [22] conclude that AR-assisted methods and traditional teacher-led methods of basic guitar instruction are both feasible for learning and they observe that users of their AR application were “more motivated” by the use of the application.

While AR enables associating in-place visualizations with notes, authoring AR tutorials can be a time-consuming process. The involved complexity is entirely dependent on the chosen guidance and visualization methods. Therefore, we propose an authoring tool for AR-guitar tutorials, which can utilize the vast amount of existing online learning content. In addition, we design a framework that uses the generated content for helping a user learn guitar basics via visual guides and live feedback on performance.

## 2 RELATED WORK

The music industry is no stranger to AR as an education tool. Devices such as LOOG Guitars<sup>1</sup> are increasing in popularity. Guitar basics are taught by an AR application, which superimposes instructions on where to place one's fingers onto the guitar neck. However, it requires the instrument to be rigidly held in place, as its position is not tracked.

**Visualization on Guitars.** Liarokapis and Anderson [11] discuss AR scenarios for higher education. The authors highlight the pedagogical advantages that the technology enables, such as intuitive integration of learning content into real environments. Importantly, they remark that the effectiveness of an AR system significantly depends on its implementation. In an earlier work, Liarokapis [10] investigates the correlation between the effectiveness of AR-based self-learning systems and the robustness of their integration. The author specifically proposes a method for teaching guitar basics by superimposing markers onto the guitar neck where learners should place their fingers. However, this implementation is solely focused on instructing a learner on what to play and the user's performance does not factor into the lesson at all.

Motokawa and Saito [19] propose a method which tracks a guitar and displays visual guides to the user on a monitor, superimposing virtual information over the image of the user. The pose and finger placement is communicated to the user in the form of a computer-generated representation of a hand, allowing a user to easily mimic the simulated pose. The authors comment, however, that not all users found the 3D hand representation entirely helpful.

Rio-Guerra et al. [22] observe that the challenges self-teaching guitar students face can be significant obstacles in their learning experience. These are commonly resolved by face-to-face music instruction, where a teacher might correct a newcomer's mistakes and provide further feedback. However, this can lead to students becoming disincorporated, as they can feel like they are not engaging actively in the learning process. Their proposed AR application teaches users guitar chords by superimposing colored markers on the instrument's neck. These colors correspond to colored stickers attached to a user's fingers and indicate where they should depress the strings in order to produce each chord. The authors conclude that learning using traditional methods and using their AR application produced similar results when comparing the time it took for users to learn new skills. Gutierrez et al. [13] observe that users can accurately play chords by imitating the movement and positioning of a 3D-animated hand superimposed onto their guitar's neck. Furthermore, newcomers without experience playing musical instruments were motivated by the AR application and expressed desire to continue learning the guitar in the future.

While the aforementioned works can accurately provide information to the user, they do not take the user's performance into account. By capturing user input, we can provide real-time feedback to what the user is playing on the instrument.

**Embedded Lessons.** Patzer et al. [20] discuss the implementation of an AR guitar, namely the Optek Fretlight<sup>2</sup> guitar. This instrument houses LEDs within the fretboard, which light up to indicate to the user where to place their fingers. The authors conclude that participants who trained with the AR-enabled guitar showed a marked increase in knowledge retention, when compared to participants who used traditional notation methods. Keebler et al. [8] note the same effect with the Fretlight guitar. By embedding the learning material within the instrument, the barrier of entry for learning is significantly lowered and newcomers performed better overall when using AR-assisted learning methods rather than traditional diagram-based tutorials. While embedding learning content into a real guitar is an immersive and effective method of visual guidance,

it does presuppose the purchase of specialized instruments. This reduces the appeal of the system, as users may prefer to use their own instrument.

Torres and Figueroa [26] describe an AR application which combines simulated face-to-face guitar lessons and AR-enhanced information visualization. Students learn guitar basics step-by-step, while the application visualizes the learning goal, such as which frets to press. A 3D avatar, integrated into the AR scene, provides motion-captured instructions and virtual information. The authors compare this method of self-learning to the more common practice of learning basic skills via video tutorial. Users were satisfied with the AR application, however test subjects experienced discomfort after prolonged use of the HoloLens, as the application suffers from the optical see-through device's narrow field-of-view.

Kumaravel et al. present Loki [9], a system which consists of two symmetric spaces containing a pass-through AR display each. The physical environments and objects around each user are captured via depth cameras and the resulting live point clouds are streamed to one another together with audio and optional annotations. This method of Mixed Reality (MR) telepresence is proposed for a multitude of applications, including musical instruction. The authors specifically highlight one-on-one guitar lessons as a potential scenario, but the approach requires depth cameras for tutorial streaming or recording. Additionally, the superimposed content is not registered to the user's guitar. By tailoring our system to guitar content and creating a solution for intuitive visualization of the learning material, our system is uniquely suitable for learning guitar basics.

**Authoring Instructions.** Harrison [5] investigates the challenges involved with learning guitar basics. Due to the nature of the guitar's layout, single notes can be played in a variety of ways, using many possible fingerings. Thus, notes in traditional staff notation can be translated to different fretboard locations. For this reason, many guitarists choose tablature notation, which directly maps numbers on specific guitar strings to a corresponding fret position being played. In addition, fretboard charts provide a popular visualization. They present chord fingerings as a diagram of fret positions, though they also require the user process of translating abstract notations to physical actions, making them difficult for beginners to understand.

Chow et al. [2] use AR to address the difficulty in identifying a piano key based on its written notation, as well as the issue of students lacking motivation. Lessons are stored in MIDI files, which contain timing data for each note in a song. Their system uses a head-mounted display (HMD) to overlay each key with lines symbolizing which notes to play, as well as feedback based on user performance captured via a USB MIDI interface. The authors implement controls for users to adjust the tempo of a lesson, and offer a mode which only continues playback once a user played the correct key. Most users preferred playing the piano based on the indicated notes in AR rather than reading the written notes. Das et al. [3] propose a setup with a similar visualization method, using a Microsoft HoloLens and a MIDI-over-Bluetooth enabled keyboard. Instead of loading content via MIDI files, we can generate lessons by parsing guitar tablature instead.

Mohr et al. [15, 16] and Yamaguchi et al. [27] present approaches to automatically author AR instructions from existing 2D data, such as image and video tutorials. These approach focus on extracting 3D instructions by registering and tracking the known 3D object in the 2D tutorial material. However, tracking the user's hands in guitar tutorials is challenging due to hand interactions with the guitar. Recently, Mohr et al. [17] introduced an approach for replacing the 3D model with a light field. The authors demonstrate the advantages of an image based object representation over conventional geometric reconstruction. Our approach extends this work to represent dynamic scenarios, such as a guitar lesson, using multi-view video data.

<sup>1</sup><https://loogguitars.com>

<sup>2</sup><https://fretlight.com>

### 3 REQUIREMENTS

The focus of this work is to develop an application that guitar beginners can use to get familiar with guitar tablature and chords, by learning in an immersive environment as an alternative to following video tutorials. Therefore, we determine several objectives, which we define as our design goals.

#### 3.1 Content Creation

Self-taught guitar students nowadays benefit from the wealth of learning material available on the Internet, such as on the popular website Ultimate Guitar<sup>3</sup>, which contains community-created tablature for numerous popular songs. The disadvantage of this notation, however, is in its ambiguity due to a lack of standardization and its missing timing information. The possibility to convert this kind of content to an interactive lesson with little effort can make our system more appealing to newcomers. Our goals are to provide a guidance tool that is easily expandable with more lessons and to make the creation of these lessons intuitive and possible without the need for additional software. Finally, keeping the storage and generation of lesson details simple and easy-to-use allows users to easily write their own tablature and extend their library of lessons.

While this method of authoring content is fast and simple, its major drawback is in its lack of fidelity. As any note on the guitar can be played with any finger, we can only approximate the intention of the teacher. Since the lesson material contains only information on which note to play and at which time, our system assigns virtual fingers to each fret depending on their location, in ascending string and then fret order. As one of our visual guidance solutions includes an animated virtual hand, correct placement of the fingers is instrumental, therefore we include a dataset of the most common chord poses for animating the hand, and procedurally animate the fingers if a given pose is not found in our chord collection.

For uncommon chords or more complicated poses, this may produce incorrect finger assignments that could result in incorrect visualizations. As an alternative, we instead can capture expert performances in 3D with multiple cameras and play back the lesson as an animated light field.

#### 3.2 Visual Guidance

The main design goal when choosing methods of visual guidance was to enable users to perceive the lesson content without the need to translate abstract notations such as tablature diagrams or fretboard charts. Our findings indicated that one of the most common methods of indicating to the user where to place their fingers was to highlight the regions of importance on the fretboard. As this method does not convey subtle details such as the intended finger shape, we additionally implement a method which displays an animated virtual hand, which the user can mimic. Offering multiple alternatives of visual guidance can help users achieve a broader understanding of the learning material and allow for a more tailored experience. In addition, the guidance in the form of traditional guitar tablature is still made available to the user but is not necessary for understanding the content. Representing the content both ways may help newcomers to the subject matter learn how to read the traditional notation.

By capturing user input while playing the guitar, we can indicate to the user when they make mistakes. Visually highlighting the error region immediately alerts the user and additionally saves them the time and potential frustration of searching for their own mistake.

#### 3.3 Hardware

One of our major design considerations was the choice of display method. The most common AR scenario in related literature was for virtual content and captured video to be displayed on a monitor.

<sup>3</sup><https://www.ultimate-guitar.com>

This display solution is simple and provides the user with high-fidelity AR view. The disadvantage to this method, however, is the fact that displayed video must be mirrored, which may confuse users who may already be used to seeing the image of a guitar fretboard. For this reason, we implement an alternative display method, which not only gives the user more options for selecting their preferred setup, but also showcases the broad application of our framework in varied MR applications. The alternative relies on an HMD, as this can supply AR content alongside the user's natural view. This has the added benefit that displaying real-world scaled guidance visualizations in first-person view is often considered more intuitive [29]. In order to be able to see the virtual instructions on top of the guitar while holding it, and at the same time maintain a natural position of the head and neck, we require a HMD with a large field-of-view, which excludes most commonly available optical-see-through devices.

### 4 AUTHORING

**Tablature Parsing** As mentioned in Section 3, it is important that additional content is easy to produce for our application. The most common notation method for guitar notes, specifically for beginners, is guitar tablature, where each line represents a string, and numbers indicate a note based on which fret on the string should be pressed to produce it. This became the basis for our method of storing guitar lessons.

Our system parses common text files which contain information on pitch and timing in the form of numbers and dashes, representing notes and pauses respectively, across six lines of text, which correspond to the individual guitar strings (Figure 2).

```
En4 | |----0-----|----0-----2---|----0-----|
Bn3 | |----1-----|----1-----3---|----1-----|
Gn3 | |--2-----|--0-----2---|--2-----|
Dn3 | |-----0-----|-----|
An2 | |0-----|3-----|0-----|
En2 | |-----|-----|-----|

|CRD| |G---G-G---G-G---|G---G-G---G-G---|e---e---e---e---|
```

Figure 2: Example content of text lessons, represented as ASCII tablature. The top lines stand for each string individually, together with the notes to be played. The bottom line represents an alternative notation, containing chords denoted by single characters.

Every measure of a song is delimited by a horizontal line, which makes visually separating the song into bars easier. Requiring the same number of characters between measures implicitly supplies timing information for the notes as well. The time signature of a song is 4/4 or “common time” by default, where each measure contains four beats, though this can be adapted by adding a line to the start of the text file which indicates the number of beats in each measure. The tuning of the guitar strings can be adapted by prefacing the lines which represent the guitar strings with their tuning written in scientific pitch notation [28], e.g. “En4” for E natural in octave four, or “Bb3” for B flat in octave 3.

Alternatively to creating lessons note-by-note, it is possible to define a song using only chord names. With this method, instead of lines which represent each string, only one line with the preface “CRD” is needed and the chords are denoted by upper- or lower-case letters, representing major and minor chords, respectively. In order to represent each chord using just one character, only the major and minor chords of the seven natural notes are compatible with this method.

**Virtual Hands.** From a 3D human model library, MakeHuman<sup>4</sup>, we generated a collection of left hand models with different sizes and finger shapes, as users prefer the option to customize

<sup>4</sup><http://www.makehumancommunity.org>



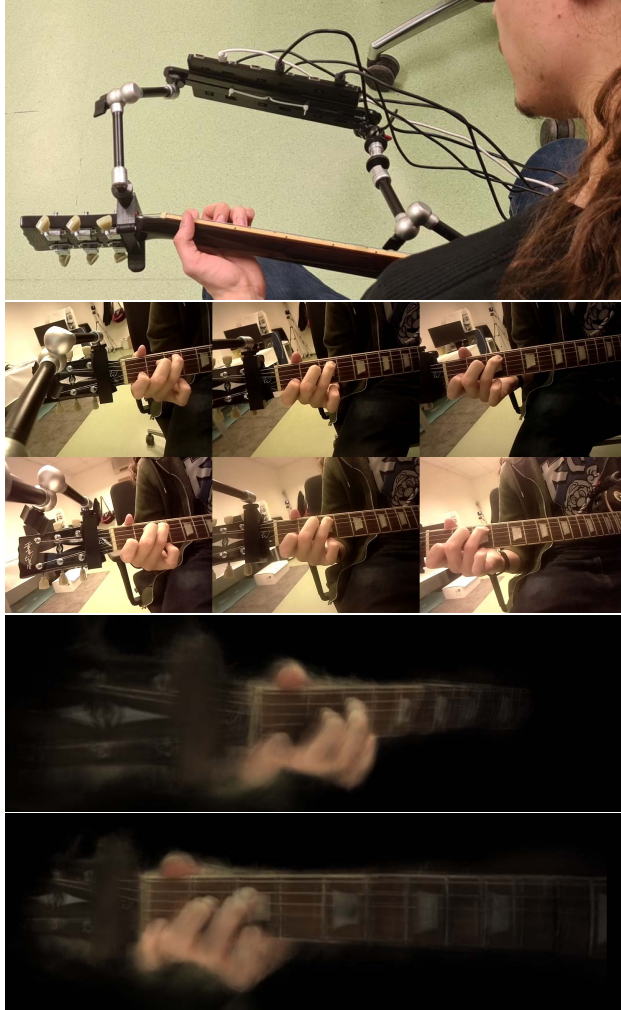


Figure 3: Light field capture rig and example images. Camera rig consisting of six cameras (top), captured multi-view images captured produced by rig (middle), example light field renderings from two different viewpoints (bottom).

the appearance of the virtual representation of their own hand [12]. We apply an inverse kinematics (IK) model to animate the hand in physically correct finger placements on the guitar. We include the transformations which produce the poses corresponding to the most common fingerings of the major and minor C, D, E, and G chords. These were the most common chords encountered during testing, though the data set can always be extended in the future. The transformations are used for animating the virtual hand whenever the lesson includes one of these chords. If a specific pose is not included in our data set, the pose is procedurally animated instead. Here the virtual location for each finger is determined based on the lesson data and the hand is deformed accordingly using the IK model.

**3D Video.** Alternatively to generating lessons by parsing tablature, we can record and playback actual performances in 3D with a portable light field camera rigged on the guitar headstock. Our camera rig consists of six cameras powered by Raspberry Pi Zero W computers<sup>5</sup> (Figure 3), which send their video streams individually to the main computer over USB. The relative camera positions

<sup>5</sup><https://www.raspberrypi.com/products/raspberry-pi-zero-w/>

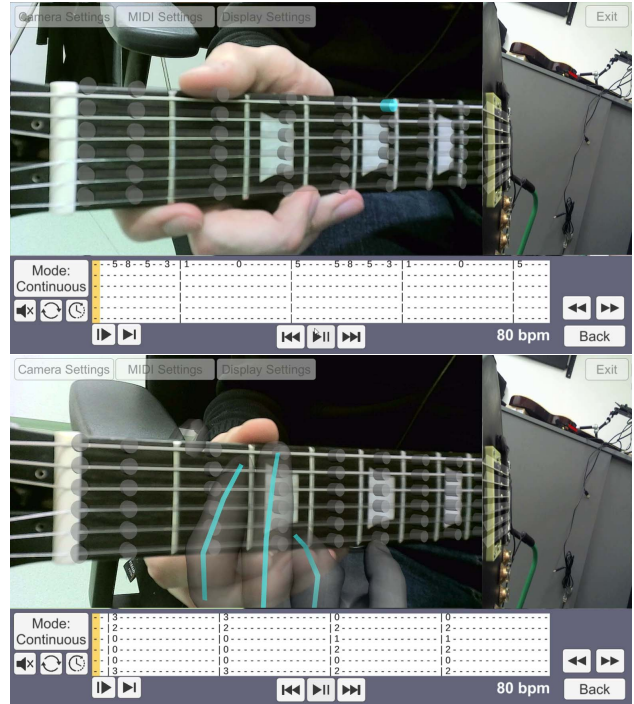


Figure 4: Visualizations to indicate finger placement on the guitar neck: using virtual fret markers that light up to indicate finger positions (top), and employing a computer-generated 3D hand model showing the full hand pose (bottom).

are determined using the COLMAP feature extractor and matcher pipeline [23, 24]. The images are fed to a light field reconstruction algorithm, generating a representation that enables us to view the captured scene later from free viewing angles.

## 5 INTERACTIVE VISUALIZATION

The guitar tablature is displayed to the user separately, to convey which notes are coming next, or to familiarize the user with a more traditional notation approach, but it can optionally be entirely disregarded.

**Fret Highlighting.** Concerning the user interface for finger placement instructions, Liarokapis [10] proposes a method for displaying the necessary placement of each finger directly onto the fret board, a concept that is used similarly in Optek’s Fretlight guitar [20]. We endeavor to display the information in a similar way by embedding the material to be learned within the instrument itself. Our system superimposes markers onto the guitar neck at the points between fret bars, as visualized in Figure 4. Coloring these markers communicates to the user which frets to depress. We additionally superimpose the strings with three-dimensional representations thereof and light these up to communicate to the user when to pluck each string.

**Virtual Hands.** A 3D virtual hand registered on the guitar neck provides an intuitive visual clue for the learner on how to pose the hand and fingers [19]. The benefit of the virtual hand, instead of seeing a physical hand on the teacher’s guitar, is that the learners can directly trace the shape on their own guitar neck. We render a 3D hand model as seen in Figure 4. Originally, the virtual hand had an opaque material, which made it difficult to see the positions of the fingertips. Instead, we apply a custom semitransparent shader with a Fresnel effect, which enhances edges. This makes the contour of each finger more visible, while letting users see the

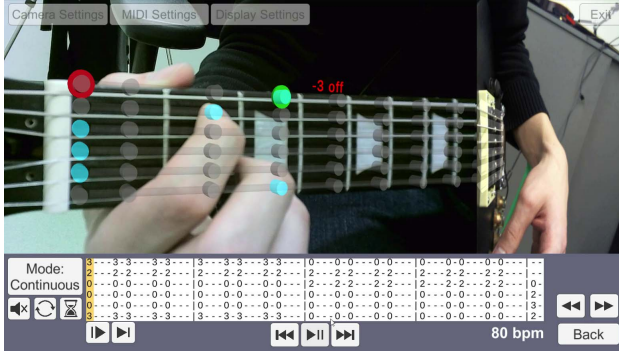


Figure 5: Error feedback visualized as a red indicator, annotated with a floating text, in order to inform the user on false inputs. This should allow the user to see and correct their mistakes quickly. In this example, an open string was played instead of the third fret.

fretboard and their own hand beneath the virtual representation. To further highlight the fingertips, light blue lines are rendered along the trajectory of each finger, making the pose even easier to mimic.

**Video Captured Hands.** The 3D virtual hand communicates the shape and position of each finger. However, it is only an abstraction of the teacher’s intention. The fingers are automatically assigned to predetermined fretboard locations since the lesson material only contains information of notes and timing to play. However, any finger can be assigned to any note. This may not always result in the correct assignment and can trigger the hand animation unexpectedly. We therefore implement an alternative solution to record the teacher’s hand as a 3D video using the aforementioned light field camera rig. We feed the calibrated images per frame to a 3D Convolutional Neural Network of the Local Light Field Fusion approach [14] to calculate per-camera multiplane images (MPIs). For compression, we encode each image in the MPIs in the PNG format. We project every MPI from furthest to the closest by applying over alpha composition and switching the loaded MPI frame by frame for a video. Some MPI layers, e.g., layers assigned to the background, can be filtered out depending on the depth to keep only the relevant information. Consequently, the resultant rendering is a guitar performance in a free-viewpoint video (Figure 3). We implemented the light field renderer using Unity 3D<sup>6</sup>.

**Playback Control** The lessons typically go through each beat one-by-one and the interval of time between each beat is determined by the program’s metronome. The rate of new notes is displayed in beats-per-minute (bpm) and time is kept by the system’s clock. Instead of simply working through a guitar lesson from start to finish, users can control how and when the lesson ends or loops. In addition, users can toggle a responsive mode. In this mode, each beat which contains notes to be played by the user is halted until the user plays the correct notes, which adjusts the content to the pace of the user [21] and can help them familiarize themselves with the content. As previously mentioned, we implemented a range of looping modes that affect the behavior of the lesson. A continuous mode, where each lesson is played from beginning until the end, a mode where it is played measure-by-measure, and a mode where the user can set custom starting and ending points for a loop.

**Feedback.** Chow et al. [2] describe a method by which user input is provided by a MIDI interface and compared to a reference MIDI file. User performance is then graded according to accuracy in pitch, timing, and dynamics. Similarly, using the previously described lesson files, we can tell which note should be played at what

<sup>6</sup><https://unity.com/>



Figure 6: Color cameras are attached rigidly to the guitar. One camera is mounted on the guitar’s headstock oriented towards the guitar’s neck (left), while a second camera is mounted on the fretboard, to capture the strumming (right).

time. We can calculate each note’s frequency  $f$  based on its distance to  $Cn0$ , which has a frequency of 16.352 Hz, applying Formula 1, where  $n$  is the number of half-steps between each note and  $Cn0$  [25].

$$f = 16.352 * 2^{\frac{n}{12}} \quad (1)$$

The notes plucked by a user are captured by a Fishman TriplePlay Connect<sup>7</sup> MIDI interface, which passes the played note on each string to a PC via a separate MIDI channel. The input is parsed by our system with a modified implementation of Keijiro Takahashi’s MidiJack<sup>8</sup> C# library. The played frequencies can then be compared against the previously calculated frequencies. Upon playing a wrong note, the note that should have been played is highlighted in red, indicating that the input was incorrect. Additionally, floating text is displayed to the user in the error region, as can be seen in Figure 5. This text spells out how many frets higher or lower their finger should be placed. This instant feedback is designed to save the user the time commonly spent searching for and correcting their own mistakes.

## 6 DISPLAY

Integrating lessons into the instrument itself is one of our main goals, as our plan is to reduce the user’s need to interpret musical notations. In our application, we implement two separate AR scenarios:

**Magic Mirror.** Our desktop-based setup is similar to a “magic mirror” [4], where the camera view and AR content are visible on a monitor. We rigidly attach two webcams to a guitar, one with a view of the fretboard, and another one looking down at the bridge (Figure 6). The two images are displayed side-by-side on the monitor, simulating a panoramic view of the guitar, with the frets and strings clearly visible. The user registers four distinct points on the guitar fret board by mouse click, which are used by the OpenCV library [1] as input points for its Perspective-n-Point solver. This calculates the 3D-2D point correspondences, allowing us to align the three-dimensional virtual fret board with the image, the result of which can be seen in Figure 7 (top). The controls, tablature, and settings are displayed under the view of the guitar and can be interacted with through a mouse.

The virtual cameras are calibrated such that the reprojection of the user-defined points should succeed without significant error.

<sup>7</sup><https://www.fishman.com/portfolio/tripleplay-connect-midi-guitar-controller>

<sup>8</sup><https://github.com/keijiro/MidiJack>



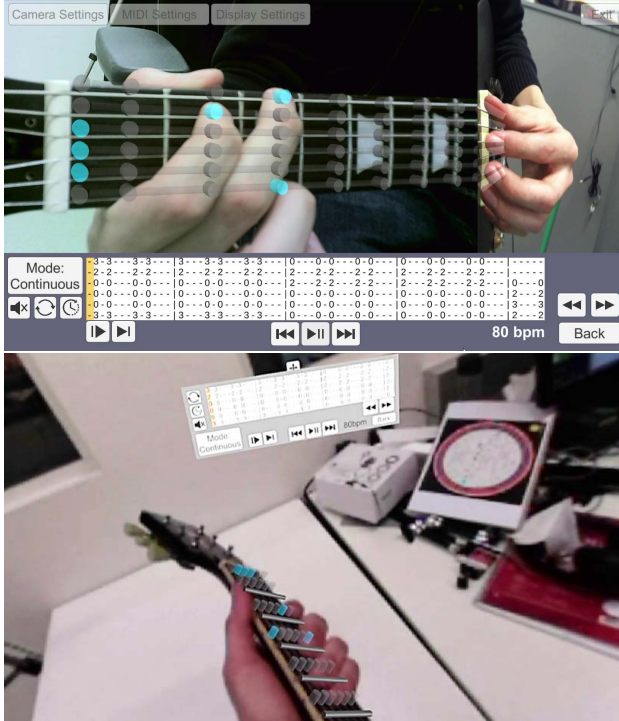


Figure 7: Examples of supported displays environments. We use a “magic mirror” setup with a screen space menu on a monitor (top), and a first-person experience on a video-see through HMD with a spatially registered interface (bottom). Both methods show registered fret markers for guiding the user.

However, in the case of deviations, the virtual camera intrinsics can be manually adjusted. The “Display Settings” menu contains sliders for tweaking the focal length in the x and y directions. The re-projections of the four selected points are made visible during this procedure, which helps with the adjustment of the virtual camera parameters.

**Video See-Through HMD.** Originally, we planned to implement an interface for the Microsoft HoloLens. However, as Torres and Figueroa [26] point out, prolonged use is not comfortable for the user. This is, in part, due to the fact that displaying information on the guitar neck becomes difficult when holding the instrument, as the narrow field-of-view makes lining up the guitar with the virtual content difficult. Another issue with this method was that it was challenging to use the device’s spatial mapping feature to track the guitar itself, since the instrument is likely to move freely in the hands of the user, and thus, requires an independent tracking solution. We tried to overcome this issue by using the Vuforia framework<sup>9</sup> to track an image marker attached to the guitar. However, unobtrusively positioning the marker and headset in such a way that its built-in camera could both track the marker and overlay information onto the guitar neck was uncomfortable for the user.

Due to these shortcomings, our HMD display technique is achieved via video see-through (VST) display, in our case with an HTC Vive Pro<sup>10</sup>. Using a VST display greatly increases the effective field-of-view when compared to optical-see-through methods, which is necessary for this application. VST displays have the drawback of introducing minimal latency, but this was not noticeable during our current implementation. We rigidly attach an HTC

Vive Tracker to the guitar allowing it to be tracked within the scene. The frets and strings are immediately visible to the user and superimposed onto the real guitar (Figure 7). The controls, tablature, and settings are displayed in interactive panels and can be manipulated using a separate HTC Vive Controller, which casts a ray pointer into the scene.

One issue we are faced with when using this method of tracking the guitar is that vibrations, caused in our case by plucking certain notes on the guitar, affect the inertial measurement unit (IMU) performance<sup>11</sup>. This becomes noticeable during use, as the virtual overlay can vibrate or drift off. The misalignment is quickly rectified by the absolute tracking of the HTC Vive Base Stations.

## 7 CONCLUSION

By gearing our implementation towards solving the aforementioned issues with related work, we designed a robust framework for guitar tutoring that is user-friendly and appealing to newcomers. Considering our design requirements, we can identify the ways in which we met our goals, and also observe where we could not meet expectations.

Authoring lessons by parsing tablature is a fast and reliable way to generate learning content. Online tablature only requires slight modification to become compatible with our application, and once a lesson is ready, it is memory-effective and easily shareable. The drawback to this method is its lack of precision when the application must approximate which finger should be assigned to which note. The alternative method of light field generation captures the teacher’s intention exactly but requires more time to prepare. In addition, the fidelity of the reconstruction highly depends on the quality of the captured images.

When it comes to visualization, displaying lesson content by highlighting the frets is a precise method of conveying information, but requires additional perceptual effort by the user, as it is an abstraction of the teacher’s intent. The animated virtual hand better represents the intended action but lacks visual clarity due to potential self-occlusion. Other issues include the IK calculations failing to produce a lifelike deformation of the hand model, as well as the aforementioned possibility of incorrect finger placement. These can be addressed by the light field reconstruction method.

The “magic mirror” display solution requires less hardware than the HMD-based method, making it more accessible than the alternative. However, one user who was more familiar with the instrument considered this representation confusing, as it mirrors the guitar image. The alternative VST display method is more intuitive, as it projects lesson content directly into the user’s real view but requires precise calibration of the tracker and more computational resources. In addition, our setup relies on the low-resolution cameras of the HTC Vive Pro, which may negatively affect the user experience.

## 8 FUTURE WORK

We intend to improve on the fidelity of our system by implementing several improvements. Improving the IK-transformations, as well as adding more poses to the database can further improve the look and realism of the virtual hand. In addition, optimizing the distance and angles between cameras for light field capture will produce MPIs that allow for steeper viewing angles, which can further increase their viability for MR integration. We will furthermore improve the visual clarity by expanding our VST setup with a higher-resolution stereo camera rig, and we plan building a mobile solution based on TrackCap [18] for localizing the guitar at runtime.

Finally, we intend to perform a user study in order to determine users’ acceptance of the system, and to investigate its effectiveness in teaching new users by comparing with more common self-teaching methods such as video tutorials and text descriptions.

<sup>9</sup><https://developer.vuforia.com>

<sup>10</sup><https://www.vive.com/eu/product/vive-pro/>

<sup>11</sup>HTC VIVE Tracker (2018), Developer Guidelines, Ver. 1.0

## ACKNOWLEDGEMENTS

This work was enabled by the Austrian Science Fund FWF (grant no. P30694), as well as by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2075 - 390740016.

## REFERENCES

- [1] G. Bradski. The OpenCV Library. In *Dr. Dobb's Journal of Software Tools*, 2000.
- [2] J. Chow, H. Feng, R. Amor, and B. C. Wünsche. Music Education using Augmented Reality with a Head Mounted Display. In *Proc. Australasian User Interface Conference (AUIC)*, volume 139, pages 73–79, 2013.
- [3] S. Das, S. Glickman, F. Y. Hsiao, and B. Lee. Music Everywhere-Augmented Reality Piano Improvisation Learning System. In *Proc. International Conference on New Interfaces for Musical Expression (NIME)*, pages 511–512, 2017.
- [4] M. Fiala. Magic Mirror System with Hand-held and Wearable Augmentations. In *Proc. IEEE Virtual Reality (VR)*, pages 251–254, 2007.
- [5] E. Harrison. Challenges Facing Guitar Education. In *Music Educators Journal*, volume 97(1), pages 50–55, 2010.
- [6] S. Henderson and S. Feiner. Exploring the Benefits of Augmented Reality Documentation for Maintenance and Repair. In *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, volume 17(10), pages 1355–1368, 2010.
- [7] J. R. Keebler, T. J. Wiltshire, D. C. Smith, and S. M. Fiore. Picking Up Steam: Educational Implications for Teaching With an Augmented Reality Guitar Learning System. In *International Conference on Virtual, Augmented and Mixed Reality (VAMR)*, pages 170–178. Springer, 2013.
- [8] J. R. Keebler, T. J. Wiltshire, D. C. Smith, S. M. Fiore, and J. S. Bedwell. Shifting the Paradigm of Music Instruction: Implications of Embodiment Stemming From an Augmented Reality Guitar Learning System. In *Frontiers in Psychology*, volume 5. Frontiers Research Foundation, 2014.
- [9] B. T. Kumaravel, F. Anderson, G. Fitzmaurice, B. Hartmann, and T. Grossman. Loki: Facilitating Remote Instruction of Physical Tasks Using Bi-directional Mixed-reality Telepresence. In *Proc. ACM Symposium on User Interface Software and Technology (UIST)*, pages 161–174. Association for Computing Machinery, Inc, 10 2019.
- [10] F. Liarokapis. Augmented Reality Scenarios for Guitar Learning. In *Proc. of Theory and Practice of Computer Graphics (TPCG)*, pages 163–170, 2005.
- [11] F. Liarokapis and E. F. Anderson. Using Augmented Reality as a Medium to Assist Teaching in Higher Education. In *Eurographics 2010*, pages 9 – 16. Eurographics Association, 2010.
- [12] L. Lin, A. Normoyle, A. Adkins, Y. Sun, A. Robb, Y. Ye, M. Di Luca, and S. Jörg. The Effect of Hand Size and Interaction Modality on the Virtual Hand Illusion. In *Proc. IEEE Virtual Reality (VR)*, pages 510–518, 03 2019.
- [13] J. Martin-Gutierrez, M. S. D. R. Guerra, V. Lopez-Chao, R. H. S. Gastelum, and J. F. V. Bojórquez. Augmented reality to facilitate learning of the acoustic guitar. In *Applied Sciences*, volume 10(7). Multidisciplinary Digital Publishing Institute (MDPI), 4 2020.
- [14] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar. Local Light Field Fusion: Practical View Synthesis with Prescriptive Sampling Guidelines. In *ACM Transactions on Graphics (TOG)*, volume 38(4), pages 1–14, 2019.
- [15] P. Mohr, B. Kerbl, M. Donoser, D. Schmalstieg, and D. Kalkofen. Retargeting Technical Documentation to Augmented Reality. In *Proc. ACM Conference on Human Factors in Computing Systems (CHI)*, pages 3337–3346, 2015.
- [16] P. Mohr, D. Mandl, M. Tatzgern, E. Veas, D. Schmalstieg, and D. Kalkofen. Retargeting Video Tutorials Showing Tools With Surface Contact to Augmented Reality. In *Proc. ACM Conference on Human Factors in Computing Systems (CHI)*, pages 6547–6558. ACM, 2017.
- [17] P. Mohr, S. Mori, T. Langlotz, B. H. Thomas, D. Schmalstieg, and D. Kalkofen. Mixed Reality Light Fields for Interactive Remote Assistance. In *Proc. ACM Conference on Human Factors in Computing Systems (CHI)*, pages 1–12, New York, NY, USA, 2020. Association for Computing Machinery.
- [18] P. Mohr, M. Tatzgern, T. Langlotz, A. Lang, D. Schmalstieg, and D. Kalkofen. TrackCap: Enabling Smartphones for 3D Interaction on Mobile Head-Mounted Displays. In *Proc. ACM Conference on Human Factors in Computing Systems (CHI)*, pages 1–11, 2019.
- [19] Y. Motokawa and H. Saito. Support System for Guitar Playing Using Augmented Reality Display. In *Proc. International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 243–244, 2006.
- [20] B. Patzer, D. C. Smith, and J. R. Keebler. Novelty and Retention for Two Augmented Reality Learning Systems. In *Proc. Human Factors and Ergonomics Society Annual Meeting*, volume 58(1), pages 1164–1168. Human Factors and Ergonomics Society Inc., 2014.
- [21] S. Pongnumkul, M. Dontcheva, W. Li, J. Wang, L. Bourdev, S. Avidan, and M. F. Cohen. Pause-and-Play: Automatically Linking Screencast Video Tutorials with Applications. In *Proc. ACM Symposium on User Interface Software and Technology (UIST)*, pages 135–144. Association for Computing Machinery, 2011.
- [22] D. Rio-Guerra, M. Sylvia, J. Martin-Gutierrez, V. A. Lopez-Chao, R. Flores Parra, and M. A. Ramirez Sosa. AR Graphic Representation of Musical Notes for Self-learning on Guitar. In *Applied Sciences*, volume 9(21), page 4527. Multidisciplinary Digital Publishing Institute (MDPI), 2019.
- [23] J. L. Schönberger and J.-M. Frahm. Structure-from-Motion Revisited. In *Proc. Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016.
- [24] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm. Pixelwise View Selection for Unstructured Multi-View Stereo. In *Proc. European Conference on Computer Vision (ECCV)*. Springer, 2016.
- [25] B. Suits. Frequencies for Equal-tempered Scale. In *Physics of Music—Notes*. Physics Department, Michigan Technological University, 1998.
- [26] C. Torres and P. Figueroa. Learning How to Play a Guitar with the HoloLens: A Case Study. In *Proc. XLIV Latin American Computer Conference (CLEI)*, pages 606–611, 2018.
- [27] M. Yamaguchi, S. Mori, P. Mohr, M. Tatzgern, A. Stanescu, H. Saito, and D. Kalkofen. Video-Annotated Augmented Reality Assembly Tutorials. In *Proc. ACM Symposium on User Interface Software and Technology (UIST)*, pages 1010–1022, 2020.
- [28] R. W. Young. Terminology for Logarithmic Frequency Units. In *The Journal of the Acoustical Society of America*, volume 11(1), pages 134–139. Acoustical Society of America, 1939.
- [29] X. Yu, K. Angerbauer, P. Mohr, D. Kalkofen, and M. Sedlmair. Perspective Matters: Design Implications for Motion Guidance in Mixed Reality. In *Proc. International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 577–587, 2020.