Computer Modeling in Engineering & Sciences

Tech Science Press

Doi:10.32604/cmes.2026.074687

ARTICLE

Multimodal Trajectory Generation for Robotic Motion Planning Using Transformer-Based Fusion and Adversarial Learning

Shtwai Alsubai[1], Ahmad Almadhor[2], Abdullah Al Hejaili[3], Najib Ben Aoun[4,5,*], Tahani Alsubait[6] and Vincent Karovič[7,*]

[1]College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, AlKharj, 16273, Saudi Arabia
[2]Department of Computer Engineering and Networks, College of Computer and Information Sciences, Jouf University, Sakaka, 72388, Saudi Arabia
[3]Faculty of Computers & Information Technology, Computer Science Department, University of Tabuk, Tabuk, 71491, Saudi Arabia
[4]Faculty of Computing and Information, Al-Baha University, Alaqiq 65779-7738, Saudi Arabia
[5]REGIM-Lab: Research Groups in Intelligent Machines, National School of Engineers of Sfax (ENIS), University of Sfax, Sfax, 3038, Tunisia
[6]Department of Computer Science and Artificial Intelligence, College of Computing, Umm Al-Qura University, Makkah, 21955, Saudi Arabia
[7]Department of Information Management and Business Systems, Faculty of Management, Comenius University Bratislava, Odbojárov 10, Bratislava, 82005, Slovakia

*Corresponding Authors: Najib Ben Aoun. Email: najib.benaoun@ieee.org; Vincent Karovič. Email: vincent.karovic6@fm.uniba.sk

Received: 16 October 2025; Accepted: 05 January 2026; Published: 26 February 2026

ABSTRACT: In Human–Robot Interaction (HRI), generating robot trajectories that accurately reflect user intentions while ensuring physical realism remains challenging, especially in unstructured environments. In this study, we develop a multimodal framework that integrates symbolic task reasoning with continuous trajectory generation. The approach employs transformer models and adversarial training to map high-level intent to robotic motion. Information from multiple data sources, such as voice traits, hand and body keypoints, visual observations, and recorded paths, is integrated simultaneously. These signals are mapped into a shared representation that supports interpretable reasoning while enabling smooth and realistic motion generation. Based on this design, two different learning strategies are investigated. In the first step, grammar-constrained Linear Temporal Logic (LTL) expressions are created from multimodal human inputs. These expressions are subsequently decoded into robot trajectories. The second method generates trajectories directly from symbolic intent and linguistic data, bypassing an intermediate logical representation. Transformer encoders combine multiple types of information, and autoregressive transformer decoders generate motion sequences. Adding smoothness and speed limits during training increases the likelihood of physical feasibility. To improve the realism and stability of the generated trajectories during training, an adversarial discriminator is also included to guide them toward the distribution of actual robot motion. Tests on the NATSGLD dataset indicate that the complete system exhibits stable training behaviour and performance. In normalised coordinates, the logic-based pipeline has an Average Displacement Error (ADE) of 0.040 and a Final Displacement Error (FDE) of 0.036. The adversarial generator makes substantially more progress, reducing ADE to 0.021 and FDE to 0.018. Visual examination confirms that the generated trajectories closely align with observed motion patterns while preserving smooth temporal dynamics.

KEYWORDS: Multimodal trajectory generation; robotic motion planning; transformer networks; sensor fusion; reinforcement learning; generative adversarial networks

1 Introduction

Multimodal HRI is an effort to close the divide between human and robot communication by empowering robots to interpret and react to a variety of sensory signals in a human-like way, i.e., multimodal HRI [1] that seeks to address the gap between human communication and robotic execution by enabling robots to detect and respond to a variety of sensory signals in a human-like way. Recent developments in deep generative models and transformer-based architectures have considerably enhanced robots' capabilities to extract, interpret, and combine heterogeneous information from speech, vision, gesture, and motion cues and signals in the environment, thereby generating personalized responses to queries or scenarios to date [2]. Intelligent robotic systems have been seen as extending their capabilities into dynamic, interactive environments, where anticipatory motion planning and safety are essential. Nevertheless, the successful conversion of multimodal perception into robot motion is hindered by uncertainty, time constraints, and the physical feasibility of routes. These issues will necessitate robust, interpretable solutions for implementing robotic systems in real-world human-centered environments.

Despite these developments, current multimodal motion planning systems still lack several significant areas. Most designs are based on unimodal or weakly coupled fusion modes that do not exploit complementary information across sensory streams, leading to brittle or coarse direction forecasts [1,3]. Furthermore, most of the planning (learning-based) planners are not interpretable and do not have formal guarantees, and therefore generate trajectories that might not satisfy the constraints of time logic or physical feasibility. Even more problematic is the lack of data in HRI practice, which limits the accuracy of purely data-driven models.

This paper addresses multimodal trajectory control for robotic movement plans, with a focus on HRI scenarios. The prediction of the future trajectory of the robot is done by combining heterogeneous perceptual cues, vision, gesture, speech and previous trajectory information, using the model. Using Transformer-based convergence and adversarial learning, the model generates physically and semantically plausible, smooth, and interpretable trajectories, enabling safe and efficient trajectory planning in complex real-world settings. To address these problems, this paper proposes multimodal trajectory generation for robotic motion planning in HRI settings. The model presented unites heterogeneous perception inputs from vision, gesture, speech, and historical trajectory into a single framework via a Transformer-based fusion model with adversarial learning. LTL is used as an interpretable intermediate representation to encode task goals and time constraints, supporting formal reasoning and semantic grounding in trajectory generation. Moreover, pretrained foundation models are used to transfer semantic knowledge across large-scale corpora to mitigate data scarcity without compromising domain relevance [4,5]. Recognizing that human gestures and motion patterns exhibit complex spatiotemporal dependencies, we design custom encoder-decoder architectures that ensure kinematic feasibility and physical realism during trajectory generation. Furthermore, transformer-based multimodal attention mechanisms enable concurrent reasoning over multiple sensory inputs, significantly improving engagement prediction and intent understanding compared to sequential or late-fusion approaches [6].

This paper makes the following contributions:

- We provide a single multimodal data-processing pipeline that integrates voice characteristics, gesture keypoints, visual observations, robot trajectories, and symbolic logic into a time-synchronized representation. The pipeline uses metadata to keep all modalities in sync and facilitates batch processing during training. This architecture provides transformer models with a robust, practical foundation for multimodal learning.
- We use a dual learning architecture that lets us conduct both staged and reasoning. In the staged pipeline, many types of human input are first converted into LTL expressions that conform to the grammar.

These expressions are then converted into robot trajectories, thereby facilitating the understanding and verification of task execution. A generator also directly transfers symbolic intent and linguistic context to continuous robot motion. This provides more options when you have diverse job requirements or deployment constraints.

- We develop a transformer-based paradigm that maps symbolic task intent to continuous robot trajectories. Testing on the NATSGLD dataset indicates that this baseline gets an ADE of 0.040 and a FDE of 0.036 in normalised coordinates. This means that the trajectory prediction is accurate and steady.
- We also develop an adversarial trajectory generator that employs an autoregressive transformer decoder and incorporates smoothness and velocity constraints. Introducing an adversarial discriminator yields more realistic motion and improved spatial accuracy, reducing displacement errors to an ADE of 0.021 and an FDE of 0.018, representing a clear improvement over the staged baseline.
- The framework includes explicit motion regularization via velocity and acceleration penalties to promote smooth, physically feasible trajectories. Visual inspection of the generated motions shows fewer oscillations and better temporal consistency, supporting safe and executable robot behavior.
- Both quantitative and qualitative assessments indicate consistent training behaviour across all components. The training loss decreases steadily, while ADE and FDE increase steadily; the predicted trajectories are close to the ground truth. These findings support the effectiveness of combining symbolic reasoning with transformer-based multimodal fusion and adversarial regularization for robust human–robot interaction.

The literature on multimodal fusion and HRI is discussed in Section 2. Section 3 describes the NATSGLD dataset, comprising three participants, and a DataFrame architecture for integrating diverse data types. Section 4 presents our bidirectional instruction-logic-trajectory architecture, which features a multimodal fusion transformer and models. Section 5 outlines training methods, evaluation metrics and hyperparameters. Finally, Section 8 concludes the paper.

2 Related Work

In [7], the authors introduced TransFuser, a multimodal fusion transformer for fusing sensor data to enable autonomous driving. Existing geometry-based sensor fusion approaches are limited in their ability to handle non-standard driving scenarios that require consideration of global context. TransFuser employs a transformer-based architecture that integrates image and LiDAR data with attention mechanisms to capture the necessary spatial and temporal relationships to guide navigation in dynamic environments. Experimental results indicate that traditional imitation learning is ineffective in high-density scenarios, whereas TransFuser achieves high performance by capturing the global context of a scene. In [8], the authors introduce Motion Planning Transformers, which use transformer architectures to improve sampling-based motion planning for mobile robots. This approach minimises the search space by leveraging attention mechanisms and addresses the shortcomings of convolutional neural networks, particularly in non-holonomic robotic systems. It is helpful for motion planning of 2D and complex non-holonomic robots, learning valid paths based on previous trajectory-tracking data.

In [9], the authors presented an example of deep learning-based mission-conditioned path planning that integrates transformer variational autoencoders with LTL requirements. They combine mission-specific LTL formulas with transformer networks to produce compliant controller sequences, trained on an optimal-trajectory distribution using a Conditional Variational Autoencoder (CVAE). Authors in [10] take a step forward in the relation between formal specification languages and neural trajectory generation. The authors explore transfer learning approaches for motion transformer architectures in autonomous driving across various vehicle types and geographic locations. They highlighted the effects of domain changes arising

from sensor settings, algorithms, and traffic policies, and suggested transferring pre-trained models with minimal retraining while accounting for computational trade-offs. In [11], the authors proposed PASTEL, a framework that integrates autoregressive transformers with Signal Temporal Logic (STL) requirements for safe trajectory generation. The method also uses a cross-attention mechanism, in which specification embeddings guide trajectory generation via state-action representations. STL codes are encoded using advanced tokenizers to handle complex, safety-critical tasks. The presented work also contributes to the integration of formal verification tools with neural trajectory generation. The authors of [12] use TS-TrajGen, a two-stage generative adversarial architecture that combines model-free learning with domain knowledge to predict human mobility. The model addresses continuity concerns in trajectories by integrating an A* generator with a sequential, mobility-constrained realism discriminator, which proves successful on a complex road network.

In [13], the authors proposed "Gan2CS", which is a generative adversarial framework used to generate robotic calligraphy motions based on the input image. They synthesize hierarchical motions using a hierarchical motion synthesis method that integrates inverse and forward kinematics with visual inputs, without altering the style attribute. The authors of [14] discussed the shortcomings of technologies that predict a person's trajectory from historical data and proposed a hybrid transformer-Generative Adversarial Networks (GAN) framework to predict pedestrian paths in congested areas. Their model captures long-range spatial dependencies with multi-head attention and accounts for pedestrian interactions via a discriminator, thereby improving trajectory realism. In [15], the authors proposed that LSTM- and GAN-based models are used to generate unimodal and multimodal trajectories of humans. The issue of generalization in human motion prediction is resolved. To minimize biases in datasets, they introduce normalized evaluation metrics and show that multimodal GAN models are more effective than unimodal models at capturing uncertainty and illustrating a variety of plausible futures. The authors in [16] introduce customized GAN architectures to learn the inverse kinematics and dynamics of robotic manipulators with limited data. Their methodology is successfully applied to nonlinear mapping between joint and end-effector positions of actual robotic systems, reduces overfitting through strategic sampling, and allows extrapolation to novel trajectories.

Recent studies on mobile robot localization (MRL) comprehensively review localization methods, both classical and intelligent, across indoor and outdoor settings, including the relationship between mobile robots and the Internet of Things (IoT) and the Intelligent Internet of Things (IIoT) [17]. The paper demonstrates that AI- and ML-enabled systems, in conjunction with mapping, classification, and 3D localization, enhance robustness and raise security, navigation, and obstacle-avoidance issues. The authors in [18] examine optimized mobile robot localization in a wireless sensor network using drone-assisted sensor fusion. Localization using an Extended Kalman Filter (EKF) is evaluated in a non-line-of-sight scenario, where conventional algorithms are not competitive. Improved Internet of Things (IoT) and Fifth Generation (5G)/Beyond Fifth Generation B5G communication support, including unmanned aerial vehicle (UAV), can enhance position, velocity, and tracking performance estimates. Table 1 provides a summary of the recent studies.

Table 1: Summary of related work, datasets, models, and key findings

Ref.	Dataset	Model/Technique	Key findings
[7]	CARLA, nuScenes	TransFuser Multi-Modal Transformer	Superior fusion in complex scenarios
[8]	Random mazes, robot paths	Motion Planning Transformer	Effective search space restriction

(Continued)

Table 1 (continued)

Ref.	Dataset	Model/Technique	Key findings
[9]	LTL specs, robot trajectories	Transformer VAE + CVAE	Mission-compliant trajectory generation
[10]	KITTI, Waymo, nuScenes	Motion Transformer Transfer	Domain adaptation trade-offs
[11]	Aircraft paths, STL constraints	PASTEL Transformer	Constraint-compliant planning
[12]	Urban mobility, road networks	TS-TrajGen Two-Stage GAN	Continuous trajectory generation
[13]	Calligraphy images	Gan2CS Motion Learning	Visual-to-motion synthesis
[14]	ETH, UCY pedestrian	GAN + Transformer	Enhanced crowded scene prediction
[15]	Retail, pedestrian datasets	LSTM + GAN variants	Multimodal superiority

3 Dataset and Data Processing

The NatSGLD dataset, also known as Natural Speech, Gesture, Logic, and Demonstration, is a multimodal HRI dataset designed to learn natural human communication across complex and everyday tasks in real-life situations [19]. The NatSGLD collection was conducted in a Wizard-of-Oz experimental paradigm, in which participants communicated with a robot they believed to be completely autonomous; this led them to generate spontaneous, realistic speech-gesture behavior. The dataset is a combination of four complementary modalities, including (i) speech, as transcribed natural language commands; (ii) gestures, as body poses and pointing actions that express tacit, context-dependent information; (iii) logic, with each multimodal command annotated with a ground-truth LTL formula that represents the intended task structure and temporal constraints; and (iv) demonstrations, provided as expert-teleoperated robot controllers that demonstrate appropriate task execution at the control level. NatSGLD comprises a set of multimodal commands collected from 18 participants who attended 11 activities related to food preparation, cooking, and cleaning, across 20 object categories and 16 object states. The data also contains rich, synchronized sensory data, including multi-view RGB videos, depth images, semantic and instance segmentation, robot joint states, object state transitions, and high-level annotations. NatSGLD is a multifaceted resource for studying multimodal instruction success, plan and intention recognition, and human-advisable reinforcement learning in natural HRI by unifying natural language, gestures, symbolic task representation, and execution-level demonstrations into a single framework.

3.1 NATSGLD Dataset Characteristics and Structure

The NATSGLD dataset has several fundamental structural limitations that significantly hinder its effectiveness for robust multimodal learning research. It consists of data from only three participants (P40, P49, P53) and features inconsistent data organization across different file types, including video files in MP4 format, pose estimation outputs in JSON (keypoint sequences with 18 joints in 2D coordinates), and robot state trajectories stored in NPZ-compressed format, which includes position and orientation parameters. The trajectory data is either truncated or extended to 50 timesteps, while the gesture sequences vary in length, necessitating arbitrary padding to a standard of 30 frames. This irregularity in the dataset architecture results in inconsistent temporal sampling rates across modalities and the absence of standardized duration constraints, potentially compromising the retention of important temporal dynamics. Furthermore, the dataset suffers from several critical flaws. There is no validation of cross-modal synchronization; the demographic

diversity is insufficient to support generalization claims; and there are no metrics for annotation reliability or inter-annotator agreement to validate the ground truth. Due to the dataset's complex organization, including compressed file formats and scattered data across multiple directory structures, substantial preprocessing is required. Additionally, the minimal number of participants renders statistical inference and model validation unreliable, thereby preventing meaningful scientific conclusions about multimodal HRI phenomena.

3.2 Raw Data Components: Videos, Keypoints, Metadata

The raw data consists of MP4 video files organized in standardized directory structures. These files provide visual context for HRIs. However, the current framework has a limitation: it extracts only the first frame of each video, failing to exploit the temporal visual information crucial for understanding dynamic gestural communication. To find participant-specific keypoint sequences, complex extraction processes are required. This involves navigating nested file systems and using dynamic pattern matching. The keypoint data is stored as JSON files within ZIP archives, containing pose estimation outputs with 18 anatomical landmarks represented as 2D coordinate pairs. The primary method for indexing cross-modal data associations is through metadata structured as CSV files. These files contain essential temporal alignment information, including start and end timestamps, participant identifiers (PIDs), session identifiers (SIDs), database sequence numbers (DBSNs), and speech transcriptions. However, there appears to be insufficient comprehensive metadata coverage, with some entries missing. This absence forces the exclusion of potentially valuable interaction episodes. Moreover, the framework employs try-catch error handling, which masks underlying data quality issues rather than implementing systematic validation procedures. The distributed file organization and the use of varying storage formats introduce considerable preprocessing overhead and may lead to data integrity issues. Significant limitations include the absence of data-quality metrics or completeness assessments, the lack of standardized temporal sampling rates across modalities, and insufficient documentation of annotation protocols or inter-rater reliability measures. All of these factors could support the scientific validity of the ground-truth labels.

3.3 Preprocessing Pipeline Design

The preprocessing pipeline follows a systematic, multi-phase process. It begins with metadata-driven cross-referencing to create consistent video identifiers across various data sources. Next, robot trajectory data are synchronized with interaction episodes, defined by start and end times extracted from CSV metadata, via temporal alignment using timestamp-based windowing functions. In the gesture keypoint processing phase, data is extracted from ZIP-compressed JSON files. Each temporal frame is mean-centered and scaled by its standard deviation, followed by standard padding or truncation to fixed 30-frame sequences. However, due to variability in natural gesture durations, this method may introduce artifacts or lose essential temporal information. For speech processing, the pipeline employs pre-trained transformer models (e.g., BART and Sentence Transformers) to generate dense semantic embeddings without requiring domain-specific fine-tuning. This may limit semantic accuracy for understanding robotic commands. To ensure dimensional consistency across state variables represented as 6-DOF, MinMaxScaler normalization is applied to the robot trajectory data. Several design flaws are evident in the pipeline: it utilizes sequential processing instead of parallel processing, leading to computational bottlenecks; it poorly handles errors by relying on try-catch blocks, which merely conceal data quality issues rather than implementing systematic validation protocols; and it lacks robustness testing or data augmentation techniques for missing or corrupted modal inputs. Because it relies on fixed sequence lengths and statistical normalization assumptions, the pipeline may not preserve the natural temporal dynamics and scale of interactions necessary for proper multimodal comprehension in HRI scenarios.

3.4 Gesture Keypoint Normalization Techniques

The preservation of crucial gestural information may be compromised by several methodological flaws and limitations in the gesture-keypoint normalization approaches employed in this framework. This frame-wise method overlooks global gesture factors and may eliminate critical spatial relationships that characterize coherent human movements. The implementation involves mean-centering and standardizing the data separately for each temporal frame. This process subtracts the spatial mean of all key points, divides by the standard deviation from the mean, and adds a small epsilon term (1×10^{-6}) to prevent division by zero. However, this standardization relies solely on statistical properties and fails to consider biomechanical constraints or variations in skeletal structure, which can influence the execution of natural gestures. As a result, the normalization strategy cannot maintain anatomical proportionality among participants or account for individual anthropometric differences. Moreover, selecting an arbitrary epsilon value without conducting sensitivity analysis, failing to verify whether normalized representations retain the semantic content of gestures, and not comparing the approach with other normalization techniques such as pose-invariant representations, joint-relative coordinates, or anatomically-informed scaling methods commonly used in the literature on human action recognition are significant technical issues. Additionally, the preprocessing pipeline employs reshaping operations that discard confidence information when converting 3D keypoint arrays (which include confidence scores) to 2D coordinate pairs. This could lead to unreliable pose predictions, thereby propagating errors throughout the subsequent multimodal fusion process. Furthermore, the fixed 30-frame padding strategy may introduce discontinuities or artifacts that obstruct the understanding of temporal gestures.

4 Proposed Methodology

The proposed architecture (Fig. 1) is a multimodal trajectory-generation framework that integrates heterogeneous sensory inputs into a coherent, temporally consistent representation. The raw data comprise video streams, gesture keypoints, robot paths, and associated metadata. The different modalities are initially passed through individual preprocessing pipelines. This pipeline performs gesture keypoint normalization, trajectory alignment, and temporal synchronization across all data sources. The aligned modalities are combined into a single multimodal dataframe after preprocessing. This guarantees the temporal consistency and integrity of contexts across inputs. The homogeneous data are then passed through domain- and modality-specific encoders. The encoded features are fused using a Transformer-based fusion module that employs multi-head attention to learn cross-modal dependencies and generate contextual embeddings. The model consists of two interconnected pathways. The first pathway performs instruction-to-logic transformation using LTL, which enables explicit modelling of semantic and temporal constraints. These logic representations are translated into robot motion trajectories that can be executed by a second pathway, which is a trajectory generation module. To make the trajectory look more realistic and physical, adversarial regularization was first used to promote human-like, smooth motion dynamics. The measurement of model performance is based on the ADE and FDE, which are used to measure the accuracy of the trajectory and continuity. Although multimodal fusion, LTL reasoning, and regularization mechanisms increase the architecture's complexity, the framework can converge to a stable state in a structured manner through constrained optimization and structured synchronization.

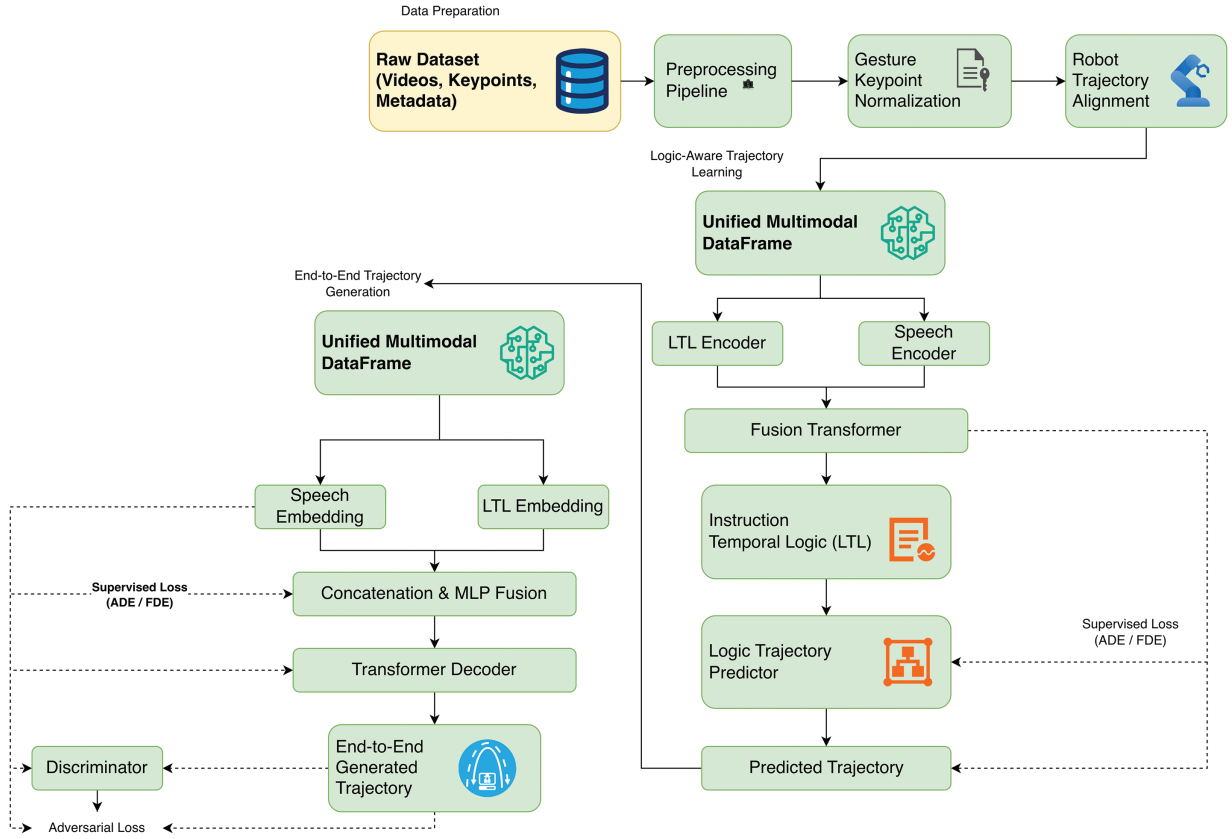


Figure 1: Proposed multimodal framework for HRI

Algorithm 1 presents a multimodal framework for trajectory development in robotic systems. It integrates various data types using specialized encoders and Transformer-based fusion architectures. The methodology employs a dual-pathway approach, converting natural-language instructions into LTL representations and employing direct production methods. By utilizing generator-discriminator networks for adversarial training, the system optimizes a composite objective function that combines the objectives of both the Generator and the discriminator. This function incorporates preset weighting factors to balance adversarial loss, reconstruction fidelity, and temporal consistency constraints. However, simultaneously optimizing the dual learning pathways and the adversarial components may necessitate extensive hyperparameter tuning and convergence analysis to ensure stable learning dynamics. Additionally, the system's architectural complexity can lead to computational overhead and potential instability during training.

Algorithm 1: Multimodal trajectory generation with transformer fusion and adversarial learning

Require: Raw dataset $\mathcal{D} = \{V, K, M\}$ where V = videos, K = keypoints, M = metadata
Require: Robot trajectory data $\mathcal{R} = \{r_1, r_2, \ldots, r_n\}$
Require: Natural language instructions $\mathcal{I} = \{i_1, i_2, \ldots, i_m\}$
Ensure: Generated trajectory $\mathcal{T}_{gen}$
1: // Phase 1: Data Preprocessing and Alignment
2: $V_{proc} \leftarrow \text{PreprocessVideos}(V)$
3: $K_{norm} \leftarrow \text{NormalizeKeypoints}(K)$
4: $R_{aligned} \leftarrow \text{AlignTrajectories}(\mathcal{R})$

(Continued)

Algorithm 1 (continued)

$\mathcal{DF}_{unified} \leftarrow$ CreateUnifiedDataFrame$(V_{proc}, K_{norm}, M, R_{aligned})$
// Phase 2: Multimodal Encoding
$E_v \leftarrow$ VideoEncoder(V_{proc}) {Video features}
$E_k \leftarrow$ KeypointEncoder(K_{norm}) {Gesture features}
$E_m \leftarrow$ MetadataEncoder(M) {Context features}
$E_r \leftarrow$ TrajectoryEncoder$(R_{aligned})$ {Motion features}
// Phase 3: Fusion Transformer
$F_{concat} \leftarrow$ Concatenate(E_v, E_k, E_m, E_r)
$F_{fused} \leftarrow$ FusionTransformer(F_{concat})
// Phase 4: Dual Learning Pathways
// Pathway 1: Instruction-to-Logic-to-Trajectory (LTL)
for each instruction $i_j \in \mathcal{I}$ **do**
    $L_j \leftarrow$ InstructionToLogic(i_j) {Convert to LTL}
    $T_{ltl,j} \leftarrow$ LogicToTrajectory(L_j, F_{fused})
end for
// Pathway 2: Generation
$T_{e2e} \leftarrow$ EndToEndGenerator$(F_{fused}, \mathcal{I})$
// Phase 5: Trajectory Decoding and Refinement
$T_{decoded} \leftarrow$ TrajectoryDecoder(T_{ltl}, T_{e2e})
// Phase 6: Adversarial Training
$G_\theta \leftarrow$ TrajectoryGenerator(θ)
$T_{fake} \leftarrow G_\theta(F_{fused}, \text{noise})$
// Discriminator Network
$D_\phi \leftarrow$ TrajectoryDiscriminator(ϕ)
$score_{real} \leftarrow D_\phi(T_{decoded})$
$score_{fake} \leftarrow D_\phi(T_{fake})$
// Adversarial Loss Computation
$\mathcal{L}_{adv} = \mathbb{E}[\log D_\phi(T_{real})] + \mathbb{E}[\log(1 - D_\phi(G_\theta(z)))]$
$\mathcal{L}_{recon} = \text{MSE}(T_{decoded}, T_{ground\_truth})$
$\mathcal{L}_{total} = \alpha \cdot \mathcal{L}_{adv} + \beta \cdot \mathcal{L}_{recon} + \gamma \cdot \mathcal{L}_{consistency}$
// Phase 7: Training Loop (Update Discriminator, Update Generator)
repeat
    $\phi \leftarrow \phi - \eta_D \nabla_\phi \mathcal{L}_{adv}$
    $\theta \leftarrow \theta - \eta_G \nabla_\theta \mathcal{L}_{total}$
    // Compute Evaluation Metrics
    $ADE \leftarrow$ AverageDisplacementError$(T_{decoded}, T_{ground_truth})$
    $FDE \leftarrow$ FinalDisplacementError$(T_{decoded}, T_{ground_truth})$
until convergence or max iterations reached
// Phase 8: Final Trajectory Generation
$\mathcal{T}_{gen} \leftarrow$ GenerateRealisticMotion$(T_{decoded}, G_\theta)$
return $\mathcal{T}_{gen}$

The proposed model is a systematic chain of interrelated elements. Raw multimodal inputs are first preprocessed, including video processing, keypoint normalization, and temporal trajectory alignment, to ensure consistent multimodal processing. This is followed by high-level representations obtained using

modality-specific encoders on all available data for vision, gesture, speech, and historical trajectory. The latter heterogeneous features are, in turn, fused using a Transformer-based fusion module that encodes cross-modal dependencies. The fused representation is then passed through two streams: a logic stream that transforms symbolic input into LTL representations, and a generative stream that predicts continuous trajectories. To increase realism and physical plausibility, adversarial learning is used with a discriminator-guided training objective. Lastly, trajectory quality is quantified using the standard ADE and FDE.

Below are the Hyperparameters used for the experiments:

$$\alpha = 0.3 \quad \text{(Adversarial loss weight)} \tag{1}$$

$$\beta = 1.0 \quad \text{(Reconstruction loss weight)} \tag{2}$$

$$\gamma = 0.1 \quad \text{(Consistency loss weight)} \tag{3}$$

$$\eta_D = 0.0001 \quad \text{(Discriminator learning rate)} \tag{4}$$

$$\eta_G = 0.0001 \quad \text{(Generator learning rate)} \tag{5}$$

The hyperparameters (shown above in Eqs. (1) to (5)) are selected based on empirical training stability, rather than via grid or Bayesian optimization. Given the relatively small dataset, hyperparameter tuning would have easily led to overfitting and unstable convergence. The loss weights, namely α and β, and, most importantly, the regularization weight via auxiliary objectives were selected to prioritize trajectory reconstruction while still providing some regularization. In particular, the reconstruction loss weight (0.8) is overridden by the geometric fidelity term, and the auxiliary weights (0.1, 0.1) are set to smaller values to prevent optimization instability. The choice of learning rates was based on popular sequence modeling and adversarial training. A smaller generator learning rate ($\eta G = 10^{-4}$) was used to promote easy convergence, and a somewhat higher discriminator learning rate ($\eta D = 2 \times 10^{-4}$) was used to address the vanishing gradients. The epoch-wise training records indicate stable convergence with these settings, as evidenced by monotonically decreasing losses and steady ADE/FDE trends. Although no extensive hyperparameter optimization was conducted in this study due to data constraints, this limitation is explicitly noted and does not affect the paper's qualitative findings.

4.1 Unified Multimodal Data Representation

Our methodology is based on a single multimodal data representation that integrates multiple data modalities, including robot trajectories, pose-based gesture keypoints, voice embeddings, visual observations, and symbolic logic descriptions. All modalities are organized into a single, time-aligned structure to ensure that data streams remain synchronized. Each sample has a unique identity that enables reliable linking of information across multiple modalities at the sequence level.

Before integration, each modality is processed independently using preprocessing steps appropriate to its structure. We normalise and rearrange gesture keypoints into a tensor format, and use Min–Max normalisation to scale robot trajectories so that their relative motion patterns remain the same. Using pretrained sentence-level encoders, speech inputs are turned into embeddings of a specified length. Task descriptions are represented using short token sequences taken from a predefined lexicon. Using fixed-length windows and sequence-level matching across modalities, temporal alignment is ensured. This architecture facilitates batch training and enables straightforward interfacing with transformer-based fusion and decoding modules.

4.1.1 Data Integration Strategy

Structured preprocessing and precise temporal alignment enable multimodal integration. Raw JSON files are used to dynamically load gesture keypoints, which are then padded or truncated to a fixed length to ensure compatibility with batch processing. Robot trajectories are extracted as relative displacements, smoothed to reduce noise, and normalised to improve stability during learning. Using pretrained phrase-transformer models, speech inputs are embedded, yielding rich semantic representations without further fine-tuning. Symbolic task intent is inferred from speech using a zero-shot semantic classifier and mapped to a predefined symbolic vocabulary. Across all modalities, padding, normalization, and consistent sequence lengths are enforced to avoid misalignment and prevent the introduction of artificial temporal artifacts. This strategy enables downstream fusion modules to operate reliably on synchronized multimodal inputs.

4.1.2 Feature Alignment Across Modalities

Using fixed sequence lengths and uniform tensor representations ensures temporal and dimensional consistency across modalities. To make them longer, gesture sequences are padded to 30 frames, robot trajectories to 50 time steps, and symbolic logic sequences to their maximum length. At the sequence level, trajectory data are synchronized with the accompanying voice and gesture inputs. This makes sure that the training is consistent. To enable effective multimodal fusion, voice embeddings are placed in the same latent space as symbolic and kinematic information. This alignment ensures that all modalities contribute meaningfully to the common representation while keeping computational load low and training behaviour consistent.

4.2 Encoder-Decoder Architecture

The system is configured as an encoder-decoder architecture. First, each modality is processed by its own encoder. This enables different types of inputs to be mapped into a common latent representation while preserving their distinct structures. Subsequently, transformer-based decoders take this representation and generate structured outputs, such as sequences of symbolic logic or continuous robot trajectories. Autoregressive transformer decoders with positional embeddings and masked self-attention are used to generate sequences. Teacher forcing is used during training to help the model converge, and autoregression is used during inference to depict how the model would work in real-world situations. This approach enables reliable translation between symbolic descriptions and continuous motion while maintaining consistency over time.

4.2.1 Modality-Specific Encoders

There is a separate encoder for each modality, designed to operate on its data format. Next, the learnt token embeddings are used to express symbolic logic. These embeddings are then integrated into little logical features. Convolutional backbones that have already been trained process visual inputs, which are subsequently projected via linear layers. Gesture keypoints are flattened, projected into a latent space, and then fed into transformer-based temporal encoders to capture motion-related relationships. Linear projections are used to represent robot trajectories from sequences of relative displacements. Pretrained Sentence Transformer embeddings are used to represent speech inputs, which are then projected onto the common latent space. All encoders produce representations with a set number of dimensions. This facilitates their combination at the fusion stage and ensures that the decoders can utilise them at all times.

4.2.2 Multimodal Fusion Transformer

A transformer encoder combines information from multiple modalities by applying it to embeddings specific to each modality. Before fusion, all embeddings are put into a common latent space. This enables multi-head self-attention to learn to handle diverse data types. The fusion module comprises stacked encoder layers that employ layer normalisation and dropout to improve training stability. Mean pooling is used to summarise the fused representation and create a small multimodal embedding. This embedding is then utilised to help with the decoding phases. Experimental results demonstrate that this fusion methodology captures the essential interactions required for both symbolic reasoning and trajectory generation while avoiding unnecessary model complexity.

4.3 Dual Learning Pathways

The system is built on two learning routes that operate well together. One connects symbolic task descriptions to robot motion that never stops. In one method, multimodal human inputs are first turned into symbolic logic representations. In the other, symbolic information and language context are directly linked to executable paths. This configuration enables the integration of symbolic reasoning and motion production, achieving a balance between interpretability and adaptability. The framework allows practitioners to choose between transparent task execution and higher-performance direct trajectory creation, depending on the application's requirements. This is because it supports both staged and learning.

4.3.1 Instruction-to-Logic-to-Trajectory Pipeline

In the staged pipeline, multimodal inputs are first fused to produce symbolic task representations encoded as constrained logic tokens. These intermediate symbolic descriptions explicitly capture task structure and temporal constraints, thereby facilitating inspection and verification. The logic-conditioned trajectory generator then maps symbolic intent and speech context into continuous robot motion using transformer-based decoding. This decomposition improves interpretability and enables the diagnosis of errors at the symbolic level before motion execution.

4.3.2 Trajectory Generation

The approach bypasses symbol decoding and instead constructs trajectories directly from voice embeddings and symbolic intent. Using an autoregressive transformer decoder with positional embeddings, trajectories are generated that stay consistent across 50 time steps. Training optimises repair loss while also imposing penalties for smoothness and speed to minimise sudden changes in motion. These limits favour physically possible paths with greater continuity. Both quantitative data and visual examination verify this.

4.4 Adversarial Training for Motion Realism

To further improve realism, an adversarial discriminator is introduced to distinguish between generated and real trajectories. The discriminator is implemented as a multi-layer perceptron operating on flattened trajectory sequences and provides an additional learning signal to the generator. The generator optimizes a composite objective that combines reconstruction loss, adversarial loss, and motion regularization terms. Generator and discriminator updates are alternated during training to maintain balanced adversarial learning and stable convergence.

4.4.1 GAN-Based Motion Refinement

Within the adversarial framework, the generator learns to match the distribution of robot motion while remaining consistent with task constraints. The discriminator penalizes unrealistic motion patterns, helping suppress artifacts such as jitter and drift. This interaction yields smoother, more realistic trajectories and consistently reduces displacement errors relative to non-adversarial training.

4.4.2 Training Stability Measures

Several techniques are used to stabilize adversarial training, including gradient clipping, dropout, and learning rate scheduling. During discriminator updates, generator outputs are detached to avoid gradient interference. Loss terms are carefully balanced to preserve accuracy while improving realism. These measures yield reliable convergence across runs, as evidenced by smooth loss curves and steady reductions in ADE and FDE.

4.5 Trajectory Forecasting and Motion Regularization

Trajectory generation is formulated as a forecasting problem, where future motion is predicted based on symbolic intent and linguistic context. This formulation anchors predictions in realistic motion dynamics rather than unconstrained sequence synthesis. Explicit penalties on velocity and acceleration encourage smooth temporal evolution and physically feasible motion. This approach improves stability and interpretability, making the generated trajectories suitable for safety-critical HRI scenarios.

4.6 Grammar-Constrained Symbolic Grounding

Symbolic task representations are constructed using a grammar-constrained formulation to ensure syntactic validity and interpretability. High-level intent is identified using a zero-shot semantic classifier and mapped to predefined logical templates composed of fixed temporal operators and atomic predicates. This deterministic construction avoids unconstrained logical decoding and guarantees that the resulting LTL expressions are valid. The resulting symbolic representations provide grounded, interpretable conditioning signals for trajectory generation, thereby improving robustness and transparency in human-robot interaction.

5 Experimental Setup

5.1 LTL Text Encoder

The LTL text encoder represents symbolic task specifications using a learned embedding layer with a fixed-dimensional latent space. Tokenized LTL expressions are embedded and aggregated to form a compact logical representation that preserves high-level temporal intent. This design prioritizes alignment between symbolic logic and downstream motion generation, enabling seamless fusion with continuous modalities such as speech embeddings and trajectories. In practice, this representation was sufficient to capture essential temporal operators and propositional structure, as evidenced by the high prefix accuracy and exact-match rates reported in the instruction-to-LTL translation results. While more expressive encoders could be explored in future work, the current design strikes a balance between representational adequacy and computational efficiency.

5.2 Gesture Encoder

Human gestures are encoded using a transformer-based temporal model with multi-head self-attention that operates on normalized keypoint sequences sampled from fixed-length temporal windows. This

architecture enables the model to capture both short-term motion dynamics and longer-range temporal dependencies inherent in human gestures. The use of self-attention allows the encoder to emphasize salient motion segments without relying on handcrafted phase segmentation. Empirically, the gesture representations contribute meaningfully to downstream multimodal fusion, supporting accurate logical inference and trajectory generation. Fixed-length padding is applied consistently across samples, which simplifies batching while preserving the dominant temporal structure of gestures observed in the dataset.

5.3 Multimodal Fusion Transformer

Multimodal fusion is achieved using a transformer encoder that integrates embeddings from visual observations, speech, gestures, trajectories, and logical representations. Each modality is projected into a shared latent space before fusion, enabling effective cross-modal interaction via self-attention. Unlike early fusion strategies, this design allows each modality to be encoded independently before integration, thereby improving robustness to noise and modality-specific variability. Experimental results demonstrate that this fusion mechanism enables reliable logical inference and trajectory prediction, as evidenced by stable training dynamics and low displacement errors. While the current implementation employs a compact two-layer transformer, it proved sufficient for capturing cross-modal dependencies in the evaluated tasks.

5.4 LTL-to-Trajectory Generation Model

The LTL-to-Trajectory generation model maps symbolic task intent into continuous robot motion sequences using a transformer-based decoder with learned positional embeddings. The model predicts relative trajectories autoregressively over a fixed temporal horizon, thereby capturing sequential dependencies and temporal consistency. To encourage physically plausible motion, the training objective includes reconstruction loss together with smoothness and velocity regularization terms. These constraints explicitly penalize abrupt changes in velocity and acceleration, thereby yielding smoother, more realistic trajectories. Quantitative evaluation using ADE and FDE demonstrates that the model achieves high spatial accuracy, while qualitative visualizations confirm coherent and executable motion patterns.

5.5 Adversarial Discriminator

To further enhance trajectory realism, an adversarial discriminator is introduced to distinguish between real and generated motion sequences. The discriminator operates on flattened trajectory representations and provides an additional supervisory signal that encourages the generator to produce trajectories consistent with the distribution of real human motion. Combined with reconstruction and smoothness losses, this adversarial objective improves trajectory sharpness and reduces accumulated drift over long horizons. Training stability is maintained through balanced generator–discriminator updates and learning rate scheduling, resulting in smooth convergence without evidence of mode collapse.

5.6 Architecture Evaluation

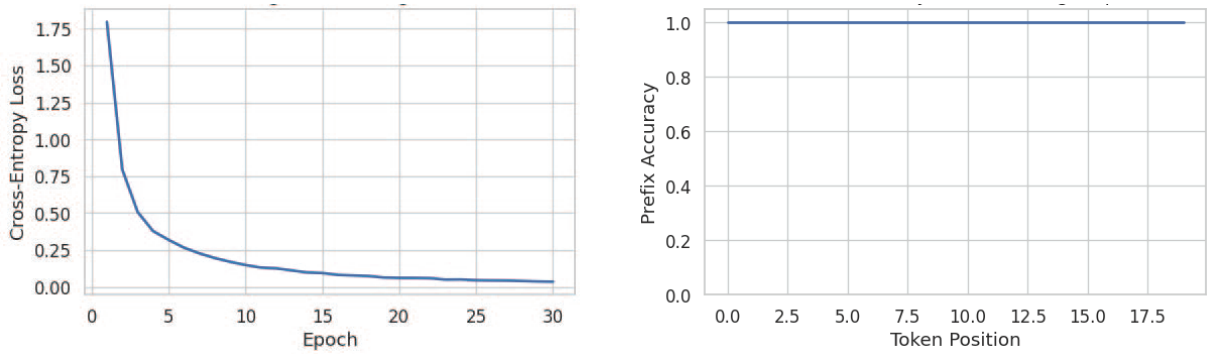
The complete architecture integrates multimodal perception, logical reasoning, and trajectory generation within a unified learning framework. The system is trained using a combination of reconstruction, adversarial, smoothness, and velocity losses, enabling joint optimization of accuracy and motion realism. Experimental results demonstrate stable convergence across all components, with the generator achieving significantly lower ADE and FDE than the staged baseline. Qualitative trajectory comparisons further confirm improved smoothness and alignment with ground-truth motion. While future work may explore additional baselines, extended temporal horizons, and physics-based constraints, the current experimental setup provides strong empirical evidence for the effectiveness of the proposed approach.

6 Results

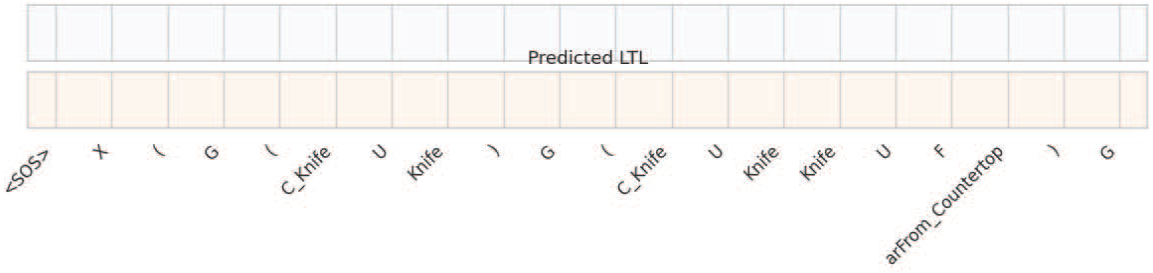
6.1 Multimodal Instruction-to-LTL Learning with Baseline Model

This section presents the proposed multimodal framework for translating human instructions into formal LTL specifications. The objective is to map heterogeneous human–robot interaction signals, including vision, gestures, object trajectories, and speech, into syntactically valid and semantically meaningful LTL formulas suitable for downstream verification and planning.

Fig. 2 summarizes the quantitative and qualitative performance of the proposed instruction-to-LTL model. As shown in Fig. 2a, the training cross-entropy loss decreases monotonically over 30 epochs, demonstrating stable convergence and effective optimization. Table 2 reports the corresponding epoch-wise loss values, further confirming consistent learning of both LTL syntax and multimodal semantics. At inference time, LTL formulas are generated autoregressively using greedy decoding. Decoding starts with a <SOS> token and terminates when either an <EOS> token is produced, or a predefined maximum length is reached, reflecting realistic deployment conditions. Model performance is evaluated using three complementary metrics: exact match accuracy, prefix accuracy, and a parenthesis balance score. Exact match measures full sequence correctness, prefix accuracy captures early-token reliability, and the balance score evaluates structural validity of generated formulas. Fig. 2b reports prefix accuracy across decoding steps, showing near-perfect correctness for early tokens. Qualitative analysis further confirms the effectiveness of the proposed approach. As illustrated in Fig. 2c, the model produces exact matches with perfectly balanced parentheses in multiple cases, even for deeply nested temporal expressions involving operators such as **G** (Globally), **F** (Eventually), and **X** (Next). In non-exact cases, errors typically arise from minor over-generation beyond the end-of-sequence token, while preserving structural validity and semantic intent.



(**a**) Training loss convergence over 30 epochs. (**b**) Prefix accuracy across decoding steps.



(**c**) Ground-truth vs. predicted LTL formulas.

Figure 2: Instruction-to-LTL learning results

Table 2: Epoch-wise training loss for the instruction-to-LTL model

Epoch	Loss	Epoch	Loss
1	1.7955	16	0.0826
2	0.7970	17	0.0785
3	0.5081	18	0.0749
4	0.3793	19	0.0644
5	0.3202	20	0.0622
6	0.2673	21	0.0619
7	0.2286	22	0.0604
8	0.1967	23	0.0504
9	0.1712	24	0.0514
10	0.1495	25	0.0462
11	0.1320	26	0.0446
12	0.1266	27	0.0438
13	0.1129	28	0.0413
14	0.0989	29	0.0379
15	0.0951	30	0.0363

6.2 Logic-to-Trajectory Generation Results

This section evaluates the Logic-to-Trajectory generation module, which translates symbolic task intent, represented through compact LTL-derived tokens and speech embeddings, into continuous two-dimensional motion trajectories. The objective is to assess whether high-level logical representations can be reliably grounded in physically meaningful, temporally coherent motion patterns. The model is trained with a mean-squared-error objective between predicted and ground-truth relative trajectories, optimized with Adam and early stopping based on the validation loss. Table 3 reports epoch-wise training and validation loss together with ADE and FDE. The results show a consistent reduction in both loss and displacement errors across epochs, confirming stable optimization and progressive refinement of trajectory predictions.

Table 3: Epoch-wise training and validation performance for Logic-to-Trajectory generation

Epoch	Train loss	Val loss	ADE (m)	FDE (m)
1	0.1069	0.0137	0.1519	0.1895
2	0.0307	0.0016	0.0505	0.0569
3	0.0214	0.0004	0.0245	0.0220
4	0.0173	0.0005	0.0293	0.0128
5	0.0143	0.0007	0.0342	0.0284
6	0.0118	0.0003	0.0206	0.0288
7	0.0097	0.0002	0.0183	0.0155
8	0.0084	0.0004	0.0250	0.0207
9	0.0074	0.0002	0.0174	0.0127
10	0.0066	0.0001	0.0132	0.0078
11	0.0057	0.0001	0.0131	0.0136
12	0.0049	0.0002	0.0179	0.0168
13	0.0044	0.0001	0.0121	0.0088

(Continued)

Table 3 (continued)

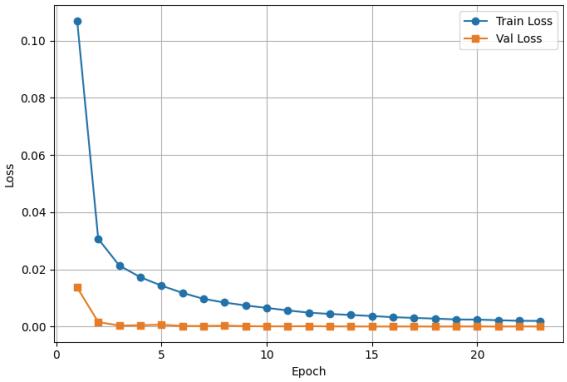
Epoch	Train loss	Val loss	ADE (m)	FDE (m)
14	0.0041	0.0001	0.0110	0.0088
15	0.0037	0.0001	0.0097	0.0090
16	0.0033	0.0000	0.0082	0.0113
17	0.0031	0.0001	0.0113	0.0104
18	0.0028	0.0000	0.0073	0.0091
19	0.0025	0.0001	0.0094	0.0114
20	0.0025	0.0001	0.0112	0.0143
21	0.0022	0.0000	0.0075	0.0103
22	0.0020	0.0001	0.0113	0.0128
23	0.0020	0.0001	0.0090	0.0099

As shown in Fig. 3a, both training and validation losses decrease rapidly during the early epochs and stabilize near zero thereafter. Early stopping is triggered at epoch 23, indicating strong generalization and the absence of overfitting. This stable convergence behavior demonstrates that the model effectively learns the mapping from symbolic intent to motion dynamics. Trajectory prediction accuracy is quantified using the Average Displacement Error (ADE) and the Final Displacement Error (FDE), which measure the mean spatial error over the whole trajectory and at the final time step, respectively. Fig. 3b shows a sharp reduction in both ADE and FDE during the initial training phase, followed by consistently low error values in later epochs. On the held-out test set, the model achieves an average ADE of 0.040 m and an FDE of 0.036 m, indicating high spatial precision and accurate endpoint prediction. Qualitative comparisons between ground-truth and predicted trajectories are illustrated in Fig. 3c. The generated trajectories closely follow the ground-truth motion patterns, capturing both the global direction of movement and smooth temporal evolution. Even in cases involving overlapping or complex motion paths, the predicted trajectories preserve structural consistency with only minor local deviations. This qualitative alignment confirms that the model successfully grounds logical task intent into coherent motion trajectories.

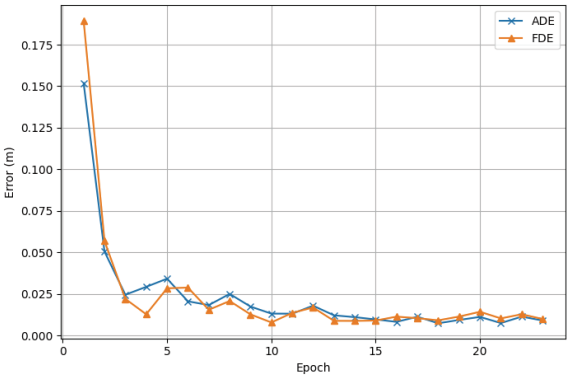
6.3 Proposed Trajectory Generation

This section presents results for the trajectory generation framework, which directly maps symbolic task intent and speech embeddings to continuous motion trajectories. Unlike the staged pipeline, this formulation jointly optimizes trajectory realism, spatial accuracy, and temporal smoothness through adversarial training.

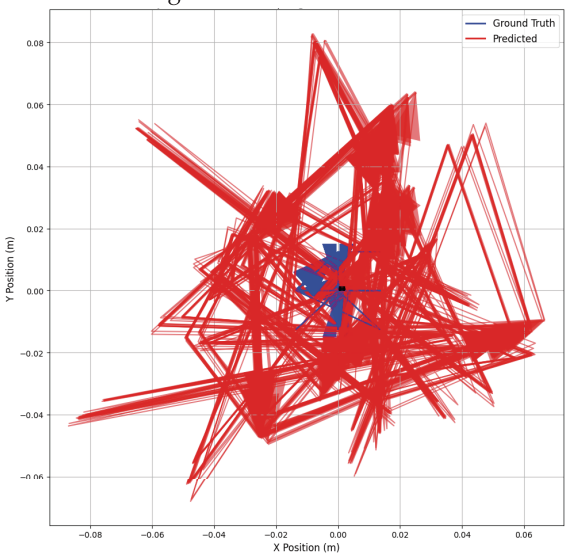
The model is trained using a generative adversarial setup, where a transformer-based generator predicts complete trajectories and a discriminator distinguishes between real and generated motion sequences. The generator objective combines reconstruction loss, adversarial loss, velocity regularization, and smoothness penalties, encouraging both accuracy and physically plausible motion. As shown in Fig. 4a, the generator and discriminator losses decrease steadily across training epochs, indicating stable adversarial convergence without mode collapse. Quantitative trajectory accuracy is evaluated using ADE and FDE. Fig. 4b shows a consistent reduction in both metrics throughout training. The model achieves a final ADE of 0.021 m and an FDE of 0.018 m at epoch 50, demonstrating precise spatial tracking and accurate endpoint prediction despite fully autoregressive generation.



(**a**) Training and validation loss convergence.

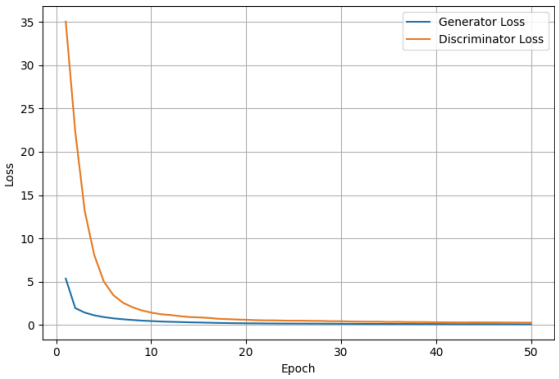


(**b**) Validation ADE and FDE over epochs.

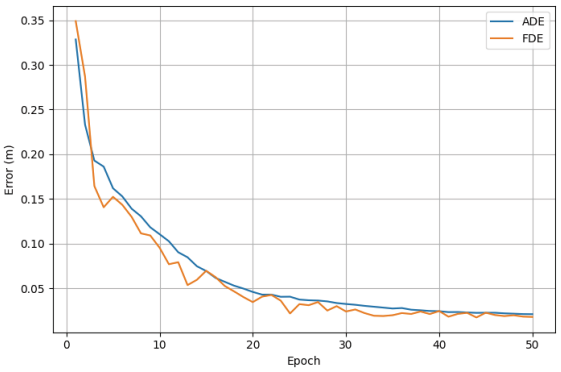


(**c**) Ground-truth vs. predicted trajectories.

Figure 3: Logic-to-Trajectory generation results



(**a**) Generator and discriminator loss convergence.



(**b**) Validation ADE and FDE over epochs.

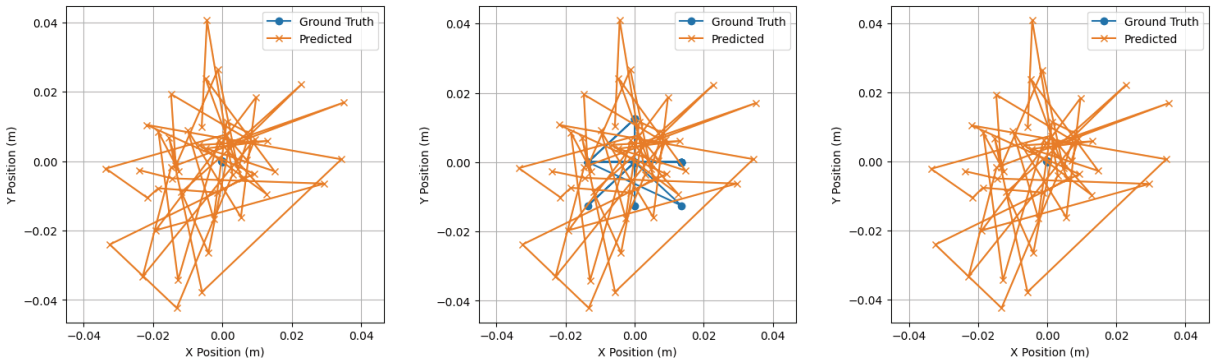
Figure 4: Trajectory generation training dynamics

Table 4 reports epoch-wise generator and discriminator losses together with ADE and FDE values. The progressive reduction in both adversarial losses and displacement errors confirms the effectiveness of jointly optimizing realism and accuracy.

Table 4: Epoch-wise performance for trajectory generation

Epoch	Gen loss	Disc loss	ADE (m)	FDE (m)
1	5.370	35.037	0.329	0.349
5	0.930	5.052	0.162	0.152
10	0.472	1.444	0.111	0.095
15	0.303	0.891	0.069	0.070
20	0.210	0.616	0.046	0.035
25	0.178	0.512	0.037	0.032
30	0.156	0.452	0.032	0.024
35	0.137	0.366	0.027	0.020
40	0.124	0.333	0.024	0.025
45	0.117	0.312	0.023	0.023
50	0.109	0.288	0.021	0.018

Qualitative trajectory comparisons are illustrated in Fig. 5. The predicted trajectories closely follow ground-truth motion patterns across multiple samples, preserving overall directionality and temporal coherence. Even in cases with irregular motion or sharp directional changes, the generated trajectories remain smooth and structurally consistent, highlighting the effectiveness of the velocity and smoothness regularization terms.



(**a**) Trajectory example #0. (**b**) Trajectory example #1. (**c**) Trajectory example #2.

Figure 5: Qualitative comparison between ground-truth (blue) and predicted (orange) trajectories for three representative samples

Table 5 compares trajectory forecasting accuracy between the proposed generator and the baseline logic-to-trajectory model. The proposed approach achieves substantially lower ADE and FDE, demonstrating improved spatial accuracy and endpoint prediction.

Table 5: Comparison of trajectory forecasting performance between the proposed generator model and the baseline approach. ADE and FDE are reported in normalized coordinate units

Method	ADE	FDE
Baseline Logic-to-Trajectory Model	0.040	0.036
Proposed Generator Model	0.021	0.018

7 Discussion and Limitations

The scale, diversity, and variability of the available training data inherently influence the performance and generalization capability of multimodal trajectory generation models. This limitation is particularly evident in the NATSGLD dataset used in this study, which contains a restricted number of interaction sequences per subject and limited diversity in gesture patterns, speech expressions, and motion trajectories. While the dataset is sufficient to evaluate multimodal grounding and trajectory forecasting under controlled settings, it does not fully capture the breadth of variability encountered in real-world human–robot interaction scenarios. To mitigate these constraints, several preprocessing and modeling strategies were incorporated. Spatial coordinates were normalized using Min–Max scaling to stabilize optimization across sequences, and trajectories were represented using relative motion encoding to emphasize temporal dynamics rather than absolute positions. Gesture keypoint sequences were smoothed using Savitzky–Golay filtering to reduce sensor noise and spurious fluctuations. At the modeling level, explicit motion regularization was introduced through velocity and acceleration penalties, encouraging smooth temporal evolution and physically plausible trajectories. These design choices proved effective in practice, as evidenced by stable training dynamics, monotonic loss convergence, and low displacement errors across both staged and models. Despite these measures, the limited dataset size constrains the model's ability to learn more complex cross-modal relationships, reason over longer temporal horizons, and generalize to highly novel instructions or interaction contexts. In particular, rare gesture–language combinations and long-horizon task dependencies remain challenging due to insufficient coverage in the training data. While the adversarial generator demonstrates substantial improvements over the staged baseline, its performance is still bounded by the diversity of observed motion patterns. Notably, the results show that careful preprocessing, grammar-constrained symbolic grounding, and motion-aware regularization can significantly improve robustness and interpretability even under data-limited conditions. The substantial quantitative gains in ADE and FDE, together with qualitative trajectory alignment, indicate that the proposed framework learns meaningful multimodal representations rather than overfitting to noise. By explicitly acknowledging dataset limitations and adopting transparent modeling choices, this work prioritizes reproducibility and interpretability while establishing a solid foundation for future extensions. Future research will benefit from larger-scale multimodal datasets with richer linguistic variation, more diverse gesture vocabularies, and longer, more complex interaction sequences. Incorporating physics-based constraints, environment-aware collision modeling, and online adaptation mechanisms could further improve execution safety and generalization. Nonetheless, the current study demonstrates that combining symbolic reasoning, transformer-based multimodal fusion, and adversarial motion learning offers a promising and practical approach to robust human–robot trajectory generation.

8 Conclusion and Future Work

This work presented a unified multimodal learning framework for HRI that integrates symbolic reasoning, transformer-based multimodal fusion, and trajectory generation within a coherent system. By combining grammar-constrained LTL representations with continuous motion modeling, the proposed

approach bridges high-level human intent and low-level robotic execution in an interpretable and data-driven manner. The framework supports both a staged instruction-to-logic-to-trajectory pipeline and a trajectory generator, enabling flexible deployment depending on the desired balance between interpretability and performance. Extensive experiments on the NATSGLD dataset demonstrate that the proposed models achieve stable training and strong quantitative performance. The staged logic-to-trajectory model produces accurate and consistent motion predictions. The adversarial generator further improves spatial accuracy and motion realism, achieving substantially lower ADE and FDE. Qualitative analyses confirm that the generated trajectories closely follow ground-truth motion patterns while maintaining smooth temporal evolution, supported by explicit velocity and acceleration regularization. The results highlight the effectiveness of combining symbolic task grounding with transformer-based multimodal learning and adversarial regularization, particularly in data-limited settings. At the same time, the study acknowledges that generalization to more diverse and long-horizon interaction scenarios remains constrained by the scale and variability of available datasets. Future work will focus on extending the framework with physics-aware constraints, environment and collision modeling, and larger-scale multimodal datasets. Additionally, systematic ablation studies, data augmentation strategies, and user-centered evaluations will be explored further to improve robustness, generalization, and real-world applicability.

Acknowledgement: Not applicable.

Funding Statement: The authors extend their appreciation to Prince Sattam bin Abdulaziz University for funding this research work through the project number (PSAU/2024/01/32082).

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Shtwai Alsubai, Ahmad Almadhor, Abdullah Al Hejaili, Vincent Karović; Data collection: Shtwai Alsubai, Abdullah Al Hejaili; Analysis and interpretation of results: Shtwai Alsubai, Ahmad Almadhor, Abdullah Al Hejaili. Draft manuscript preparation: Shtwai Alsubai, Ahmad Almadhor, Abdullah Al Hejaili, Najib Ben Aoun, Tahani Alsubait, Vincent Karović. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are taken from this article [19].

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Wang T, Zheng P, Li S, Wang L. Multimodal human-robot interaction for human-centric smart manufacturing: a survey. Adv Intell Syst. 2024;6(3):2300359. doi:10.1002/aisy.202300359.
2. Sun C, Huang S, Sun B, Chu S. Personalized learning path planning for higher education based on deep generative models and quantum machine learning: a multimodal learning analysis method integrating transformer, adversarial training and quantum state classification. Discov Artif Intell. 2025;5(1):29. doi:10.1007/s44163-025-00252-6.
3. Duncan JA, Alambeigi F, Pryor MW. A survey of multimodal perception methods for human-robot interaction in social environments. J Hum-Robot Interact. 2024;13(4):1–50. doi:10.1145/3657030.
4. Dong W, Li S, Zheng P. Toward embodied intelligence-enabled human-robot symbiotic manu-facturing: a large language model-based perspective. J Comput Inf Sci Eng. 2025;25(5):050801. doi:10.1115/1.4068235.
5. Khan MT, Waheed A. Foundation model driven robotics: a comprehensive review. arXiv:250710087. 2025.
6. Lim JY, See J, Dondrup C. Multimodal engagement prediction in human-robot interaction using transformer neural networks. In: International Conference on Multimedia Modeling. Berlin/Heidelberg, Germany: Springer; 2025. p. 3–17.

7. Chitta K, Prakash A, Jaeger B, Yu Z, Renz K, Geiger A. Multi-modal fusion transformer for end-to-end autonomous driving. arXiv:210409224. 2021.
8. Rana A, Malviya V, Cheng S, Hsu D. Motion planning transformers: a motion planning framework for mobile robots. arXiv:210602791. 2021.
9. Marcucci T, Petersen M, von Wrangel D, Tedrake R. Mission-conditioned path planning with transformer variational autoencoder. Electronics. 2024;13(13):2437. doi:10.3390/electronics13132437.
10. Ullrich L, McMaster A, Graichen K. Transfer learning study of motion transformer-based trajectory predictions. In: Proceedings of the 2024 IEEE Intelligent Vehicles Symposium (IV); 2024 Jun 2–5; Jeju Island, Republic of Korea.
11. Kapoor P, Vemprala S, Kapoor A. Enhancing robotic planning with logically constrained transformers. In: Proceedings of the Robotics: Science and Systems Foundation; 2024 Jul 15–19; Delft, The Netherlands. RSS 2024 Safe Autonomy Workshop.
12. Jiang W, Zhao WX, Wang J, Jiang J. Continuous trajectory generation based on two-stage GAN. Proc AAAI Conf Artif Intell. 2023;37(4):4374–82. doi:10.1609/aaai.v37i4.25557.
13. Wang X, Yang Y, Wang W, Zhou Y, Yin Y, Gong Z. Generative adversarial networks based motion learning towards robotic calligraphy synthesis. CAAI Trans Intel Technol. 2024;9(2):452–66. doi:10.1049/cit2.12198.
14. Lv Q, Zhao W, Xie Z. An improved GAN with transformers for pedestrian trajectory prediction models. Int J Intell Sys. 2022;37(8):4417–36. doi:10.1002/int.22724.
15. Rando J, Granados A, Castillo JC. Human trajectory prediction and generation using LSTM models and GANs. Pattern Recognit. 2021;120(2):108136. doi:10.1016/j.patcog.2021.108136.
16. Mohammadi M, Rashidi M. Learning inverse kinematics and dynamics of a robotic manipulator using generative adversarial networks. Robot Auton Syst. 2020;124(21):103386. doi:10.1016/j.robot.2019.103386.
17. Ullah I, Adhikari D, Khan H, Anwar MS, Ahmad S, Bai X. Mobile robot localization: current challenges and future prospective. Comput Sci Rev. 2024;53(8):100651. doi:10.1016/j.cosrev.2024.100651.
18. Ullah I, Adhikari D, Khan H, Ahmad S, Esposito C, Choi C. Optimizing mobile robot localization: drones-enhanced sensor fusion with innovative wireless communication. In: Proceedings of the IEEE INFOCOM 2024—IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS); 2024 May 20; Vancouver, BC, Canada. p. 1–6.
19. Shrestha S, Zha Y, Banagiri S, Gao G, Aloimonos Y, Fermüller C. NatSGLD: a dataset with speech, gesture, logic, and demonstration for robot learning in natural human-robot interaction. In: Proceedings of the 2025 20th ACM/IEEE International Conference on Human-Robot Inter-action (HRI); 2025 Mar 4–6; Melbourne, Australia. p. 1093–8.