# DRUMS: Drummer Reconstruction Using Midi Sequences

Theodoros Kyriakou
University Of Cyprus
Nicosia, Cyprus
CYENS - Centre of Excellence
Nicosia, Cyprus
t.kyriakou@cyens.org.cy

Panayiotis Charalambous
CYENS - Centre of Excellence
Nicosia, Cyprus
p.charalambous@cyens.org.cy

Andreas Aristidou
University Of Cyprus
Nicosia, Cyprus
CYENS - Centre of Excellence
Nicosia, Cyprus
a.aristidou@ieee.org

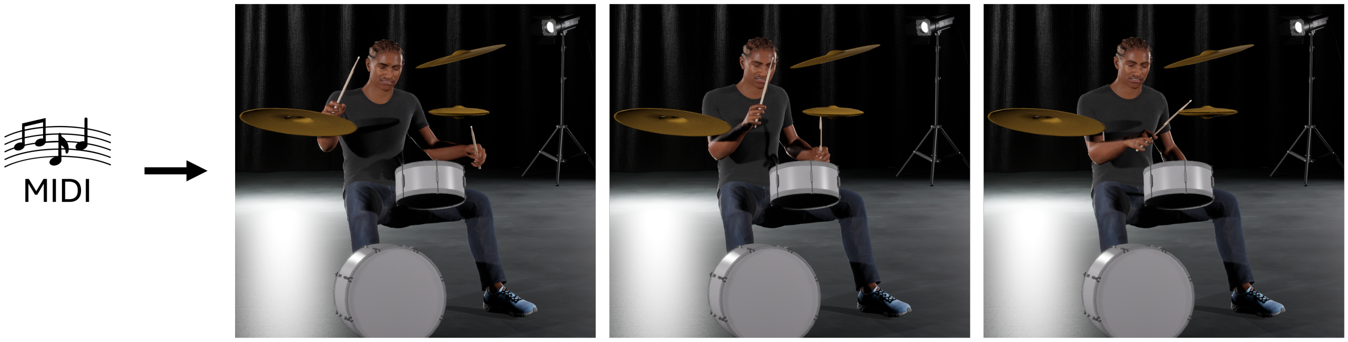

**Figure 1: Given as input a MIDI of drums (music signal), our method predicts full-body expressive animation.**

## Abstract

We present a system for generating expressive, full-body drumming performances from MIDI input, combining rhythmic precision with lifelike motion. Unlike prior work that focuses on limited gestures or audio-driven models, our approach produces coordinated animations of the entire performer, including hands, torso, legs, and facial expressions, driven solely by symbolic MIDI. Our system integrates a Bi-directional LSTM to predict fine-grained 3D hand trajectories, using sticks parented to the hands and synchronized with MIDI events. It also includes a retrieval-based module that generates expressive upper-body and facial motion conditioned on musical phrasing, and a pedal enforcement component that procedurally animates the feet. Our method addresses the unique challenges of drumming, where rhythm is both heard and seen in dynamic, physically grounded motion. To the best of our knowledge, this is the first system to generate full-body drum performances from raw MIDI. Our approach enables new applications in virtual concerts, immersive training, game animation, and digital avatar performance.

## CCS Concepts

• **Computing methodologies** → **Animation**; **Machine learning**; • **Applied computing** → **Performing arts**; *Sound and music computing*.

*MIG '25, Zurich, Switzerland*

ACM ISBN 979-8-4007-2236-3/25/12
https://doi.org/10.1145/3769047.3769066

## Keywords

Animation, Drums Performance, MIDI-to-Motion, Motion Capture

**ACM Reference Format:**
Theodoros Kyriakou, Panayiotis Charalambous, and Andreas Aristidou. 2025. DRUMS: Drummer Reconstruction Using Midi Sequences. In *The 18th ACM SIGGRAPH Conference on Motion, Interaction, and Games (MIG '25), December 03–05, 2025, Zurich, Switzerland.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3769047.3769066

## 1 Introduction

Expressive musical performance is a deeply multimodal experience, where subtle body movements, hand gestures, and facial expressions come together to bring music to life. In recent years, the creation of realistic virtual musicians has gained attention, supported by advances in motion capture, generative models, and controllable character animation. Virtual performers are no longer science fiction. They have headlined major digital events, such as Metallica's in-game world tour on Fortnite [Fortnite 2024]. Whether on a virtual concert stage or guiding a student in VR, audiences now expect the same expressive nuance they see in live musicians. This requires full-body movement and facial expression that convey groove, effort, and emotion.

Recent work has begun to holistically motion capture a variety of musical instruments [Huang et al. 2024; Kyriakou et al. 2025], resulting in open datasets that enable new generative approaches for synchronizing body animation to audio or symbolic representations such as MIDI. However, generating physically and expressively plausible full-body performances from MIDI remains an open challenge. This is particularly true for drumming: when a drummer plays, rhythm is not only heard. It is seen in every swing of the arms, flex of the fingers, and nod of the head. Capturing this coordination between rhythm, motion, and expression is technical challenging.

While MIDI is a powerful and compact representation for analyzing and controlling musical structure, it lacks spatial and expressive information. Generating believable motion from such abstract input requires models that understand both the precision of musical timing and the subtlety of embodied performance. Prior work has explored gesture generation from audio or symbolic inputs, but these systems often focus on limited body parts (e.g., fingers or bow) [Bogaers et al. 2021; Shlizerman et al. 2018; Zakka et al. 2023] or treat gesture as a secondary byproduct [Guo et al. 2021; Li et al. 2018; Xu et al. 2022]. These approaches also typically rely on audio alone, which lacks precise hit-level timing and instrument-specific intent. More recent methods incorporate symbolic inputs such as MIDI to align hand motion with note onsets [Guo et al. 2021; Li et al. 2018], but still omit full-body coordination and facial expressivity.

In this work, we introduce a MIDI-driven system for generating full-body drumming performances, where the output is both rhythmically precise and visually expressive. The generated motion includes detailed stick strikes, hand dynamics, upper-body movement, and facial animation (see Figure 1). To the best of our knowledge, despite drums being a central rhythm-setting instrument in most ensembles, our work is the first system to generate expressive, full-body animations of a seated drummer from raw MIDI input. It addresses this challenge in three stages: (1) **A Bidirectional LSTM** (BiLSTM) [Graves and Schmidhuber 2005] directly predicts detailed 3D trajectories and orientations for both hands from MIDI input, effectively modeling the complex temporal dependencies involved in multi-limb coordination while ensuring accurate hit timing (Section 3.3). (2) A **MIDI-matching module** retrieves realistic upper-body and facial expressions based on the current musical phrase, capturing head nods, body lean, and facial tension aligned with expressive drumming (Section 3.4). (3) A **procedural foot module** enforces pedal articulation aligned with the MIDI (Section 3.5). These components are then integrated by a modular inverse kinematics (IK) layer, producing a coherent, full-body performance.

Unlike prior work that narrowly targets gesture synthesis or treats the performer as a passive character, our method models the drummer as a holistic, expressive figure, capable of believable solo or ensemble performance from symbolic input alone. This unlocks applications in digital performance, immersive training, avatar animation, and beyond. Results demonstrate that our approach successfully synthesizes expressive, full-body character animation that is both visually convincing and rhythmically aligned with the musical performance.

## 2 Related Work

Audio-driven motion generation has been widely studied, especially for dance. Generative models such as LSTMs and transformers have been used to map audio input to dance motions aligned with the rhythm and style of music [Aristidou et al. 2023; Chen et al. 2021b]. More recently, diffusion models have emerged as the dominant paradigm for audio-driven motion generation [Alexanderson et al. 2023; Tseng et al. 2023; Wang et al. 2025]. However, diffusion models typically require large-scale training datasets, which are not available in our case. Moreover, generating motion for musical instrument performance introduces new challenges: unlike dance, where rhythmic alignment and stylistic movement are often sufficient, instrument performance demands high precision in contact points, especially finger placement on keys, strings, or surfaces.

Most dance-related research focuses on full-body motion. Although hands and face are important for expressivity, they are often underrepresented due to limited data. In contrast, instrument playing places greater emphasis on fine finger control and facial cues, while full-body motion tends to be more constrained but still meaningful. For example, a drummer's performance involves rhythm not only in sound but also in visible arm swings, finger movements, and head nods. Generating such coordinated multimodal motion, across body, face, and hands, poses significant technical challenges. A comprehensive survey of virtual instrument performance techniques is presented in Kyriakou et al. [2024].

A major challenge in this domain is the limited availability of large, high-quality datasets. Prior to the deep learning era, early methods relied on rule-based systems or optimization techniques to model fingering, particularly for keyboard instruments [Kugimoto et al. 2009; Sekiguchi and Eiho 2000; Yamamoto et al. 2010; Zhu et al. 2013]. These approaches were typically instrument-specific, focusing on placing fingers on the correct notes but neglecting full-body animation and expressivity, often limiting motion to the hands or upper torso. Similarly, ElKoura and Singh [2003] used k-nearest neighbors to model guitar finger transitions, while Kim et al. [2000] employed a neural network to optimize fingering for violin performance.

The rise of deep learning brought data-driven models like LSTMs, which are well-suited for temporal motion sequences. LSTMs can align with rhythmic cues and retain long-term context, making them effective for modeling expressive musical motion. They are also compatible with multimodal input, such as audio features from Convolutional Neural Networks (CNNs) or symbolic embeddings.

In piano animation, the focus has been on mapping audio or MIDI input to finger motion while maintaining correct rhythm and note contacts. Li et al. [Li et al. 2018] used CNNs and LSTMs to generate upper-body motion from MIDI. Shlizerman et al. [2018] showed that partial body motion can be predicted from audio using LSTMs, and Bogaers et al. [2021] generated expressive upper-body motion using musical audio features. Guo et al. [2021] developed a MIDI-based system for animating piano fingering using Hidden Markov Models, optimized for augmented reality training.

For violin, similar techniques have been used but with added complexity from bowing. Lin et al. [2020] proposed a real-time system using DTW-aligned audio and LSTM-based pose prediction. Kao and Su [2020] extended the LSTM-based method proposed in [Shlizerman et al. 2018] with attention and beat tracking for better bow control. Liu et al. [2020] used CNNs for bowing (since it successfully identifies bowing direction changes) and Convolutional Recurrent Neural Networks (CRNN) for expression modeling. Hirata et al. [2022] improved realism by learning bowing and full-body dynamics from audio. More recently, Nishizawa et al. [2025] used a CRNN for fingering/bowing and a two-layer BiLSTM for full-body motion generation.

Other instruments are less studied. Shirai and Sako [2021] modeled double bass performance using LSTMs. However, again, these methods focus only on local gestures and ignore expressive cues such as facial motion and torso involvement.

More recent work explores generative models: for instance, Chen et al. [2021a] used an adversarial networks (GAN) to synthesize upper-body motion for Guzheng performance, while Shrestha et al. [2022] applied Transformers for violin gesture generation. MOSA, by Huang et al. [2024], also used a Transformer to animate violin and piano from audio, with separate modules for hands and torso, refined via IK. More recently, Qiu et al. [2025] used a diffusion-based framework for cello performance full-body motion generation. However, these methods typically omit facial animation and symbolic control.

Reinforcement learning (RL) offers an alternative by optimizing actions through trial and reward. It allows real-time adaptation and exploration beyond fixed datasets. Xu et al. [2022] trained piano-playing agents using touch-based reinforcement learning, while Zakka et al. [2023] extended the approach to robotic hands. Wang et al. [2024] further refined motion-generated data through inverse kinematics, using high-resolution MIDI key-press data. More recently, Xu and Wang [2024] applied physics-based rules to enable cooperative control for guitar playing. While these methods generate skillful hand motion, they usually they remain restricted to local gestures without addressing coordinated full-body motion or expressive behaviors such as style and facial movement.

To our knowledge, no prior work has addressed animation synthesis specifically for drum performance. This is a notable gap given the central role of drumming in music, where it provides the rhythmic foundation, drives the groove, and shapes the expressive dynamics of a piece. Drum performance demands complex multi-limb coordination, dynamic torso articulation, and precise temporal alignment. Challenges include accurate synchronization with percussive hits, realistic contact point fidelity across varied drum setups, and hand alternation timing. MIDI input for drums also lacks hand-specific hit information, making generation harder. These challenges make drums particularly demanding for motion synthesis and are unaddressed in current literature.

## 3 Method Overview

This section describes our method, which consists of three primary components: a *BiLSTM* network for hand motion prediction, a *MIDI-matching* module for synthesizing head, spine, and facial expressions, and a *pedal enforcement* component that procedurally animate the feet. The outputs of these components are integrated using IK to generate expressive full-body animations.

### 3.1 Dataset

A critical component of our approach is the use of high-quality, multimodal performance data. For this purpose, we utilize the Multi-Modal Instrument Performances (MMIP) dataset [Kyriakou et al. 2025], which captures professionally recorded musical performances with synchronized multimodal data. MMIP includes full-body 3D motion capture (including hand and facial animations), MIDI, high-fidelity audio, and multi-angle video recordings. For this work, we focus on the drum performances, comprising approximately 35 minutes of data (≈126K frames). Motion is provided in FBX format at 60 FPS and has been converted to a structured JSON representation. Each frame includes the 3D joint positions and quaternion rotations for the full skeleton, including finger joints and facial animation. The dataset also includes corresponding MIDI data, which has been converted from .mid to .json format and quantized at 60 FPS. MIDI encodes symbolic performance data, including note onsets, pitch, velocity (intensity), and timing.

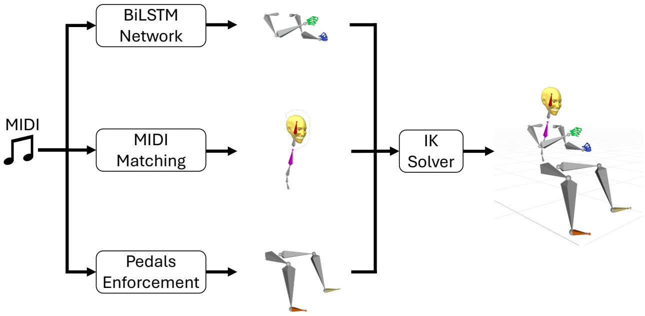

**Figure 2: Our method generates full-body animation from a MIDI sequence. A BiLSTM network predicts hand motion (green/blue), while a MIDI-matching module infers spine (pink), head (red), and facial expressions (yellow). The feet are procedurally animated using the Pedal Enforcement module (orange/dark yellow). These outputs are merged and refined using IK to produce expressive full-body animation.**

### 3.2 Problem Formulation

Our goal is to generate expressive, full-body character animation of an avatar playing drums, using only MIDI input. Drum performance has unique characteristics, including limited lower-body articulation, especially in the feet, while the player remains seated with minimal translation or maneuvering. Yet, the generated animation must be precisely synchronized with the MIDI, ensuring each note triggers the correct motion at the right frame. Beyond temporal accuracy, spatial precision is also critical, with hands striking the correct drum surfaces.

A key difficulty is that MIDI does not indicate which hand plays each note. This ambiguity complicates motion generation, especially during fast or alternating hand patterns. Expressive timing styles like swing, which follow a long-short rhythmic feel rather than strict quantization, add further complexity and require flexible interpretation. Finally, realistic animation requires more than technical accuracy. Stylistic and emotional qualities, expressed through posture, head motion, and facial cues, are crucial for realism in drums playing.

To address these challenges, our system breaks down the task into modular components, each specialized in generating motion for different parts of the body based on the musical input. Figure 2 provides an overview of the proposed system.

*Input Representation:* We use a fixed vocabulary of $N = 16$ drum note numbers:

```
NOTE_NUMS = [36, 37, 38, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 55, 58, 59]
```

Each MIDI file is preprocessed into a binary piano-roll matrix $X \in \{0,1\}^{T \times N}$, where $X_{t,n} = 1$ indicates that note $n$ is active at frame $t$, and $T$ is the total number of time frames in the sequence.

*Output Representation:* The system consists of three components that independently generate motion for the hands, upper body (head, spine, face), and feet.

- *Hand Motion (BiLSTM Module):* BiLSTM predicts the motion of both hands. At each frame $t$, the output is a vector $a_t \in \mathbb{R}^{14}$, representing the 3D position ($\mathbb{R}^3$) and orientation as a quaternion ($\mathbb{R}^4$) for both the left and right hands.
- *Upper Body and Facial Motion (MIDI-Matching Module):* This retrieval-based module selects motion-captured templates based on MIDI note patterns and phrasing. The output at frame $t$ is a vector $h_t \in \mathbb{R}^{66}$, which consists of two subsets $K_t$ and $f_t$, where:
  - $K_t \in \mathbb{R}^{14}$ encodes 3D position ($\mathbb{R}^3$) and quaternion rotation ($\mathbb{R}^4$) for both the spine and head bones.
  - $f_t \in \mathbb{R}^{52}$ is the facial expression vector, representing ARKit blendshape weights at time $t$.
- *Foot Motion (Pedal Enforcement Module):* This component procedurally generates foot motion using pedal-related MIDI events. Each frame produces a vector $r_t \in \mathbb{R}^{14}$, containing the 3D position and quaternion rotation for both feet.

These outputs are then integrated using an *IK solver* to generate a coherent full-body pose for a 57-bone skeleton. The solver combines hand motion ($a_t$), upper-body and facial motion ($h_t$), and foot motion ($r_t$) to produce expressive, temporally aligned animation that dynamically responds to the MIDI input.

## 3.3 BiLSTM Network

In this work, we use a BiLSTM network to generate hand motion from MIDI sequences, as it captures both past and future contexts, enabling smoother and more coherent motion by effectively learning temporal dependencies. It also better resolves hand ambiguities caused by the lack of labeling information on the MIDI input through full-sequence awareness.

*Data Handling.* Each training sample consists of an input–output pair $(X, A)$ (as introduced in §3.2). To regularize training and expose the model to varied rhythmic contexts, we apply random temporal cropping as a form of data augmentation. For $T > W$, where $W = 200$ frames, a random contiguous subsequence of length $W$ is sampled:

$$X' = X_{s:s+W}, \quad A' = A_{s:s+W}, \quad \text{where } s \sim \text{Uniform}(0, T - W)$$

If $T \leq W$, the full sequence is used, padded with zeros to match the window size for batching.

*Model Architecture.* We use a two-layer BiLSTM network followed by a fully connected layer:

$$Z = \text{BiLSTM}(X') \in \mathbb{R}^{T \times 2H}$$
$$\hat{A}' = \text{Linear}(Z) \in \mathbb{R}^{T \times 14}$$

where $H$ is the hidden dimension (default $H = 128$), and $Z$ contains the intermediate hidden representations. The output $\hat{A}' \in R^{14}$ is the predicted hand motion sequence. To reduce overfitting, we apply dropout with probability 0.1 between LSTM layers during training.

*Loss Function.* To train the network, we minimize a masked Mean Squared Error (MSE) loss between the predicted and ground-truth hand motion sequences.

Since input sequences may vary in length, we apply a binary mask to ensure that only valid (unpadded) time steps contribute to the loss. Let $B$ be the batch size (default 32), and let $T_b$ be the actual sequence length of the $b$-th sample in the batch. For each frame $t \in [1, T_b]$, the model predicts $\hat{a}_t^{(b)} \in \mathbb{R}^{14}$, and compares it to ground-truth $a_t^{(b)} \in \mathbb{R}^{14}$. This vector includes the 3D position and quaternion orientation for both the left and right hands. The masked MSE loss is defined as:

$$\mathcal{L} = \frac{1}{\sum_{b=1}^{B} T_b} \sum_{b=1}^{B} \sum_{t=1}^{T_b} \left\| \hat{a}_t^{(b)} - a_t^{(b)} \right\|^2$$

To enable effective training on variable-length sequences, we define a binary mask $M_{b,t}$ to exclude padding frames from the loss computation. Specifically, $M_{b,t} = \mathbb{I}[t \leq T_b]$ where $\mathbb{I}[\cdot]$ is the indicator function, which returns 1 if the condition inside is true, and 0 otherwise. In this case, $M_{b,t} = 1$ if frame $t$ is part of the valid sequence for sample $b$, and $M_{b,t} = 0$ if it falls beyond the actual sequence length (i.e., in the padded region).

*Training.* We split the dataset into training and validation subsets using a fixed 90/10 ratio. For our evaluation (Section 4.1), we report results on a separate test set containing unseen performances that were not included in either training or validation. The model is trained using the AdamW optimizer with a learning rate of $\eta = 10^{-3}$, and we apply gradient clipping with a maximum norm of 1.0 to stabilize training. Training runs for 500 epochs. After each epoch, the model is evaluated on the validation set using the same masked MSE loss. We track and save the parameters $\theta^*$ that yield the lowest validation loss across all epochs: $\theta^* = \arg\min_\theta \mathcal{L}_{\text{val}}(\theta)$. This helps prevent overfitting by selecting the version of the model that generalizes best to unseen data.

## 3.4 MIDI Matching

To generate realistic motion for the spine, head, and facial expressions, we adopt a retrieval-based MIDI matching approach. Given the limited size of our dataset and the high dimensionality of the motion data, particularly the facial blendshapes, early experiments with generative models revealed difficulties in learning reliable correlations between MIDI input and the corresponding animation. Thus, instead of synthesizing these components directly, we retrieve motion fragments from previously recorded performances that closely match the input MIDI sequence. This strategy ensures both physical plausibility and expressive consistency, especially for subtle upper-body gestures and facial movements, which are difficult to model with small datasets.

Each input MIDI sequence is divided into non-overlapping windows of fixed length, and each frame is encoded as a binary activation vector over a fixed vocabulary of $N = 16$ drum note numbers (as defined in §3.2). Let $Q \in \{0,1\}^{L \times N}$ represent a query window of length $L = 240$. Each candidate window $C \in \{0,1\}^{L \times N}$ from the database is compared against $Q$ using a vectorized cost function:

$$\text{cost}(Q, C) = w_{\text{miss}} \cdot M + w_{\text{sil}} \cdot S + w_{\text{extra}} \cdot E$$

where: $M$ is the number of missing note activations ($Q_{t,n} = 1$ and $C_{t,n} = 0$), $S$ is the number of silent frames in $Q$ that are non-silent in $C$, $E$ is the number of extra notes ($C_{t,n} = 1$ and $Q_{t,n} = 0$), and $w_{\text{miss}}, w_{\text{sil}}, w_{\text{extra}}$ are manually tuned penalty weights.

To ensure robust retrieval, we adopt a progressive matching strategy. The system begins with strict matching criteria, applying high penalties for missing ($w_{\text{miss}}$) and silent ($w_{\text{sil}}$) note violations. If no candidate falls below a predefined cost threshold, the system progressively relaxes the constraints by reducing $w_{\text{miss}}$ and $w_{\text{sil}}$, allowing for softer matches. If all candidates still fail to meet the threshold, the matching process is repeated using smaller window sizes, which facilitate more localized alignment and increase the chances of finding a viable match. Naturally, the larger the dataset, the higher the likelihood of retrieving an exact or near-exact match.

Once a suitable match is found, the corresponding motion segment $h_t$, comprising spine, head, and facial motion, from the selected candidate window is transferred to the output sequence. Frames are blended on the window edges to ensure temporal coherency. This window-by-window retrieval process continues over the entire input MIDI sequence, producing an animation that preserves both the rhythmic structure and expressive intent of the original performance. For all experiments, candidate windows are retrieved only from the training subset, ensuring that no motion fragments from validation or test sets are used during generation.

Note that, we also experimented with applying MIDI-based retrieval to hand animation. While visually plausible results can be achieved, precise synchronization between the retrieved hand motion and the MIDI input remains a significant challenge. Slight timing mismatches in note alignment can lead to unsynchronized hand movements. Reducing the size of the retrieval window improves alignment and yields more rhythmically accurate motion, but at the cost of expressiveness and motion fluidity. For hand animation, accurate note execution is critical, while for upper-body motion, rhythmic similarity and expressive quality are prioritized over exact timing.

## 3.5 Pedals Enforcement

Lower-body motion is procedurally generated based on pedal-related MIDI activity, rather than learned from data, due to high noise and frequent foot sliding observed in the original mocap data. Rather than relying on captured leg trajectories, we extract pedal events from the MIDI input to drive foot IK targets.

For the right foot, each kick drum activation (MIDI note 36) triggers a forward stepping gesture applied to the right foot IK controller. This motion consists of a brief lift and backward tilt initiated 2–5 frames before the drum hit, timed to accommodate rapid, continuous strikes, followed by a return to the rest pose at the moment of impact.

For the left foot, open hi-hat events (note 46) initiate a lift-and-hold gesture, which remains elevated until either a closed hi-hat event (note 42) occurs or a maximum hold duration of approximately 20 frames is reached. This mechanism simulates common foot-controlled hi-hat techniques, enabling musically synchronized and expressive leg motion.

Overall, this approach produces clean, responsive leg gestures that remain tightly synchronized with the MIDI percussion input, while avoiding the instability present in raw mocap leg data.

## 3.6 IK Solver

Final full-body animation is synthesized using an IK solver that integrates motion signals from three components: the BiLSTM network (hands), the MIDI Matching module (spine, head, face), and the Pedals Enforcement module (legs). These signals drive six IK targets mapped to a 57-bone character rig.

More specifically, the predicted hand trajectories $a_t$ are used to control the left and right arm IK chains, enabling accurate and expressive upper-limb motion. Spine and head motion $K_t$ are applied to torso and head IK targets to capture posture and upper-body dynamics, while leg motion $r_t$ is enforced via pedal-driven IK. Facial animation, integrated through retrieved blendshape weights, completes the full-body output. By combining learned, retrieved, and procedural signals, the system generates coherent, physically plausible character animation aligned with the musical performance.

# 4 Results

This section presents our experimental results, implementation details, and evaluation of the generated motion in terms of visual quality, synchronization with the music, and temporal smoothness.

*Implementation Details.* Our model and training pipeline are implemented in PyTorch. Data preprocessing, sequence collation, and padding are handled via a custom Python script. Procedural lower-body (foot) motion is implemented in Blender through scripts that read MIDI-aligned JSON files and keyframe the left and right foot IK targets accordingly. For evaluation, we train our model on 90% of the dataset and test on the remaining 10%, ensuring that the test MIDI sequences correspond to unseen songs.

*Results.* Figure 3 illustrates the predicted animation for an unseen MIDI input, showing smooth, natural full-body motion that is well synchronized with the musical structure. The virtual drummer exhibits realistic coordination between limbs, including hand and foot interactions with specific drum elements. Figure 1 shows a rendered frame of the virtual drummer holding drumsticks and performing. Additional qualitative results are provided in the accompanying video, demonstrating the expressiveness and fidelity of the generated animations.

## 4.1 Evaluation

We evaluate our method on two dimensions: *Motion Realism*, measuring the fidelity of generated hand trajectories against ground-truth motion capture, and *Beat Accuracy*, assessing temporal alignment between hand motion and the musical score.

*4.1.1 Motion Realism Evaluation.* To quantitatively assess motion realism, we compare our model's predictions against ground-truth 3D hand trajectories using the following metrics:

- **L1 Distance:** Mean absolute error between predicted and ground-truth hand positions over all frames.
- **Dynamic Time Warping (DTW):** Measures the minimal cumulative distance between time series, accommodating local time shifts and tempo variations.

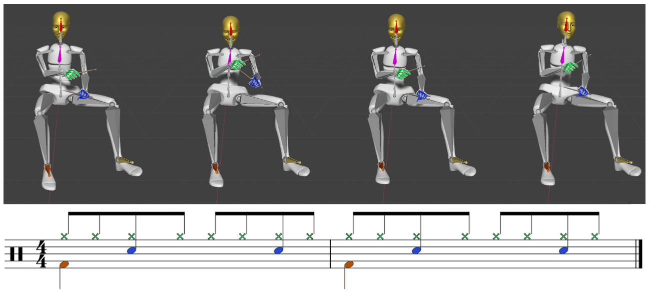

**Figure 3: Predicted motion from unseen MIDI input. Top: animation frames of the virtual drummer. Bottom: corresponding MIDI-derived score. R hand (green) plays hi-hat, L hand (blue) plays snare, and R foot (orange) plays bass drum.**

- **Jerk (RMS):** Root mean square of the third derivative of position, representing motion smoothness. Lower values indicate more natural, fluid movement.

We evaluate each hand independently using a custom script that computes frame-by-frame error. Results are presented in Table 1. Our model achieves low spatial error, strong temporal alignment, and smooth dynamics, indicating that the generated motion closely resembles real human performance with a high degree of realism.

**Table 1: Quantitative comparison of predicted hand and upper body motion with ground-truth.**

| | L1↓ | DTW↓ | Jerk (RMS)↓ |
|---|---|---|---|
| Left Hand | 0.0499 | 0.0465 | 0.0237 |
| Right Hand | 0.0550 | 0.0492 | 0.0228 |
| Average | 0.0524 | 0.0479 | 0.0233 |
| Head | 0.0442 | 0.0445 | 0.0138 |
| Spine | 0.0497 | 0.0516 | 0.0126 |
| Average | 0.0470 | 0.0481 | 0.0126 |
| Tot. Average | 0.0497 | 0.0479 | 0.0182 |

### 4.1.2 *Beat Accuracy Evaluation.*

We assess beat synchronization by comparing motion-derived peaks in hand speed to drum onset times from the MIDI input. Only hand-played instruments (e.g., snare, toms, hi-hats) are considered; foot-triggered notes (e.g., kick, MIDI 36) are excluded, as they are by default aligned with the beat that initiates the animation.

Hand velocity is computed from 3D trajectories and smoothed using a Savitzky-Golay filter. Local maxima in speed are detected as candidate impact points. Each motion peak is matched with the nearest MIDI onset within a frame window ±3 (approximately 50, ms at 60, FPS), and the delay is calculated as: $\Delta t = (t_{\text{peak}} - t_{\text{onset}}) \cdot \frac{1000}{\text{fps}}$. Negative delays indicate anticipatory motion, which is typical in drumming due to physical contact preceding the MIDI onset.

For a representative evaluation clip, the predicted motion yields a mean delay of −6.1 ms, a standard deviation of 22.1 ms, and a median delay of 0.0 ms. In comparison, the ground-truth motion capture exhibits a mean delay of −16.3 ms, a standard deviation of 31.7 ms, and a median of −16.7 ms. These results indicate that the synthesized motion closely mirrors real performance timing. A visualization is shown in Figure 4, which plots hand speed curves, detected motion peaks, and corresponding MIDI onset events.

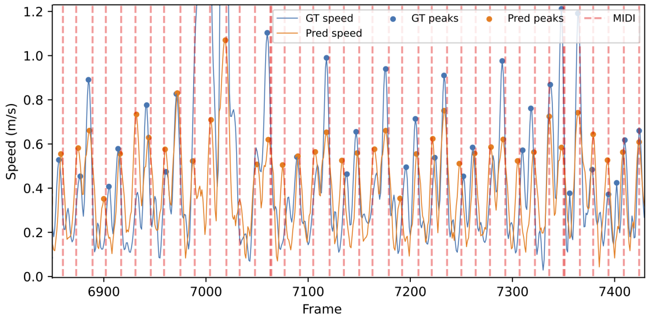

**Figure 4: Beat-to-motion synchronization. Blue/orange curves: 3D speed of ground-truth/predicted right hand; dots/circles: detected peaks; red dashed lines: MIDI onsets.**

## 5 Conclusions

We presented a system for generating expressive full-body drum performance animations directly from MIDI input. Our approach combines a BiLSTM network for hand motion prediction, a MIDI-matching module for upper-body and facial expressions, a procedural foot motion generator, and an IK solver that integrates all components into the final animation. Quantitative evaluation shows that the system produces hand trajectories that are temporally aligned with musical beats and statistically consistent with mocap data.

*Limitations.* While the system achieves accurate rhythm synchronization, its generalization is limited. In particular, performance degrades when encountering complex drum fills and patterns not well represented in the training set. Additionally, the current architecture does not scale to different drum configurations or setups with varying spatial layouts, as the model has implicitly learned the fixed setup present in the dataset. Another limitation is the lack of explicit spatial hit accuracy, since the dataset provides performer motion but no drum kit geometry, preventing verification of whether predicted movements intersect the drum surfaces.

*Future Work.* Addressing these limitations through larger datasets and more flexible, configuration-aware models is a promising direction for future work.

## Acknowledgments

This project has received funding from the European Commission's Horizon Europe program under grant agreements 101178362 and 101061303; the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No 739578 and the Government of the Republic of Cyprus through the Deputy Ministry of Research, Innovation and Digital Policy.

## References

Simon Alexanderson, Rajmund Nagy, Jonas Beskow, and Gustav Eje Henter. 2023. Listen, Denoise, Action! Audio-Driven Motion Synthesis with Diffusion Models. *ACM Trans. Graph.* 42, 4, Article 44 (July 2023), 20 pages.

Andreas Aristidou, Anastasios Yiannakidis, Kfir Aberman, Daniel Cohen-Or, Ariel Shamir, and Yiorgos Chrysanthou. 2023. Rhythm is a Dancer: Music-Driven Motion Synthesis with Global Structure. *IEEE Transactions on Visualization and Computer Graphics* 29, 8 (aug 2023), 3519–3534.

Alysha Bogaers, Zerrin Yumak, and Anja Volk. 2021. Music-Driven Animation Generation of Expressive Musical Gestures. In *Companion Publication of the 2020 International Conference on Multimodal Interaction* (Virtual Event, Netherlands) *(ICMI '20 Companion)*. ACM, NY, USA, 22–26.

Jiali Chen, Changjie Fan, Zhimeng Zhang, Gongzheng Li, Zeng Zhao, Zhigang Deng, and Yu Ding. 2021a. A Music-Driven Deep Generative Adversarial Model for Guzheng Playing Animation. *IEEE Transactions on Visualization and Computer Graphics* 29 (2021), 1400–1414. Issue 2.

Kang Chen, Zhipeng Tan, Jin Lei, Song-Hai Zhang, Yuan-Chen Guo, Weidong Zhang, and Shi-Min Hu. 2021b. ChoreoMaster: Choreography-Oriented Music-Driven Dance Synthesis. *ACM Trans. Graph.* 40, 4, Article 145 (July 2021), 13 pages.

George ElKoura and Karan Singh. 2003. Handrix: Animating the Human Hand. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (San Diego, California) *(SCA '03)*. Eurographics Association, Goslar, DEU, 110–119.

Fortnite. 2024. Fortnite Festival Season 4: Prepare for battle with Metallica! https://www.fortnite.com/news/fortnite-festival-season-4-prepare-for-battle-with-metallica Accessed: July 2025.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks* 18, 5 (2005), 602–610. Proccedings of IJCNN 2005.

Ruoxi Guo, Jiahao Cui, Wanru Zhao, Shuai Li, and Aimin Hao. 2021. Hand-by-Hand Mentor: An AR based Training System for Piano Performance. In *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. 436–437.

Asuka Hirata, Keitaro Tanaka, Masatoshi Hamanaka, and Shigeo Morishima. 2022. Audio-Driven Violin Performance Animation with Clear Fingering and Bowing. In *ACM SIGGRAPH 2022 Posters* (Vancouver, BC, Canada) *(SIGGRAPH '22)*. ACM, NY, USA, Article 7, 2 pages.

Yu-Fen Huang, Nikki Moran, Simon Coleman, Jon Kelly, Shun-Hwa Wei, Po-Yin Chen, Yun-Hsin Huang, Tsung-Ping Chen, Yu-Chia Kuo, Yu-Chi Wei, Chih-Hsuan Li, Da-Yu Huang, Hsuan-Kai Kao, Ting-Wei Lin, and Li Su. 2024. MOSA: Music Motion with Semantic Annotation Dataset for Cross-Modal Music Processing. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* (2024), 1–14.

Hsuan-Kai Kao and Li Su. 2020. Temporally Guided Music-to-Body-Movement Generation. In *Proceedings of the 28th ACM International Conference on Multimedia* (Seattle, WA, USA) *(MM '20)*. ACM, NY, USA, 147–155.

Junhwan Kim, Frederic Cordier, and Nadia Magnenat-Thalmann. 2000. Neural network-based violinist's hand animation. In *Proceedings Computer Graphics International 2000*. 37–41.

Nozomi Kugimoto, Rui Miyazono, Kosuke Omori, Takeshi Fujimura, Shinichi Furuya, Haruhiro Katayose, Hiroyoshi Miwa, and Noriko Nagata. 2009. CG Animation for Piano Performance. In *SIGGRAPH '09: Posters* (New Orleans, Louisiana) *(SIGGRAPH '09)*. ACM, NY, USA, Article 3, 1 pages.

Theodoros Kyriakou, Merce Alvarez de la Campa Crespo, Andreas Panayiotou, Yiorgos Chrysanthou, Panayiotis Charalambous, and Andreas Aristidou. 2024. Virtual Instrument Performances (VIP): A Comprehensive Review. *Comput. Graph. Forum* 43, 2 (2024), 35–58.

Theodoros Kyriakou, Andreas Aristidou, and Panayiotis Charalambous. 2025. Multi-Modal Instrument Performance (MMIP): A musical database. *Comput. Graph. Forum* 44, 2 (2025), e70025.

Bochen Li, Akira Maezawa, and Zhiyao Duan. 2018. Skeleton Plays Piano: Online Generation of Pianist Body Movements from MIDI Performance.. In *Proceedings of the International Society for Music Information Retrieval Conference*. 218–224.

Yuen-Jen Lin, Hsuan-Kai Kao, Yih-Chih Tseng, Ming Tsai, and Li Su. 2020. A Human-Computer Duet System for Music Performance. In *Proceedings of the 28th ACM International Conference on Multimedia* (NY, USA). ACM, 772–780.

Jun-Wei Liu, Hung-Yi Lin, Yu-Fen Huang, Hsuan-Kai Kao, and Li Su. 2020. Body Movement Generation for Expressive Violin Performance Applying Neural Networks. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 3787–3791.

Hiroki Nishizawa, Keitaro Tanaka, Asuka Hirata, Shugo Yamaguchi, Qi Feng, Masatoshi Hamanaka, and Shigeo Morishima. 2025. SyncViolinist: Music-Oriented Violin Motion Generation Based on Bowing and Fingering. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV'25)*. 5419–5428.

Zhiping Qiu, Yitong Jin, Yuan Wang, Yi Shi, Chao Tan, Chongwu Wang, Xiaobing Li, Feng Yu, Tao Yu, and Qionghai Dai. 2025. ELGAR: Expressive Cello Performance Motion Generation for Audio Rendition. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers (SIGGRAPH Conference Papers '25)*. ACM, NY, USA, Article 54, 9 pages.

Hiroyuki Sekiguchi and Shigeru Eiho. 2000. Generating the human piano performance in virtual space. In *Proceedings 15th International Conference on Pattern Recognition (ICPR'00, Vol. 4)*. 477–481 vol.4.

Takeru Shirai and Shinji Sako. 2021. 3D skeleton motion generation of double bass from musical score. In *15th International Symposium on Computer Music Multidisciplinary Research (CMMR)*.

Eli Shlizerman, Lucio Dery, Hayden Schoen, and Ira Kemelmacher-Shlizerman. 2018. Audio to Body Dynamics. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7574–7583.

Snehesh Shrestha, Cornelia Fermüller, Tianyu Huang, Pyone Thant Win, Adam Zukerman, Chethan M Parameshwara, and Yiannis Aloimonos. 2022. AIMusicGuru: Music Assisted Human Pose Correction. arXiv:2203.12829 [cs.CV]

Jonathan Tseng, Rodrigo Castellon, and C. Karen Liu. 2023. EDGE: Editable Dance Generation From Music. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ruocheng Wang, Pei Xu, Haochen Shi, Elizabeth Schumann, and C. Karen Liu. 2024. FürElise: Capturing and Physically Synthesizing Hand Motion of Piano Performance. In *SIGGRAPH Asia 2024 Conference Papers* (Tokyo, Japan) *(SA '24)*. ACM, NY, USA, Article 77, 11 pages.

Sen Wang, Jiangning Zhang, Xin Tan, Zhifeng Xie, Chengjie Wang, and Lizhuang Ma. 2025. MMoFusion: Multi-modal co-speech motion generation with diffusion model. *Pattern Recognition* 169 (2025), 111774.

Huazhe Xu, Yuping Luo, Shaoxiong Wang, Trevor Darrell, and Roberto Calandra. 2022. Towards Learning to Play Piano with Dexterous Hands and Touch. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 10410–10416.

Pei Xu and Ruocheng Wang. 2024. Synchronize Dual Hands for Physics-Based Dexterous Guitar Playing. In *SIGGRAPH Asia 2024 Conference Papers* (Tokyo, Japan) *(SA '24)*. ACM, NY, USA, Article 143, 11 pages.

Kazuki Yamamoto, Etsuko Ueda, Tsuyoshi Suenaga, Kentaro Takemura, Jun Takamatsu, and Tsukasa Ogasawara. 2010. Generating natural hand motion in playing a piano. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 3513–3518.

Kevin Zakka, Philipp Wu, Laura Smith, Nimrod Gileadi, Taylor Howell, Xue Bin Peng, Sumeet Singh, Yuval Tassa, Pete Florence, Andy Zeng, and Pieter Abbeel. 2023. RoboPianist: Dexterous Piano Playing with Deep Reinforcement Learning. In *Conference on Robot Learning (CoRL)*.

Yuanfeng Zhu, Ajay Sundar Ramakrishnan, Bernd Hamann, and Michael Neff. 2013. A system for automatic animation of piano performances. *Computer Animation and Virtual Worlds* 24 (9 2013), 445–457. Issue 5.