

SLASH: AUDIO CHORD ESTIMATION WITH HARMONIC FEATURES AND STRUCTURED DECODING

Mehmet Akif VARDAR

Independent Researcher

mehmetakifvrdr@gmail.com

ABSTRACT

We present SLASH, our system for the MIREX 2026 Audio Chord Estimation task. SLASH follows a structured, component-based approach to large-vocabulary chord recognition. A harmonic constant-Q transform feeds a convolutional front end and six Conformer blocks. Ten output heads predict the root, triad, fifth, sixth, seventh, ninth, eleventh, thirteenth and bass of each frame, together with an auxiliary major/minor class. A structured decoder selects base chords with a key-aware Viterbi search and adds extensions and bass notes only when their scores pass fixed thresholds. The network has 12,608,910 parameters and was trained from scratch on 728 recordings annotated in the MARL-Chords, Isophonics and public Billboard 2012 sets. On 62 song-disjoint held-out recordings, SLASH reaches 86.14% major/minor and 70.88% sevenths accuracy in a local `mir_eval` evaluation; running the official MIREX evaluation code locally on the same outputs gives 85.54% and 70.37%. Adding the Billboard data improved major/minor and sevenths accuracy on two unseen Western-pop sets. The system runs offline and needs about 3.7% of the audio duration on a laptop GPU.

1. INTRODUCTION

Audio chord estimation (ACE) maps a recording to time intervals labelled with chord symbols [1]. Large chord vocabularies are difficult: rare qualities, extensions and inversions have few training examples, and a single wrong component makes the whole label wrong. Structured methods address this by predicting the components of a chord rather than treating every label as an unrelated class [2, 3].

SLASH belongs to this family. Its main features are: (i) a harmonic constant-Q transform (HCQT) [4] as input to a Conformer [5] network; (ii) ten component heads, including separate fifth and sixth heads and an auxiliary 25-class major/minor head, trained with a class-weighted focal loss [6]; and (iii) a conservative decoder that fixes the base chord first and adds extensions and bass only above fixed score thresholds. We also report local results and the effect of adding public Billboard 2012 training data.

MIREX 2026 Audio Chord Estimation system description.

2. RELATED WORK

McFee and Bello [2] introduced structured training, in which a network predicts the root, pitch classes and bass alongside the chord label. Jiang et al. [3] decomposed each chord into root, triad, bass, seventh, ninth, eleventh and thirteenth components, decoded them with a linear conditional random field (CRF), and re-weighted rare component classes. Park et al. [7] applied a bi-directional Transformer to chord recognition, and ChordFormer [8] combined Conformer blocks with a structured chord representation and a re-weighted loss.

Like ChordFormer, SLASH uses Conformer blocks, and like Jiang et al. it predicts chord components. It differs in its harmonic input, its finer set of output heads and its focal loss. Its decoder runs a key-aware Viterbi search over base chords and then makes thresholded extension and bass decisions, instead of decoding complete chord labels with a linear-chain CRF as in [3, 8]. Its class weights follow the re-weighting form of Jiang et al. with $\gamma = 0.5$.

3. SYSTEM

3.1 Features

Audio is converted to mono and resampled to 22,050 Hz. Three CQT channels use harmonic multipliers 1, 2 and 3, each with 144 bins at 24 bins per octave. Each spans six octaves, starting at its multiplier times C1 (32.70 Hz). The hop is 2,048 samples (92.88 ms). Magnitudes are compressed with $\log(1 + |\text{CQT}|)$ and standardized per channel over the recording's time and frequency dimensions. Values are clipped to $[-12, 12]$ and stored as float16, as in training; each frame gives 432 input values.

3.2 Network and Targets

The front end uses padded 5×5 and 3×3 convolutions with 64 channels, batch normalization and SiLU (Figure 1). A projection to width 256 is followed by learned absolute positional embeddings and six Conformer blocks with eight attention heads, feed-forward width 1,024 and depthwise convolution kernel length 31. Backbone dropout is 0.15, and feed-forward residual contributions are scaled by 0.5.

Ten linear heads predict root, triad, fifth modification, sixth, seventh, ninth, eleventh, thirteenth, bass relative to the root, and major/minor. The auxiliary head has 25 classes: 12 major, 12 minor and N. The thirteenth head

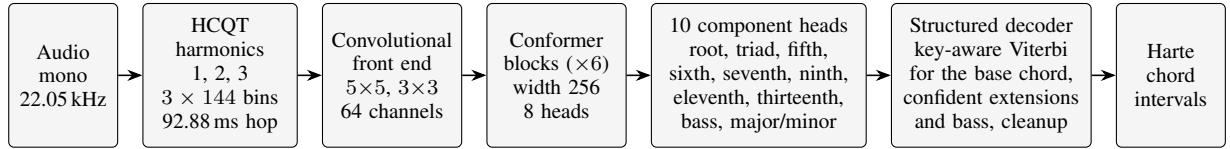


Figure 1. Overview of SLASH. The network predicts chord components frame by frame; the decoder turns them into Harte chord labels [1] with start and end times.

serves as an additional training target; the decoder suppresses its output. Annotation components that are unknown or cannot be represented are masked from their losses.

3.3 Structured Decoding

The network processes 256-frame windows advanced by 130 frames. At each interior window edge, 63 frames of context are discarded before the logits are joined.

The decoder first chooses the base chord. It scores candidate base chords from the root, triad, fifth and major/minor outputs, with a fixed penalty for base chords other than major and minor. A global key prior then favours the major or minor scale whose seven diatonic roots carry the most root probability over the whole song: candidates on those roots are boosted and the others damped. A Viterbi search over the 32 most likely labels per frame (at most 85 per song) selects the base-chord sequence with a fixed bonus for staying on a chord and a penalty for changing, and base-chord segments shorter than three frames are merged into their neighbours. Extensions and the bass note are added only when their head prefers a non-default class with a softmax score of at least 0.5 (extensions) or 0.475 (bass). Component runs shorter than three frames are removed without crossing root or triad boundaries, and a slash bass is kept only when it is a chord tone. Thirteenth extensions are never emitted. A final cleanup merges eligible symbol runs shorter than four frames into neighbouring labels but never changes root, triad, fifth, bass or the major/minor projection, so it adds no universal four-frame minimum. Consecutive identical labels are merged, and the last interval ends at the end of the audio. Beat snapping is not used.

4. DATA, TRAINING, AND SELECTION

Training and validation use the MARL-Chords and Iso-phonics annotations [9, 10] and the public part of the McGill Billboard annotations [11] (Table 1). MARL-Chords annotates songs from the RWC Popular Music database [12] and the USPop2002 collection [13]; the Iso-phonics recordings are mostly by the Beatles and Queen. Billboard recordings were limited to public identifiers up to 1300: none of the 161 protected Billboard 2013 identifiers was used, and as a precaution we also excluded recordings whose identifiers appear in one 2013 system’s outputs. All splits are song-disjoint.

Training starts from random initialization (seed 2026) and runs for 50 epochs and 187,850 optimizer steps.

Corpus	Train	Val.	Held-out	BB holdout
MARL-Chords	225	30	30	–
Iso-phonics	126	33	32	–
Billboard 2012	377	53	–	51
Total	728	116	62	51
Hours	47.11	5.66	2.92	2.71

Table 1. Recordings and audio hours per split. The held-out and Billboard (BB) holdout sets are used only for evaluation.

Batches contain two 256-frame windows, sampled uniformly without replacement within each epoch. Pitch shifts from -3 to $+3$ semitones are applied in the feature domain, with the root and major/minor targets transposed to match. AdamW uses learning rate 2×10^{-4} , weight decay 0.003 and $\beta = (0.9, 0.95)$, with a one-epoch warmup and a cosine schedule floored at 5% of the initial rate. Training uses automatic mixed precision and gradient clipping at norm 1.

The objective sums masked, class-weighted focal cross-entropy terms [6] with focal exponent 1.5 and label smoothing 0.03. Inverse square-root-frequency class weights are clamped and normalized to mean one. Major/minor, root, triad and bass loss weights are 3.0, 1.5, 1.0 and 0.5; the other component weights range from 0.03 to 0.35.

We keep the weights from epoch 41, which had the best full-song validation major/minor accuracy (82.51%) under the training-time decoder. The decoder settings were fixed before this model was trained, on development data that included the MARL-Chords and Iso-phonics validation recordings, Robbie Williams [14] and JAAH [15] songs and the 15 IdolSongsJp recordings [16]. The held-out recordings were not used to train SLASH, select its checkpoint or tune its decoder, although some fixed decoder constants were inherited from an earlier in-house system that had seen most of them.

5. RESULTS

Table 2 reports local `mir_eval` accuracy [17]: for each metric, scores are weighted by the duration that the metric can evaluate, and reference intervals outside its vocabulary are excluded. Values in parentheses give the change relative to the same recipe trained without Billboard data. We chose the submitted model after scoring both versions on these sets, so these are post-selection numbers, not official MIREX results.

Running the unmodified official MIREX evaluation code [18], which uses the MusOO evaluator [19], gives

Metric (%)	Validation (116)	IdolSongsJp (15)	Billboard holdout (51)	Held-out (62)
Root	83.90	80.22 (+0.39)	83.44 (+1.87)	86.88 (+2.00)
MajMin	83.51	79.89 (−0.22)	81.74 (+2.13)	86.14 (+2.07)
MajMinBass	80.39	77.19 (+0.80)	79.25 (+2.61)	83.07 (+2.72)
Sevenths	66.96	60.89 (−1.13)	64.77 (+3.16)	70.88 (+5.40)
SeventhsBass	64.80	58.91 (−0.74)	62.69 (+3.15)	68.72 (+5.69)

Table 2. Local accuracy of the submitted system (MIREX metric names). In parentheses: change relative to the same recipe trained without Billboard data.

85.54% major/minor and 70.37% sevenths on the held-out set, and 80.71% and 63.96% on the Billboard holdout. The Billboard data helped both Western-pop sets: 99.8% of paired song-bootstrap resamples showed a major/minor gain on the Billboard holdout, and 99.8% on the held-out set. On the J-pop set the two versions are about level. Sevenths accuracy remains about 15 points below major/minor accuracy on the held-out set.

6. SUBMISSION

The Linux package contains the weights, the decoder and separate preparation and chord-identification commands, as required by the task. Inference needs no network connection. On an RTX 3050 Laptop GPU, a 15-recording check took 3.7% of the audio duration, including per-file start-up, well within the task’s limit of five times the audio duration. Chords without a third that MusOO cannot map, such as `C:5`, are written as `C:sus4`, because MusOO otherwise aborts the whole major/minor or sevenths run; both forms count as the same mismatch, so scores are unchanged.

7. CONCLUSION

SLASH combines harmonic input features, a Conformer network with component heads and a cautious structured decoder. Adding public Billboard 2012 annotations raised accuracy on Western pop, most for sevenths, and left the J-pop set about unchanged. The held-out recordings come from the same corpora, and partly the same artists, as the training data, so they do not measure generalization to other genres or recording styles. Sevenths accuracy remains well below major/minor accuracy. With 728 training recordings, SLASH now uses about as many as the roughly 730 per fold of Jiang et al. [3] and ChordFormer [8]; adding further permitted annotated data is a natural next step.

8. REFERENCES

- [1] C. Harte, M. Sandler, S. Abdallah, and E. Gómez, “Symbolic representation of musical chords: A proposed syntax for text annotations,” in *Proc. ISMIR*, 2005, pp. 66–71.
- [2] B. McFee and J. P. Bello, “Structured training for large-vocabulary chord recognition,” in *Proc. ISMIR*, 2017, pp. 188–194.
- [3] J. Jiang, K. Chen, W. Li, and G. Xia, “Large-vocabulary chord transcription via chord structure decomposition,” in *Proc. ISMIR*, 2019, pp. 644–651.
- [4] R. M. Bittner, B. McFee, J. Salamon, P. Li, and J. P. Bello, “Deep salience representations for F_0 estimation in polyphonic music,” in *Proc. ISMIR*, 2017, pp. 63–70.
- [5] A. Gulati *et al.*, “Conformer: Convolution-augmented Transformer for speech recognition,” in *Proc. Interspeech*, 2020, pp. 5036–5040, doi: 10.21437/Interspeech.2020-3015.
- [6] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proc. ICCV*, 2017, pp. 2999–3007, doi: 10.1109/ICCV.2017.324.
- [7] J. Park, K. Choi, S. Jeon, D. Kim, and J. Park, “A bi-directional Transformer for musical chord recognition,” in *Proc. ISMIR*, 2019, pp. 620–627.
- [8] M. W. Akram, S. Dettori, V. Colla, and G. C. Buttazzo, “ChordFormer: A Conformer-based architecture for large-vocabulary audio chord recognition,” *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 34, pp. 581–595, 2026, doi: 10.1109/TASLPRO.2025.3646468.
- [9] Music and Audio Research Lab, New York University, “Chord annotations,” GitHub repository. [Online]. Available: <https://github.com/tmc323/Chord-Annotations>. Accessed Oct. 4, 2026.
- [10] Centre for Digital Music, “Reference annotations,” Isophonics. [Online]. Available: <https://isophonics.net/content/reference-annotations>. Accessed Oct. 4, 2026.
- [11] J. A. Burgoyne, J. Wild, and I. Fujinaga, “An expert ground truth set for audio chord recognition and music analysis,” in *Proc. ISMIR*, 2011, pp. 633–638.
- [12] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, “RWC Music Database: Popular, classical, and jazz music databases,” in *Proc. ISMIR*, 2002, pp. 287–288.
- [13] A. Berenzweig, B. Logan, D. P. W. Ellis, and B. Whitman, “A large-scale evaluation of acoustic and subjective music-similarity measures,” *Computer Music Journal*, vol. 28, no. 2, pp. 63–76, 2004, doi: 10.1162/014892604323112257.

- [14] B. Di Giorgi, M. Zanoni, A. Sarti, and S. Tubaro, “Automatic chord recognition based on the probabilistic modeling of diatonic modal harmony,” in *Proc. Int. Workshop on Multidimensional Systems (nDS)*, 2013.
- [15] Music Technology Group, “Jazz Audio-Aligned Harmony (JAAH) dataset documentation.” [Online]. Available: <https://mtg.github.io/JAAH/>. Accessed Oct. 4, 2026.
- [16] H. Suda, J. Koguchi, S. Yoshida, T. Nakamura, S. Fukayama, and J. Ogata, “IdolSongsJp corpus: A multi-singer song corpus in the style of Japanese idol groups,” in *Proc. ISMIR*, 2025, pp. 647–654.
- [17] C. Raffel *et al.*, “mir_eval: A transparent implementation of common MIR metrics,” in *Proc. ISMIR*, 2014, pp. 367–372.
- [18] ismir-mirex, “mirex-evaluation,” GitHub repository, commit b9fa0b0. [Online]. Available: <https://github.com/ismir-mirex/mirex-evaluation>. Accessed Oct. 4, 2026.
- [19] J. Pauwels and G. Peeters, “Evaluating automatically estimated chord sequences,” in *Proc. ICASSP*, 2013, pp. 749–753, doi: 10.1109/ICASSP.2013.6637748.