AI-Driven Dancing Robots: Synchronizing Motion with Music Generated by LLMs

Final Report
COMP8851 (2025-S1)

Supervisor(s):

AUTUMN WU
XIAOHAN YU
YUANKAI QI

Project Team Member(s):

KARTHIK NARAYAN VENKATASUBRAMANIAN - 48004545
GISHOR THAVAKUMAR - 48032875
MIR SADIA AFRIN - 47965495
GOPI ABHIRAM VISHAL CHONGALA - 48017892

Certification

Student Name	Big Picture - Idea & Discussion	Problem Formulation and Analysis	Identification of Research Methods	Research, Solution Development	Report Writing	Report Checking, providing Feedback Editing	Slide Preparation, Discussion on F2F Presentation	Signature
KARTHIK	10	30	30	20	40	20	20	Karthik V.
GISHOR	30	30	30	40	20	20	20	
SADIA	30	30	10	20	20	20	40	
ABHIRAM	30	10	30	20	20	40	20	
								[sign here]
								[sign here]
								[sign here]
								[sign here]
								[sign here]
								[sign here]
Total	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	**100%**	

Supervisor Signature Yanqiu Wu

Abstract: This research proposes a novel AI-driven robotic choreography system that integrates AI-generated multilingual music with real-time expressive and synchronized dancing robot via a robotic operating system - based control framework. The system autonomously generates dance movements synchronized to music features such as beat, intensity, and lyrical structure. This approach demonstrates a unique fusion of generative music and robotics, highlighting the significance of cross-modal AI integration for creative robotic performance. The robot was successfully deployed in a real-world performance, providing practical validation of the system's robustness and expressiveness. Overall, the project represents a significant step forward in AI-driven robotic performance art.

Keywords: *Robotic choreography, AI-generated music, multilingual lyrics, GPT-4, MusicGen, Suno AI, ROS2, JetRover, beat synchronization, real-time control, emotion-aware robotics, audio feature extraction, phonetic-responsive motion.*

1 Introduction

The convergence of generative artificial intelligence and robotics offers novel opportunities for advancing human-robot interaction through expressive, multimodal performance. This project investigates the development of a robotic system capable of synchronising physical motion with AI-generated multilingual music, including structured lyrics and vocal synthesis. The core objective is to construct a real-time choreography framework in which robotic movements are informed by rhythmic, phonetic, and emotional cues derived from AI-generated audio content.

While significant progress has been made in generative modelling for text, music, and speech, their integration with robotic control particularly for live, adaptive performance remains an underexplored area. Existing robotic dance systems are typically limited by deterministic, beat-aligned routines that lack responsiveness to linguistic variation and semantic content. This limitation is particularly evident in multicultural or multilingual contexts, where expressive performance must align with both rhythm and meaning.

To address this gap, the proposed system incorporates a modular ROS2-based control architecture for the Jet Rover robotic platform, real-time audio feature extraction, and beat-driven motion mapping. Music and lyrics are generated using commercial large language models (LLMs) and audio synthesis tools such as Suno AI, while extracted features such as tempo, beat structure, and loudness are used to drive expressive and context-sensitive robot behaviour.

This report details the final pipeline, encountered challenges, design decisions, and project outcomes. It presents a step-by-step account of how real-time, culturally aware robotic dance can be achieved by bridging AI-generated soundtracks with embodied physical motion. The system lays the foundation for future research into emotion-aware and phonetic-responsive choreography.

1.1 Aim

To build a robotic system that can perform real-time synchronised dance to AI-generated music, complete with multilingual lyrics and vocal synthesis. To design a modular choreography framework responsive to rhythm, and extensible toward phonetics and emotion in music. To enable expressive, real-world robot performances through beat-driven synchronisation and adaptive motion mapping.

1.2 Background

As artificial intelligence reshapes creative fields, the convergence of music generation and robotics is opening new frontiers for interactive performance. Today's large language models (LLMs) like ChatGPT and similar systems can produce emotionally rich, culturally diverse lyrics across multiple languages. In parallel, advanced tools for audio synthesis allow these lyrics to be rendered into expressive vocals with instrumental backing. However, robotic systems have not kept pace with this creative intelligence.

Most existing dance bots rely on pre-scripted motion sequences, tied only to fixed beats or tempo, and are usually limited to English-language songs. This severely restricts their emotional range and cross-cultural relevance.

Meanwhile, advances in motion planning, and gesture modelling point toward possibilities for more intelligent robotic expression. Yet, such approaches are either simulation-bound or overlook the semantic and emotional layers embedded in music especially in multilingual or emotionally nuanced contexts. Our project bridges this gap by integrating AI-generated multilingual music with real-time robotic choreography, focusing on physical embodiment, cultural awareness, and responsive movement.

To ground the project in real-world implementation, we utilize Jet Rover which is a robot platform with a servo-actuated arm and control it using ROS2, an open-source robotics middleware. The primary challenge lies in synchronizing these systems to not only follow rhythmic cues but also adapt to lyrical meaning, language-specific, and dynamic musicality.

1.3 Research Problems and the Context

This project addresses the core research question:

> **Can a robot be made to dance expressively and synchronously to AI-generated multilingual music, capturing both rhythmic and semantic content in real time?**

While existing systems can synchronize movement to beats, they fall short of capturing the deeper musical intent such as emotional tone, linguistic nuance, and melodic shifts especially across diverse languages like English, Hindi, and Tamil. Moreover, integrating such intelligence on physical platforms like Jet Rover introduces additional complexity: the hardware is limited by sparse documentation, unreliable servo behaviour, and inconsistent basic structural implementation.

Initially, we planned to rely on beat-tracking and amplitude-based motion. However, as the project progressed, it became clear that this would result in robotic motion that felt repetitive and emotionless. We recognized that truly expressive choreography must account for mood, language, and lyrical meaning. Consequently, we reframed our approach:

- Used commercial LLMs to generate lyrics in English, Hindi, and Tamil.
- Composed custom music and generated vocals independently to allow emotional control.
- Extracted beat-level, loudness, and amplitude features to inform body and arm motion.
- Overhauled the robot's software layer using ROS2, enabling real-time control of limbs and wheels for synchronized, semantically aware dance routines.

These adjustments reflect the technical goal and research depth of the project, particularly in aligning generative AI, robotics, and real-time control in a single system. More than just music-to-motion, this project explores how a robot can become an expressive performer.

1.4 (Expected) Outcomes

- A full AI-to-robot pipeline that converts lyrics and music into beat-based choreography without any preprogrammed moves for Jet Rover platforms.
- A music generation pipeline combining LLM-based lyrics with audio synthesis and multilingual voice.
- A modular robot control framework supporting expressive movement of wheels, arms, and other actuators.

- A synchronisation mechanism mapping rhythmic and loudness features into real-time dance, with extensibility for phonetic and emotional mapping in future work.

1.5 Benefits and Significance

This project redefines robotic performance by enabling machines to respond to the rhythmic, and ultimately semantic complexity of human-generated art. Beyond its technical contribution to AI and robotics, it showcases a new medium for cultural storytelling and public engagement. In educational, entertainment, and artistic settings, such systems could foster inclusive, multilingual experiences where robots become not just tools or performers, but interpreters of diverse musical traditions.

2 Literature Review

The problem of enabling robots to perform expressive, synchronised dance to AI-generated multilingual music draws on several well-established fields of research, including automatic music generation, cross-modal alignment of sound and motion, robot dance synthesis, and risk-aware motion planning. Considerable progress has been made in each area, yet existing work still leaves key challenges unaddressed particularly the development of a unified, real-time pipeline suitable for deployment on physical robots interacting with dynamic, diverse musical inputs. This section reviews the most relevant conceptual and technical foundations.

2.1 Brief Literature Review

This section surveys key research areas that intersect in our problem: generative music, robotic dance, cross-modal alignment, and adaptive control. We highlight representative work in each area and identify how they inform or contrast with our approach.

2.1.1 Automatic Music Generation and Cross-Modal Alignment

In recent years, AI-driven music generation has advanced significantly, with models evolving from handcrafted feature extraction and autoregressive token models to latent diffusion generators guided by cross-attention mechanisms. This shift allows for the integration of multiple modalities such as motion, rhythm, and semantics into the music generation process. A key insight from current literature is that fusing beat-level features with emotional content leads to audio outputs that are both rhythmically aligned and thematically coherent (Ji et al., 2025).

Our system applies these insights by combining beat timing with lyrical mood and semantics derived from LLM-generated lyrics, ensuring that the generated music serves as an effective driver for robot choreography.

2.1.2 Robot Dance Synthesis

Early efforts, such as those by Santiago et al. (2011), relied on pre-programmed routines, enabling robots to move in sync with fixed beats but lacking flexibility and real-time responsiveness. Moreover, traditional robot-dance work either reused human trajectories or required tedious motion-library authoring. Ahn (2024) proposed the first framework that lets non-humanoid bodies learn to dance directly from human videos by (i) learning an optical-flow/music similarity reward via contrastive learning and (ii) optimising a reinforcement-learning policy to maximise that reward. The study validates two principles we adopt:

- Optical-flow-based “visual rhythm” is a lightweight but effective representation for musical beat structure which is ideal for real-time onboard computation on Jetson-based rovers.

- Reward shaping with learned audio-visual similarity allows the same policy to generalise across radically different sound structure (cart-pole). Similarly, we can train a small network to score how well the rover's motion matches the music (audio–visual similarity).

Hou et al. (2025) proposed a method for enabling legged robots to synchronize walking patterns with rhythmic beats. Additionally, Liu et al. (2022) enhanced gesture synthesis using self-supervised learning for conducting motions, focusing on synchronization with classical music. However, their model lacked adaptability to multilingual or emotionally varied music genres.

2.1.3 Music-Aligned Motion Generation

It is now well established that temporal audio features can effectively condition robotic movement policies. Frameworks such as Music-Driven Robot Primitives Choreography (MDRPC) demonstrate that physics-based RL agents can learn dance gestures synchronised to musical beats and tempo (Guan et al., 2024).

However, these studies typically focus on humanoid avatars in simulation, and do not address whole-body coordination or deployment on real hardware. Our project addresses this gap by implementing a hardware-validated system on JetRover, combining arm and wheel coordination in real time, an area largely unexplored in current work.

2.1.4 Integrated Locomotion and Manipulation

Research on whole-body RL control for mobile platforms shows that robots can simultaneously manage locomotion and manipulation (Wang et al., 2024). However, these systems are typically optimised for utility-driven tasks such as object retrieval or navigation not for aesthetic synchronisation with musical stimuli. As a result, timing precision and expressive fluency, both critical in performance contexts remain underexplored. Our work focuses precisely on this aspect, developing a system where timing accuracy and aesthetic synchronisation with AI-generated music are primary objectives.

2.1.5 Environment-Aware Robust Motion

It is well recognised that expressive motion must also be robust to environmental variation. Research on perception-conditioned control demonstrates that robots can adapt movement in response to changing visual input (Sekkat et al., 2021). However, such techniques have largely been applied to static tasks like pick-and-place operations, not to dynamic, rhythm-driven choreography requiring continuous realignment of pose and movement.

Our system is designed to extend these principles to dynamic performance settings, where the robot must remain synchronised with music while adapting to its environment—an essential capability for live robotic dance.

2.2 Research Problem

While the literature demonstrates progress in music generation, robot dance synthesis, audio-conditioned motion, and risk-aware control, key gaps remain when viewed through the lens of our project. Current cross-modal music generation techniques (Ji et al., 2025) focus on producing aligned audio, but do not extend to controlling physical robot motion in real time. Similarly, robot dance synthesis via (Ahn, 2024) is typically confined to simulation or humanoid platforms, and often relies on pre-scripted movements.

Work on music-driven locomotion (Guan et al., 2024) shows that audio features can condition robot policies, but remains limited to simulated humanoids without real-world validation. Studies on integrated manipulation (Wang et al., 2024) optimize for utility tasks, not aesthetic synchronisation, while perception-conditioned control (Sekkat et al., 2021) has yet to be applied to dynamic, beat-driven choreography.

In contrast, our project integrates:

- AI-generated multilingual music
- Real-time beat perception
- Algorithmic motion mapping and adaptation
- Hardware-synchronised choreography

This extends existing work from simulation to real-world platforms, and from basic beat-sync toward architectures enabling future semantic and emotional alignment. The problem is well-motivated and fills a clear gap not already solved, but a natural next step in the field. Having established the state of the art and our project's unique position, we next discuss our research plan and methodology to address these gaps.

3 Research Plan and Methodology

This project adopted a multi-phase, data-driven experimental methodology designed to bridge generative AI with real-time robotic choreography. The research process evolved significantly from its original design due to both practical hardware limitations and emerging insights gained during development. What began as an ambitious vision of multi-robot, emotion-aware choreography was progressively refined into a more focused, functional, and technically grounded system, capable of multilingual beat-aligned robotic dance on real-world platforms.

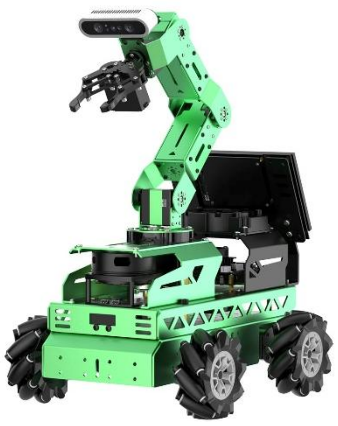


Figure 1: JetRover robot used for deployment of choreography model

As shown in Figure 1, the JetRover robot served as the primary deployment platform. Understanding its mechanical structure, articulated arm, and onboard computing unit was fundamental to designing a modular control architecture, selecting compatible AI components, and planning robust fallback strategies across development phases.

3.1 Analysis:

The main problem was decomposed into six interdependent stages, as illustrated in Figure 2. Initially, our project methodology encompassed three converging research pathways:

- **Experimental Research** for AI-driven music generation.
- **Data-Driven Experimental** methods for feature extraction and motion mapping.
- **Empirical Robotics Deployment** for synchronization and environmental feedback.

The original roadmap included advanced concepts like computer vision-based motion imitation and reinforcement learning (RL) for real-time adaptive dance. However, as implementation challenges surfaced, most notably, unpredictable hardware communication interfaces, ROS2-to-servo latency, and lack of structured training datasets which we revised our methodology. The adapted and now fully realized framework is shown in Figure 2. By narrowing our focus, we were able to produce a system that still validates our core research problem.

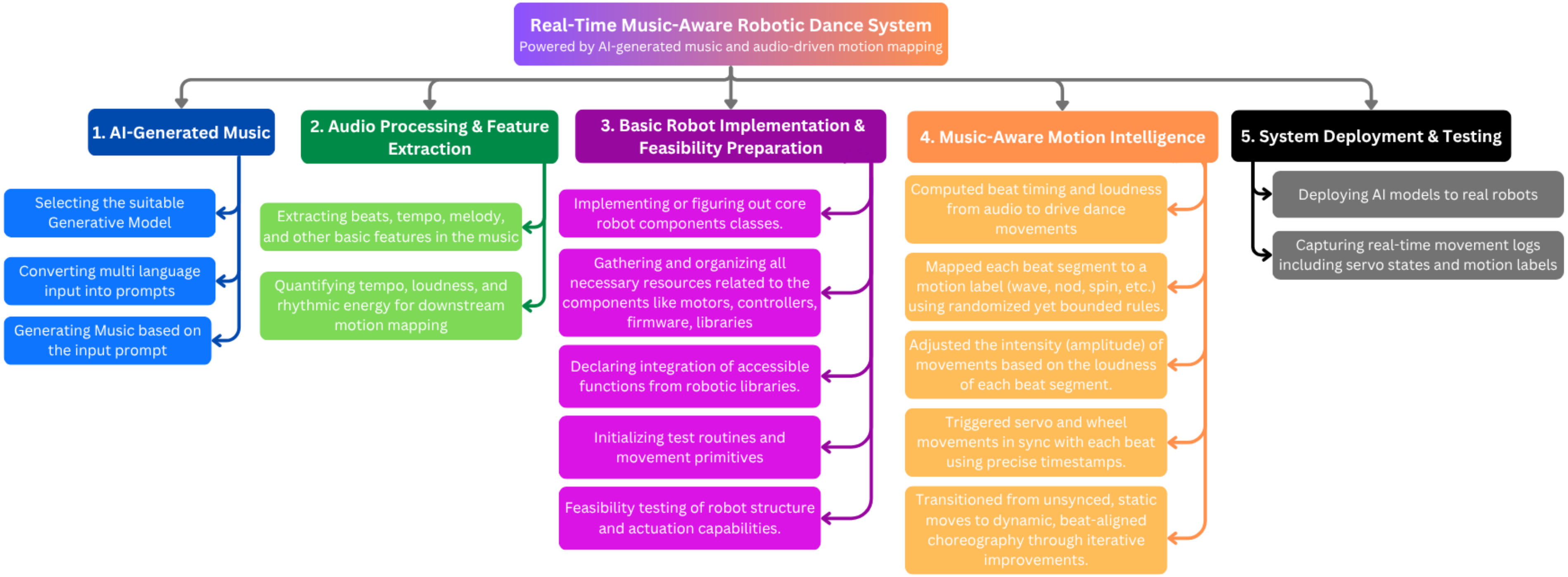

Real-Time Music-Aware Robotic Dance System
Powered by AI-generated music and audio-driven motion mapping
1. AI-Generated Music
Selecting the suitable Generative Model
Converting multi language input into prompts
Generating Music based on the input prompt
2. Audio Processing & Feature Extraction
Extracting beats, tempo, melody, and other basic features in the music
Quantifying tempo, loudness, and rhythmic energy for downstream motion mapping
3. Basic Robot Implementation & Feasibility Preparation
Implementing or figuring out core robot components classes.
Gathering and organizing all necessary resources related to the components like motors, controllers, firmware, libraries
Declaring integration of accessible functions from robotic libraries.
Initializing test routines and movement primitives
Feasibility testing of robot structure and actuation capabilities.
4. Music-Aware Motion Intelligence
Computed beat timing and loudness from audio to drive dance movements
Mapped each beat segment to a motion label (wave, nod, spin, etc.) using randomized yet bounded rules.
Adjusted the intensity (amplitude) of movements based on the loudness of each beat segment.
Triggered servo and wheel movements in sync with each beat using precise timestamps.
Transitioned from unsynced, static moves to dynamic, beat-aligned choreography through iterative improvements.
5. System Deployment & Testing
Deploying AI models to real robots
Capturing real-time movement logs including servo states and motion labels

Figure 2: Research methodology flow showing experimental, data-driven, and empirical stages from AI music generation to real-world robot testing and performance evaluation.

Our adapted methodology consists of five structured modules, each chosen to address a critical sub-problem using the most suitable and scalable technique available.

1. AI-Generated Music

Why: We selected LLM-based lyric generation (e.g., GPT-4) to support multi-language fluency, a core requirement for our culturally inclusive goal. This allowed for emotional tone and phonetic rhythm generation in English, Hindi, and Tamil.

How: Prompts were carefully structured to control language, mood, and structure. Tools like Suno AI then synthesize musical compositions and vocals.

Justification: LLMs are the de facto standard for generative text across domains. Their ability to produce semantically rich and linguistically diverse content aligned perfectly with our ambition for cross-cultural musical expression.

2. **Audio Processing & Feature Extraction**
   Why: Rather than relying on raw audio for motion control, we extracted beat, tempo, loudness, and energy profiles to map to robotic movements. Tools like Librosa provided spectral features needed for rhythmic interpretation.

   How: We implemented beat-tracking and energy quantification algorithms to convert audio into timestamped motion triggers.

   Justification: Libraries based feature extraction is a robust, real-time-compatible method, widely used in audio-reactive systems. It enabled us to decouple motion control from raw waveform processing and allowed modular data-driven control downstream.

3. **Basic Robot Implementation & Feasibility Preparation**
   Why: Given Jet Rover's under documented implementation and usage, we adopted a bottom-up robotics integration strategy. This involved writing custom ROS2 nodes for wheel motion, arm servo control, and movement synchronization.

   How: We modularized robot control into three packages: Arm Controller, Robot Mover, and central Dance Manager (DanceNode). Each was developed and tested in isolation before system-level integration.

   Justification: ROS2 was selected for its support for real-time communication, node modularity, and hardware abstraction, all critical for synchronizing separate physical components like arms and wheels. The feasibility stage helped define limits of servo speed, motion resolution, and network latency.

4. **Music-Aware Motion Intelligence**
   Why: We originally planned to use RL or motion capture data for learning dance movements, but hardware and data constraints led us to design a rule-based, randomized choreography engine.

   How: Beat segments were mapped to labeled movements ("wave," "spin," "nod," etc.) with intensities modulated by loudness curves. Transitions were made smoother via interpolation and timestamp queuing.

   Justification: In the absence of real motion datasets and high-precision actuation, a heuristic approach ensured both safety and expressiveness. It also allowed us to preserve semantic cues like rhythm energy while remaining fully real-time.

5. **System Deployment & Testing**
   Why: Evaluation on a physical platform was critical to assess real-world feasibility. We tested various combinations of languages, tempos, and choreographies.

   How: Logs were captured from ROS2 topics for wheel velocities and servo angles, which were analyzed alongside video footage.

   Justification: While we couldn't quantify dance quality numerically, beat alignment, execution latency, and motion variety were used as primary indicators of system responsiveness and sync.

Our methodology was iterative. We tested each stage (e.g., generating a sample song and making the robot respond) before integrating them, which aligns with an experimental prototyping approach.

3.2 Synthesis

The synthesis of these modules was achieved through a modular pipeline, where each sub-system communicates through well-defined data flows and timestamps. Figure below illustrates the architecture of this final integrated system.

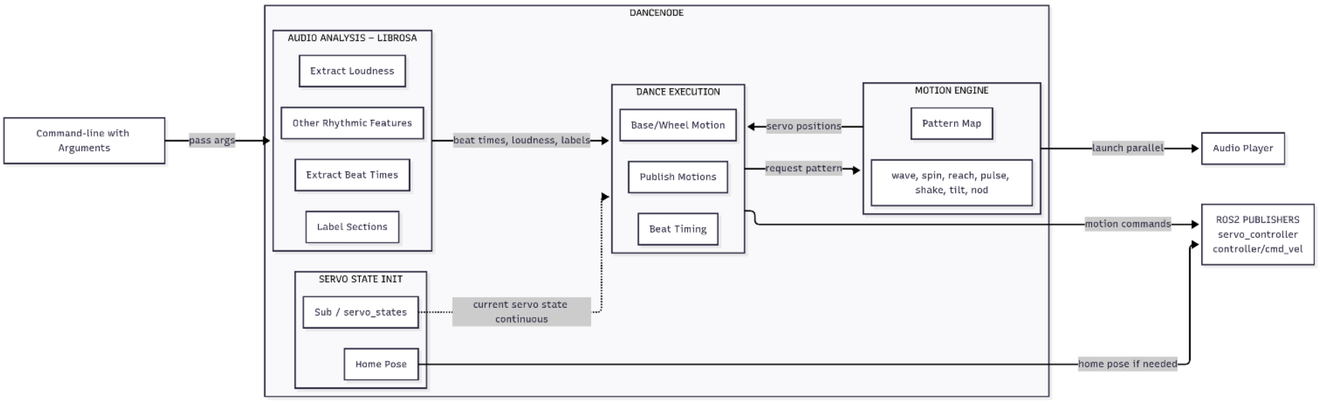


Figure 3: Internal architecture of the DanceNode, the central ROS2 module responsible for real-time choreography execution. It integrates audio feature inputs (beat timing, loudness, section labels) from Librosa with continuous servo state monitoring. Based on these inputs, the Dance Execution unit selects appropriate motion patterns from the Motion Engine (e.g., wave, spin, nod) and publishes synchronized motion commands to ROS2 publishers. This design ensures dynamic, beat-aligned movements with precise timing across both base and arm actuators.

- The pipeline begins with the AI-generated music module, where multilingual prompts are used to produce both lyrics and instrumental music. This establishes the emotional and linguistic context of the performance.
- The music is passed to the audio processing module, which extracts rhythmic features such as beats, tempo, and loudness using signal processing techniques. These features form the temporal anchors for downstream choreography.
- Simultaneously, the robot implementation layer establishes motor control primitives by interfacing Jet Rover's wheelbase and articulated arm with custom ROS2 nodes. This foundational layer exposes movement logics that can be called asynchronously by the choreography controller.
- The motion intelligence module then maps each beat to randomized motion labels (e.g., wave, nod, spin) and modulates the amplitude based on the beat's loudness. These instructions are queued and precisely time-triggered to ensure visual alignment with the music's rhythm.
- Finally, the deployment layer manages real-time execution, synchronizing arm and wheel movements on beat events while capturing performance logs and system diagnostics.

This integrated solution was designed to be modular, real-time responsive, and scalable. However, during integration, we discovered additional challenges not fully foreseen in the original design particularly the need for latency buffering and motion smoothing to avoid jitter during beat-aligned transitions. These were addressed through timestamp interpolation, preloading command queues, and tuning control loop frequencies across ROS2 nodes.

Consequently, through coordinated integration of data-driven audio processing, modular control nodes, and timestamp-based motion planning, we were able to synthesize a system that not only performs synchronized dance to music but lays the foundation for future expansions such as emotion-aware choreography, phonetic gesture alignment, and multi-robot interaction.

3.3 Reframing the Problem

The project initially assumed that simple beat matching would suffice for achieving realistic robotic synchronisation to music. Early plans focused on tempo and beat extraction from AI-generated multilingual songs, with the belief that rhythmic alignment alone would yield convincing choreography.

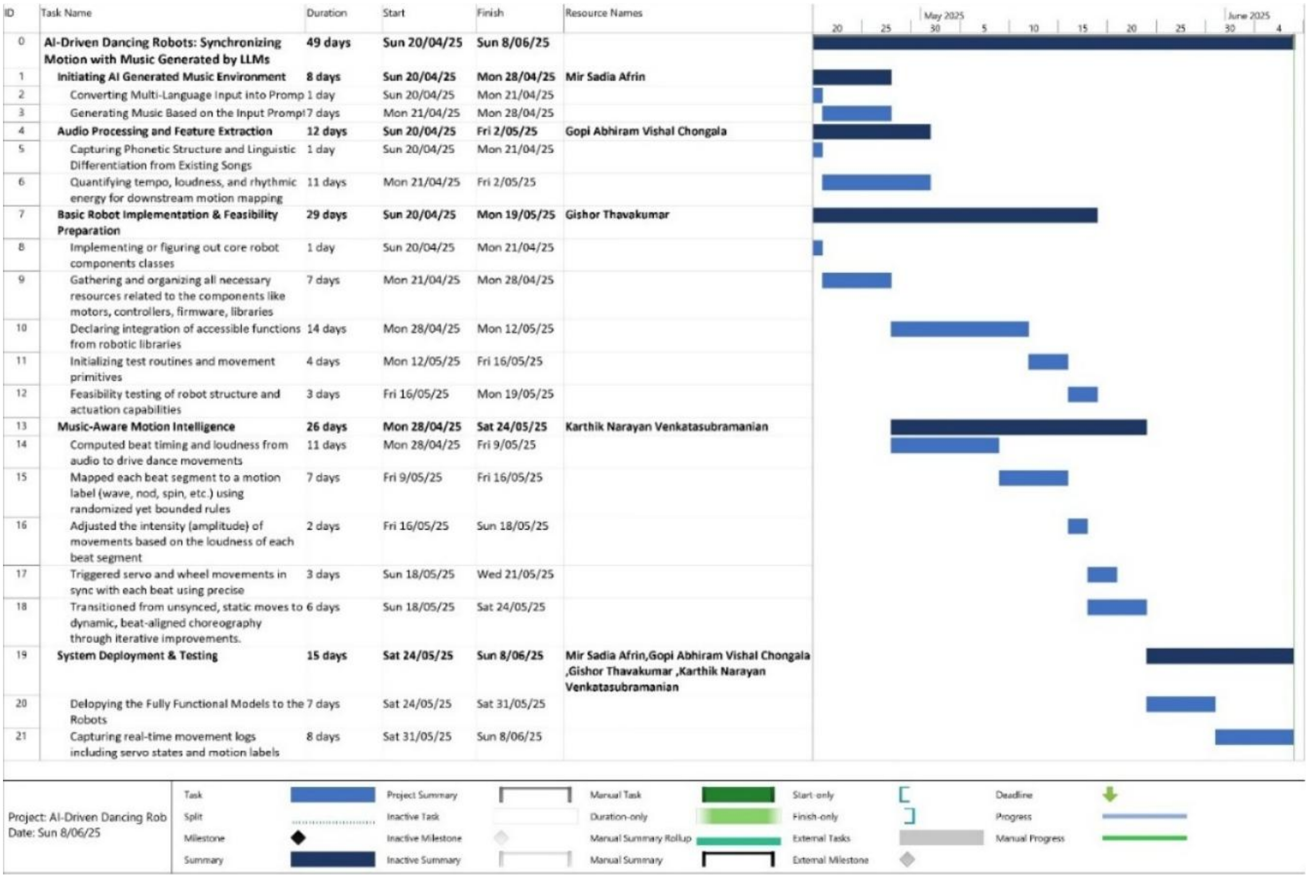
However, during the audio processing stage, it became clear that beat matching alone was inadequate. Music, particularly in multilingual contexts, carries meaning through phonetic patterns, emotional tone, and lyrical phrasing where elements are not captured by rhythm alone. To create more natural and expressive choreography, we expanded our approach to incorporate linguistic and emotional cues, complementing rhythmic features.

Simultaneously, practical challenges arose when deploying to the Jet Rover platform. We discovered that existing low-level control functions were incomplete and poorly documented, limiting our ability to execute synchronised choreography. This led us to redesign the robot control architecture. We developed modular ROS2 classes to manage:

- Basic locomotion (forward, backward, rotation)
- Arm articulation with 6 Degree of Freedom (DoF)
- Multi-joint synchronisation, coordinated via a central Manager class

This redesign improved the system’s robustness and flexibility. As a result, the project evolved from a narrow "beat-to-motion mapping" problem to building a more adaptable choreography framework which one capable of supporting music-aware and semantic extensions in future work.

3.4 Timeline



ID	Task Name	Duration	Start	Finish	Resource Names
0	**AI-Driven Dancing Robots: Synchronizing Motion with Music Generated by LLMs**	**49 days**	**Sun 20/04/25**	**Sun 8/06/25**	
1	**Initiating AI Generated Music Environment**	**8 days**	**Sun 20/04/25**	**Mon 28/04/25**	**Mir Sadia Afrin**
2	Converting Multi-Language Input into Promp	1 day	Sun 20/04/25	Mon 21/04/25	
3	Generating Music Based on the Input Prompt	7 days	Mon 21/04/25	Mon 28/04/25	
4	**Audio Processing and Feature Extraction**	**12 days**	**Sun 20/04/25**	**Fri 2/05/25**	**Gopi Abhiram Vishal Chongala**
5	Capturing Phonetic Structure and Linguistic Differentiation from Existing Songs	1 day	Sun 20/04/25	Mon 21/04/25	
6	Quantifying tempo, loudness, and rhythmic energy for downstream motion mapping	11 days	Mon 21/04/25	Fri 2/05/25	
7	**Basic Robot Implementation & Feasibility Preparation**	**29 days**	**Sun 20/04/25**	**Mon 19/05/25**	**Gishor Thavakumar**
8	Implementing or figuring out core robot components classes	1 day	Sun 20/04/25	Mon 21/04/25	
9	Gathering and organizing all necessary resources related to the components like motors, controllers, firmware, libraries	7 days	Mon 21/04/25	Mon 28/04/25	
10	Declaring integration of accessible functions from robotic libraries	14 days	Mon 28/04/25	Mon 12/05/25	
11	Initializing test routines and movement primitives	4 days	Mon 12/05/25	Fri 16/05/25	
12	Feasibility testing of robot structure and actuation capabilities	3 days	Fri 16/05/25	Mon 19/05/25	
13	**Music-Aware Motion Intelligence**	**26 days**	**Mon 28/04/25**	**Sat 24/05/25**	**Karthik Narayan Venkatasubramanian**
14	Computed beat timing and loudness from audio to drive dance movements	11 days	Mon 28/04/25	Fri 9/05/25	
15	Mapped each beat segment to a motion label (wave, nod, spin, etc.) using randomized yet bounded rules	7 days	Fri 9/05/25	Fri 16/05/25	
16	Adjusted the intensity (amplitude) of movements based on the loudness of each beat segment	2 days	Fri 16/05/25	Sun 18/05/25	
17	Triggered servo and wheel movements in sync with each beat using precise	3 days	Sun 18/05/25	Wed 21/05/25	
18	Transitioned from unsynced, static moves to dynamic, beat-aligned choreography through iterative improvements.	6 days	Sun 18/05/25	Sat 24/05/25	
19	**System Deployment & Testing**	**15 days**	**Sat 24/05/25**	**Sun 8/06/25**	**Mir Sadia Afrin,Gopi Abhiram Vishal Chongala ,Gishor Thavakumar ,Karthik Narayan Venkatasubramanian**
20	Delopying the Fully Functional Models to the Robots	7 days	Sat 24/05/25	Sat 31/05/25	
21	Capturing real-time movement logs including servo states and motion labels	8 days	Sat 31/05/25	Sun 8/06/25	

Project: AI-Driven Dancing Rob
Date: Sun 8/06/25

Figure 4: This is the updated Gantt Project. This Gantt chart outlining the detailed timeline of the pending tasks, task distribution, and resource allocation for the mentioned tasks. The chart highlights all key phases with assigned team responsibilities and deadlines.

3.5 Risk Mitigation

Through initial planning and implementation, we identified that the highest risks were concentrated in the robotic implementation stages, particularly given the hardware limitations of the Jet Rover platform. To ensure consistent progress, we defined fallback strategies for each project phase previously. Throughout the project, predefined fallback strategies proved critical in navigating unforeseen technical challenges.

AI-Generated Music: Our fallback to commercial LLMs was necessary. Initial open-source models lacked diversity and control. As planned, we transitioned to GPT-4 and Suno AI for higher-quality multilingual lyrics and music generation. This ensured creative, fluent, and culturally adaptive outputs.

Audio Processing & Feature Extraction: When phonetic and linguistic extraction became infeasible within time limits, we reverted to low-level audio features, beat and loudness. This fallback was successful and directly enabled the robot to synchronise movements to tempo dynamics.

Machine Learning & Motion Mapping: The fallback of using annotated datasets failed due to data scarcity and poor motion generalisation. To overcome this, we designed a custom algorithmic pipeline that generated dance movements using logical rules mapped from beat features. This "music-aware motion intelligence" was a breakthrough, producing varied, synchronised, and visually pleasing choreography.

Real-Time Synchronisation with Environmental Adaptiveness: The rule-based fallback failed due to ROS driver and power issues. Instead, we resolved LiDAR conflicts by fixing underlying driver bugs and were inspired by ROS Nav2 principles to program obstacle-aware navigation. The robot could then dynamically respond to its surroundings, partially fulfilling the adaptiveness goal.

System Deployment & Testing: Due to hardware and time constraints, we prioritized the fallback of a robust single-robot demo, ensuring successful deployment of the core system.

Computer Vision for Followed Dance: As planned, this module was not prioritised due to scope limits along with limitation in time.

Team Risks: While fallback strategies were in place, no interpersonal issues emerged, and all team members collaborated smoothly.

These mitigations were pivotal to project success, allowing recovery from critical roadblocks while preserving core goals.

4 Results/Outcomes and Analysis/Discussion

Following the research plan and methodology described earlier, we successfully built and deployed a working AI-driven choreography system on the Jet Rover platform. The final system allows the robot to perform real-time synchronised dance movements to AI-generated multilingual music, in alignment with the modular architecture and staged development described in Section 3.

Throughout this process, several key refinements and practical insights emerged, prompting adjustments to the original plan and informing future directions.

4.1 Application of Methods and Progress on Sub-Problems

Robot Kinematics and Control Architecture

Consistent with the Basic Robot Implementation & Feasibility Preparation phase, our first critical step was understanding the robot's kinematic capabilities. This involved detailed analysis of the robot's wheel-base dynamics and arm articulation (6 DoF). We validated how each servo and motor responded under ROS2 control, which informed realistic constraints for choreography design.

Control Module Development

To overcome the gaps identified in native JetRover ROS Environment, we followed through on the plan to implement a custom modular control architecture:

Arm Controller - Allows precise, synchronised articulation of all servo joints in the robot's arms.

Robot Mover - Controls linear and rotational wheel motion, synchronised with beat timing.

Robot Manager - Integrates arm and wheel movements for coordinated expressive choreography based on incoming beat-segment data.

The final choreography loop, implemented in dance_attempt.py, reads real-time beat segments from Librosa based audio analysis, maps them to motion labels, and drives robot movement, accordingly, achieving real-time beat-synchronised performance.

4.2 Results and Achievements → *What was accomplished*

The JetRover robot can now reliably perform synchronised motion sequences combining base movement and arm articulation, driven by AI-generated music. Real-time beat perception using Librosa is fully integrated, with beat intervals mapped to predefined motion labels (wave, spin, tilt, nod, etc.). Arm and wheel coordination is handled via ROS2 messaging, with demonstrated live choreography aligning robot actions with musical rhythm and partially aware of the environment with the help of LiDAR. The software framework is now modular and extensible, providing a strong foundation for future emotion and semantic-aware extensions.

4.3 Technical Implementation → *How it was built (system design, modules, tools, etc.)*

The AI-driven robotic dance system was implemented using a ROS2-based software architecture with a modular node structure. The central component is the `DanceNode` which is a Python node responsible for audio analysis, choreography generation, and command orchestration while dedicated ROS2 nodes handle lower-level motor control (e.g., a servo controller node for the robot's articulated joints and a wheel controller node for the mobile base). This design follows a publish/subscribe paradigm: the `DanceNode` publishes motion commands to actuation topics and subscribes to sensor/feedback topics (such as servo state feedback), ensuring loose coupling between perception, planning, and actuation. The modular ROS2 architecture not only enhances separation of concerns (each node focuses on a specific task) but also facilitates real-time performance by running tasks in parallel and communicating asynchronously.

Sequence diagram in Figure 5 describing real-time interaction among system components from launch to dance execution. The process begins with a CLI command that launches the DanceNode, which analyzes the audio for beat times and loudness using Librosa. The music-aware motion intelligence module then assigns motion labels and generates servo targets for each beat. The node sets up ROS2 publishers for servo and wheel commands and subscribes to servo state updates.

Once ready, it triggers audio playback via an external system. For each detected beat, the DanceNode publishes a synchronized servo motion command with target positions and durations. Every 4 beats, a subtle "wiggle" command is sent to the steering servo, and on every 8th beat if loudness exceeds a threshold, a wheel move command nudges the robot forward or backward. These commands are precisely timed to match musical beats, creating coordinated dance movements. Throughout execution, the node logs and responds to servo state feedback. At the end, it sends a home-position and stop command to conclude the routine smoothly.

During each beat interval, the motion mapping logic uses a set of parametric motion primitives (defined in generate_motion) to translate motion labels and amplitude into joint positions. For instance, "wave" alternates servos, "reach" extends the arm, and "nod" tilts forward and back. Louder beats yield more intense movement, while softer beats result in subtle gestures. These joint positions are wrapped in a ServosPosition message and published to the servo controller, which smoothly interpolates the movement over the beat duration. Timing is managed precisely by using non-blocking waits and beat timestamps, ensuring consistent synchronization.

Base movement is handled similarly. Steering oscillations every 4 beats and short velocity bursts every 8 beats (on strong beats) add expressive movement. Commands are brief and well-timed to preserve responsiveness and balance. ROS2's asynchronous messaging and modular control nodes allow simultaneous wheel and arm motion, resulting in fluid and expressive robotic choreography aligned tightly to the music.

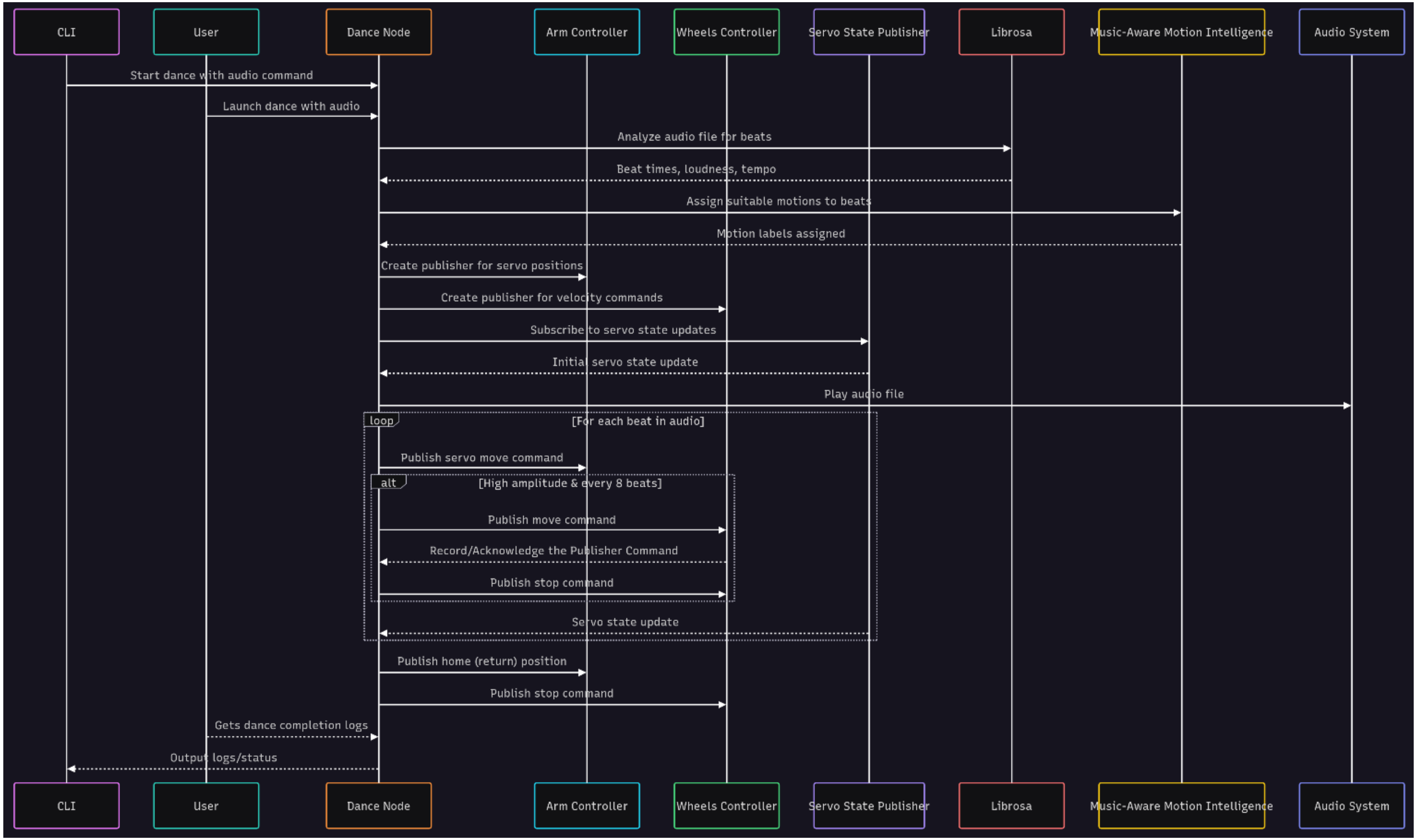
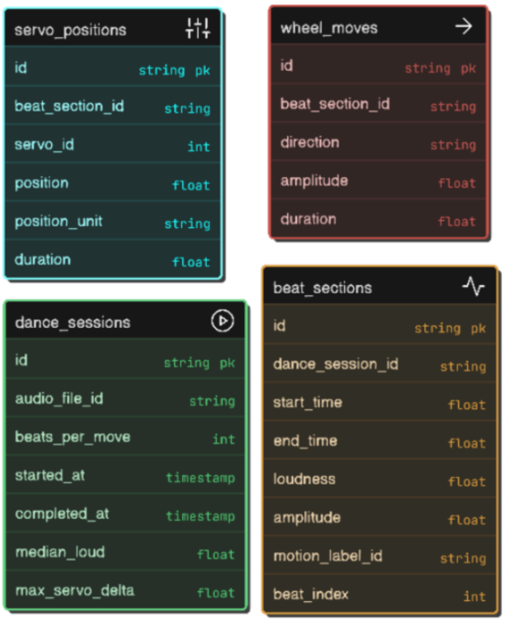

CLI
User
Dance Node
Arm Controller
Wheels Controller
Servo State Publisher
Librosa
Music-Aware Motion Intelligence
Audio System
Start dance with audio command
Launch dance with audio
Analyze audio file for beats
Beat times, loudness, tempo
Assign suitable motions to beats
Motion labels assigned
Create publisher for servo positions
Create publisher for velocity commands
Subscribe to servo state updates
Initial servo state update
Play audio file
loop
[For each beat in audio]
Publish servo move command
alt
[High amplitude & every 8 beats]
Publish move command
Record/Acknowledge the Publisher Command
Publish stop command
Servo state update
Publish home (return) position
Publish stop command
Gets dance completion logs
Output logs/status
CLI
User
Dance Node
Arm Controller
Wheels Controller
Servo State Publisher
Librosa
Music-Aware Motion Intelligence
Audio System

Figure 5: Sequence diagram illustrating the real-time execution flow of the DanceNode system. It captures the interactions between audio analysis (via Librosa), motion assignment, ROS2 controllers, and audio playback, resulting in synchronized robotic choreography. Click here to view full-resolution version



servo_positions	
id	string pk
beat_section_id	string
servo_id	int
position	float
position_unit	string
duration	float

wheel_moves	
id	string pk
beat_section_id	string
direction	string
amplitude	float
duration	float

dance_sessions	
id	string pk
audio_file_id	string
beats_per_move	int
started_at	timestamp
completed_at	timestamp
median_loud	float
max_servo_delta	float

beat_sections	
id	string pk
dance_session_id	string
start_time	float
end_time	float
loudness	float
amplitude	float
motion_label_id	string
beat_index	int

Figure 6: Presenting core schema used to structure robotic choreography data.

Throughout development, a structured data schema was designed to organize choreography sessions and related metadata (Figure 6). Instead of relying on scattered values or ad-hoc logic, formal data representations were used to define and manage each dance routine, helping both debugging and analysis. At the core, the dance_sessions table logs each session's metadata, including audio file reference, beats-per-move, and timestamps. Each session links to multiple beat_sections, capturing timing, loudness, amplitude, and motion labels for every beat. Corresponding actuator commands are stored in servo_positions (per joint, per beat) and wheel_moves (logging base motion). Additional tables like servos describe individual motor IDs and characteristics. This schema, implemented via Python data classes and optionally serializable to JSON, ensures clarity and extensibility, allowing motion tracking, playback, or analysis across sessions with ease.

4.4 Experimental Trials and Challenges Faced

4.4.1 EDGE-Based Human-to-Robot Motion Mapping Experiment

The first experimental approach explored leveraging the **E**ditable **D**ance **GE**neration from Music (**EDGE**) model, a state-of-the-art framework for generating 3D human dance motion from audio input (Tseng et al., 2023). The core idea was to translate these human-centric dance movements into corresponding robotic actions using a logical spatial mapping strategy tailored for a rover-based robot.

A bounding box was conceptually defined around the animated human avatar to monitor positional shifts and gestures. Forward steps by the dancer were mapped to linear wheel motions on the robot, while angular or lateral body movements, such as stepping diagonally or turning, triggered directional shifts or wheel pivots. To emulate arm gestures, the spinal and upper body movements from the avatar's backbone were translated to the robot's single articulated arm, allowing for expressive gestures like nods, waves, or reaches, despite the robot's limited degrees of freedom.

This method aimed to bring a natural, human-like fluidity to robotic dance, potentially enabling robots to mirror complex full-body choreography grounded in real dance motion data. However, this pathway was not pursued further due to significant computational demands of running EDGE on limited hardware. In addition, the resulting choreography lacked interactivity and real-time energy responsiveness. The output, although visually smooth, failed to match the beat-reactive and lively experience achieved through the primary music-aware choreography pipeline.

In essence, this approach laid foundational ideas for future human-to-robot motion translation but proved unsuitable for immediate deployment under current hardware constraints and project timeframes.

4.4.2 LLM-Based Effective Dance Generation

In an experimental trial, we implemented a pipeline using a large language model (GPT-4.1) to generate generic dance action tokens from song lyrics and textual prompts.

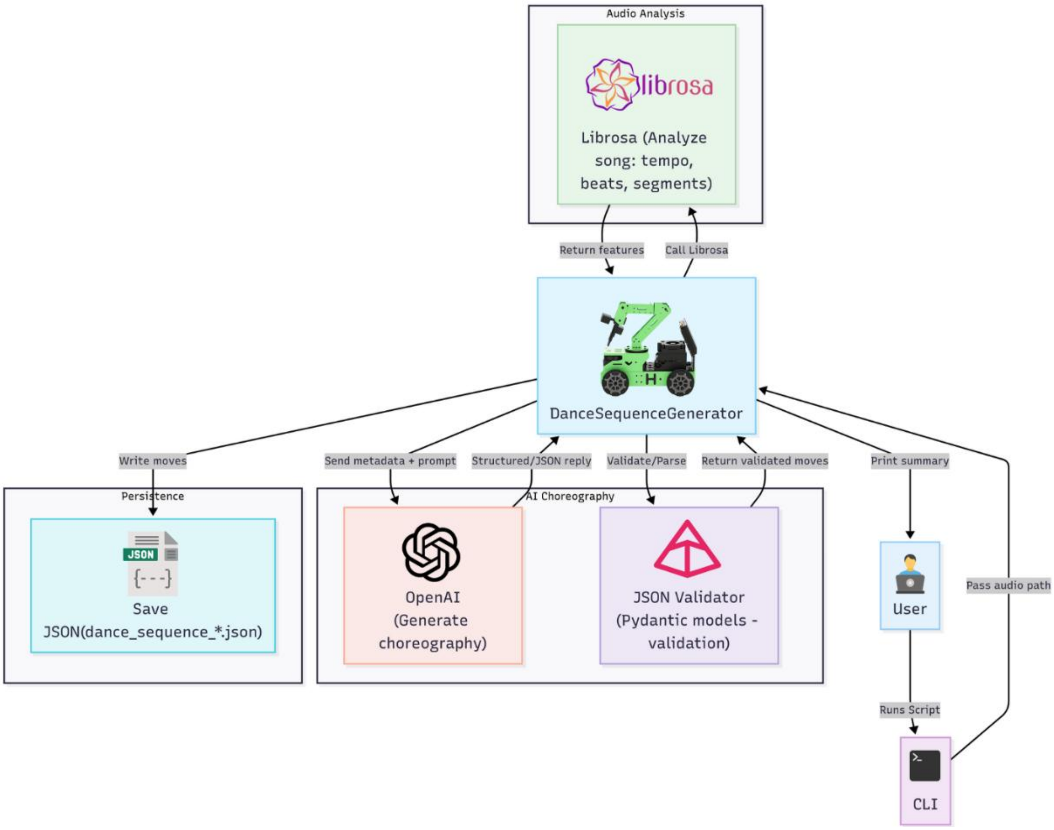


Figure 7: System architecture for LLM-based symbolic dance generation, where LLMs generate action tokens from lyrics.

The architecture (Figure 7) used musical analysis (e.g., beat and tempo extraction via Librosa) combined with lyric content to prompt the LLM, producing a structured sequence of high-level dance moves (action tokens). Each generated token corresponded to a proper move or pose, which was then mapped to parameterized motion commands (servo angles and wheel movements) for the robot. This approach aimed to leverage GPT-4.1's generative capabilities to craft choreographies in response to different lyrical themes and song segments.

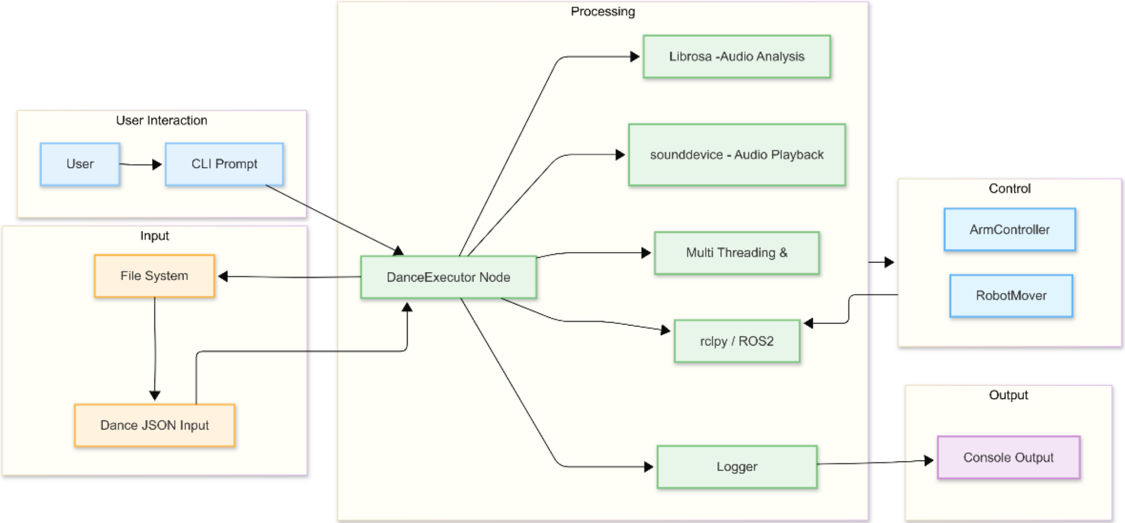


Figure 8: Execution architecture for mapping symbolic action tokens to robotic motion commands.

The generated dance tokens were executed via a choreography control node that synchronized music playback with robot motions. As shown in Figure 8, the system translated LLM-generated symbolic moves into real-time motor commands using ROS2, controlling both arm and wheel actions in sync with the audio. However, this LLM-based method lacked diversity and context sensitivity, the model frequently repeated similar moves regardless of lyrical changes. This limited variation resulted in monotonous choreography, reducing the expressive value intended through lyric-driven motion. Due to these shortcomings, the subsystem was not deployed on the physical robot. Instead, we continued with the more effective music-aware motion intelligence pipeline described earlier. That approach, detailed in Section 4.3, responds directly to beat and intensity features from the audio, enabling more dynamic and contextually relevant performances. This contrast will be explored further in the evaluation section.

4.5 System Evaluation and Results

To evaluate the choreography pipeline, we tested both the Music-Aware Motion Intelligence system and the LLM-based effective dance generation approach on the same AI-generated multilingual song (~140 seconds). This ensured that comparisons were drawn under identical musical conditions, as validated by the matching end timestamps in both sets of plots.

4.5.1 Music Aware Motion Intelligence (Deployed Model)

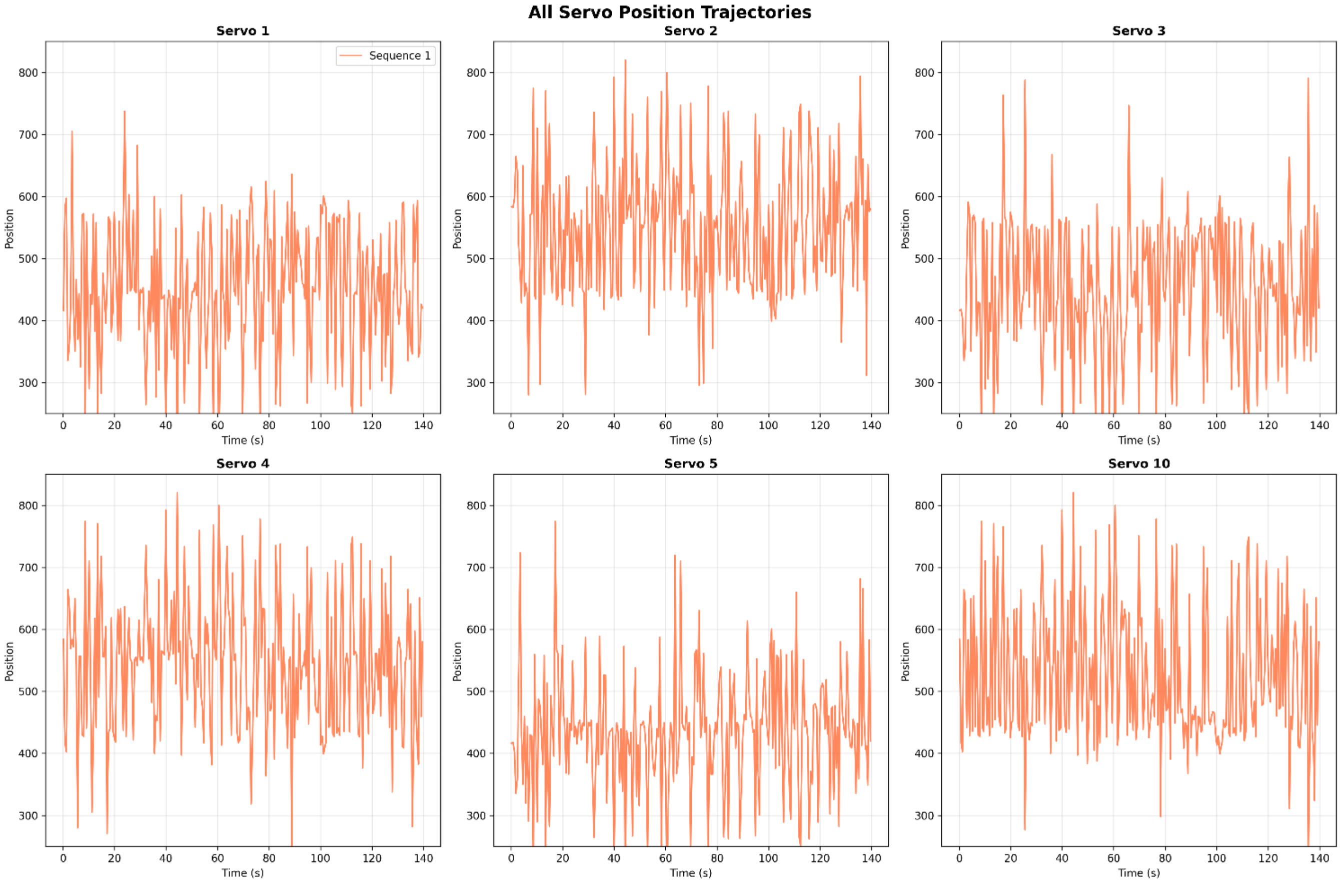


Figure 9: Multiple panel plot showing all six servo position trajectories over time (0–140 s) for the deployed Music-Aware Motion Intelligence system. Each panel corresponds to one servo, indicating how every joint moves continuously in response to the music.

Plots in Figure 9 shows servo position trajectories for six active servos. Each joint exhibits diverse motion patterns, some oscillate rapidly, others gradually sweep across a wide range confirming that the robot produces non-repetitive, expressive gestures throughout the performance.

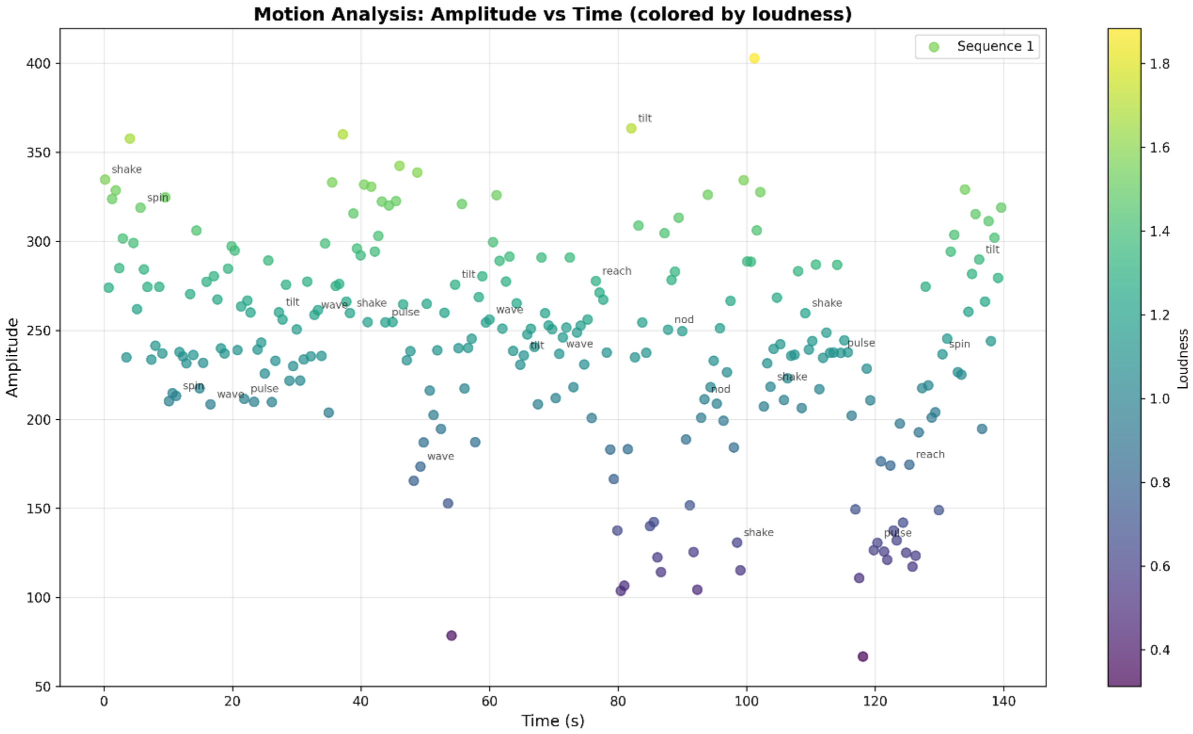


Figure 10: Scatter plot of the robot's motion amplitude vs. time for the Music-Aware approach, with each motion event labelled (e.g., wave, tilt, shake) and points coloured by the music loudness at that moment.

Visualization above offers a temporal view of the robot's motion amplitudes, put over with motion labels and colored by corresponding audio loudness. Strong correlations emerge tilt and spin gestures are more prominent in louder sections (yellow-green), while nod or pulse dominate during quieter phases (blue-purple). This alignment demonstrates the system's ability to modulate motion intensity in sync with the music, achieving context-sensitive choreography with dynamic variation.

4.5.2 LLM-Based Effective Motion Generator (Simulated)

The LLM-based choreography system was tested under two distinct controlled settings to evaluate its responsiveness and generative creativity. In the first configuration, we used a minimally engineered prompt with a low temperature setting (0.3), aiming for deterministic, structured outputs. In the second, we applied heavily prompt-engineered inputs including song context, mood features, and dance intent with a higher temperature (0.7) to produce more creative motion sequences.

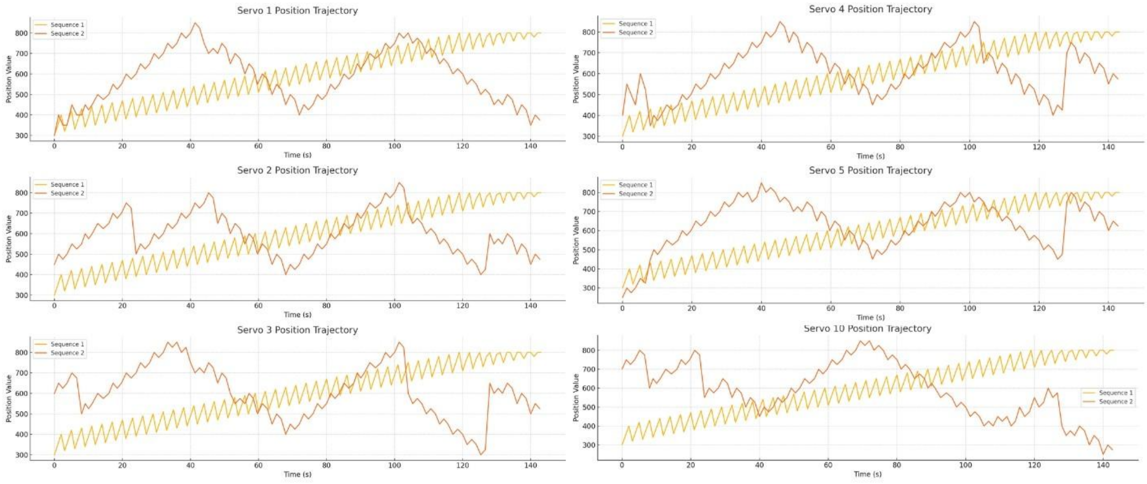


Figure 11: LLM-based output, which progresses monotonically with minimal motion diversity. Comparative servo position trajectories across six joints for both choreography approaches using the same song and pipeline.

Despite these variations, the generated output in both scenarios exhibited highly repetitive and uniform motion patterns which are incapable to be assumed as a dance movement. As seen in Figure 11, servo trajectories followed linear, sawtooth-like paths across all joints, lacking variation in both speed and form. Even with detailed prompts, the LLM tended to reissue similar symbolic actions at regular intervals, independent of song dynamics. Due to the absence of expressive variance and contextual awareness, the LLM-generated motions were not deployed on the robot. Though functional in simulation, the approach lacked the nuance, energy, and timing necessary for compelling dance performance specially when compared to the real-time responsiveness of the Music-Aware system.

4.6 Challenges and Weaknesses

Several challenges were encountered including Incomplete robot ROS launch required substantial reverse engineering and reimplementation of core control functions like the LiDAR. Hardware latency and responsiveness limitations necessitated tuning of motion speeds and synchronisation intervals to maintain alignment with the audio.

Due to these constraints, the current system focuses on beat-driven synchronisation; deeper semantic and emotional alignment remains as future work. Moreover, advanced approaches like 3D human motion mapping via the EDGE model were considered but not pursued due to their computational demands, which exceeded the JetRover's on-board capabilities.

4.7 Retrospective Insights

In hindsight, building a solid robot control layer earlier would have streamlined the entire project. Much time was spent resolving low-level motor, servo behaviors and other peripherals, delaying choreography refinement.

A better strategy would have followed a clearer staged pipeline:

- Start with validating motor and servo control
- Then build in rhythm-based choreography logic
- Finally, experiment with emotion and semantic enhancements

Additionally, although LiDAR-based safety was implemented and worked independently, it failed to run smoothly alongside music playback likely due to threading or resource conflicts. While this is likely solvable with improved synchronization or ROS2 executor configuration, time constraints and limited access to robot deployment prevented a fix before the final testing window.

4.8 New Research Directions

During this project, several new research avenues emerged:

- **Emotion-aware choreography-** Extending the pipeline to extract emotional tone from AI-generated music and map it to motion style and dynamics.
- **Phonetic-aware motion mapping-** Leveraging linguistic patterns to inform gestures aligned with lyrical content.
- **Environment-aware adaptive dance-** Integrating depth camera feedback to allow robots to adapt choreography dynamically in live performance spaces.
- **Integrating a Proper User Interface-** Designing an intuitive graphical user interface (GUI) that abstracts away command-line operations, enabling non-technical users to upload music, configure choreography options, and launch dance routines with a single click. This direction focuses on democratizing access to robotic performance systems, making them operable by artists, educators, or general users without requiring shell or command line expertise.

These directions are significant for advancing robotic performance from simple rhythm synchronisation toward truly expressive, context-sensitive dance systems, an exciting frontier in AI-robotics interaction.

5 Individual Contribution and Reflection

5.1 Gishor Thavakumar

Area	Contribution
Control System	Designed and implemented a modular Python-based control system for JetRover (arm + wheel).
Integration	Integrated control with beat-driven logic and audio pipeline; collaborated with Gopi (audio) and Karthik (motion mapping).
LiDAR Safety	Successfully implemented LiDAR-based obstacle detection; music playback conflicted due to threading limitations, unresolved due to time and hardware access.
Experimental Trials	Initiated and led: **1.** LLM-based choreography (action tokens) **2.** EDGE-based 3D human-to-rover motion translation. Both ideas and developed by contributor.
Documentation	Authored core technical sections in the report and assisted with reviewing and editing the report; created architectural, sequence, and data schema diagrams.
Project Planning	Contributed to research methodology, project timeline, and early literature review (RL and adaptive performance).
Evaluation	Conducted system evaluation and model comparisons across all approaches.
Music-Aware System	Assisted in logic-layer design and implementation; full architecture developed by Karthik.
Reflective Impact	The project enhanced the ability to design, debug, and deploy real-time AI-robotic systems, while bridging creative choreography with technical feasibility. It highlighted the value of experimental thinking, system integration, and adaptive problem-solving in real-world robotics.

5.2 Karthik Narayan Venkatasubramanian

Area	Contribution
Proposal Development	Drafted key sections of the initial proposal: Aims, Background, Research Problem, Context, Expected Outcomes, Benefits & Significance, and Conclusion. Played a critical role in shaping the project's research vision and technical scope.
AI-Generated Lyrics	Evaluated multiple LLMs (LLaMA(Grattafiori et al., 2024) , Mistral (Jiang et al., 2023), Falcon (Almazrouei et al., 2023), GPT-NeoX (Black et al., 2022)) for multilingual lyric generation. Identified limitations in coherence and emotional quality, which helped pivot the team to more capable commercial LLMs.
Music Generation	Explored the Suno music generation API and discovered API limitations (web-only access). Findings directly impacted the design and feasibility of the audio generation pipeline.
Music-Aware Motion Intelligence	Designed and rigorously tested architecture for mapping musical features to robotic movement. Iteratively improved system responsiveness and choreography accuracy based on real-world observations and beat-timing validations.
Pipeline Integration	Coordinated the alignment of AI-generated lyrics and music with motion logic modules. Collaborated with Gishor (robotic control) and Gopi (audio processing) to ensure a cohesive end-to-end pipeline.

Vision for Future Work	Proposed future enhancements, including emotion-aware choreography, semantic-driven movement from lyrics, and genre-adaptive motion profiles. Expressed interest in integrating vision feedback for dynamic and contextual performances.
Reflective Impact	Gained practical understanding of cross-modal AI integration with robotics. Demonstrated leadership in bridging generative AI output with physical motion systems, contributing to one of the project's core innovations.

5.3 Mir Sadia Afrin

Phase	Contribution
Proposal & Planning	Soley define the project's vision and Big Picture for AI-driven robotic choreography. Shaped the research problem, emphasizing multilingual and emotionally rich content. This contributed significantly to refining the presentation and final report structure and messaging.
AI Music & Lyric Pipeline	Led the development of the AI lyric and music generation pipeline. Evaluated multiple LLMs (GPT-2, GPT-3.5, GPT-4, GPT-4o, GPT-NeoX) for multilingual lyric generation, selecting GPT-4 as the most effective. Experimented with models like MusicGen-small (Copet et al., 2023), medium, melody, and So-VITS-SVC (Zhou et al., 2023) for audio, identifying their limitations in stability and control. Adopted Suno AI for high-quality song generation with vocals.
Technical Innovation	Refined prompt engineering and sampling strategies for better lyric-music alignment.
Documentation & Support	Maintained project documentation and supported report writing during all phases. Played an active role in ideation for presentations and assisted in shaping narratives for formal submission.
System Integration	Collaborated with Gopi (audio processing) and Karthik (motion mapping) to align AI-generated content with beat-to-motion execution. Contributed during live testing loops to ensure timing consistency between the music and robotic outputs.
Reflective Impact	Gained experience in bridging creative AI outputs with embodied robotics. This role deepened insight into applying generative models for physical interaction and performance, guiding future interest in emotion-aware and genre-specific dance generation.

5.4 Gopi Abhiram Vishal Chongala

Phase	Contribution
Proposal & Planning	Led the formulation of the project's research problem, focusing on enabling dynamic synchronization of robot motion with AI-generated multilingual music. Identified limitations in existing pre-scripted choreography systems. Contributed to significance and impact framing.
Audio Processing Pipeline	Developed a Python-based audio processing pipeline using Librosa to extract tempo, beat timings, and pitch. Structured outputs into JSON compatible with ROS-based systems. This enabled music data to drive choreography. Integrated visualization tools to verify alignment.
Robot Feasibility & Testing	Assisted in evaluating hardware feasibility and participated in testing the robot's responsiveness to audio features. Supported ROS control analysis and joint-level actuation testing.
System Integration	Collaborated with Karthik (motion mapping) and Gishor (robot control) to ensure audio features were accurately translated into robot choreography. Contributed during synchronization tests and performance tuning.
Music Tool Evaluation	Assessed compatibility of tools like Mureko AI and AudioCraft for the audio pipeline. Provided insights into how tempo and song structure influence robot motion.

Reflective Impact	Looking ahead, interested in extending the current pipeline to support emotion-aware motion mapping, where extracted emotional tone and lyrical content can drive more expressive and adaptive choreography. Also interested in real-time performance adaptation, allowing the robot to respond dynamically to audience interaction or environment.

6 Conclusion and Future Outlook

The project initially set out to develop a real-time robotic dance system capable of performing to AI-generated multilingual music. The intended research problems included:

- Generating lyrics and music in English, Hindi and Tamil using LLMs and generative audio models
- Mapping musical features (rhythm, emotion, phonetics) to robotic motion
- Demonstrating synchronised dance on JetRover platforms

During development, practical challenges particularly around hardware limitations and control integration led to an important reframing of scope. The project shifted focus toward building a robust, modular control framework and achieving beat-driven synchronisation as a foundation for future extensions.

6.1 Key Achievements

Despite hardware constraints, the project achieved meaningful progress:

- **End-to-End Integration:** Built a complete AI-to-robot pipeline, from lyric generation to audio, beat feature extraction, and ROS2-based robotic choreography.
- **Control Architecture:** Developed a modular Python-ROS2 system for coordinating JetRover's wheels and arm, driven by musical features.
- **Real-Time Demo:** Successfully executed beat-synchronized choreography on the robot.
- **Experimental Trials:** Explored LLM-driven choreography and EDGE-based human motion mapping as alternative pipelines.

These results demonstrate the potential of fusing generative AI and robotics to create intelligent, creative systems.

6.2 Limitations & Future Directions

Challenges included time constraints and system complexity:

- Emotion-based and phonetic choreography modules were not fully implemented.
- Environmental adaptation via LiDAR was only partially achieved due to driver conflicts.
- Multi-robot coordination was out of scope for this cycle.

Future Opportunities:

- Develop emotion-aware and phonetic-aware motion mapping.
- Integrate real-time vision or LiDAR feedback for adaptive choreography.
- Scale to multi-robot performances.
- Implement a user-friendly GUI for wider accessibility.

Final Reflection: This project built a strong foundation for future AI-driven robotic performances by integrating music, choreography, and real-time control in a unified system.

Bibliography

Ahn, H. (2024). May the Dance be with You: Dance Generation Framework for Non-Humanoids. *arXiv [cs.CV]*. http://arxiv.org/abs/2405.19743

Almazrouei, E., Alobeidli, H., Alshamsi, A., Cappelli, A., Cojocaru, R., Debbah, M., Goffinet, É., Hesslow, D., Launay, J., Malartic, Q., Mazzotta, D., Noune, B., Pannier, B., & Penedo, G. (2023). The Falcon Series of Open Language Models. *arXiv [cs.CL]*. http://arxiv.org/abs/2311.16867

Black, S., Biderman, S., Hallahan, E., Anthony, Q. G., Gao, L., Golding, L., He, H., Leahy, C., McDonell, K., Phang, J., Pieler, M. M., Prashanth, U. S., Purohit, S., Reynolds, L., Tow, J., Wang, B., & Weinbach, S. (2022). GPT-NeoX-20B: An Open-Source Autoregressive Language Model. *arXiv*, *abs/2204.06745*.

Copet, J., Kreuk, F., Gat, I., Remez, T., Kant, D., Synnaeve, G., Adi, Y., & D'efossez, A. (2023). Simple and Controllable Music Generation. *arXiv*, *abs/2306.05284*.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., Yang, A., Fan, A., Goyal, A., Hartshorn, A., Yang, A., Mitra, A., Sravankumar, A., Korenev, A., Hinsvark, A., . . . Ma, Z. (2024). The Llama 3 Herd of Models. *arXiv [cs.AI]*. http://arxiv.org/abs/2407.21783

Guan, H., Wei, X., Long, W., Yang, D., Zhai, P., & Zhang, L. (2024, 28-30 Oct. 2024). MDRPC: Music-Driven Robot Primitives Choreography. 2024 IEEE 36th International Conference on Tools with Artificial Intelligence (ICTAI), https://doi.org/10.1109/ICTAI62512.2024.00111

Hou, T., Zhang, Y., Wei, X., Dong, Z., Yi, J., Zhai, P., & Zhang, L. (2025). Music-Driven Legged Robots: Synchronized Walking to Rhythmic Beats. *arXiv [cs.RO]*. http://arxiv.org/abs/2503.04063

Ji, S., Wu, S., Wang, Z., Li, S., & Zhang, K. (2025). A Comprehensive Survey on Generative AI for Video-to-Music Generation. *arXiv [eess.AS]*. http://arxiv.org/abs/2502.12489

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. L., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. e. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., & Sayed, W. E. (2023). Mistral 7B. *arXiv*, *abs/2310.06825*.

Liu, F., Chen, D.-L., Zhou, R.-Z., Yang, S., & Xu, F. (2022). Self-Supervised Music Motion Synchronization Learning for Music-Driven Conducting Motion Generation. *Journal of Computer Science and Technology*, *37*(3), 539-558. https://doi.org/10.1007/s11390-022-2030-z

Santiago, C., Oliveira, J., Reis, L., & Sousa, A. (2011). *Autonomous robot dancing synchronized to musical rhythmic stimuli*.

Sekkat, H., Tigani, S., Saadane, R., & Chehri, A. (2021). Vision-Based Robotic Arm Control Algorithm Using Deep Reinforcement Learning for Autonomous Objects Grasping. *Applied Sciences*, *11*(17), 7917. https://www.mdpi.com/2076-3417/11/17/7917

Tseng, J., Castellon, R., & Liu, C. K. (2023, 17-24 June 2023). EDGE: Editable Dance Generation From Music. 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), https://doi.org/10.1109/CVPR52729.2023.00051

Wang, Z., Jia, Y., Shi, L., Wang, H., Zhao, H., Li, X., Zhou, J., Ma, J., & Zhou, G. (2024, 14-18 Oct. 2024). Arm-Constrained Curriculum Learning for Loco-Manipulation of a Wheel-Legged Robot. 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), https://doi.org/10.1109/IROS58592.2024.10802062

Zhou, Y., Chen, M., Lei, Y., Zhu, J., & Zhao, W. (2023). VITS-Based Singing Voice Conversion System with DSPGAN Post-Processing for SVCC2023. *2023 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 1-8.