REVISITING NUMERICAL FORECASTING MODELS FOR LANGUAGE-BASED TRAJECTORY PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Language-based trajectory predictors represent coordinates as discrete tokens and learn auxiliary tasks such as destination and group reasoning. This formulation enables the model to capture behavioral intent and social context beyond coordinate dynamics alone. However, token-level objectives provide only indirect guidance for continuous coordinate-space dynamics. To address this limitation, we introduce **MoRE** (**M**ixture **O**f **R**eward **E**xperts), a refinement framework that transfers numerical forecasting priors into a pretrained language-based predictor through reinforcement learning. Five frozen numerical predictors provide complementary coordinate-level knowledge of motion and interactions. Their predictions are converted into expert rewards and combined through an uncertainty-weighted consensus that penalizes disagreement. A ground-truth reward anchors the prediction to the target trajectory. To focus refinement on difficult cases, MoRE refines the policy using the top 1% of training samples ranked by predictive entropy. Expert predictions are computed once and cached before PPO training, so the experts are not run during policy updates or inference. In this way, MoRE combines the contextual modeling of the language-based predictor with coordinate-level feedback from numerical experts. On ETH-UCY, MoRE reduces ADE from 0.22 to 0.20 m and FDE from 0.32 to 0.29 m. Relative to the base policy, ADE decreases by 17.9% on SDD and 12.7% on NBA. On ETH-UCY, MoRE also reduces collision rates and better matches ground-truth pedestrian spacing, without increasing measured inference memory or latency.

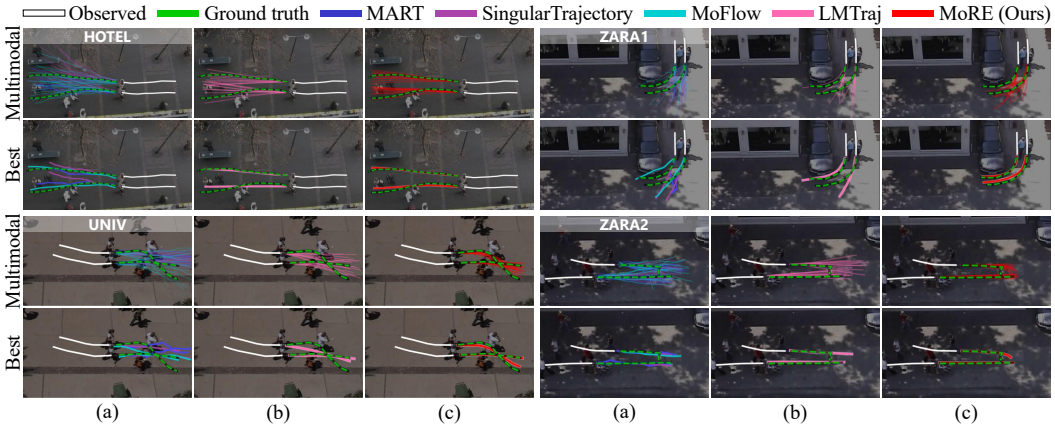

Figure 1: Prediction examples on ETH-UCY. "Multimodal" shows 20 samples, and "Best" shows the closest sample. (a) Numerical baselines and (b) LMTraj exhibit different spatial patterns. (c) MoRE refines the language-based predictions using numerical feedback.

1 INTRODUCTION

Forecasting human trajectories requires modeling motion, interactions, and uncertainty about future behavior. Numerical trajectory predictors model human motion directly in continuous coordinate

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

space and have progressively incorporated specialized priors for local interactions, group behavior, motion patterns, and destination goals (Mohamed et al., 2020; Bae et al., 2022a; 2024b; Zhao & Wildes, 2021). This coordinate-based formulation provides strong spatial accuracy, particularly for the best sample under standard best-of-K evaluation (Fig. 1(a), Best). However, coordinate-based numerical predictors do not directly leverage the contextual knowledge captured by language-based models. As a result, stochastic numerical predictors often generate high-variance, scattered predictive distributions while optimizing coordinate-level regression objectives (Fig. 1(a), Multimodal).

Language-based predictors provide a complementary formulation by representing trajectories and related reasoning tasks as discrete tokens. LMTraj (Bae et al., 2024a) jointly learns destination, group, collision, and other auxiliary tasks, allowing the model to capture behavioral intent and social context more explicitly. Compared with the dispersed multimodal predictions of the numerical baselines in Fig. 1(a), LMTraj produces more spatially consistent multimodal paths in Fig. 1(b). However, token-level likelihood does not directly reflect metric errors in continuous coordinates, so its best prediction can remain spatially inaccurate (Fig. 1(b), Best). This motivates incorporating coordinate-level numerical feedback during refinement. Our goal is to improve the coordinate accuracy of a pretrained language-based predictor using numerical forecasting knowledge, without requiring numerical experts at inference.

We propose MoRE (Mixture of Reward Experts), a framework that transfers complementary knowledge from pretrained numerical predictors to a language-based predictor during training. Rather than directly combining expert outputs, MoRE converts their predictions into reward signals, integrating the language-based model's contextual knowledge with the complementary motion and social-interaction priors learned by numerical predictors. The expert rewards are combined through a consensus objective and used to refine the policy with reinforcement learning. To keep the process efficient, refinement focuses on difficult samples identified by predictive uncertainty, with expert predictions computed and cached in advance.

MoRE combines the contextual representations of the language-based predictor with complementary knowledge from multiple numerical experts. This integration improves prediction accuracy across ETH-UCY, SDD, GCS, and NBA, with larger gains on challenging non-linear motion and cross-dataset transfer. Overall, MoRE transfers knowledge from multiple numerical experts into a single language-based predictor during training, without requiring the experts at inference or increasing the measured inference memory and latency of the base policy.

2 Related Work

2.1 Numerical Trajectory Prediction

Numerical trajectory prediction has progressed from hand-crafted social interaction models (Helbing & Molnar, 1995; Pellegrini et al., 2009) to data-driven approaches that learn interactions and multimodal futures (Alahi et al., 2016; Gupta et al., 2018; Mangalam et al., 2020; Salzmann et al., 2020). Recent methods introduce different inductive biases to capture human motion more effectively. Spatio-temporal graph models focus on interaction structure (Mohamed et al., 2020; Huang et al., 2019; Bae & Jeon, 2021), while GP-Graph and GroupNet explicitly model group-level relations (Bae et al., 2022a; Xu et al., 2022a). Other methods capture characteristic motion patterns through compact trajectory representations (Bae et al., 2023; 2024b), or guide future motion using destination information (Dendorfer et al., 2020; Zhao & Wildes, 2021). Recent generative and relational approaches further improve multimodal forecasting through diffusion, transformer, and flow-based modeling (Gu et al., 2022; Mao et al., 2023; Lee et al., 2024; Fu et al., 2025). These numerical predictors already model social interactions, group structure, and diverse motion patterns (Kothari et al., 2021), but they encode such knowledge through different architectures and objectives. MoRE leverages this diversity rather than introducing another numerical predictor. We use models with complementary strengths as numerical experts and transfer their specialized coordinate-level knowledge to a language-based predictor.

2.2 Language-Based Trajectory Prediction

Language-based trajectory prediction represents coordinates as tokens and formulates forecasting as sequence generation. LMTraj (Bae et al., 2024a) jointly learns trajectory prediction with auxiliary

tasks involving destinations, directions, groups, collisions, and mimicry, while its multimodal extension incorporates visual information (Bae et al., 2025). Subsequent approaches explore language-guided numerical forecasting (Chib & Singh, 2025), goal-oriented prompting and chain-of-thought reasoning (Kim et al., 2025), visual-linguistic reasoning (Shenkut & Kumar, 2025), autoregressive multimodal generation (Wang et al., 2026), and reinforcement learning for language-based trajectory prediction (Xu et al., 2026). These methods aim to strengthen contextual and behavioral reasoning, but token-only objectives provide only indirect supervision for continuous coordinate accuracy. MoRE combines the contextual modeling of the language-based predictor with coordinate-level feedback from numerical predictors while retaining a single language-based predictor at inference.

2.3 Expert Transfer and Reward-Based Refinement

Knowledge from multiple models can be transferred in several ways. Knowledge distillation trains a student to reproduce the predictions of one or more teacher models (Hinton et al., 2015; You et al., 2017). Mixture-of-experts methods instead maintain subnetworks and select or combine them within the model (Jacobs et al., 1991; Shazeer et al., 2017; Fedus et al., 2022). These approaches transfer expert knowledge through supervised targets or architectural composition. Reward-based learning provides another form of supervision by evaluating generated outputs and optimizing the policy from this feedback (Christiano et al., 2017; Ouyang et al., 2022). Since a reward can be an imperfect proxy for the desired behavior, aggressive optimization may lead to reward overoptimization (Gao et al., 2023). Reward-model ensembles reduce reliance on a single reward signal, and Coste et al. (2024) investigate conservative aggregation strategies including uncertainty-weighted optimization. MoRE builds on this reward-based approach but uses pretrained numerical predictors as domain experts. Their forecasts provide coordinate-level rewards for refining a language-based predictor, rather than serving as direct supervision targets or inference-time modules.

3 Method

MoRE builds on a pretrained language-based forecasting architecture and refines it using rewards from numerical experts. We first describe the tokenized trajectory prediction setup (Sec. 3.1), followed by the multi-expert consensus reward (Sec. 3.2), PPO-based refinement (Sec. 3.3), and uncertainty-driven sample selection (Sec. 3.4). Implementation details are provided in Sec. 3.5.

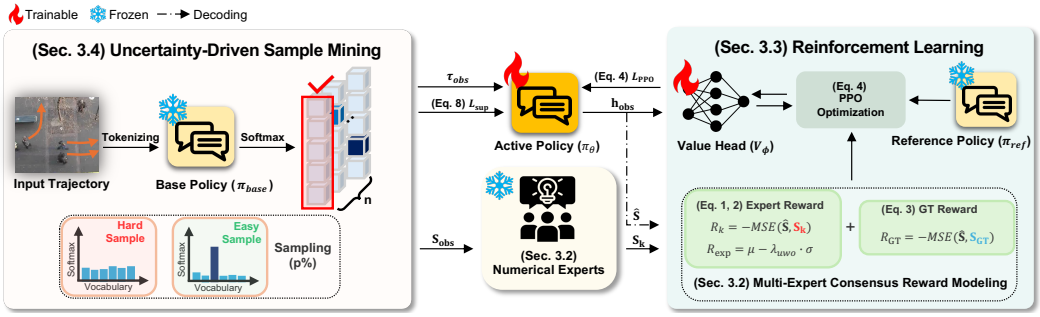

Figure 2: MoRE training pipeline. A frozen base policy ranks training samples by predictive entropy. Numerical expert predictions for the selected subset are cached once. Decoded policy samples receive expert-consensus and ground-truth rewards for PPO refinement, interleaved with supervised learning. Only the refined forecasting policy is needed at inference.

3.1 Preliminaries

Problem Definition. Given observed trajectories of N pedestrians over T_{obs} timesteps, our goal is to predict their future trajectories over T_{pred} timesteps. Formally, let $\mathbf{p}_t^n = [x_t^n, y_t^n]^\top \in \mathbb{R}^2$ denote the 2D coordinate vector of pedestrian n at time t. The observed trajectory and the ground truth future trajectory for pedestrian n are defined as matrices $\mathbf{S}_{\text{obs}}^n = [\mathbf{p}_1^n, \ldots, \mathbf{p}_{T_{\text{obs}}}^n]^\top \in \mathbb{R}^{T_{\text{obs}} \times 2}$ and $\mathbf{S}_{\text{GT}}^n = [\mathbf{p}_{T_{\text{obs}}+1}^n, \ldots, \mathbf{p}_{T_{\text{obs}}+T_{\text{pred}}}^n]^\top \in \mathbb{R}^{T_{\text{pred}} \times 2}$, respectively. Consequently, our model aims to learn the parameterized conditional distribution across all N pedestrians, formulated as $p_\theta(\{\mathbf{S}_{\text{GT}}^n\}_{n=1}^N \mid$

$\{\mathbf{S}_{\text{obs}}^n\}_{n=1}^N$). For the pedestrian benchmarks, we use $T_{\text{obs}} = 8$ and $T_{\text{pred}} = 12$. NBA follows its benchmark-specific evaluation protocol described in Sec. 4.1.

Base Policy. Our framework builds upon LMTraj (Bae et al., 2024a), which formulates trajectory prediction as a text-based sequence generation task (Raffel et al., 2020). A numerical tokenizer maps continuous trajectories into a discrete token space, and we initialize from its pretrained weights. However, next-token likelihood does not directly penalize the magnitude of errors in decoded coordinates. Formally, the tokenizer maps the continuous observation matrix $\mathbf{S}_{\text{obs}}$ to a discrete text prompt τ_{obs}, and the ground-truth future trajectory $\mathbf{S}_{\text{GT}}$ to a target token sequence $\tau_{\text{GT}} = [\tau_1, \ldots, \tau_L]$. The policy models the discrete conditional distribution $\pi_\theta(\tau_k \mid \tau_{\text{obs}}, \tau_{<k})$.

3.2 Multi-Expert Consensus Reward Modeling

Numerical Reward Formulation. We define a composite reward that integrates complementary knowledge from multiple numerical experts. We use five specialized predictors with different strengths: (i) a *local interaction expert* for local pedestrian interactions, (ii) a *structural relation expert* for relational dynamics, (iii) a *social grouping expert* for group behavior, (iv) a *motion pattern expert* for characteristic motion patterns, and (v) a *goal expert* for destination information. Each expert contributes its own prediction as coordinate-level guidance, allowing MoRE to preserve their complementary knowledge rather than forcing them to agree on a single behavior. At the same time, individual predictions can contain noisy or unreliable guidance. MoRE therefore applies a consensus reward in which larger disagreement among expert scores receives a stronger penalty, reducing the influence of strongly inconsistent signals while retaining the guidance from each expert. The same framework can incorporate other pretrained numerical predictors, and results with additional experts and alternative expert configurations are provided in Sec. C.

Reward Aggregation via Uncertainty-Weighted Optimization. Given a text-based observation prompt, the language-based policy autoregressively generates a sequence of discrete tokens, which is decoded into a continuous coordinate matrix $\hat{\mathbf{S}}$. For each selected observation, each frozen numerical expert $k \in \{1, \ldots, K\}$ produces a continuous prediction $\mathbf{S}_k$ from the historical trajectory, and these predictions are cached before policy refinement. We then define the reward from expert k as the negative mean squared error between the decoded policy prediction and the cached expert prediction:

$$R_k = -\text{MSE}(\hat{\mathbf{S}}, \mathbf{S}_k). \tag{1}$$

To combine expert rewards, we follow the conservative aggregation principle of Uncertainty-Weighted Optimization (UWO) (Coste et al., 2024). For each generated trajectory, the mean reward represents its overall agreement with the numerical experts, while the standard deviation measures disagreement among their scores. We penalize this disagreement so that trajectories receiving inconsistent evaluations from the experts obtain lower consensus rewards. The resulting consensus reward is defined as:

$$R_{\text{exp}} = \mu - \lambda_{\text{uwo}} \cdot \sigma, \quad \text{where } \mu = \frac{1}{K}\sum_{k=1}^{K} R_k, \;\; \sigma = \sqrt{\frac{1}{K}\sum_{k=1}^{K}(R_k - \mu)^2}. \tag{2}$$

The penalty coefficient λ_{uwo} reduces the influence of noisy or inconsistent expert signals by penalizing disagreement among expert rewards, encouraging more stable and physically grounded predictions. Here, μ is the mean expert reward and σ is the standard deviation of expert rewards for the same generated trajectory. Setting $\lambda_{\text{uwo}} = 0$ recovers mean aggregation.

Total Reward Formulation. To balance expert guidance with accuracy to the ground-truth trajectory, we use the composite reward:

$$R = R_{\text{GT}} + \lambda_{\text{exp}} \cdot R_{\text{exp}}, \quad \text{where } R_{\text{GT}} = -\text{MSE}(\hat{\mathbf{S}}, \mathbf{S}_{\text{GT}}). \tag{3}$$

While R_{GT} anchors the prediction to the target trajectory, R_{exp} incorporates complementary knowledge from the numerical experts. The expert reward encourages agreement with their forecasts rather than directly enforcing physical constraints.

3.3 REINFORCEMENT LEARNING VIA PPO

PPO Optimization. The policy π_θ and the value network V_ϕ are jointly optimized by minimizing a composite surrogate loss, which integrates a clipped policy objective, a value regression loss, and a KL divergence penalty to prevent catastrophic forgetting:

$$\begin{aligned}\mathcal{L}_{\text{PPO}} = &- \mathbb{E}\left[\frac{1}{L}\sum_{t=1}^{L}\min\left(\rho_t(\theta)\hat{A}, \text{clip}(\rho_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}\right)\right] + c_v\mathbb{E}\left[(V_\phi(\mathbf{h}_{\text{obs}}) - R)^2\right] \\ &+ \beta\hat{D}_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}}), \quad \text{where} \quad \rho_t(\theta) = \frac{\pi_\theta(\tau_t \mid \tau_{\text{obs}}, \tau_{<t})}{\pi_{\text{old}}(\tau_t \mid \tau_{\text{obs}}, \tau_{<t})}.\end{aligned} \tag{4}$$

The token-level importance ratio $\rho_t(\theta)$ is computed for each generation step t relative to the behavior policy π_{old}. The sequence-level advantage $\hat{A} = R - V_\phi(\mathbf{h}_{\text{obs}})$ is calculated as the direct difference between the composite reward R and the expected return. The value function V_ϕ processes the mean-pooled encoder hidden states of the observation context ($\mathbf{h}_{\text{obs}}$) and is jointly trained via Mean Squared Error (MSE) regression on the observed returns R. Further mathematical details are provided in Sec. A.

3.4 UNCERTAINTY-DRIVEN SAMPLE MINING

We focus PPO refinement on training samples with high predictive entropy under the frozen base policy. Entropy is used to rank sample uncertainty, and numerical expert rewards are applied to the selected top-$p\%$ subset.

Sample Difficulty Scoring. We assess the difficulty of each trajectory sample by measuring the predictive uncertainty of the base policy π_{base}. Rather than relying on ground-truth predictive error, we compute the Shannon entropy of the model's Softmax output distribution to reveal inherent model ambiguity. The difficulty score $\mathcal{S}_i$ for sample i is defined as the sequence-averaged entropy across the vocabulary space $\mathcal{V}$:

$$\mathcal{S}_i = -\frac{1}{L_i}\sum_{k=1}^{L_i}\sum_{v\in\mathcal{V}}\pi_{\text{base}}(v \mid \tau_{i,\text{obs}}, \tau_{i,<k})\log\pi_{\text{base}}(v \mid \tau_{i,\text{obs}}, \tau_{i,<k}), \tag{5}$$

where v represents a candidate token and L_i is the sequence length. Here, a high $\mathcal{S}_i$ indicates a relatively flat Softmax distribution, indicating greater uncertainty in the base policy's token distribution. Based on this difficulty distribution across the entire training set $\mathcal{D}$, we define a selection threshold γ_p to identify the top $p\%$ of samples. We then apply a binary selection mask M_i to focus the optimization processes on these high-uncertainty cases:

$$\gamma_p = \text{Percentile}(\{\mathcal{S}_i\}_{i=1}^{|\mathcal{D}|}, 100-p), \quad M_i = \mathbb{1}[\mathcal{S}_i \geq \gamma_p]. \tag{6}$$

Overall Objective. The refinement process combines supervised learning with PPO on the selected subset. The total objective is defined as:

$$\mathcal{L}_{\text{total}} = \frac{1}{|M|}\sum_{i\in M}\left(\mathcal{L}_{\text{sup}}^{(i)} + \lambda_{\text{RL}}\mathcal{L}_{\text{PPO}}^{(i)}\right), \tag{7}$$

where $M = \{i : M_i = 1\}$ denotes the identified subset of hard samples. The supervised term $\mathcal{L}_{\text{sup}}^{(i)}$ acts as a regularizer to maintain the model's fundamental sequence-to-sequence mapping capabilities. It is defined as the standard negative log-likelihood (NLL) against the ground-truth token sequence τ_i^*:

$$\mathcal{L}_{\text{sup}}^{(i)} = -\frac{1}{L_i}\sum_{k=1}^{L_i}\log\pi_\theta(\tau_{i,k}^* \mid \tau_{i,\text{obs}}, \tau_{i,<k}^*). \tag{8}$$

The PPO term provides reward-based feedback from the ground-truth and numerical expert rewards. During training, PPO and supervised updates are interleaved, with one PPO update every ten supervised steps. This schedule maintains the original token-prediction objective while introducing coordinate-level guidance through PPO.

3.5 Implementation Details

We adopt the pretrained LMTraj (Bae et al., 2024a) as the base policy and refine it with LoRA (Hu et al., 2022) of rank $r = 16$, updating 0.3% of the base-policy parameters. The five frozen numerical experts are Social-STGCNN (Mohamed et al., 2020), DMRGCN (Bae & Jeon, 2021), GP-Graph (Bae et al., 2022a), SingularTrajectory (Bae et al., 2024b), and Expert-Trajectory (Zhao & Wildes, 2021). We set $\lambda_{\text{exp}} = 0.5$, $\lambda_{\text{uwo}} = 0.05$, and $\lambda_{\text{RL}} = 0.5$. PPO uses $\epsilon = 0.2$, $c_v = 0.5$, and $\beta = 0.1$. We refine the top 1% of training samples ranked by predictive entropy using AdamW (Loshchilov & Hutter, 2017) with a learning rate of 10^{-4} and an effective batch size of 1,024. Refinement takes about one hour on eight NVIDIA RTX PRO 6000 GPUs.

4 Experiments

4.1 Experimental Setup

Benchmarks and protocol. We evaluate on ETH-UCY (Pellegrini et al., 2009; Lerner et al., 2007), Stanford Drone Dataset (SDD) (Robicquet et al., 2016), Grand Central Station (GCS) (Yi et al., 2015), and NBA. ETH-UCY comprises ETH, HOTEL, UNIV, ZARA1, and ZARA2 and uses leave-one-out evaluation. SDD contains aerial trajectories, while GCS includes dense station scenes with up to 289 concurrent pedestrians. These pedestrian benchmarks use eight observed and twelve future frames. NBA evaluates highly interactive motion under the protocol of Wong et al. (2025). Cross-dataset evaluation trains on ETH-UCY and tests on SDD without fine-tuning, assessing whether numerical refinement improves spatial accuracy while retaining the cross-dataset transfer capability of the language-based predictor.

Metrics. Average Displacement Error (ADE) measures mean Euclidean position error over future timesteps, while Final Displacement Error (FDE) measures endpoint error. Following the standard best-of-20 evaluation protocol for stochastic trajectory prediction (Gupta et al., 2018), we report the minimum ADE and FDE over 20 sampled trajectories.

Social spacing and collisions. On ETH-UCY, we additionally report collision rate (COL) (Liu et al., 2021) and pedestrian-spacing statistics based on Hall's proxemic zones (Hall, 1966). These distance ranges provide a simple way to compare whether predicted pedestrian spacing resembles real interactions beyond displacement error alone. We use three zones: intimate (< 0.45 m), per-

Table 1: Best-of-20 ADE / FDE (ETH-UCY in meters, with SDD and GCS in pixels). Bold and underline indicate best and second-best values, including ties. Model references are in Sec. C.1.

Model	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG	SDD	GCS
Numerical models								
Social GAN	0.77/1.40	0.43/0.88	0.75/1.50	0.35/0.69	0.36/0.72	0.53/1.04	13.6/24.6	15.9/32.6
Social-STGCNN	0.65/1.10	0.50/0.86	0.44/0.80	0.34/0.53	0.31/0.48	0.45/0.75	20.8/33.2	14.7/23.9
PECNet	0.61/1.07	0.22/0.39	0.34/0.56	0.25/0.45	0.19/0.33	0.32/0.56	10.0/15.9	17.1/29.3
Trajectron++	0.61/1.03	0.20/0.28	0.30/0.55	0.24/0.41	0.18/0.32	0.31/0.52	11.4/20.1	12.8/24.2
AgentFormer	0.46/0.80	0.14/0.22	0.25/0.45	0.18/0.30	0.14/0.24	0.23/0.40	8.7/14.9	10.2/16.9
MID	0.57/0.93	0.21/0.33	0.29/0.55	0.28/0.50	0.20/0.37	0.31/0.54	7.6/14.3	10.7/18.2
EqMotion	0.40/0.61	0.12/0.18	0.23/0.43	0.18/0.32	0.13/0.23	0.21/0.35	7.9/11.9	7.6/13.1
GP-Graph	0.43/0.63	0.18/0.30	0.24/0.42	0.17/0.31	0.15/0.29	0.23/0.39	9.1/13.8	7.8/13.7
LED	0.39/0.58	**0.11**/0.17	0.26/0.43	0.18/**0.26**	0.13/0.22	0.21/0.33	8.5/11.7	9.0/11.4
MART	**0.35**/0.47	0.14/0.22	0.25/0.45	0.17/0.29	0.13/0.22	0.21/0.33	7.4/11.8	10.6/14.1
SingularTraj.	**0.35**/0.42	0.13/0.19	0.25/0.44	0.19/0.32	0.15/0.25	0.21/0.32	7.58/12.1	7.9/13.3
MoFlow	0.40/0.57	**0.11**/0.17	0.23/0.39	**0.15/0.26**	**0.12**/0.22	**0.20**/0.32	7.5/12.0	9.1/11.6
PCHGCN	0.42/0.65	0.17/0.28	0.21/0.38	0.17/0.31	0.13/0.23	0.22/0.37	N/A	N/A
ViTE	**0.35**/0.49	**0.11**/0.17	0.23/0.42	0.18/0.30	0.13/0.22	**0.20**/0.32	7.4/11.9	N/A
Language-based models								
LMTraj-ZERO	0.80/1.64	0.20/0.37	0.37/0.77	0.33/0.66	0.24/0.50	0.39/0.79	10.9/21.0	12.7/25.5
LMTraj-SUP	0.41/0.50	0.12/0.16	0.22/0.34	0.20/0.32	0.17/0.27	0.22/0.32	7.8/10.1	7.1/9.6
GUIDE-CoT	0.38/0.43	0.13/0.15	0.34/0.48	0.19/0.29	0.17/**0.21**	0.24/0.31	N/A	N/A
W2W	**0.35/0.41**	0.12/0.15	**0.20/0.32**	0.19/0.29	0.17/0.26	0.21/**0.29**	7.4/10.1	N/A
MoRE (Ours)	0.36/0.42	**0.11/0.14**	0.21/**0.32**	0.18/0.28	0.17/0.26	**0.20/0.29**	**6.4/9.5**	**6.8/8.5**

sonal (0.45–1.2 m), and social (1.2–3.7 m). For each zone, we measure the percentage of pedestrians with at least one other pedestrian within that range. A pedestrian may have neighbors in multiple zones, so the percentages do not need to sum to 100%. Values closer to the ground truth indicate more realistic spacing, while lower COL is better.

4.2 Evaluation Results

Forecasting accuracy. Tab. 1 compares numerical and language-based predictors. At the reported precision, MoRE achieves tied-best average ADE and FDE on ETH-UCY and the lowest ADE and FDE on both SDD and GCS among the compared models. Relative to LMTraj-SUP, ETH-UCY ADE decreases from 0.22 to 0.20 m and FDE from 0.32 to 0.29 m, corresponding to approximately 9.1% and 9.4% reductions. SDD ADE decreases from 7.8 to 6.4 pixels by approximately 17.9%, while GCS FDE decreases from 9.6 to 8.5 pixels by approximately 11.5%. Together, these results show consistent improvements across benchmarks, with MoRE matching the best average accuracy on ETH-UCY and yielding larger relative gains on SDD and GCS.

Table 2: Social interaction statistics on ETH-UCY (%). Intimate, personal, and social values are compared with the ground truth, while lower collision rate (COL) is better.

Method	Intimate	Personal	Social	COL↓
Ground truth	22.48	62.61	12.93	N/A
GP-Graph	26.02	58.91	13.82	2.44
LED	27.06	58.81	13.72	2.62
SingularTraj.	27.10	57.25	12.12	2.96
MART	26.71	58.14	12.14	<u>2.23</u>
LMTraj-SUP	<u>25.72</u>	<u>59.79</u>	<u>12.58</u>	2.54
MoRE (Ours)	**23.41**	**60.49**	**13.08**	**2.11**

Table 3: Trajectory prediction results on the NBA dataset using best-of-20 ADE / FDE (meters).

Method	ADE	FDE
MemoNet	1.25	1.47
E-V2-Net	1.26	1.64
SocialCircle+	1.14	1.37
Resonance	1.12	1.38
LMTraj-SUP	<u>1.10</u>	<u>1.21</u>
MoRE (Ours)	**0.96**	**1.11**

Interaction-heavy motion and social spacing. On the highly interactive NBA benchmark, MoRE achieves the lowest ADE and FDE among the compared methods (Tab. 3). Relative to LMTraj-SUP, ADE decreases from 1.10 to 0.96 m and FDE from 1.21 to 1.11 m, reductions of approximately 12.7% and 8.3%. On ETH-UCY, we further evaluate whether the predicted trajectories reproduce observed pedestrian spacing. MoRE is closest to the ground truth in all three distance zones (Tab. 2). The intimate-zone percentage decreases from 25.72% to 23.41%, closer to the ground-truth value of 22.48%. COL also decreases from 2.54% to 2.11%, the lowest among the compared methods. Together, these results show that MoRE improves trajectory accuracy while producing sampled predictions with spacing patterns closer to real pedestrians and fewer collisions.

Qualitative comparison. Fig. 1, Fig. 3, and Fig. 4 compare sampled predictions across ETH-UCY and SDD. In the illustrated cases, MoRE reduces the spatial offset of the language-based predictions and concentrates samples around paths closer to the ground truth. These qualitative observations complement the displacement and pedestrian-spacing measurements. Additional visual comparisons on ETH-UCY, SDD, and GCS are provided in Sec. E.

4.3 Ablation Studies

Expert integration and reward design. Tab. 4 examines both how numerical expert knowledge is integrated and how the reward is constructed. Additional SFT on the same hard-sample subset does not improve the base policy, while distillation remains above MoRE. Within PPO, maximum and mean aggregation give 0.218 / 0.314 m and 0.217 / 0.307 m, respectively, whereas UWO reaches 0.204 / 0.285 m. Removing the ground-truth reward increases the error to 0.211 / 0.297 m. Together, these results show that the gain depends on both the integration strategy and the proposed reward design, rather than additional fine-tuning or direct expert transfer alone. Output ensembling, additional non-PPO integration strategies, and per-scene results are reported in Tab. 12.

Individual expert contributions. Tab. 5 evaluates refinement with individual experts on four behavioral subsets following prior work. These subsets may overlap. GP-Graph performs best among

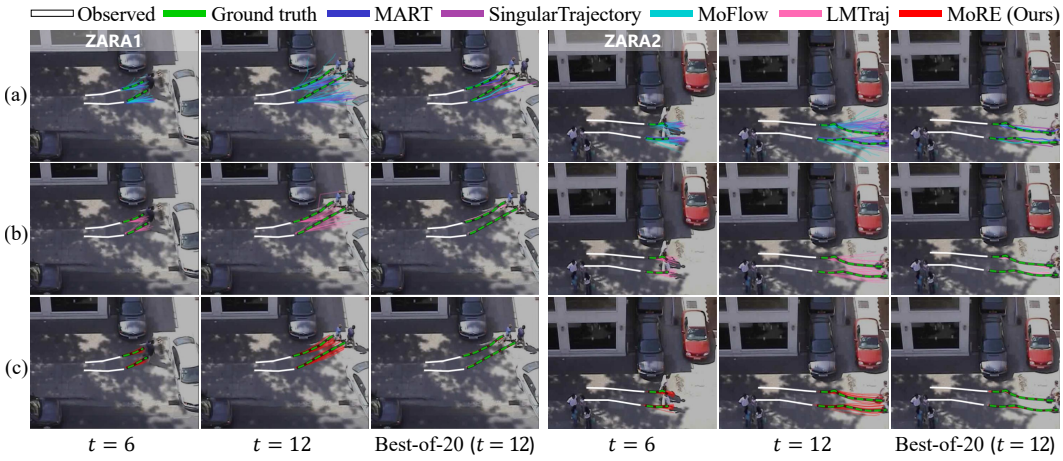

Figure 3: Sampled predictions on ZARA1 and ZARA2 at $t = 6$ and $t = 12$, with the best-of-20 trajectory. Rows show (a) numerical baselines, (b) LMTraj, and (c) MoRE. Additional qualitative results are provided in Sec. E.

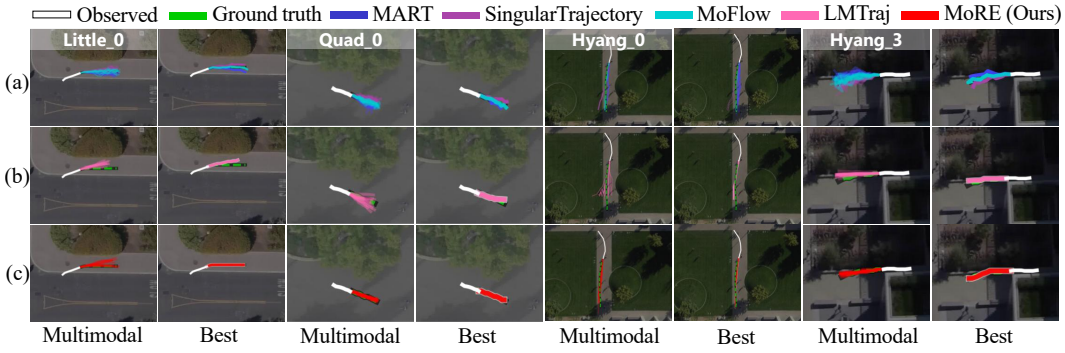

Figure 4: Sampled predictions on four SDD scenes, showing 20 trajectories and the best-of-20 trajectory. From top to bottom, rows show numerical baselines, LMTraj, and MoRE.

Table 4: Expert integration and reward ablation on ETH-UCY (ADE / FDE, meters). GT denotes ground truth.

Method	**ADE**	**FDE**
Base policy	0.224	0.318
Additional SFT	0.227	0.321
Distillation	0.218	0.313
PPO + Max	0.218	0.314
PPO + Mean	0.217	0.307
MoRE w/o GT	<u>0.211</u>	<u>0.297</u>
MoRE (Ours)	**0.204**	**0.285**

Table 5: Refinement with individual experts on ETH-UCY behavioral subsets (ADE, meters). Subsets may overlap; All combines the five experts.

Expert	**Group**	**Mimicry**	**Non-lin.**	**Coll.**
None	0.206	0.164	0.322	0.200
STGCNN	0.202	0.158	0.312	0.215
DMRGCN	0.198	0.148	0.318	0.205
GP-Graph	<u>0.192</u>	0.152	0.310	<u>0.192</u>
SingularTraj.	0.212	<u>0.145</u>	<u>0.292</u>	0.202
ExpertTraj.	0.210	0.160	0.308	0.198
All (Ours)	**0.188**	**0.142**	**0.282**	**0.185**

individual experts on group walking and collision avoidance, while SingularTrajectory performs best on mimicry and non-linear motion. The effect of each expert varies by behavior. For example, DMRGCN improves mimicry but does not improve the collision subset over the base policy. Combining all five experts gives the lowest ADE in every reported subset, showing the benefit of their complementary strengths. Full ADE / FDE and aggregate results are provided in Tab. 10.

Sample selection and refinement. Tab. 6 compares tuning strategies and data fractions. On the same 1% subset, LoRA gives 0.204 / 0.285 m, compared with 0.213 / 0.297 m for full fine-tuning. Using 0.1% or 100% of the data raises LoRA's ADE to 0.223 m or 0.221 m. The 1% setting is best among those shown. Tab. 7 examines samples before refinement. The top 1% has base-policy ADE

Table 6: Refinement on ETH-UCY (meters). Data is the training fraction used. Lower ADE / FDE is better.

Tuning	Data (%)	ADE / FDE
Full FT	1	0.213/0.297
LoRA	100	0.221/0.303
LoRA	0.1	0.223/0.288
LoRA (Ours)	**1**	**0.204/0.285**

Table 7: Training samples before refinement. Base-policy errors are in meters. Coll.-inv. denotes collision-involved samples.

Partition	ADE/FDE	Non-linear (%)	Coll.-inv. (%)
Top 1%	0.40/0.60	82	25
Random 1%	0.24/0.35	45	13
Bottom 1%	0.05/0.08	2	1
All	0.22/0.32	41	9

/ FDE of 0.40 / 0.60 m, compared with 0.22 / 0.32 m overall. Non-linear and collision-involved samples account for 82% and 25%, versus 41% and 9% overall. These results show that entropy ranking selects samples where the base policy struggles more and where non-linear and collision-related interactions occur more frequently. We therefore focus refinement on the top-entropy samples instead of applying numerical expert feedback to the entire training set. The random and bottom-entropy rows are included only to characterize sample difficulty, not as refinement baselines.

Table 8: Additional evaluation. Forecasting entries are ADE / FDE. Non-linear errors are in meters and ETH-UCY-to-SDD transfer errors are in pixels. Inference is measured on an RTX 4090. The reference is the base policy for non-linear subsets and LMTraj-SUP for transfer and cost.

Evaluation	Reference	MoRE (Ours)
Non-linear group walking	0.297 / 0.323	**0.211 / 0.274**
Non-linear mimicry walking	0.195 / 0.277	**0.151 / 0.217**
Non-linear collision avoidance	0.251 / 0.288	**0.198 / 0.264**
Non-linear subset average	0.248 / 0.296	**0.187 / 0.252**
ETH-UCY $\rightarrow$ SDD	10.03 / 17.93	**9.14 / 15.32**
Inference memory (MB)	1,401	1,401
Inference latency (ms)	18.3	18.3

Hard cases, transfer, and inference cost. Tab. 8 evaluates MoRE only on challenging non-linear motion and interaction cases, using behavioral subsets that are distinct from the entropy-ranked training subset. MoRE improves both metrics in all three subsets, reducing the reported average ADE by approximately 24.6%. This focused evaluation shows that the gains remain substantial on challenging non-linear motion and interaction cases. In ETH-UCY-to-SDD transfer without fine-tuning, FDE decreases from 17.93 to 15.32 pixels, approximately 14.6%, showing that the refinement also transfers beyond the source dataset. Reported RTX 4090 measurements remain 1,401 MB and 18.3 ms, confirming that the numerical experts introduce no measured inference-time overhead. Sec. C provides full per-scene results, expert configurations, collision ablations, coefficient sweeps, and deterministic comparisons.

5 Conclusion

MoRE refines a pretrained language-based trajectory predictor using cached coordinate-level rewards from complementary numerical experts, uncertainty-weighted consensus, and entropy-based sample selection. Across ETH-UCY, SDD, GCS, and NBA, MoRE improves stochastic forecasting accuracy, including non-linear subsets and cross-dataset transfer. On ETH-UCY, it also yields pedestrian spacing closer to the ground truth and fewer collisions. Because experts are used only during training, measured inference memory and latency remain unchanged, while ablations show that expert composition and reward aggregation both matter.

Limitations. MoRE depends on numerical experts, which may share errors or provide inaccurate forecasts. Expert disagreement can reflect valid alternative futures, so consensus does not guarantee safe or calibrated predictions. Deterministic gains remain small because available deterministic experts provide limited improvement, motivating stronger deterministic forecasters. Future work will investigate stronger deterministic experts and adaptive expert weighting to provide more reliable guidance. We also plan to extend MoRE to broader domains and evaluate multimodal coverage under larger distribution shifts.

AI USE STATEMENT

Generative AI tools were not used for any activities requiring disclosure, including synthetic data generation, theoretical model development, the formulation or proof of mathematical claims, research methodology design, method implementation, or result interpretation. The authors drafted all text and used generative AI tools solely to improve its readability. All edited text was reviewed by the authors, who take full responsibility for the final content of this work.

ETHICS STATEMENT

This work uses only existing, publicly available datasets for trajectory prediction. These datasets may have limited coverage of motion patterns and environments, and the proposed method may inherit biases present in the source data. Predicted trajectories are uncertain estimates rather than guaranteed outcomes of individual behavior. Deployment in safety-critical or privacy-sensitive settings would require further validation and appropriate safeguards, particularly where predictions could support the tracking or monitoring of individuals. Any materials released with this work will comply with the applicable licenses and redistribution restrictions of the source datasets.

REPRODUCIBILITY STATEMENT

To support reproducibility, Sec. 3 presents the model formulation, learning objectives, sample selection, and implementation settings. Sec. A provides further details on the RL formulation, value network, optimization, hyperparameters, and numerical handling. Sec. 4.1 describes the datasets, evaluation protocols, and metrics, while Sec. C reports additional ablations, sensitivity analyses, transfer experiments, and computational costs. Additional qualitative results are provided in Sec. E. We intend to release our code and evaluation scripts upon acceptance, subject to internal approval.

REFERENCES

Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Inhwan Bae and Hae-Gon Jeon. Disentangled multi-relational graph convolutional network for pedestrian trajectory prediction. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Learning pedestrian group representations for multi-modal trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022a.

Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Non-probability sampling network for stochastic human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022b.

Inhwan Bae, Jean Oh, and Hae-Gon Jeon. EigenTrajectory: Low-rank descriptors for multi-modal trajectory forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Inhwan Bae, Junoh Lee, and Hae-Gon Jeon. Can language beat numerical regression? language-based multimodal trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024a.

Inhwan Bae, Young-Jae Park, and Hae-Gon Jeon. Singulartrajectory: Universal trajectory predictor using diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b.

Inhwan Bae, Junoh Lee, and Hae-Gon Jeon. Social reasoning-aware trajectory prediction via multimodal language model. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2025.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. arXiv preprint arXiv:2204.05862, 2022.

Apratim Bhattacharyya, Michael Hanselmann, Mario Fritz, Bernt Schiele, and Christoph-Nikolas Straehle. Conditional flow variational autoencoders for structured sequence prediction. arXiv preprint arXiv:1908.09008, 2020.

Niccoló Bisagno, Bo Zhang, and Nicola Conci. Group lstm: Group trajectory prediction in crowded scenarios. In *Proceedings of the European Conference on Computer Vision Workshop (ECCVW)*, 2018.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2020.

Wangxing Chen, Haifeng Sang, and Zishan Zhao. Pchgcn: Physically constrained higher-order graph convolutional network for pedestrian trajectory prediction. *IEEE Internet of Things Journal*, 2025.

Pranav Singh Chib and Pravendra Singh. Lg-traj: LLM guided pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2017.

Thomas Coste, Usman Anwar, Robert Kirk, and David Krueger. Reward model ensembles help mitigate overoptimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.

Patrick Dendorfer, Aljosa Osep, and Laura Leal-Taixe. Goal-gan: Multimodal trajectory prediction based on goal position estimation. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2020.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research (JMLR)*, 2022.

Tharindu Fernando, Simon Denman, Sridha Sridharan, and Clinton Fookes. Soft+ hardwired attention: An lstm framework for human trajectory prediction and abnormal event detection. *Neural Networks*, 108:466–478, 2018.

Yuxiang Fu, Qi Yan, Lele Wang, Ke Li, and Renjie Liao. Moflow: One-step flow matching for human trajectory forecasting via implicit maximum likelihood estimation based distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2023.

Tianpei Gu, Guangyi Chen, Junlong Li, Chunze Lin, Yongming Rao, Jie Zhou, and Jiwen Lu. Stochastic trajectory prediction via motion indeterminacy diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Edward T. Hall. *The Hidden Dimension*. Doubleday, 1966.

Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51 (5):4282, 1995.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531, 2015.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. arXiv preprint arXiv:2203.15556, 2022.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

Yingfan Huang, Huikun Bi, Zhaoxin Li, Tianlu Mao, and Zhaoqi Wang. Stgat: Modeling spatial-temporal interactions for human trajectory prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Ronny Hug, Wolfgang Hübner, and Michael Arens. Introducing probabilistic bézier curves for n-step sequence prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

Boris Ivanovic and Marco Pavone. The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural Computation*, 1991.

Yanlin Jiang, Yuchen Liu, and Mingren Liu. Cort-predictor: Chain of risk thought autoregressive trajectory predictor for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. arXiv preprint arXiv:2001.08361, 2020.

Sungsik Kim, Janghyun Baek, Jinkyu Kim, and Jaekoo Lee. GUIDE-CoT: goal-driven and user-informed dynamic estimation for pedestrian trajectory using chain-of-thought. arXiv preprint arXiv:2503.06832, 2025.

Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, Hamid Rezatofighi, and Silvio Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. In *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2019.

Parth Kothari, Sven Kreiss, and Alexandre Alahi. Human trajectory forecasting in crowds: A deep learning perspective. *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, 2021.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2017.

Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B. Choy, Philip H. S. Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Seongju Lee, Junseok Lee, Yeonguk Yu, Taeri Kim, and Kyoobin Lee. Mart: Multiscale relational transformer networks for multi-agent trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024.

Alon Lerner, Yiorgos Chrysanthou, and Dani Lischinski. Crowds by example. *Computer Graphics Forum*, 26(3):655–664, 2007.

Ruochen Li, Zhanxing Zhu, Tanqiu Qiao, and Hubert P. H. Shum. ViTE: Virtual Graph Trajectory Expert Router for Pedestrian Trajectory Prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(21):17535–17543, 2026. doi: 10.1609/aaai.v40i21.38808.

Junwei Liang, Lu Jiang, and Alexander Hauptmann. Simaug: Learning robust representations from simulation for trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

Yuejiang Liu, Qi Yan, and Alexandre Alahi. Social nce: Contrastive learning of socially-aware motion representations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101, 2017.

Karttikeya Mangalam, Harshayu Girase, Shreyas Agarwal, Kuan-Hui Lee, Ehsan Adeli, Jitendra Malik, and Adrien Gaidon. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

Weibo Mao, Chenxin Xu, Qi Zhu, Siheng Chen, and Yanfeng Wang. Leapfrog diffusion model for stochastic trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Ramin Mehran, Alexis Oyama, and Mubarak Shah. Abnormal crowd behavior detection using social force model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Ingrid Navarro and Jean Oh. Social-patternn: Socially-aware trajectory prediction guided by motion patterns. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2022.

Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2009.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research (JMLR)*, 2020.

Eike Rehder and Horst Kloeden. Goal-directed pedestrian prediction. In *Proceedings of the IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2015.

Alexandre Robicquet, Amir Sadeghian, Alexandre Alahi, and Silvio Savarese. Learning social etiquette: Human trajectory understanding in crowded scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016.

Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Tim Salzmann, Boris Ivanovic, Punarjay Chakravarty, and Marco Pavone. Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347, 2017.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. arXiv preprint arXiv:1701.06538, 2017.

Dereje Shenkut and BVK Vijaya Kumar. Visual-linguistic reasoning for pedestrian trajectory prediction. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

Liushuai Shi, Le Wang, Chengjiang Long, Sanping Zhou, Mo Zhou, Zhenxing Niu, and Gang Hua. Sgcn: Sparse graph convolution network for pedestrian trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Liushuai Shi, Le Wang, Chengjiang Long, Sanping Zhou, Fang Zheng, Nanning Zheng, and Gang Hua. Social interpretable tree for pedestrian trajectory prediction. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2022.

Jianhua Sun, Yuxuan Li, Hao-Shu Fang, and Cewu Lu. Three steps to multimodal trajectory prediction: Modality clustering, classification and synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Anirudh Vemula, Katharina Muelling, and Jean Oh. Social attention: Modeling attention in human crowds. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

Chuhua Wang, Yuchen Wang, Mingze Xu, and David J Crandall. Stepwise goal-driven networks for trajectory prediction. *IEEE Robotics and Automation Letters (RA-L)*, 2022.

Teng Wang, Yanting Lu, and Ruize Wang. Autotraces: Autoregressive trajectory forecasting via multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. arXiv preprint arXiv:2206.07682, 2022a.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Proceedings of the Neural Information Processing Systems (NeurIPS)*, 2022b.

Conghao Wong, Ziqian Zou, and Beihao Xia. Resonance: Learning to predict social-aware pedestrian trajectories as co-vibrations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Chenxin Xu, Maosen Li, Zhenyang Ni, Ya Zhang, and Siheng Chen. Groupnet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022a.

Chenxin Xu, Robby T Tan, Yuhong Tan, Siheng Chen, Yu Guang Wang, Xinchao Wang, and Yanfeng Wang. Eqmotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Pei Xu, Jean-Bernard Hayet, and Ioannis Karamouzas. Socialvae: Human trajectory prediction using timewise latents. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022b.

Zirui Xu, Biao Yang, Rongrong Ni, Zhongkai Zhou, and Shaobo Shen. W2w: Language-model-based trajectory prediction with reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Kota Yamaguchi, Alexander C Berg, Luis E Ortiz, and Tamara L Berg. Who are you with and where are you going? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

Shuai Yi, Hongsheng Li, and Xiaogang Wang. Understanding pedestrian behaviors from stationary crowd groups. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Shan You, Chang Xu, Chao Xu, and Dacheng Tao. Learning from multiple teacher networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2017.

Ye Yuan, Xinshuo Weng, Yanglan Ou, and Kris Kitani. Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Jiangbei Yue, Dinesh Manocha, and He Wang. Human trajectory prediction via neural social physics. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

He Zhao and Richard P. Wildes. Where are you heading? dynamic trajectory prediction with expert goal examples. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

A RL FORMULATION AND IMPLEMENTATION

This appendix expands the training procedure in Sec. 3. Extended related work appears in Sec. B. Additional experiments, interpretation, and qualitative results follow in Secs. C to E. The learning architecture, numerical reward definitions, and hyperparameters are retained from the main formulation.

A.1 MDP FORMULATION FOR TOKEN GENERATION

Trajectory generation is an episodic Markov decision process. At token step t, the state is $s_t = (\tau_{\text{obs}}, \tau_{<t})$ and the action is the next token $a_t = \tau_t \in \mathcal{V}$. Numerical feedback is evaluated after the entire token sequence is decoded into coordinates. Intermediate rewards are zero. The terminal reward is the bounded ground-truth and expert return used for training:

$$r_t = \begin{cases} 0, & t < L, \\ R, & t = L. \end{cases} \tag{9}$$

With discount factor $\gamma = 1$, every token receives the same terminal return. The raw score is defined in Eq. (3), and its clipping is described below. Expert predictions are functions of the observed trajectory, not of the generated token prefix. Consequently, caching these predictions once avoids repeated expert inference during policy refinement. Decoding and coordinate-error calculations still occur for each rollout.

A.2 VALUE NETWORK AND SHARED ADVANTAGE

The critic is a two-layer MLP with a ReLU activation, attached to the mean-pooled encoder features of the observation prompt, $\mathbf{h}_{\text{obs}} \in \mathbb{R}^{d_{\text{model}}}$. It predicts the terminal return from the initial context rather than the return from every generated prefix. The shared advantage estimate is

$$\hat{A} = R - V_\phi(\mathbf{h}_{\text{obs}}). \tag{10}$$

This is a sequence-return baseline, not a learned token-state value function or a generalized-advantage estimator. The actor's token-level terms share this scalar, while the critic is trained by regression to the return. The distinction matters because terminal-only feedback does not by itself require a context-only critic. It is the chosen lightweight baseline in this implementation.

A.3 TOKEN-LEVEL POLICY OBJECTIVE

The actor uses the clipped token-level surrogate in Eq. (4). The likelihood ratio is relative to the rollout behavior policy π_{old}, whereas the KL penalty is relative to the frozen pretrained reference π_{ref}. These policies have different roles. The same sequence-level advantage is used for every token, and the actor objective is averaged over the generated length. The supervised objective instead averages over the ground-truth sequence length.

The training schedule uses one gradient update per rollout, interleaved with supervised updates. Under a strictly on-policy single-step update, with no parameter change between sampling and loss evaluation, the importance ratios initially equal one and clipping is locally inactive. Thus, using a PPO surrogate should not be interpreted as independently demonstrating a benefit from clipping. The empirical comparisons evaluate the complete reward-guided refinement procedure, including its supervised and reference-policy regularization, rather than isolating every optimizer component.

A.4 HYPERPARAMETERS AND NUMERICAL HANDLING

One PPO update is interleaved with every ten supervised steps (`rl_step_frequency`=10). The weights are $\lambda_{\text{RL}} = 0.5$, $\lambda_{\text{exp}} = 0.5$, and $\lambda_{\text{uwo}} = 0.05$. PPO uses clipping threshold $\epsilon = 0.2$, value-loss coefficient $c_v = 0.5$, and reference-KL coefficient $\beta = 0.1$. Low-rank adapters have rank 16. AdamW uses a learning rate of 10^{-4} and an effective batch size of 1,024. The distributed configuration described in the manuscript uses eight RTX PRO 6000 GPUs with 128 samples per GPU.

Rollouts use temperature 0.7, top-k sampling with $k = 50$, and a maximum length of 100 tokens. The composite reward R in Eq. (3) is computed in coordinate space and then clamped to $[-10, 0]$ before PPO optimization, with -10 assigned to malformed decoded sequences. Expert and ground-truth errors are computed in the tokenizer's normalized coordinate frame, so the same clipping range is used across benchmarks. The advantage clamp value is 5. In the optimization equations, R denotes this bounded return used for training. The reported settings should be distinguished from test-time decoding, for which deterministic and stochastic evaluations are separate protocols.

A.5 Regularization and Scope

The frozen reference, supervised token likelihood, low-rank adaptation, and interleaved update schedule are intended to limit changes to the pretrained distribution. They do not guarantee preservation of all auxiliary-task capabilities or prevent every form of reward exploitation. Our evidence consists of forecasting accuracy, pedestrian-spacing statistics, collision rates, and the evaluated transfer setting. We do not measure an independent semantic-reasoning capability before and after refinement.

A.6 Geometry of the Mean Expert Reward

The following identity is an algebraic interpretation of the existing MSE reward, not an additional learning component. Let $d = 2T_{\text{pred}}$, $\bar{\mathbf{S}} = K^{-1} \sum_k \mathbf{S}_k$, and $C = K^{-1} \sum_k \|\mathbf{S}_k - \bar{\mathbf{S}}\|_F^2 / d$. Expanding the squares gives

$$\frac{1}{K} \sum_{k=1}^{K} \text{MSE}(\hat{\mathbf{S}}, \mathbf{S}_k) = \text{MSE}(\hat{\mathbf{S}}, \bar{\mathbf{S}}) + C, \tag{11}$$

$$\mu = -\text{MSE}(\hat{\mathbf{S}}, \bar{\mathbf{S}}) - C. \tag{12}$$

The cross term vanishes because $\sum_k (\mathbf{S}_k - \bar{\mathbf{S}}) = 0$. For a fixed observation and its cached experts, C is independent of the generated trajectory. The unclipped mean expert score therefore induces the same candidate ordering as distance to the average expert trajectory. It does not preserve the full multimodal predictive distribution of each expert.

The UWO term is the standard deviation of the scalar rewards $R_k(\hat{\mathbf{S}})$, which generally depends on the candidate. It penalizes unequal expert assessments of that candidate, not the fixed pairwise dispersion of cached expert trajectories alone. This explains why the mean-aggregation control is important. The identity does not imply that distillation and policy optimization have identical gradients, optimization dynamics, or regularization. Reward and advantage clipping further qualify any such equivalence.

B Extended Related Work

B.1 Numerical Trajectory Prediction

Pioneering works (Helbing & Molnar, 1995; Pellegrini et al., 2009; Mehran et al., 2009; Yamaguchi et al., 2011) rely on physics-based formulations such as social force models that describe pedestrian dynamics through attractive and repulsive forces. With the advent of deep learning, Social-LSTM (Alahi et al., 2016) introduces data-driven recurrent modeling with social pooling to capture interactions between pedestrians. Subsequent methods extend this paradigm by employing conditional variational autoencoders (Lee et al., 2017; Bhattacharyya et al., 2020; Mangalam et al., 2020; Sun et al., 2021; Wang et al., 2022; Xu et al., 2022a;b), generative adversarial networks (Gupta et al., 2018; Sadeghian et al., 2019), and attention mechanisms (Fernando et al., 2018; Ivanovic & Pavone, 2019; Salzmann et al., 2020; Vemula et al., 2018). Graph-based architectures such as GCNs (Mohamed et al., 2020; Shi et al., 2021), GATs (Bae et al., 2022b; Huang et al., 2019; Kosaraju et al., 2019; Liang et al., 2020) and transformers (Yuan et al., 2021; Bae et al., 2022b) enable direct modeling of pairwise interactions through spatial-temporal graphs, with extensions incorporating velocity- and distance-aware affinity (Bae & Jeon, 2021). Beyond pairwise interactions, group-aware methods (Bae et al., 2022a; Xu et al., 2022a) explicitly model intra-group coherence and inter-group avoidance. To capture generalizable motion patterns, recent approaches explore basis trajectory

decomposition (Bae et al., 2023) with diffusion priors (Bae et al., 2024b) that learn low-rank representations across diverse scenarios. Meanwhile, goal-conditioned approaches (Mangalam et al., 2020; Zhao & Wildes, 2021; Dendorfer et al., 2020) leverage predicted destination information to guide trajectory interpolation, and diffusion models (Gu et al., 2022; Mao et al., 2023) further enable better stochastic prediction. Recent models further improve relational modeling and efficient forecasting (Xu et al., 2023; Lee et al., 2024; Fu et al., 2025; Chen et al., 2025; Li et al., 2026). These coordinate-based methods model interactions, groups, scene context, and motion patterns. Their different architectures provide complementary inductive biases (Kothari et al., 2021). Complementary priors include pattern-guided forecasting (Navarro & Oh, 2022), probabilistic Bézier curves (Hug et al., 2020), interpretable social trees (Shi et al., 2022), and neural social physics (Yue et al., 2022), alongside group (Bisagno et al., 2018) and goal-directed prediction (Rehder & Kloeden, 2015).

B.2 Language-based Trajectory Prediction

Recently, inspired by the success of Large Language Models (LLMs) in high-level reasoning tasks (Brown et al., 2020; Wei et al., 2022b;a) and driven by scaling laws (Kaplan et al., 2020; Hoffmann et al., 2022), there have been attempts to incorporate language priors into trajectory forecasting. These approaches typically recast trajectory sequences as text-based prompts, allowing the model to interpret coordinates as discrete tokens and formulate prediction as a generative question-answering task (Bae et al., 2024a; 2025; Kim et al., 2025). To learn contextual representations, LMTraj learns auxiliary tasks such as collision assessment and group-member estimation (Bae et al., 2024a). Its multimodal extension instead uses a modality connector to incorporate visual information (Bae et al., 2025). Recent methodologies also explore task-specific enhancements, such as fine-tuning models for egocentric pedestrian reasoning (Shenkut & Kumar, 2025) and employing goal-oriented visual prompts with chain-of-thought reasoning (Kim et al., 2025). Some frameworks leverage LLMs as a submodule to incorporate high-level motion cues for numerical forecasting (Chib & Singh, 2025). More recent approaches directly formulate trajectory prediction as autoregressive language generation (Wang et al., 2026) or optimize a language-based trajectory policy with reinforcement learning (Xu et al., 2026), with related extensions to autonomous driving (Jiang et al., 2026). MoRE focuses on transferring the predictions of frozen numerical forecasters into a pretrained language-based policy as reward feedback. For predictors trained only with token likelihood, the objective does not directly penalize distances in continuous coordinate space. MoRE supplies this feedback without changing the inference architecture.

B.3 Multi-Expert Learning and Reward-based Optimization

Integrating insights from diverse models is a well-established paradigm for enhancing representational capacity, spanning from knowledge distillation (KD) (Hinton et al., 2015; You et al., 2017) to Mixture-of-Experts (MoE) architectures (Jacobs et al., 1991; Shazeer et al., 2017; Fedus et al., 2022). These mechanisms combine experts at the level of features or outputs, whereas MoRE uses numerical predictions to score trajectories sampled from the language model. Reward learning from human preferences (Christiano et al., 2017) supplies feedback beyond supervised targets. Subsequent work applies this approach to language-model alignment (Ouyang et al., 2022; Bai et al., 2022), often using PPO (Schulman et al., 2017). While conventional alignment relies on human annotations, recent strategies increasingly formulate programmatic reward functions to explicitly enforce domain-specific constraints. A critical issue in this monolithic reward optimization process is reward overoptimization (Gao et al., 2023), wherein a policy exploits flaws in the proxy reward at the expense of the underlying task objective. To mitigate this, Coste et al. (2024) showed that ensembling reward models offers useful disagreement signals (Lakshminarayanan et al., 2017), using disagreement for conservative policy updates.

Our use of multiple experts concerns the reward construction, rather than inference-time expert routing. Consensus rewards and uncertainty-driven mining guide this transfer without inference-time experts.

C Additional Experimental Results

C.1 Sources and Reporting Conventions

The references for Tab. 1 are Social GAN (Gupta et al., 2018), Social-STGCNN (Mohamed et al., 2020), PECNet (Mangalam et al., 2020), Trajectron++ (Salzmann et al., 2020), AgentFormer (Yuan et al., 2021), MID (Gu et al., 2022), EqMotion (Xu et al., 2023), GP-Graph (Bae et al., 2022a), LED (Mao et al., 2023), MART (Lee et al., 2024), SingularTrajectory (Bae et al., 2024b), MoFlow (Fu et al., 2025), PCHGCN (Chen et al., 2025), ViTE (Li et al., 2026), LMTraj-SUP (Bae et al., 2024a), GUIDE-CoT (Kim et al., 2025), and W2W (Xu et al., 2026). The NBA comparison follows Wong et al. (2025).

Benchmark and ablation results are reproduced at their displayed precision. The main benchmark table uses the reporting precision shown for each benchmark, while detailed ablations retain higher-precision values when available. For example, the LMTraj-SUP ETH result is 0.409 / 0.501 in the detailed integration experiment and is reported as 0.41 / 0.50 in Tab. 1. Likewise, the ETH-UCY average is reported as 0.22 / 0.32 in the main benchmark table and 0.224 / 0.318 in the controlled integration comparison. Percentage reductions in the main text are approximate calculations from the displayed benchmark values. Behavioral subsets may overlap, so the reported aggregate in Tab. 10 is evaluated separately and is not the arithmetic mean of the four displayed subset columns.

C.2 Numerical Expert Selection

Table 9: Numerical expert configurations on ETH-UCY (ADE / FDE, meters). The proposed five-expert model is the reference for leave-one-out, replacement, and addition tests.

Configuration	**ADE / FDE**
(a) Single-expert alignment	
SingularTrajectory	0.211 / 0.315
MART	0.207 / 0.314
MoFlow	0.206 / 0.307
(b) Leave-one-out removal	
MoRE without Social-STGCNN	0.207 / 0.291
MoRE without DMRGCN	0.206 / 0.291
MoRE without GP-Graph	0.208 / 0.293
MoRE without SingularTrajectory	0.208 / 0.291
MoRE without Expert-Trajectory	0.206 / 0.298
(c) Replace SingularTrajectory	
SingularTrajectory $\rightarrow$ MART	0.207 / 0.289
SingularTrajectory $\rightarrow$ MoFlow	0.206 / 0.288
(d) Add a sixth expert	
MoRE + LED	0.206 / 0.290
MoRE + MART	0.205 / 0.288
MoRE + MoFlow	0.205 / 0.287
(e) Ground-truth reward ablation	
MoRE without ground-truth reward	0.211 / 0.297
MoRE (five experts)	**0.204 / 0.285**

Table 9(a) examines alignment with additional individual forecasters. In panel (b), removing any of the five experts raises both reported aggregate errors relative to the full configuration. Panels (c) and (d) show that replacing SingularTrajectory or adding a sixth expert does not improve the evaluated five-expert result. These tests support the selected combination. They do not establish that it is optimal among all available forecasters. Increasing the number of experts also increases the amount of prediction caching, even though the experts remain unnecessary at inference.

Panel (e), also summarized in Tab. 4, removes the ground-truth reward and gives 0.211 / 0.297, compared with 0.204 / 0.285 for the full reward. This supports retaining the ground-truth anchor.

It is not the converse ablation of removing all expert rewards: the two questions should not be conflated.

Table 10: Refinement with individual experts on ETH-UCY behavioral subsets (ADE / FDE, meters). Behavioral subsets may overlap, and the reported aggregate is evaluated separately from the four displayed subsets.

Expert	Group	Mimicry	Non-linear	Collision	Reported avg.
None	0.206/0.256	0.164/0.232	0.322/0.474	0.200/0.275	0.223/0.309
STGCNN	0.202/0.252	0.158/0.212	0.312/0.455	0.215/0.292	0.221/0.301
DMRGCN	0.198/0.248	0.148/0.195	0.318/0.462	0.205/0.282	0.224/0.305
GP-Graph	0.192/0.242	0.152/0.202	0.310/0.450	0.192/0.268	0.216/0.298
SingularTraj.	0.212/0.268	0.145/0.200	0.292/0.428	0.202/0.278	0.218/0.296
ExpertTraj.	0.210/0.266	0.160/0.216	0.308/0.448	0.198/0.260	0.220/0.290
All (Ours)	**0.188/0.238**	**0.142/0.192**	**0.282/0.412**	**0.185/0.255**	**0.208/0.282**

Table 10 reports both ADE and FDE for all five individual experts and their combination. Since the behavioral subsets may overlap, the reported aggregate is not obtained by averaging the four subset columns. Individual experts show different strengths across behaviors, while the combined reward gives the lowest ADE and FDE in all four displayed subsets and the lowest reported aggregate error. These subset results are distinct from the per-scene integration comparison in Tab. 12.

C.3 Collision-Avoiding Performance

Table 11: Collision rates on ETH-UCY (%). Lower is better. The baseline values also appear in Tab. 2.

Baseline	COL	Integration	COL
GP-Graph	2.44	Ensemble	3.67
LED	2.62	Knowledge distillation	2.92
MART	2.23	Maximum aggregation	2.79
SingularTrajectory	2.96	Mean aggregation	2.37
LMTraj-SUP	2.54	**UWO (MoRE)**	**2.11**

Collision rates are computed across predicted multimodal samples under the protocol of Liu et al. (2021). Table 11 compares output/target transfer with reward aggregation. UWO has the lowest reported collision rate, followed by mean aggregation among the integration strategies. Maximum aggregation improves over the displayed output-level alternatives but is worse than the base policy. Thus, reward-based training alone is not sufficient to ensure improvement. The aggregation rule matters in these comparisons. The distance-zone measurements in Tab. 2 assess agreement with observed spacing, not complete social compliance.

C.4 PER-SCENE EXPERT INTEGRATION

Table 12: Per-scene expert integration on ETH-UCY (ADE / FDE, meters). Aggregate values are reproduced at their reported precision.

Method	**ETH**	**HOTEL**	**UNIV**	**ZARA1**	**ZARA2**	**Reported avg.**
Base policy	.409/.501	.120/.159	.218/.344	.199/.318	.175/.272	.224/.318
Supervised FT	.401/.514	.108/.148	.247/.353	.211/.319	.168/.271	.227/.321
Ensemble	.421/.612	.173/.261	.292/.514	.231/.382	.203/.331	.264/.420
Distillation	.382/.492	.122/.155	.211/.338	.196/.301	.177/.277	.218/.313
Reward-weighted reg.	.388/.499	.121/.158	.215/.342	.198/.305	.173/.283	.219/.317
Minimum-risk training	.412/.515	.135/.172	.230/.355	.210/.315	.188/.298	.235/.331
Expert-guided FT	.386/.495	.124/.157	.213/.339	.196/.302	.186/.287	.221/.316
Max aggregation	.377/.503	.116/.152	.223/.340	.192/.296	.182/.279	.218/.314
Mean aggregation	.393/.482	.111/.141	.218/.345	.190/.298	.173/.268	.217/.307
UWO (MoRE)	.358/.424	.111/.140	.206/.323	.180/.275	.166/.263	.204/.285

Table 12 includes every evaluated integration strategy. It expands the aggregate comparisons in Tab. 4, including reward-weighted regression, minimum-risk training, and expert-guided fine-tuning without PPO. UWO is the only evaluated configuration that improves both ADE and FDE over the base policy in every displayed scene. Other approaches have mixed effects. For example, additional supervised fine-tuning improves HOTEL but worsens UNIV. Mean aggregation improves several scenes but slightly worsens UNIV FDE. The controlled subset comparison supports the complete MoRE procedure but does not replace a matched-compute or multi-seed study of every baseline.

C.5 REWARD-COEFFICIENT SENSITIVITY

Table 13: Reward-coefficient sensitivity on ETH-UCY (ADE / FDE, meters), with $\lambda_{\mathrm{RL}} = 0.5$. Columns vary λ_{uwo}. Zero is mean aggregation.

λ_{exp}	**0**	**0.01**	**0.05**	**0.1**	**0.5**
0.1	0.223/0.316	0.218/0.309	0.211/0.297	0.213/0.300	0.222/0.315
0.3	0.222/0.312	0.212/0.298	0.208/0.289	0.214/0.292	0.217/0.306
0.5	0.217/0.307	0.210/0.295	**0.204/0.285**	0.206/0.288	0.211/0.299
0.7	0.219/0.310	0.217/0.302	0.207/0.289	0.211/0.306	0.219/0.316
1.0	0.221/0.309	0.216/0.300	0.210/0.294	0.215/0.297	0.221/0.312

Table 14: Sensitivity to λ_{RL} on ETH-UCY (ADE / FDE, meters), with $\lambda_{\mathrm{exp}} = 0.5$ and $\lambda_{\mathrm{uwo}} = 0.05$.

λ_{RL}	**0 (SFT)**	**0.1**	**0.3**	**0.5**	**0.7**	**1.0**
ADE / FDE	.227/.321	.213/.301	.207/.290	**.204/.285**	.206/.289	.213/.303

Table 13 fixes $\lambda_{\mathrm{RL}} = 0.5$ and varies expert weight and score-disagreement penalty. Table 14 fixes $\lambda_{\mathrm{exp}} = 0.5$ and $\lambda_{\mathrm{uwo}} = 0.05$ and varies the RL weight. The 5×5 grid and six RL weights contain 30 distinct configurations because the selected setting occurs in both tables. All are reported to converge without divergence or collapse. In the local range $\lambda_{\mathrm{exp}} \in [0.3, 0.7]$ and $\lambda_{\mathrm{uwo}} \in [0.01, 0.1]$, ADE ranges from 0.204 to 0.217.

At every tested expert weight, $\lambda_{\mathrm{uwo}} = 0.05$ improves over mean aggregation ($\lambda_{\mathrm{uwo}} = 0$). Setting $\lambda_{\mathrm{RL}} = 0$ gives additional supervised fine-tuning on the same hard subset and does not explain MoRE's improvement. These experiments use the interleaved update schedule and reference regularization. They support sensitivity claims only over the tested coefficient range, not a general statement that reward balance cannot destabilize training.

C.6 Complete Refinement-Efficiency Study

Table 15 retains all six settings behind the selected comparisons in Tab. 6. The 0.1%, 1%, 10%, 50%, and 100% rows vary the fraction used for LoRA refinement. Full fine-tuning uses the same 1% subset as the selected LoRA setting. LoRA on 1% gives the lowest reported ADE and FDE. Its advantage over the 0.1% setting is larger for ADE than for FDE, whereas expanding the subset does not consistently improve either metric.

Table 15: Complete tuning and data-fraction comparison on ETH-UCY (ADE / FDE, meters). Parameter fractions refer to the forecasting backbone and exclude the critic.

Tuning	Parameters (%)	Data (%)	ADE	FDE
Full FT	100	1	<u>0.213</u>	0.297
LoRA	0.3	100	0.221	0.303
LoRA	0.3	50	0.220	0.312
LoRA	0.3	10	0.219	0.299
LoRA	0.3	0.1	0.223	<u>0.288</u>
LoRA (Ours)	0.3	1	**0.204**	**0.285**

C.7 Properties of Mined Samples

Table 16: Entropy-partitioned training samples on ETH-UCY. ADE / FDE is measured with the base policy. Constant-velocity error uses pixel coordinates. Behavioral labels follow Bae et al. (2024a).

Metric	Top 1%	Random 1%	Bottom 1%	All
Mean entropy (nats)	2.8	1.3	0.4	1.3
Base-policy ADE / FDE (m)	0.40/0.60	0.24/0.35	0.05/0.08	0.22/0.32
Constant-velocity error (px)	11.0	4.8	0.8	4.5
Non-linear (%)	82	45	2	41
Collision-involved (%)	25	13	1	9

The top-entropy subset has larger base-policy errors, a higher non-linear fraction, and more collision-involved interactions than the full training set, as shown in Tab. 16. Constant-velocity extrapolation, which is independent of learned models, gives the same ordering. Its pixel-coordinate errors should not be directly compared in magnitude with the meter-coordinate ADE / FDE entries.

Entropy is used as a relative ranking statistic. A transformation that preserves the sample ordering leaves selection unchanged. Uncertainty errors that change the ordering can affect it. The random and bottom-entropy subsets in this table characterize data difficulty. They are not post-training comparisons of MoRE fitted to those subsets, and therefore do not independently establish the optimality of entropy-based selection over random selection.

C.8 Cross-Dataset Transfer

Table 17: Cross-dataset transfer from ETH-UCY to SDD without fine-tuning on SDD. ADE / FDE is measured in pixels.

Model	ADE	FDE
LMTraj-SUP	10.03	17.93
MoRE	**9.14**	**15.32**

The transfer values in Tab. 8 are reproduced in Tab. 17. In this evaluation, the model is trained on ETH-UCY and tested on SDD without fine-tuning. The improvement from 17.93 to 15.32 pixels corresponds to a 14.6% reduction in FDE. This supports transfer of the refined supervised backbone

in this particular source-to-target setting. It is distinct from LMTraj-ZERO, which tests a general-purpose foundation model without trajectory-specific supervised training. One transfer experiment does not establish robustness to all domain shifts.

C.9 Deterministic Evaluation

Table 18: Deterministic and best-of-20 results on ETH-UCY (ADE / FDE, meters).

Model	Deterministic	Stochastic
SocialVAE	0.54 / 1.12	0.21 / 0.33
EigenTrajectory	0.51 / 1.11	0.21 / 0.34
LMTraj-SUP	0.477 / 0.880	0.22 / 0.32
MoRE	0.476 / 0.877	0.20 / 0.29
MoRE + deterministic experts	0.477 / 0.878	0.20 / 0.29

Table 18 reports deterministic and stochastic comparisons. MoRE's deterministic difference from LMTraj-SUP is 0.001 m ADE and 0.003 m FDE. Adding deterministic experts does not improve those reported values. These are small point-estimate differences rather than evidence of a substantial deterministic gain. The stochastic improvement is the main empirical result. A single sampled output should not automatically be equated with deterministic decoding. The column labels follow the reported evaluation modes.

C.10 Non-Linear Behavioral Subsets

Table 19: Non-linear behavioral subsets of ETH-UCY (ADE / FDE, meters). These subsets differ from the four categories in Tab. 5.

Method	Non-linear group walking	Non-linear mimicry walking	Non-linear collision avoidance	Subset average
Base policy	0.297/0.323	0.195/0.277	0.251/0.288	0.248/0.296
MoRE	0.211/0.274	0.151/0.217	0.198/0.264	0.187/0.252

The non-linear group-walking, mimicry, and collision-avoidance results are also summarized in Tab. 8. These subsets provide a more focused test than overall scene averages. MoRE improves both reported metrics in every category. The reported average ADE falls from 0.248 to 0.187 m, a 24.6% reduction. These additional non-linear subsets are not interchangeable with the broader Group, Mimicry, Non-linear, and Collision categories in Tab. 5.

C.11 NBA and Computational Cost

Table 20: Computational cost on ETH-UCY. Inference memory and latency are measured on an RTX 4090. Training times follow the reported setups and are not hardware-matched.

Model	Memory (MB)	Training (h)	Inference (ms)
PECNet	1,733	0.3	57.0
MID	2,929	6.9	35.0
AgentFormer	9,639	22.0	8.2
SocialVAE	1,762	2.1	73.0
LMTraj-SUP	1,401	3.8	18.3
MoRE (Ours)	1,401	3.8 + 1	18.3

The cost entries for LMTraj-SUP and MoRE also appear in Tab. 8. The NBA comparison in Tab. 3 extends evaluation to highly interactive motion and improves ADE from 1.10 to 0.96. Table 20 separately compares inference memory, latency, and reported training durations on ETH-UCY. Memory

refers to inference-time GPU memory, not the full training footprint of the policy, reference model, optimizer, critic, or expert precomputation. The training durations are not a hardware-matched speed ranking. The distributed refinement configuration is described in Sec. A.4.

C.12 Inference on Additional Hardware

Table 21: Additional inference measurements in FP32 with PyTorch and without deployment-specific optimization. These are predictor latencies, not end-to-end navigation measurements.

Platform	Latency (ms)	Rate (Hz)
NVIDIA DGX Spark	55	18.2
NVIDIA AGX Orin	90	11.1

Table 21 reports full-FP32 PyTorch inference without deployment-specific optimization. MoRE runs at 11.1 Hz on AGX Orin and 18.2 Hz on DGX Spark. These measurements characterize the predictor on the tested hardware. They do not include sensing, tracking, planning, control, or all multi-agent workloads of a deployed robot. No closed-loop navigation claim is inferred from these latency values.

D Discussion and Scope

D.1 Numerical Knowledge and Contextual Prediction

Numerical models and language-based models both learn from trajectories and may both encode social context. The relevant distinction here is the form of supervision: the base policy uses a token-level objective, whereas the experts provide decoded coordinate-space feedback. MoRE uses their architectural diversity as a source of priors. It does not identify which model understands an agent's true intention, and it does not treat visible group-like motion as proof that a model recognizes a particular social relationship.

D.2 Preservation of the Base Policy

The reference KL penalty, supervised loss, and restricted refinement subset constrain changes to the pretrained policy in different ways. The first penalizes distributional deviation, the second retains the ground-truth token objective on the selected data, and the third restricts which training contexts receive updates. These are design choices for stability, not guarantees of semantic fidelity. Because only the selected subset is used for refinement, behavior outside that subset must be assessed empirically. Good transfer on SDD is one such test, not an exhaustive guarantee.

D.3 Difference from Inference-Time Ensembling

An output ensemble combines numerical predictions during forecasting. MoRE instead computes numerical predictions before policy refinement and then deploys only the refined language-based policy. Distillation is also a training-time transfer method, so the absence of inference-time experts is not unique to MoRE. The relevant empirical question is whether the reward-based training procedure improves over these alternatives under the evaluated controls. The mean-reward identity in Sec. A.6 cautions against attributing the entire effect to information unavailable in an averaged expert target.

D.4 Remaining Evaluation Boundaries

Best-of-20 displacement metrics can improve without improving calibrated probability estimates or covering all plausible futures. Collision and distance-zone statistics add useful but partial information. Lower collision rate can also be influenced by sample diversity. Likewise, agreement between several experts does not imply correctness if their errors are correlated. The reported point estimates

do not provide confidence intervals across random seeds. These boundaries motivate future evaluation of mode coverage, expert correlation, and statistically reliable gains, rather than broader claims of semantic understanding or deployment safety.

E Additional Qualitative Results

We provide additional trajectory visualizations from ETH-UCY, SDD, and GCS. Numerical and language-based predictions can differ in both spread and spatial offset. MoRE shows improved alignment in selected examples, but these illustrations do not quantify uncertainty calibration, obstacle clearance, or an agent's latent intent. All claims about accuracy and social spacing rely on the corresponding quantitative results.

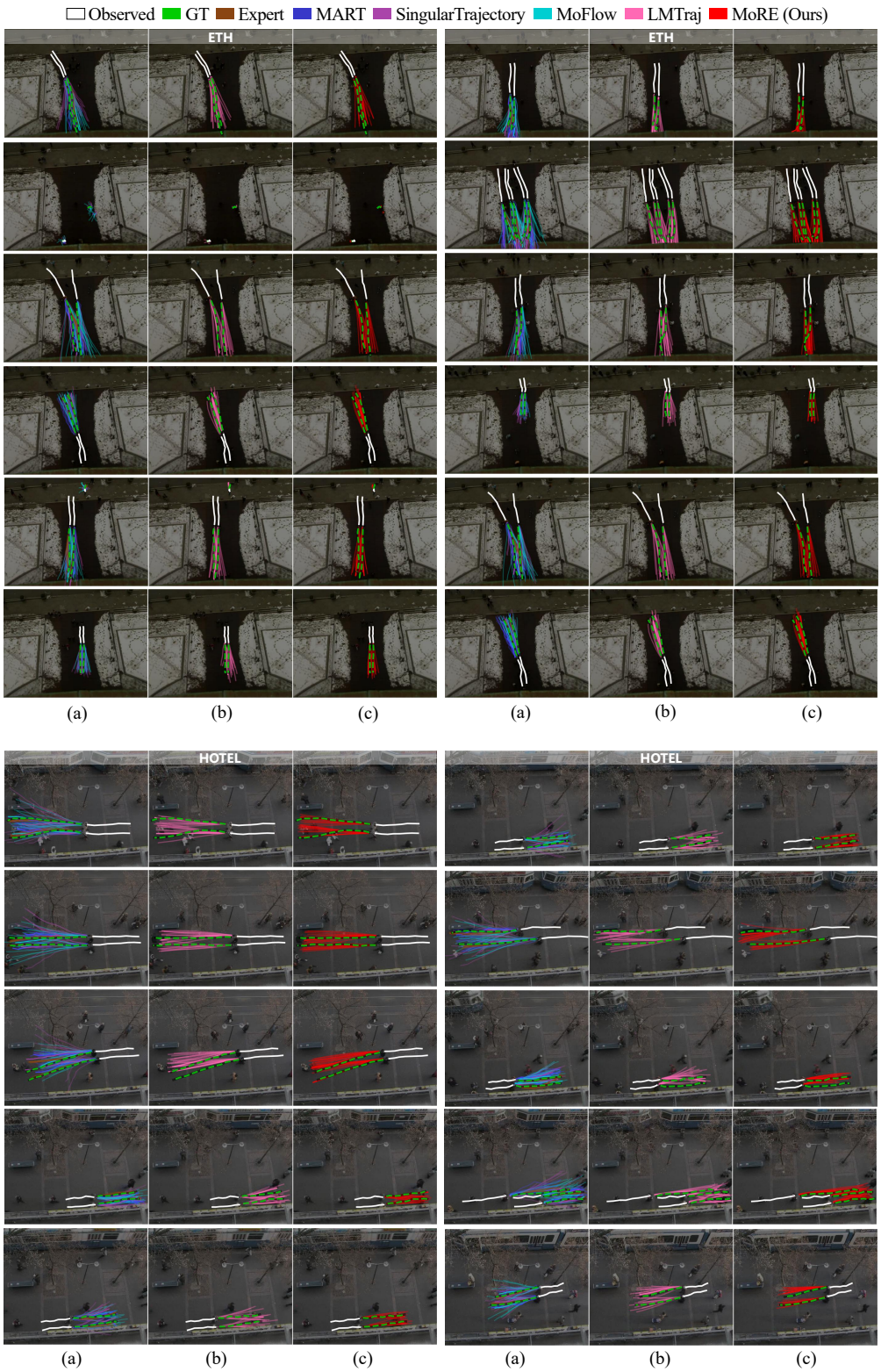

Figure 5: Visualization of prediction results on the ETH and HOTEL scenes. To aid visualization, selected agents are shown.

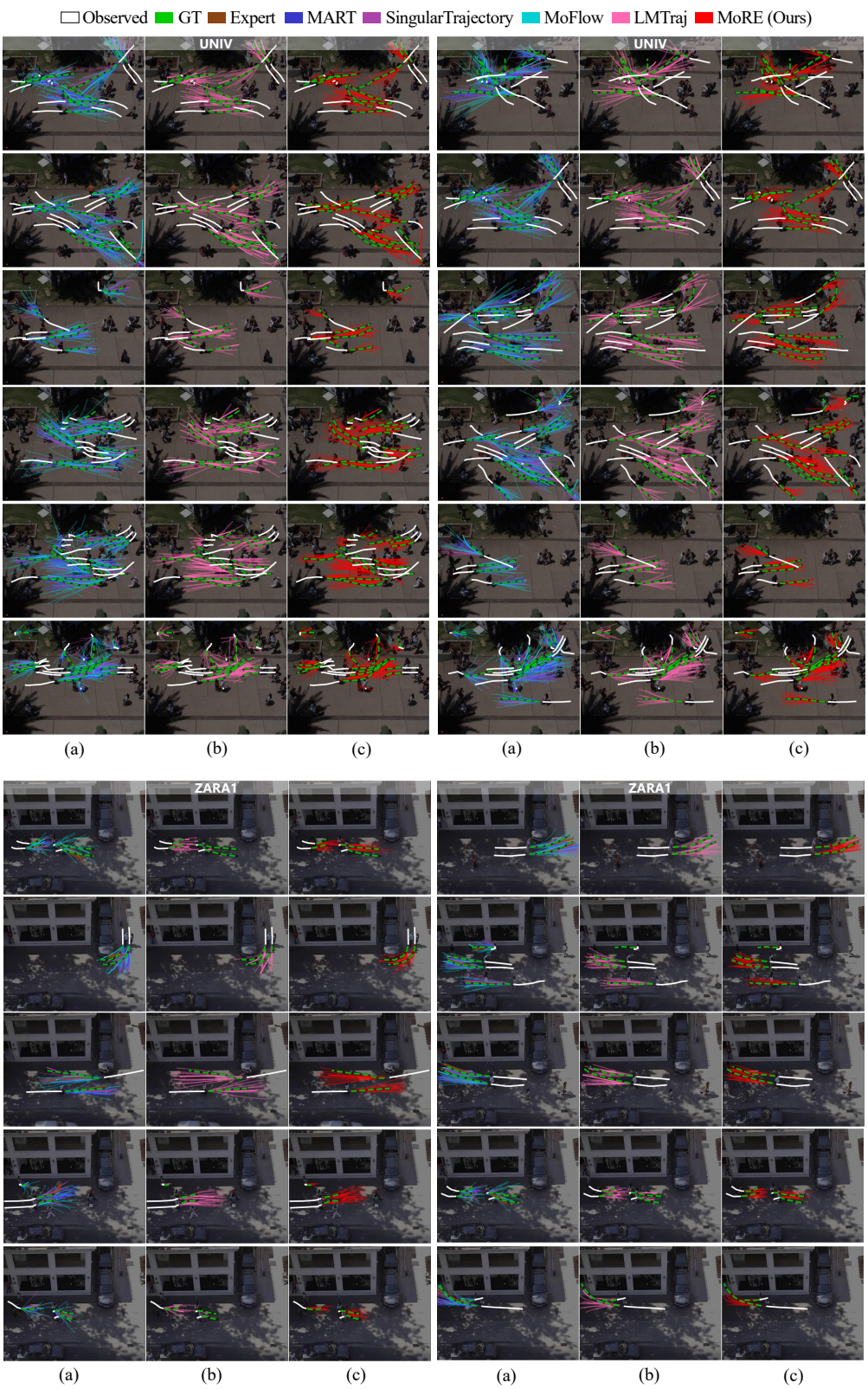

Figure 6: Visualization of prediction results on the UNIV and ZARA1 scenes. To aid visualization, selected agents are shown.

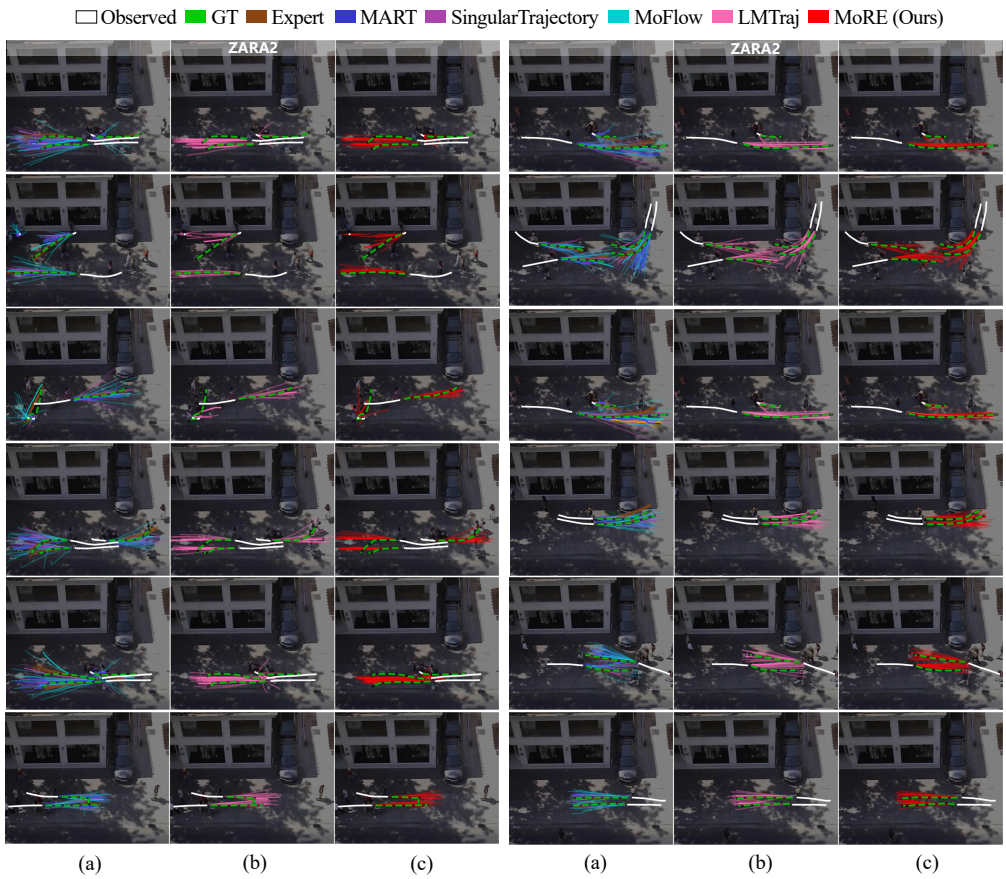

Figure 7: Visualization of prediction results on the ZARA2 scene. To aid visualization, selected agents are shown.

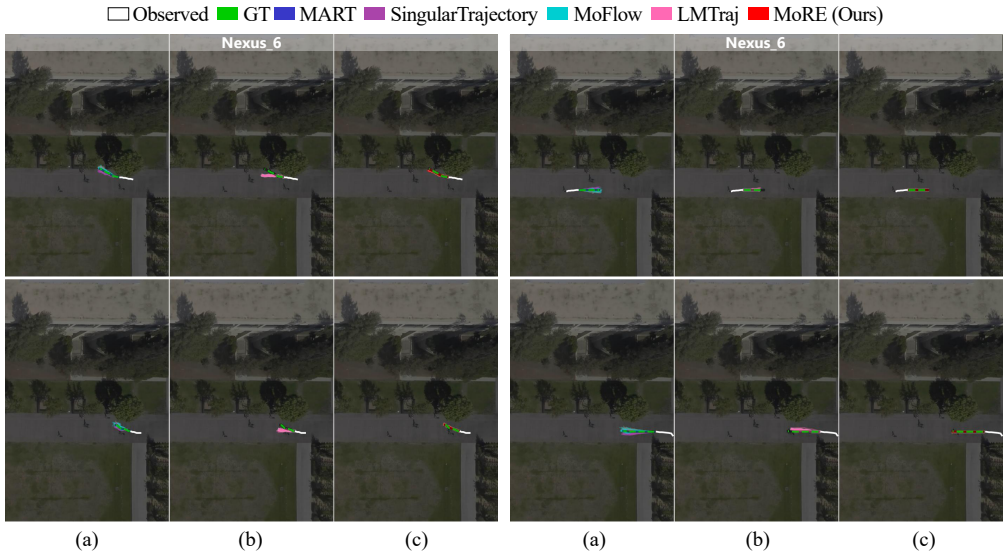

Figure 8: Visualization of trajectory predictions on nexus_6 scene. To aid visualization, selected agents are shown.

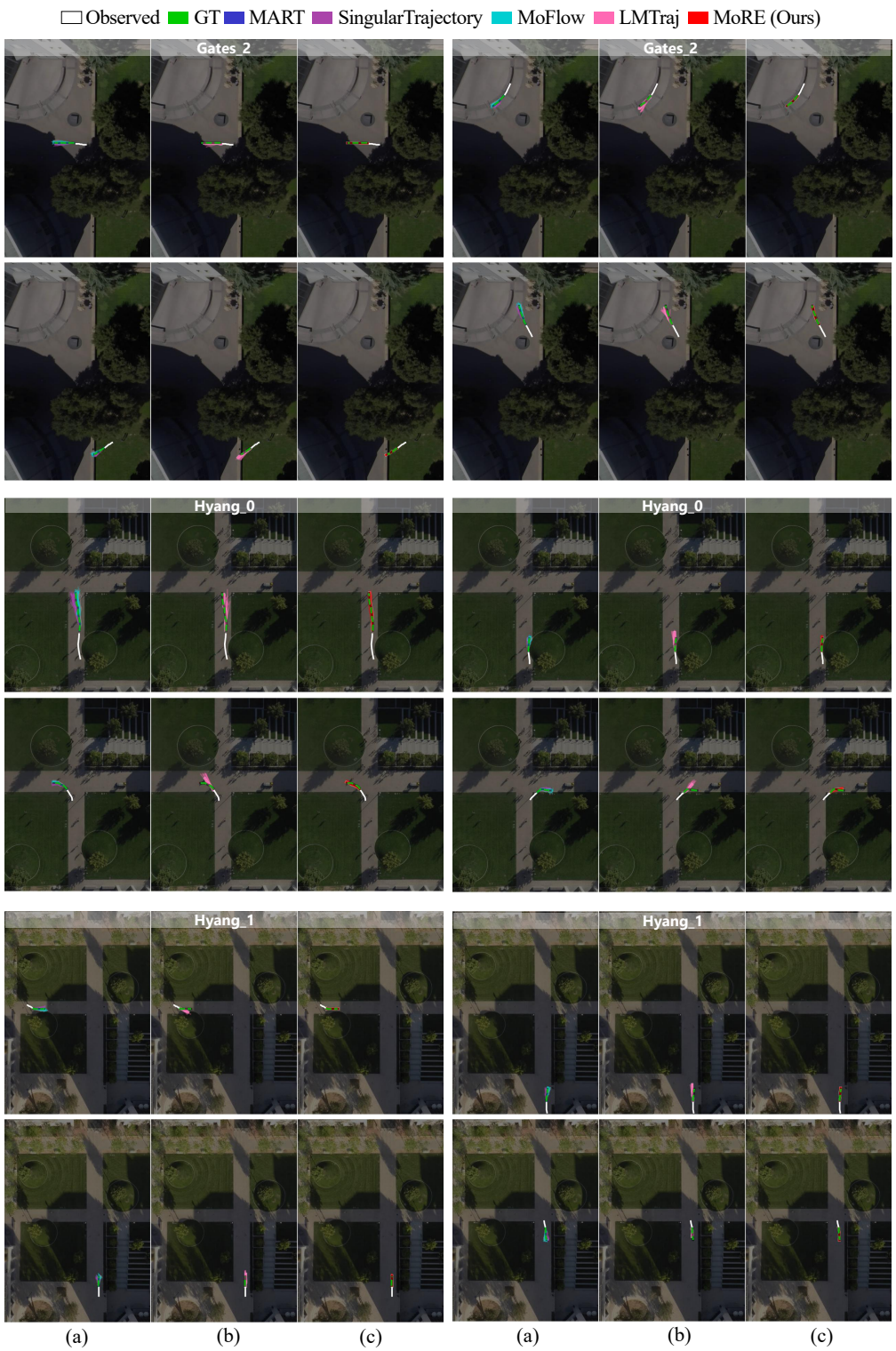

Figure 9: Visualization of trajectory predictions on gates_2, hyang_0, 1 scenes. To aid visualization, selected agents are shown.

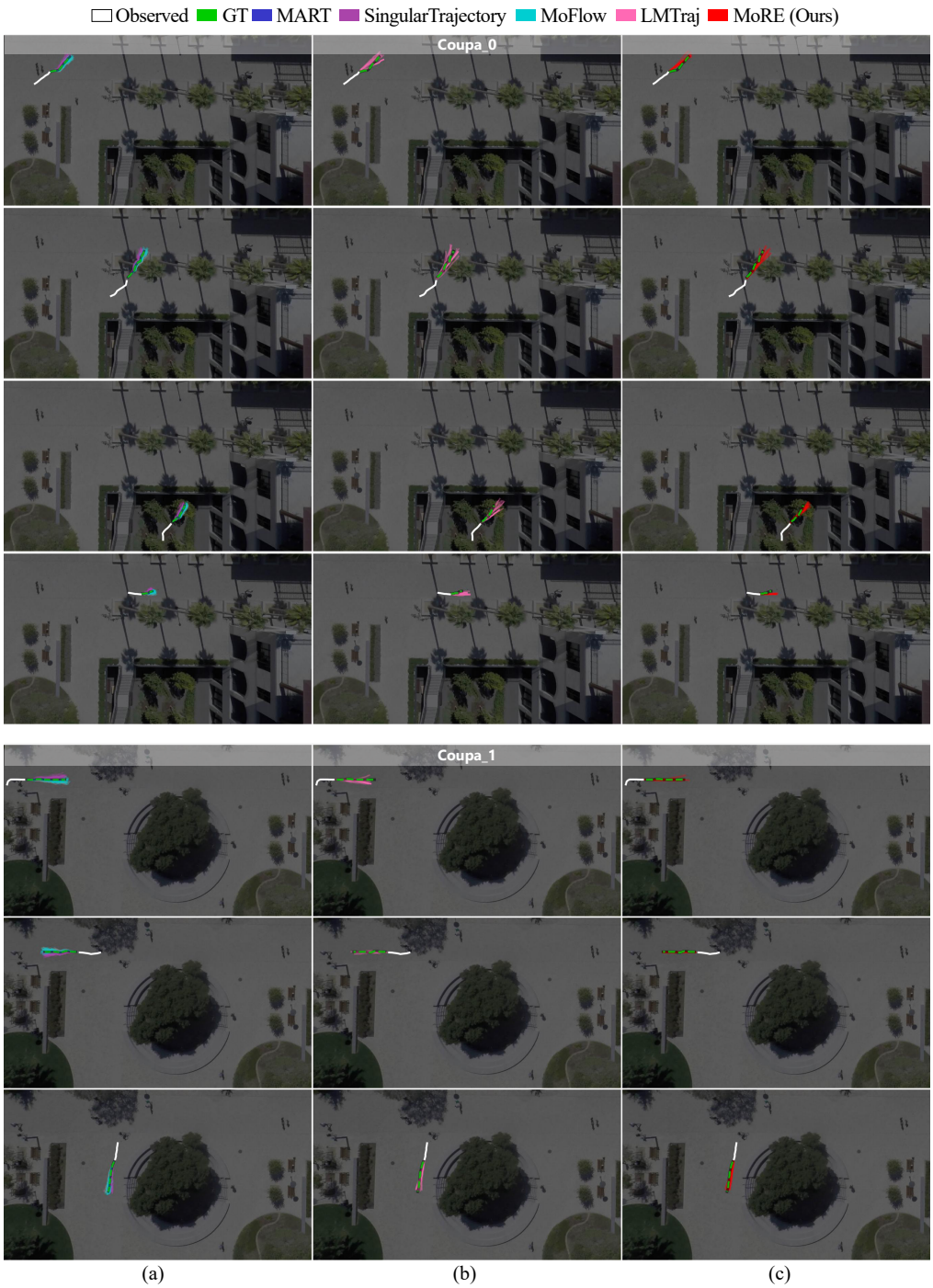

Figure 10: Visualization of prediction results on the coupa_0 and coupa_1 scenes. To aid visualization, selected agents are shown.



Figure 11: Visualization of trajectory predictions on hyang_3 and quad_0 scenes. To aid visualization, selected agents are shown.

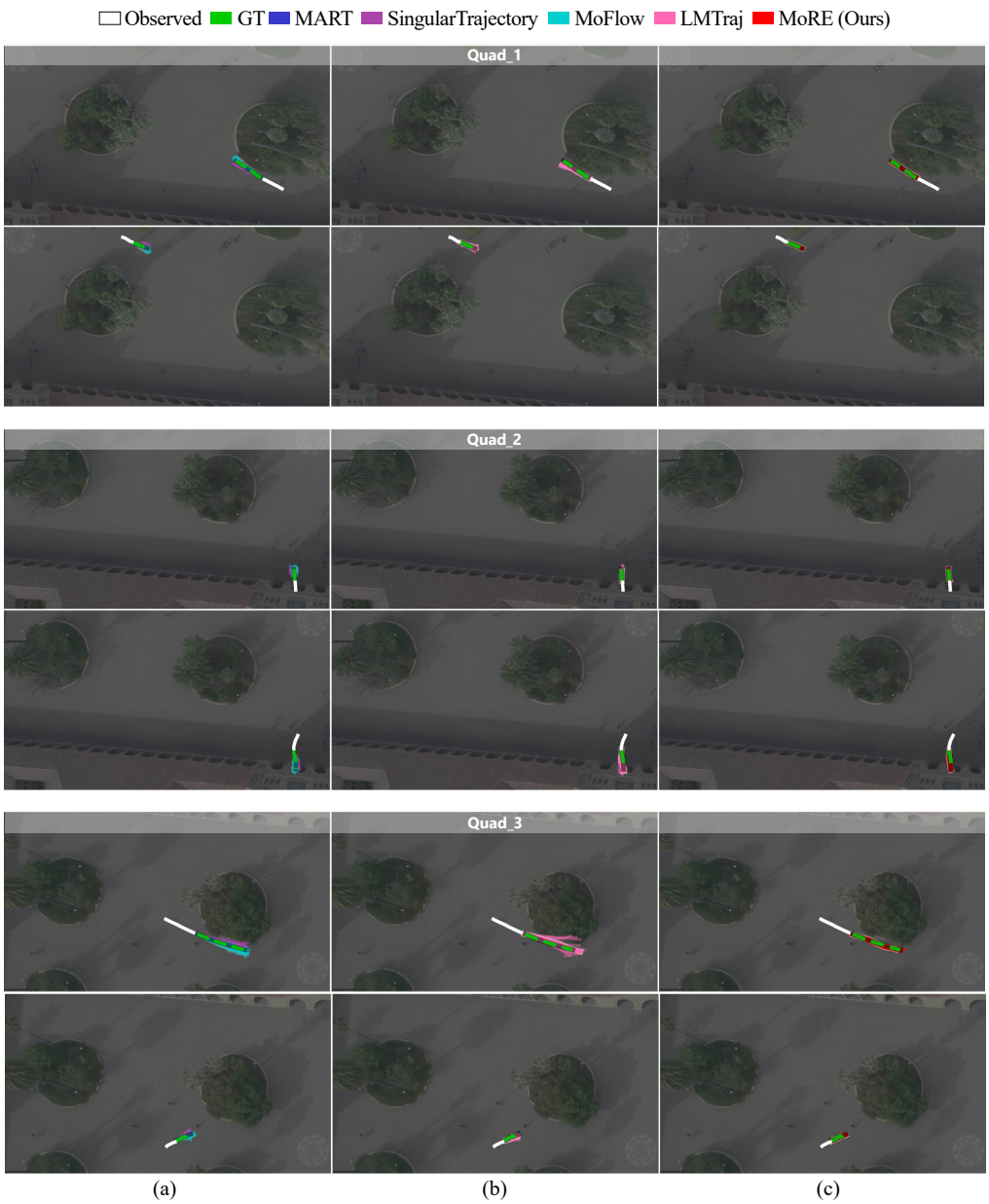

Figure 12: Visualization of prediction results on the quad_1, 2, 3 scenes. To aid visualization, selected agents are shown.

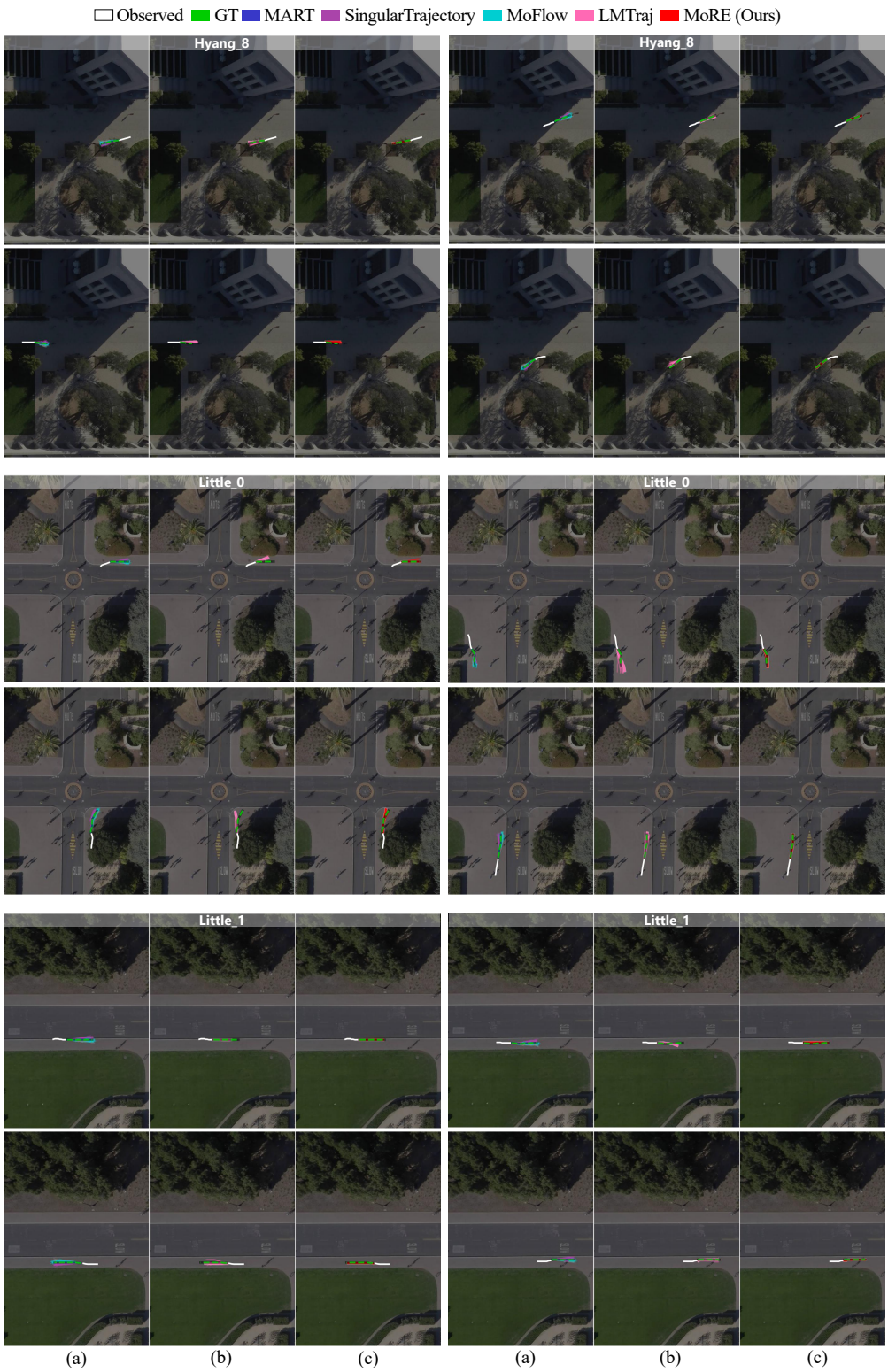

Figure 13: Visualization of prediction results on the hyang_8 and little_0, 1 scenes. To aid visualization, selected agents are shown.



Figure 14: Visualization of prediction results on the little_2, 3 and nexus_5 scenes. To aid visualization, selected agents are shown.

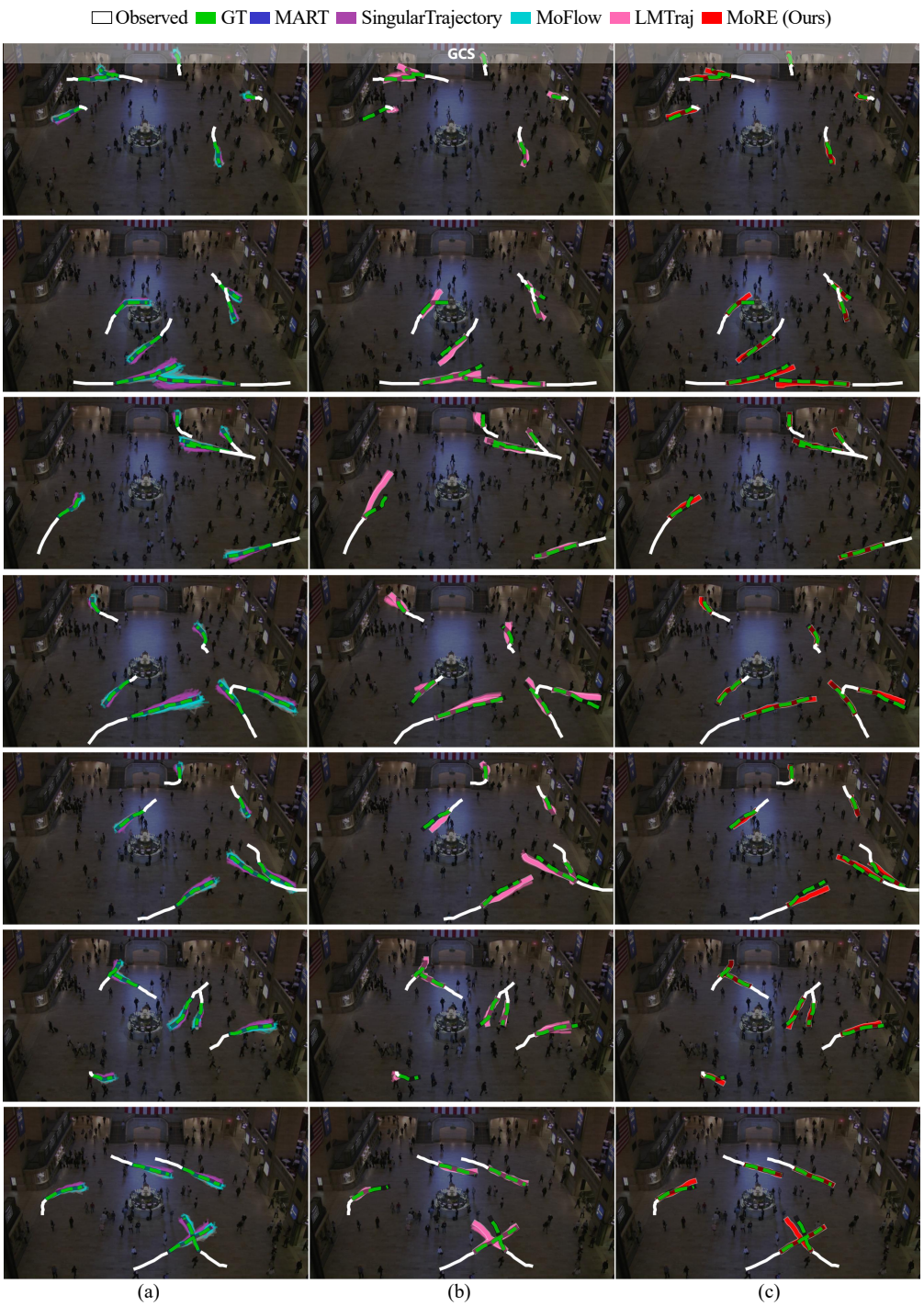

Figure 15: Visualization of prediction results on the GCS dataset. To aid visualization, selected agents are shown.