TabDetect Real-Time Viewer

The TabDetect Real-Time Viewer is a small application that can be used to visualize detected guitar tablature to an audio input using a keras model.

Motivation

Developing neural networks with a learning framework is a very abstract task. Usually, a model is defined, trained and evaluated without visible output. The only way for a user to measure performance is by introducing metrics to compare different approaches. Especially in guitar tablature detection, this can be frustrating, as the user will never really hear and see how the models perform, and in which situations they are are error-prone.
The TabDetect Real-Time Viewer was developed especially for this scenario. Its main functionality is to show detected guitar tablature while the audio is played. This can help the user to analyze chords or string/fret combinations to see if the given keras model is performant in these scenarios.

Application

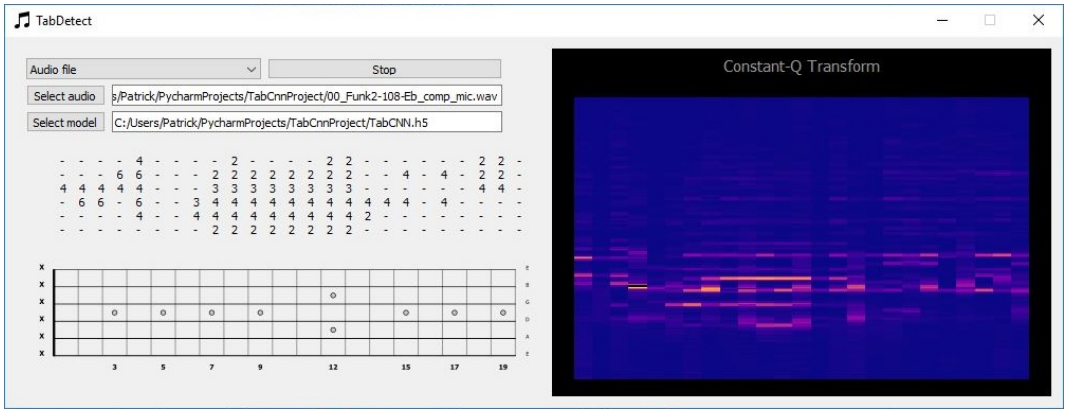
The general user interface of the TabDetect Real-Time Viewer consists of four sections:

- A setup part, where the model file, the application mode is configured and the processing can be started or stopped
- The section that shows the detected tabs and keeps them for 20 detections
- A graphical representation of current string/fret combinations
- The spectogram for the same 20 detections that are shown in the tab section

The setup section contains buttons and file selectors to configure the application. The most important parts are the selection of a keras model file and the start button.

The 20 detections show what string/fret combinations are played at each time frame, with the upper line representing the high E string of the guitar. A - shows that a string is not played.

The graphical representation can be interpreted like viewing a guitar from



The UI of the TabDetect Real-Time Viewer

above, with the hollow body on the right side. Again, the top line is the high E string. When a string is not played, a small x is displayed left of the string. In any other case, a dot appears at the played fret position for each string, containing the musical note that it produces. For better orientation, below the graphic there are numbers that show the fret for each dot marker.

The spectogram is generated using the Constant-Q Transform, which is the same transform used as the input for the model. It gives a linear representation of the frequency domain.

While the goal to detect tabs in real-time for an audio input stays the same, the TabDetect Real-Time Viewer has two different modes of operation:

Recording mode: In this mode, the detection happens for audio input from one of the systems audio drivers. To activate this mode, select an audio driver from the dropdown and a model file with the *Select model* button. Then the *Start* button can be clicked and detection begins. There is one additional feature in this mode: the audio input is recorded to the file *input_audio.wav* and the detected tabs are written to the file *input_tabs.wav*.

Playback mode: The playback mode enables the user to view the tablature for an existing audio file of a guitar recording. To activate this mode, select the option *Audio file* from the dropdown. After that, a wave audio file has to be selected with the *Select audio* button and a model file using the *Select model* button. Now, when the user clicks *Start* the audio file is played and the tabs are viewed in real-time.

Limitations

The TabDetect Real-Time Viewer currently has some limitations. The only supported input file type is wave audio. Other file types are deactivated. They could theoretically work out of the box, but are not tested.
Also, the model file must have an input shape of 192x9x1 and an output shape of 6x21. Other input shapes are currently not supported. Increasing the context window can also be hard regarding the real-time nature of the application

Findings

Visualizing the results of tablature detection networks was helpful to analyze their performance. Some of the key findings are the following:

- Frequent chords and string/tab combinations are detected much more likely than non-frequent chords. That means that currently errors can happen when irregular or rare chords are played. An option to avoid this would be to increase the size of the dataset used to train the models.
- The dataset was recorded using the same guitar for all users. This means that the models are fit exactly to the sound characteristics of that guitar and they are not performing equally well on inputs of other guitars.
- By downsampling to 22050 kHZ, Wiggins and Kim planned to reduce complexity assuming that the lost data is not very important. Using the real-time viewer, we found that a lot of errors are on the higher strings. This might or might not be improved by removing the downsampling. Experiments with the sample rate could be done in future works.

Conclusion

Summing up, the TabDetect Real-Time Viewer was a helpful tool to get insights in developed tablature detection models. Showing how the models work in a less abstract way than just quality metrics helped to find the pitfalls of tablature detection in this context.