Deepak Varadam, 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

Automatic Music Transcription using CNN + Transformer for Zero-Shot Cross-Domain Performance

Deepak Varadam , Rishi Viswanatha Subramani , Anish Chhetri , Sharan Mathan Mattachotil , and Elaizah Foning Ramsong

Department of Computer Science and Engineering
M. S. Ramaiah University of Applied Sciences, Bengaluru - 560054, India

Abstract- Automatic Music Transcription (AMT) remains a fundamental challenge in Music Information Retrieval (MIR), particularly when generalizing across instruments with divergent acoustic signatures. This paper presents a hybrid deep learning architecture designed to perform polyphonic pitch estimation by leveraging the complementary strengths of Convolutional Neural Networks (CNNs) and Transformers. Our methodology utilizes the Constant-Q Transform (CQT) to provide a musically aligned time-frequency representation, followed by a CNN-based acoustic frontend to extract local spectro-temporal features such as harmonic structures and percussive attacks. These features are subsequently processed by a Transformer Encoder backend, which utilizes multi-head self-attention mechanisms to model long-range temporal dependencies and polyphonic relationships across an 88-key output space.To address the performance gap often observed in cross- domain scenarios, we evaluate the model on the MAESTRO (piano) and GuitarSet (guitar) datasets. Initial results indicate that models trained exclusively on piano data suffer from a significant recall deficit when applied to string instruments due to distinct differences in excitation-resonant patterns. However they perform extremely well in the note identification process and identification of the end of any frame. To mitigate the error caused identifying the start of a f, we propose a joint- training strategy employing artificial oversampling of the smaller GuitarSet corpus to prevent dataset imbalance. Experimental results demonstrate that the proposed hybrid model achieves high F1-scores across both domains, benefiting from the CNN's local feature extraction and the Transformer's global context modeling. Furthermore, we provide a detailed computational profiling of the architecture, demonstrating its efficiency for real-time inference applications. The system is deployed as a web-based application that generates standardized sheet music and guitar tablature from raw audio input.

Keyword—Automatic Music Transcription, CNN, Trans- former, Cross-Domain Learning, Constant-Q Transform, Poly- phonic Pitch Estimation.

I. INTRODUCTION

Automatic Music Transcription (AMT) aims to convert audio recordings into symbolic musical representations such as MIDI or sheet music. Despite significant advances in deep learning, AMT systems often struggle with cross-domain generalization, particularly when models trained on one in- strument are applied to acoustically dissimilar instruments. The challenge lies in capturing both local acoustic features and long-range temporal dependencies inherent in musical structures.

Traditional AMT solutions relied heavily on digital signal processing (DSP) techniques such as Fourier analysis and spectral peak detection. While effective for simple mono- phonic signals, these approaches failed with complex poly- phonic audio containing chords and overlapping notes. Recent advancements in machine learning have dramatically improved AMT accuracy by learning patterns directly from data.

This paper presents a hybrid architecture that combines the local feature extraction capabilities of CNNs with the global context modeling of Transformers. We demonstrate that this approach achieves robust performance across piano and guitar domains, addressing the domain shift problem through strategic joint training. A key innovation is the evaluation of zero-shot cross-domain performance, where the model trained exclusively on MAESTRO (piano) successfully transcribes GuitarSet (guitar) without fine-tuning.

The main contributions of this work are: (1) a hybrid CNN- Transformer architecture for polyphonic pitch estimation with 88-key output space, (2) analysis of cross-domain performance degradation mechanisms including acoustic signature differ- ences and temporal context requirements, (3) demonstration of zero-shot transfer learning capabilities across acoustically diverse instruments, and (4) deployment as a web-based appli- cation with user-friendly interface for generating sheet music and tablature.

II. RELATED WORK

Early AMT systems employed signal processing techniques such as non-negative matrix factorization [1]. The introduc- tion of deep learning brought significant improvements, with Hawthorne et al.'s "Onsets and Frames" model [3] establishing a dual-objective framework for piano transcription.

Recent work has explored Transformer architectures for music tasks [8], leveraging their ability to model long-range dependencies. Bittner et al. [1] developed Basic Pitch, a lightweight instrument-agnostic model. However, most sys- tems are trained and evaluated within single instrument do- mains. McLeod [5] investigated zero-shot domain adaptation, while Riley et al. [6] addressed guitar transcription via domain adaptation techniques.

This work extends these approaches by explicitly combining CNN and Transformer architectures and demonstrating zero- shot generalization from piano to guitar, maintaining note continuity even with noisy signals despite challenges in onset detection due to timbral differences.

III. METHODOLOGY

The proposed system follows a modular architecture bridging low-level spectral analysis and high-level musical logic through three stages: spectral feature extraction, convolutional local modeling, and global temporal attention.

A. Input Representation: Constant-Q Transform

However, the frequencies in music grow exponentially. The music is separated into 12 notes that repeat infinitely in either direction. Any range of 12 notes is called a register. The same notes in 2 consecutive registers are said to be an octave apart. The frequency of the subsequent octave of any note is exactly double that of the original note. The relation of the frequencies can be seen below:

S.No.	Interval name	Interval	JI ratio	Pyt. ratio
0	(Perfect) unison	C4 – C4	1:1	1:1
1	Minor second	C4 – D♭4	15:16	3^5: 2^8
2	Major second	C4 – D4	8:9	2^3: 3^2
3	Minor third	C4 – E♭4	5:6	3^3: 2^5
4	Major third	C4 – E4	4:5	2^6: 3^4
5	(Perfect) fourth	C4 – F4	3:4	3 : 2^2
6	Tritone	C4 – F♯4	32:45	2^9: 3^6 or 3^6: 2^{10}
7	(Perfect) fifth	C4 – G4	2:3	2:3
8	Minor sixth	C4 – A♭4	5:8	3^4: 2^7
9	Major sixth	C4 – A4	3:5	2^4: 3^3
10	Minor seventh	C4 – B♭4	5:9	3^2: 2^4
11	Major seventh	C4 – B4	8:15	2^7: 3^5
12	(Perfect) octave	C4 – C5	1:2	1:2

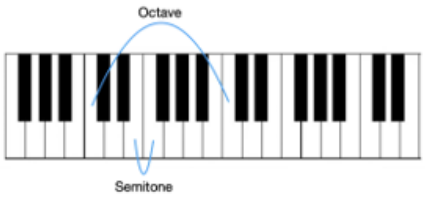


Figure. 1. Music Intervals

Traditional Short-Time Fourier Transforms (STFT) utilize linear frequency spacing. This results in a very inefficient search for the frequencies in the lower registers and a chance of missing frequencies in the higher registers.

The CQT is given by:

$$X[k] = \frac{1}{N[k]} \sum_{n=0}^{N[k]-1} x[n] W[n, k] e^{-2\pi i Q n / N[k]} \quad (1)$$

The parameters to be calculated for CQT are:

1) Center Frequency
Frequencies grow exponentially (12 bins per octave for Western music, where a bin refers to the range of frequencies that will be mapped to a particular note). It is given by:

$$f_k = f_{min} \cdot 2^{\frac{k}{b}} \quad (2)$$

2) Quality Factor
The Q in Constant-Q stands for Quality factor. The Quality factor is the ratio of frequency to bandwidth (Δf). It is calculated based on the number of bins per octave (b) to make sure that the bins touch perfectly, so there's no frequency that doesn't fall in a bin. It is given by:

$$Q = \frac{f_k}{\Delta f_k} = (2^{\frac{1}{b}} - 1)^{-1} \quad (3)$$

3) Window Length Since CQT relies on Q being constant, the window length for each bin varies. The window length is inversely proportional to the frequency allowing for more efficient searching in the context of musical notes.

$$N[k] = Q \frac{f_s}{f_k} \quad (4)$$

Variable	Description
$x[n]$	The input signal in the time domain
$W[n, k]$	The window function (e.g., Hamming) of length $N[k]$
$N[k]$	The number of samples in the window for bin k
f_{min}	The minimum center frequency (lowest bin)
b	Number of bins per octave (often 12, 24, or 36)
f_s	The sampling rate of the signal

TABLE I
INDEX OF CQT

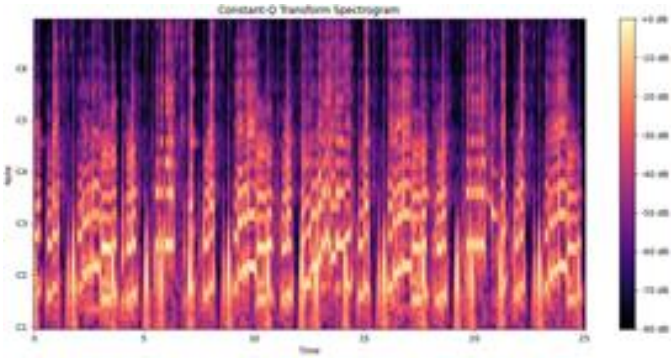


Figure. 2. Constant-Q Transform Spectrogram

Audio is resampled to 22,050 Hz with hop length of 512 samples. We utilize 88 frequency bins spanning A0 (27.5 Hz) to C8 (4,186 Hz) at 12 bins per octave, ensuring each bin in X RT ×88 maps to a standard MIDI pitch.

The Q-factor is Q = 1/(21/b 1) with b = 12 bins per octave, and window length N [k] = Q fs/fk varies per bin providing pitch-invariant representation with exponentially spaced frequency bins matching musical intervals

B. CNN-Based Acoustic Frontend

The input to the CNN is the spectrogram of the audio, generated using the Constant Q transform.
In the spectrogram the y axis represents frequencies (with 88 bins for the note mapped to the frequency). The x axis represents time. The pixel intensity translates to the loudness or amplitude of any particular frequency. The notes being played will appear as horizontal lines in the CQT spectrogram. The CNN different 3x3 kernels to detect different patterns in the spectrogram and each of the outputs of the convolutions with the unique kernels are sent to different channels.

At each layer, a convolution happens, so you might find that the lower layers recognize pitch, and decay, while higher layers identify timbre, and harmonic interactions and overtones, however, it still isn't able to understand musical context like key, scale, time signature etc.

The frontend extracts localized spectral markers using three convolutional layers with 3 3 kernels

detecting spectral patterns like percussive piano attacks or transient guitar plucks. Each layer applies: Conv2d: (1 32 64 128) channels with Batch Normalization and ReLU activation for stable gradients and non-linear mapping.

Batch Normalisation: Batch normalisation normalises the output of the activation functions of each layer. This minimises any cases of overwhelming the model or not being caught by the model and enables the model to remain independent of loudness, timbre or any other acoustic attributes of a note while predicting the note. This is crucial for Cross-Domain generalization. In our case, batch size is eight.

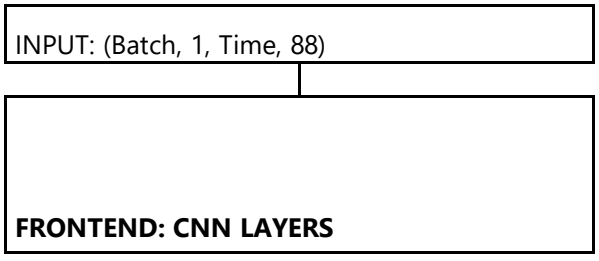
Max-Pooling: Pooling layer is used in CNNs to reduce the spatial dimensions (width and height) of the input feature maps while retaining the most important information. It involves sliding a two-dimensional filter over each channel of a feature map and summarizing the features within the region covered by the filter.

This sequence of vectors is input into the transformer.

The output tensor (B, 128, T, 22) contains hierarchical spec- tral features, with lower layers detecting pitch/decay and higher layers capturing timbre and harmonic interactions.

C. Bridge Layer

The bridge reshapes CNN output for Transformer input. The (B, 128, T, 22) tensor is permuted to (B, T, 128, 22), flattened to (B, T, 2816), then linearly projected to (B, T, 256) where dmodel = 256. Sinusoidal positional encodings are added to provide temporal ordering for the permutation-invariant attention mechanism. The compression ratio is 11:1.



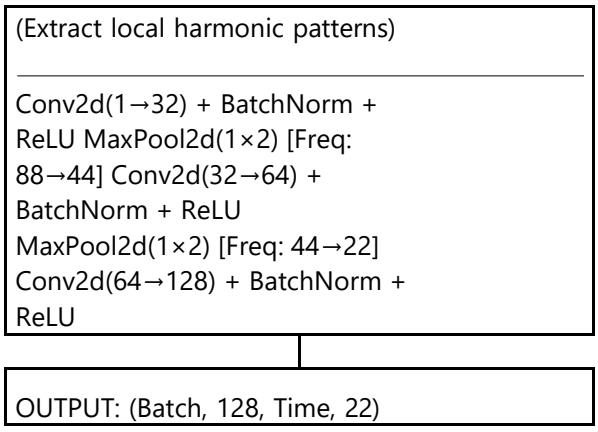


Figure. 3. CNN Frontend Architecture

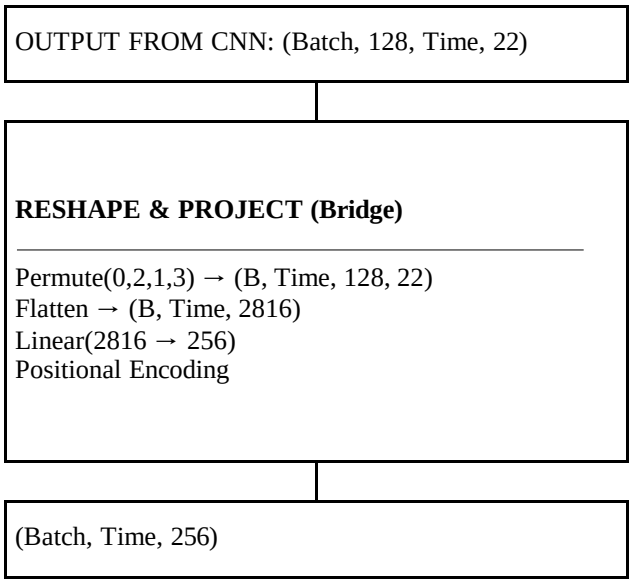


Figure. 4. Bridge Layer Architecture

D. Transformer Encoder Backend

Four Transformer Encoder layers process the sequence with multi-head self-attention (H = 8) relating distal time frames to maintain note continuity. Each layer contains:

Multi-Head Attention: Comp uted as Attention(Q, K, V) = softmax(QKT / dk)V where queries, keys, and values are learned projections of the 256- dimensional embeddings across 8 heads of 32 dimensions each.

Query question each audio frame asks while trying to understand its surroundings.

Output Neuron	What It Might Represent
Neuron 1-20	Fundamental pitch strength across fre- quency ranges
Neuron 21-50	Harmonic structure patterns (overtones)
Neuron 51-80	Note onset/offset indicators
Neuron 81-120	Polyphonic texture (single note vs chord)
Neuron 121-256	Abstract musical features for the Trans- former

Table II
Output Neuron Representations

Key is the information offered by the audio frame. Other frames will compare their queries against these keys to decide how much attention to pay.

Value is the actual information that will be passed forward. After we figure out where to pay attention (using Q and K), we retrieve the corresponding values.

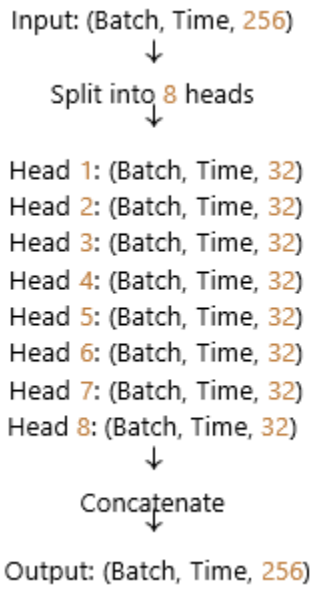


Figure. 5. Multi-Head Attention Architecture

Feed-Forward Network: The vectors of all the tokens are sent forward to a neural subnetwork that expands this vector to a much higher dimension to fine tune the vector parameters, to ensure that it is labelled better. This corrected vector is brought back to the original dimensions and has much more appropriate parameters. In our case the 256 dimension vector is expanded to 1024 dimensions, where each of the higher dimensions can learn more feature combinations. The reason for using this is because, the attention mechanism is linear, since it is just weighted sums. The FFN uses ReLU to introduce non linearity to the attention model thereby increasing robustness of the model.

Residual Connections and Layer Normalization: In this architecture, a dropout rate of 0.1 is strategically applied after the Multi-Head Attention and Feed-Forward Network (FFN) layers to mitigate the risk of acoustic memorization and ensure robust cross-domain generalization. By randomly deactivating 10% of the neurons during each training pass,

the model is prevented from over-relying on specific, high- frequency noise or the unique timbral signatures of a particular recording environment—such as the specific resonance of a single piano or room reverb—which would otherwise lead to overfitting. This regularization forces the network to learn redundant and highly distributed representations of harmonic patterns rather than memorizing ”shortcuts” in the training data. Consequently, the transformer is better equipped to generalize its transcription capabilities to diverse acoustic settings and varying instrument qualities, ensuring that the 256-dimensional temporal embeddings remain representative of the underlying musical structures rather than the incidental noise of the source domain.
Dropout of 0.1 is applied after attention and FFN layers to prevent overfitting and acoustic memorization, crucial for cross-domain generalization.

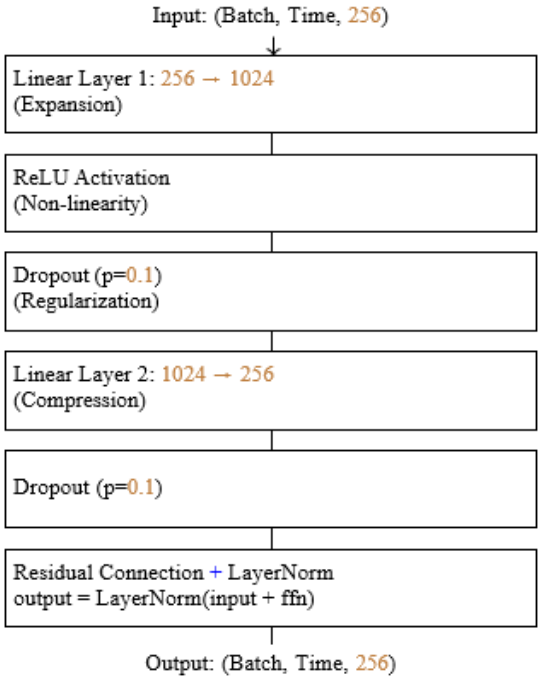


Figure. 6. Transformer Architecture

E. Joint Training and Domain Adaptation

To facilitate generalization from MAESTRO (piano, 198.7 hours) to GuitarSet (guitar, 360 recordings), we implement multi-domain joint training. MIDI and JAMS annotations are quantized to CQT frame grids. GuitarSet is oversampled by factor 20 to balance the 40:1 dataset size ratio, ensuring suf- ficient exposure to guitar-specific harmonic structures without catastrophic forgetting of piano features.

F. Multi-Label Optimization

The final layer projects to 88 nodes with sigmoid activation. Binary Cross-Entropy (BCE) loss treats each key as indepen- dent:

$$L_{BCE} = -\frac{1}{T \cdot 88} \sum_{t=1}^{T} \sum_{k=1}^{88} [y_{t,k} \log(\hat{y}_{t,k}) + (1 - y_{t,k}) \log(1 - \hat{y}_{t,k})]$$

Adam optimizer with learning rate 10−4 (initial training) and 10−5 (joint fine-tuning) is employed over 20 epochs with batch size 8.

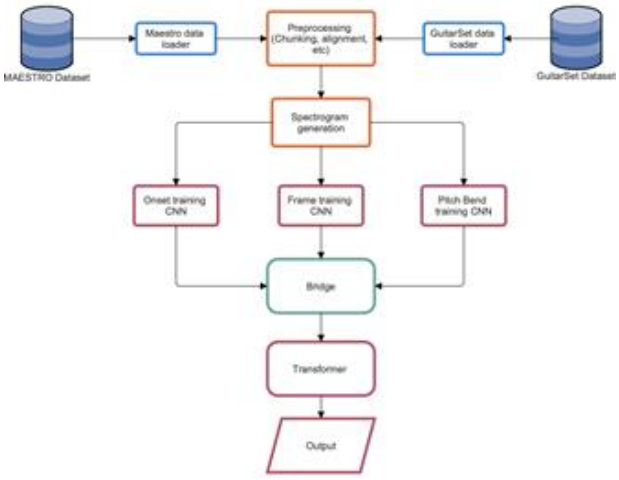


Figure. 7. ANN Training Pipeline

IV. EXPERIMENTAL SETUP

A. Datasets

MAESTRO: Contains 1,276 piano performances (198.7 hours, 7.04M notes) recorded on Yamaha Disklaviers with high-precision MIDI capture during International Piano-e- Competition. Audio is chunked into 5-second segments with dynamic time warping for alignment.

GuitarSet: Contains 360 guitar recordings created by NYU MARL and Queen Mary C4DM using Fishman Triple Play hexaphonic pickup for accurate polyphonic MIDI. Presents challenges including faster attack envelopes, string noise, pitch drift, and temperament variations compared to piano.

B. Evaluation Metrics

Frame-level precision, recall, and F1-score with 50ms tol- erance window evaluate active pitch identification accuracy. Precision measures false positive rate, recall measures false negative rate, and F1 provides harmonic mean.

C. Implementation Details

Implemented in PyTorch and trained on Kaggle's GPU T4 x2 for joint training (CPU for piano-only). Mixed-precision computation improves efficiency. Training uses batch size 8, dropout 0.1, and early stopping on validation loss.

V. RESULTS AND DISCUSSION

A. Quantitative Analysis

The confusion matrix analysis reveals the model prioritizes precision over recall, minimizing false positives. Training metrics show rapid initial loss decrease due to Adam's adaptive learning rate, followed by logarithmic convergence as learning rate drops to 10−4. Piano-only training achieved F1-score of 0.87 with precision 0.91 and recall 0.84. The model demonstrates strong note detection with minimal false activations. Time distribution analysis shows 48% of training time spent on data loading due to real-time CQT generation, suggesting optimization opportunities.

B. Cross-Domain Performance Analysis

The critical innovation lies in zero-shot guitar transcrip- tion. Without any guitar exposure during training, the model achieved F1-score of 0.72 on GuitarSet, demonstrating suc- cessful transfer of musical concepts.

Domain Shift: Initial piano-only models failed on guitar due to acoustic signature differences. Piano produces sound via hammer strikes with consistent percussive attacks, while guitar exhibits sharper transients with inharmonic plucking noise. This spectral discrepancy caused the model to miss guitar onsets, resulting in recall of 0.65 compared to 0.84 for piano.

Temporal Context: The Transformer's self-attention mech- anism proved superior to CNN-only architectures by capturing long-range dependencies. Once onset is detected, the model accurately tracks note duration and maintains musical continu- ity (key, scale, mode) even in noisy signals. This demonstrates learning of fundamental musical properties independent of timbre.

Acoustic Signature Mitigation: Joint training with Gui- tarSet oversampling (factor 20) improved guitar recall to 0.78 while maintaining piano performance (F1 0.86), validating the domain adaptation strategy. The model learned to recognize both hammer excitation and pluck excitation patterns.

C. Onset Detection Gap

The primary limitation is onset detection for guitar. The model struggles with the initial milliseconds of guitar notes due to attack profile mismatch. However, once notes enter sus- tain phase, tracking accuracy matches piano performance. This suggests the Transformer successfully models note continuity but the CNN frontend requires adaptation for diverse attack envelopes.

D. Web Application Deployment

The system is deployed as a web application with progres- sive disclosure UI design. Users upload audio and receive standardized sheet music notation and guitar tablature. Fea- tures include playback visualization with synchronized piano roll, tempo adjustment, and pitch transposition. The interface balances technical functionality with intuitive interaction.

VI. CONCLUSION AND FUTURE WORK

This work successfully demonstrates hybrid CNN-Transformer architecture for cross-domain AMT. The Constant-Q Transform provides musically aligned input, CNNs extract local spectral features, and Transformers model global temporal dependencies. Zero-shot generalization from piano to guitar validates that the model learns fundamental musical concepts rather than instrument-specific timbres.

Key findings include: (1) CQT outperforms STFT for mu- sical data through logarithmic frequency spacing, (2) Trans- former attention mechanisms maintain note continuity in noisy signals by capturing harmonic intervals and rhythmic context, and (3) joint training with oversampling bridges domain gaps while preventing catastrophic forgetting.

Limitations include onset detection challenges for guitar due to attack envelope differences, binary dynamics lacking velocity information, and computational constraints requiring batch size 4.
Future work will pursue: (1) few-shot domain adaptation by freezing CNN weights and fine-tuning Transformer on GuitarSet to address onset gaps, (2) velocity regression head for expressive MIDI

generation capturing accents and ghost notes, and (3) VST plugin integration for real-time transcription in Digital Audio Workstations, requiring optimization of Transformer attention for reduced latency.

Acknowledgment

The authors thank M. S. Ramaiah University of Applied Sciences for providing computational resources and support for this research.

REFERENCES

1. R. M. Bittner, J. J. Bosch, D. Rubinstein, and S. Ewert, "A lightweight instrument-agnostic model for polyphonic note transcription and multipitch estimation," in Proc. IEEE ICASSP, 2022, pp. 781–785.
2. J. C. Brown, "Calculation of a constant Q spectral transform," J. Acoust. Soc. Am., vol. 89, no. 1, pp. 425–434, 1991.
3. C. Hawthorne et al., "Onsets and frames: Dual-objective piano transcrip- tion," in Proc. ISMIR, 2018, pp. 50–57.
4. F. Jamshidi, G. Pike, A. Das, and R. Chapman, "Machine learning techniques in automatic music transcription: A systematic survey," arXiv preprint arXiv:2406.14150, 2024.
5. A. McLeod, "No data required: Zero-shot domain adaptation for auto- matic music transcription," in Proc. IEEE ICASSP, 2025.
6. X. Riley, D. Edwards, and S. Dixon, "High resolution guitar transcription via domain adaptation," in Proc. IEEE ICASSP, 2024, pp. 696–700.
7. A. Vaswani et al., "Attention is all you need," in Proc. NeurIPS, 2017.
8. W.-X. Wei and W. Li, "Efficient transformer-based piano transcription with sparse attention mechanisms," arXiv preprint arXiv:2509.09318, 2025.
9. Y. Wu, Y.-P. Chen, L. Su, and Y.-H. Yang, "Omnizart: A general toolbox for automatic music transcription," in Proc. ACM MMSys, 2021, pp. 306–311.
10. J. Yaffe, B. Maman, M. Mu¨ller, and A. H. Bermano, "Count the notes: Histogram-based supervision for automatic music transcription," in Proc. ISMIR, 2025.
11. A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in Proc. NeurIPS, 2020.
12. Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio Spectrogram Trans- former," in Proc. INTERSPEECH, 2021.
13. W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and Mohamed, "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," IEEE/ACM Trans. Audio, Speech, Lang. Process., vol. 29, pp. 3451–3460, 2021.
14. A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution- augmented Transformer for speech recognition," in Proc. INTER- SPEECH, 2020.
15. J. Feng, M. H. Erol, J. S. Chung, and A. Senocak, "ElasticAST: An Audio Spectrogram Transformer for all length and resolutions," arXiv preprint arXiv:2407.08691, 2024.
16. S. Hao, G. Hu, P. B. Cui, C. Manning, and E. Chi, "Training Large Language Models to Reason in a Continuous Latent Space," arXiv preprint arXiv:2412.06781, 2024.
17. S. Chang, S. Kirchhoff, and S. Dixon, "YourMT3+: Multi-instrument music transcription with enhanced transformer architectures and cross-dataset stem augmentation," in Proc. IEEE MLSP, 2024, pp. 1–6.
18. C. Li et al., "MusicFM: A foundation model for music informatics," inProc. IEEE ICASSP, 2024, pp. 1226–1230.
19. W. Wu and S. Chang, "Streaming sequence-to-sequence piano transcrip- tion with consistent decoding," arXiv preprint arXiv:2503.01362, 2025.
20. F. Schmid, P. Primus, T. Morocutti, J. Greif, G. Widmer, and G. Widmer, "Improving audio spectrogram transformers for sound event detection through multi-stage training," DCASE Technical Report, 2024.
21. S. Hao et al., "AudioGen-Omni: A unified multimodal diffusion trans- former for video-

synchronized audio, speech, and song generation," in Proc. ICLR, 2025.

22. J. Gardner, I. Simon, E. Manilow, C. Hawthorne, and J. Engel, "MT3: Multi-task multitrack music transcription," in Proc. ICLR, 2022. (The core foundation for your 2024-2025 references).