Team #16: From Strings to Sequences — Classifying and Generating Music from Acoustic Guitar Notes

Camilo Martínez 7057573
Dhimitrios Duka 7059153
Honglu Ma 7055053

1. Task and Motivation

Automatic cord recognition (ACR) consists of recognizing the chords played in a music piece. This information is quite valuable since it can later be used for music analysis, music transcription, or even fixing corrupted musical performances. ACR was first introduced in 1999 by [19] where the author utilized lisp music to perform chord recognition at the signal level. Since then, many signal-level-based approaches have been introduced. However, these methods proved to be quite challenging and not very accurate.

With the rise of deep learning, and especially computer vision, many researchers started to tackle this problem from a different perspective. They began to use spectrogram-based feature extraction methods to extract the features of the audio signal [2, 11, 18]. However, despite the success of these methods, improvements plateaued "due to the inherent shortcomings of the aural approach in handling highly timbre sounds" [7].

Building on the concept of using visual information for musical applications, Y. Kristian et al. [12] employed a Single Shot Detection (SSD) model undergirded by a MobileNetV2 [16] base model, pre-trained on the EgoHands [1] dataset to achieve fretboard detection and chords classification. Their model generates coordinates for bounding boxes that outline the fretboard which in turn is used as the input for the chord classification model.

Our work builds on top of the previously mentioned works [12] [7] and aims to improve the accuracy of chord recognition as well as implement a chord-to-audio generation model.

2. Goals

Fig. 1 illustrates a bird's eye view of our model architecture. Essentially, we aim to address the following three problems:

1) *Fretboard Detection*: Given an image or video frame, detect the bounding box that outlines the fretboard.
2) *Chord Classification*: Given an image or video frame, classify the chord being played.
3) *Seamless Audio Generation*: Given the chords being played, generate the audio of the music piece.

Y. Kristian et al. [12] focused on the first two problems, while we aim to address the third problem as well. This is a challenging task since it requires the model to seamlessly generate audio from the classified chords, effectively crossing into the *generative* side of Neural Networks. That is, we aim to move beyond chord classification to also include audio synthesis. In Section Sec. 3 and Sec. 4, we briefly cover the methods and datasets we plan to use to address these challenges.

By the mid-term, we aim to have completed data collection and pruning, the code for pre-training a YOLO architecture for the *fretboard detection model*, and the *guitar chords classifier model*.

3. Methods

We are expecting to use the following models to solve the challenges[1]:

1) **Fretboard Detection**: We plan to use YOLOv9 [20], a SOTA model in 2024, and fine-tune it for fretboard detection using the datasets mentioned in Sec. 4.
2) **Chord Classification**: We plan to use a tranformer-based approach, mainly ViT [6], and fine-tune it for guitar chord classification using the datasets mentioned in Sec. 4.
3) **Seamless Audio Generation**: We plan to try both a plain decoder architecture and a transformer-based approach, MelodyDiffusion [13].

Since the original video frames may be of different viewpoints, orientations or environments, preprocessing the image with a *fretboard detection model* will help to standardize the input for the *chord classifier model*. Although the visual transformer model has demonstrated the ability to

[1] If deemed necessary, we will also explore other models.

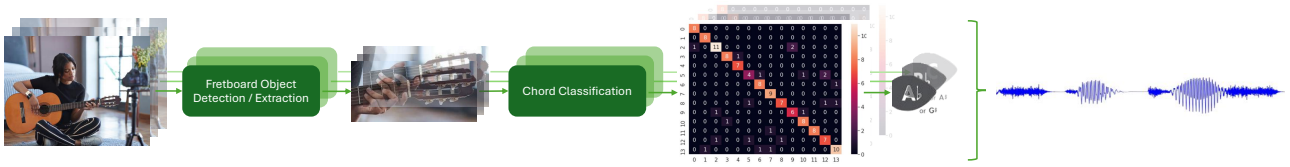

Figure 1. Overview of the Model, showcasing the 2 most important tasks: Fretboard Detection and Chord Classification, are done for each frame of an input video. Image taken from *Getty Stock Images*, confusion matrix image taken from [12].

capture features with a relatively small patch size, in our case, the fretboard cropping step will help us focus on the finger positions with more detail and reduce computational costs as well. Furthermore, by including the *seamless audio generation model*, we aim to provide a more complete solution to the problem of automatic chord recognition.

Unlike Y. Kristian et al. [12], we do not plan to use Convolutional Neural Networks (CNNs) as a backbone for our approach. They used MobileNetV2 and MobileNetV1 [8], which are CNN-based, for the fretboard detection and a Deep CNN for the chord classification.

We are planning to use mostly pre-trained models and leverage transfer learning techniques. For the *seamless audio generation model*, we are expecting a higher computational budget, as most likely we will have to train the model from scratch.

4. Datasets

We have chosen the following datasets to train and evaluate our models, depending on the specific task to address: *fretboard detection*, *chord classification*, and *seamless audio generation*. Furthermore, we follow the contributions of [12] and [9] by first using three different pre-trained datasets to start off our models and leverage transfer learning techniques to improve the overall performance.

4.1. Pre-trained Datasets

The chosen datasets have their own features, thus each one is used to pre-train a specific model.

1) **ImageNet Dataset**: This dataset is used for pre-training our *guitar chords classifier model*. It has over 14 million images that cover 20,000 types of natural objects [15].
2) **COCO Dataset**: This dataset is used for pre-training our *fretboard detection model*. The dataset is of considerable size and is dedicated to object identification. Approximately 200,000 labeled images are organized into 80 distinct categories [14]. Although somewhat comparable to ImageNet, the COCO dataset possesses a distinct emphasis.
3) **EgoHands Dataset**: Similar to ImageNet, this dataset is also used for pre-training our *guitar chords classifier model*. The dataset includes over 15,000 hand images with high-quality labels [1].
4) **GuitarSet**: This dataset is used for pre-training our *seamless audio generation model* from the classified guitar chords. It provides high-quality acoustic guitar recordings alongside time-aligned annotations including fret positions, and chords, among others [22].

4.2. Fretboard Detection

To fine-tune the previously pre-trained *fretboard detection model*, we will use the following datasets publicly available in Roboflow: [10], [17] and [5].

4.3. Chord Recognition

To fine-tune the *guitar chords classifier model*, we will use [21], [3], [4]. Since [10] contains both the fretboard and the chords, we will use it to fine-tune this model as well.

4.4. Seamless Audio Generation

To fine-tune the *seamless audio generation model*, we will use the [22].

5. Evaluation

Given the nature of our tasks (object detection, classification, and audio generation), we do not need to define our own metric for evaluation. Instead, we will use the following metrics to evaluate the performance of our models.

1) For the *fretboard detection model*, we will use the *Mean Average Precision (mAP)* and *Intersection over Union (IoU)* to evaluate the model's performance.
2) For our *guitar chords classifier model*, we will use the *Cross-Entropy Loss* to evaluate the model's performance, along with *accuracy*, which is more interpretable. Additionally, we have a baseline accuracy of 83.21% achieved by Y. Kristian et al. [12].
3) Finally, for our *seamless audio generation model*, we will use the Mean Squared Error (MSE) to evaluate the quality of the generated audio comparing it to the ground-truth audio found in *GuitarSet* [22].

References

[1] Sven Bambach, Stefan Lee, David J. Crandall, and Chen Yu. Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 1, 2

[2] Nicolas Boulanger-Lewandowski, Yoshua Bengio, and Pascal Vincent. Audio chord recognition with recurrent neural networks. In *ISMIR*, pages 335–340. Curitiba, 2013. 1

[3] GRC MASK custom dataset. Guitar chord bounding box dataset. https://universe.roboflow.com/grc-mask-custom-dataset/guitar-chord-bounding-box, jun 2024. visited on 2024-06-29. 2

[4] GRC MASK custom dataset. Guitar chord handshape dataset. https://universe.roboflow.com/grc-mask-custom-dataset/guitar-chord-handshape, apr 2024. visited on 2024-06-29. 2

[5] Done. Done dataset. https://universe.roboflow.com/done-ygt9y/done-npcll, apr 2024. visited on 2024-06-29. 2

[6] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1

[7] Yuexi Du, Ziyang Chen, Justin Salamon, Bryan Russell, and Andrew Owens. Conditional generation of audio from video via foley analogies. In *Conference on Computer Vision and Pattern Recognition 2023*, 2023. 1

[8] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 2

[9] Yogesh Jadhav, Ashish Patel, Rutvij Jhaveri, and Roshani Raut. Transfer learning for audio waveform to guitar chord spectrograms using the convolution neural network. *Mobile Information Systems*, 2022, 08 2022. 2

[10] joaomarcoscrs. Guitar chords dataset. https://universe.roboflow.com/joaomarcoscrs/guitar-chords-daewp, jun 2024. visited on 2024-06-29. 2

[11] Filip Korzeniowski and Gerhard Widmer. Feature learning for chord recognition: The deep chroma extractor. *arXiv preprint arXiv:1612.05065*, 2016. 1

[12] Yosi Kristian, Lukman Zaman, Michael Tenoyo, and Andreas Jodhinata. Advancing guitar chord recognition: A visual method based on deep convolutional neural networks and deep transfer learning. *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, 18(2):235–249, May 2024. 1, 2

[13] Shuyu Li and Yunsick Sung. Melodydiffusion: Chord-conditioned melody generation using a transformer-based diffusion model. *Mathematics*, 11(8), 2023. 1

[14] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 2

[15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2015. 2

[16] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 1

[17] SOEN357. Guitar dataset. https://universe.roboflow.com/soen357/guitar-ppfil, may 2024. visited on 2024-06-29. 2

[18] Adam M Stark and Mark D Plumbley. Real-time chord recognition for live performance. In *ICMC*, 2009. 1

[19] Fujishima Takuya. Realtime chord recognition of musical sound: Asystem using common lisp music. In *Proceedings of the International Computer Music Conference 1999, Beijing*, 1999. 1

[20] Chien-Yao Wang, I-Hau Yeh, and Hong-Yuan Mark Liao. Yolov9: Learning what you want to learn using programmable gradient information. *arXiv preprint arXiv:2402.13616*, 2024. 1

[21] My Work. Guitar chord dataset. https://universe.roboflow.com/my-work-3idwy/guitar-chord-tvon8, may 2024. visited on 2024-06-29. 2

[22] Q. Xi, R. Bittner, J. Pauwels, X. Ye, and J. P. Bello. Guitarset: A dataset for guitar transcription. In *19th International Society for Music Information Retrieval Conference (ISMIR)*, Paris, France, September 2018. 2