2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Distilling Knowledge for Short-to-Long Term Trajectory Prediction

Sourav Das[1], Guglielmo Camporese[2], Shaokang Cheng[3], and Lamberto Ballan[1]

***Abstract*— Long-term trajectory forecasting is an important and challenging problem in the fields of computer vision, machine learning, and robotics. One fundamental difficulty stands in the evolution of the trajectory that becomes more and more uncertain and unpredictable as the time horizon grows, subsequently increasing the complexity of the problem. To overcome this issue, in this paper, we propose *Di-Long*, a new method that employs the distillation of a short-term trajectory model forecaster that guides a student network for long-term trajectory prediction during the training process. Given a total sequence length that comprehends the allowed observation for the student network and the complementary target sequence, we let the student and the teacher solve two different related tasks defined over the same full trajectory: the student observes a short sequence and predicts a long trajectory, whereas the teacher observes a longer sequence and predicts the remaining short target trajectory. The teacher's task is less uncertain, and we use its accurate predictions to guide the student through our knowledge distillation framework, reducing long-term future uncertainty. Our experiments show that our proposed Di-Long method is effective for long-term forecasting and achieves state-of-the-art performance on the Intersection Drone Dataset (inD) and the Stanford Drone Dataset (SDD).**

I. Introduction

Pedestrian and vehicle trajectory forecasting has seen an increasing interest over the last few years due to the development of a large number of applications in robotics, autonomous driving, video surveillance, and embodied navigation [1], [2], [3], [4], [5]. Accurate prediction of an agent's movements is critical for these applications to ensure safety and improve efficiency. Short-term human trajectory forecasting has been extensively explored in the prior literature, and various methods have been proposed to predict the next few seconds of a person's planned intention [6], [7], [8], [9]. However, long-term human trajectory forecasting [10], [11], [12], which involves predicting human motion over a much longer time horizon, remains challenging due to the complexity of human behavior and the uncertainty of future interactions and intentions. On the other hand, the advent of transformers [13], [14], enabled a much broader long-range sequence processing, and in the context of trajectory forecasting, the long-term prediction task gained more attention with transformer-based architectures [12], [15], [16].

In addition to modeling the motion dynamics of individual agents, also contextual information from the environment semantics, from the social interaction with other agents, and from the long-term goals proved their importance in modeling trajectories [1], [17], [10]. For example, [1], [3] proposed methods that model social interactions among agents in the scene with (i) LSTMs that pool agent's information in a neighbor local region of a given location or (ii) employing a GAN for generating socially compliant trajectories. Furthermore, [10] proposed to condition the future predictions with the environment semantic maps that encoded physically compliant trajectories, thus creating a spatially and semantically-aware model. Moreover, the authors proposed to learn long-term and intermediate goals with a specialized U-Net network and to condition future forecasting with them.

In this work, we build upon these ideas and we propose the *Di-Long model* which aims to further improve the performance of long-term trajectory forecasting approaches by reducing the model's future uncertainty with a short-to-long knowledge distillation framework (see Fig. 1). The core idea of knowledge distillation [18] is to transfer knowledge from an expert large model ("teacher") to a smaller model ("student"), in order to improve the latter with the additional guidance of the expert teacher. To this end, we apply knowledge distillation to efficiently train models for long-term forecasting, with a teacher that is an expert in the

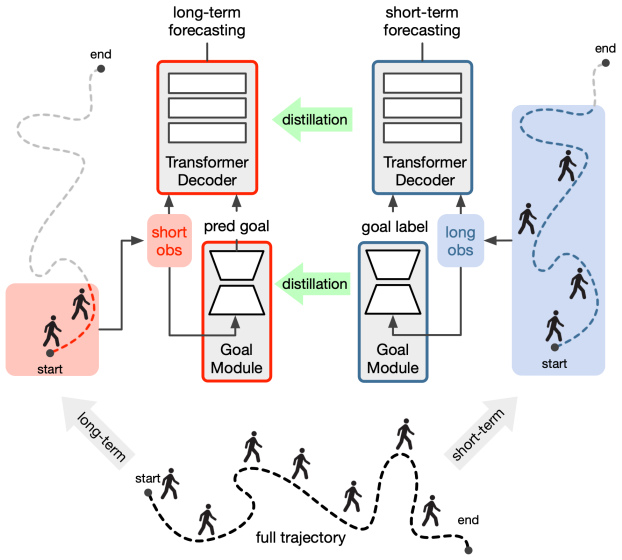

Fig. 1: **Di-Long Framework:** at the bottom, we depict the full trajectory from which the observations and targets of the student and the teacher are extracted. On top we show the components of our framework: the student transformer decoder predicts the long-term trajectory that is distilled from the teacher's prediction (based on longer sequences). The student's decoder is conditioned on the student goal module, distilled from a teacher's goal module.

[1]Sourav Das and Lamberto Ballan are with the Department of Mathematics "Tullio Levi-Civita", University of Padova, Italy. [2]Guglielmo Camporese recently earned his Ph.D. from the same department and he is currently a Research Engineer at Disney Research, Zurich. [3]S. Cheng is with the School of Automation, Northwestern Polytechnical University, China.

short-term prediction setting and guides the student on the long-term task during training. To the best of our knowledge, this idea has not been explored in the context of long-term trajectory forecasting. Our paper mostly focuses on human (pedestrian) trajectory prediction, and the Di-Long model shows state-of-the-art performance on two popular datasets, namely the Intersection Drone Dataset (inD) and the Stanford Drone Dataset (SDD), in different forecasting settings.

II. Related Work

Trajectory Forecasting. In recent years, there has been growing interest in predicting the future trajectories of an agent (e.g. pedestrian, vehicle, bicycle, etc.) as accurately as possible, under different settings. The most common is short-term trajectory forecasting, which aims to predict the next few seconds of the agent's path. Many methods have been proposed for this task, including recurrent neural networks (RNNs) [1], [19], convolutional neural networks [10], and graph neural networks (GNNs) [20], [21]. In addition to modeling the dynamics of individual agent's motion, contextual information such as social interactions has been proven to significantly influence the trajectory prediction performance [1], [3], [15], as well as other contextual features such as physical constraints and scene elements [2], [22], [23].

Although short-term trajectory forecasting models have achieved remarkable success, long-term forecasting remains challenging. The difficulty lies in the complexity of human behavior, which is affected by various factors such as the intent of other agents, randomness in the agents' decisions, the long-term goals of the agents, etc. There are very few approaches, made by the researchers to tackle this, such as estimating multi-modal long-term goals of the agents first, and then randomness is further reduced by sampling intermediate waypoints, conditioned on the final goal [10].

Goal-based Trajectory Forecasting. Some recent works [24], [25] employ destination or final goal estimation for improved trajectory prediction. Mangalam *et al.* [26] use a VAE for estimating final goals and employs a prediction network that takes as input the past trajectory and the VAE-predicted goals to predict the future trajectories. In a consequent paper [10], they take up a convolutional-based approach where positions are treated as heatmaps and final positions are sampled from 2D probability distribution maps, similar to ours. In their recent work, Gilles *et al.* [27] consider directly HD-Maps or lane graphs generated from the HD-Maps, along with the past trajectories to estimate the final goals. However, HD-Maps annotations are expensive and hard to collect.

Knowledge Distillation for Trajectory Forecasting. In the context of human trajectory forecasting, knowledge distillation has been investigated to improve the model's robustness to incorrect detection and corruption of trajectory data in crowded scenes by distilling knowledge from a teacher with uncorrupted sequence to a student with corrupted observations [28]. In [29], a two-fold knowledge distillation scheme is proposed to transfer more accurate predictability of a teacher network, which has an additional input modality of HD Maps, to a Map-less student network. However, to the best of our knowledge, knowledge distillation has not been explored for long-term trajectory forecasting. In this paper, we propose a novel method for multi-modal long-term trajectory prediction using knowledge distillation.

III. Our Method

A. Trajectory Forecasting Problem Formulation

Given a recorded scene of trajectories $\mathcal{D} = \{\mathcal{I}, \mathcal{U}\}$, where $\mathcal{I} \in \mathbb{R}^{3\times H\times W}$ is the RGB image representing the appearance of the scene, and $\mathcal{U} = \{\mathbf{u}^i\}_{i=1}^{N}$ is the collection of all the N agents' trajectories, we are interested in the forecasting of trajectories t_f seconds ahead in the future, from past observations of t_p seconds. More formally, starting from a past observed trajectory $\mathbf{u}_p^i = \{u_k^i\}_{k=1}^{K_p}$ where $(x_k^i, y_k^i) = u_k^i \in \mathbb{R}^2$ is the 2D i-th agent location at the k-th step at time $t = k/f_s$ with f_s being the sampling rate of the recording and $K_p = t_p \cdot f_s$, we are interested in the estimation of its continuation into the future of t_p seconds defined by $\mathbf{u}_f^i = \{u_k^i\}_{k=K_p+1}^{K_p+K_f}$ where $K_f = t_f \cdot f_s$. Following the long-term setting in [10], we set $t_p = 5$ sec and $t_f = 30$ sec whereas following the standard practice for the short-term setting [3], [17], [30] we set $t_p = 3.2$ sec and $t_f = 4.8$ sec.

B. The Di-Long Model Architecture

Pedestrian trajectory forecasting is a challenging task requiring modeling multiple factors influencing human motion. We stress the fact that multi-modality modeling (here instantiated in terms of goal prediction, social influence, and the spatial relation of the trajectory points w.r.t the scene semantic map) are key components of this problem. To this end, we propose a simple yet powerful architecture that consists of two main components: (i) a *goal estimation module* that predicts the likely final locations of each agent given its previously observed positions, and (ii) a multi-modal, recurrent *temporal backbone* which processes trajectory locations using a cross-attentive mechanism among inputs of different modalities. Finally, we have an auxiliary teacher network, which is composed of a similar goal module and a temporal backbone and distills the knowledge about the determinacy of the predicted trajectory to a student network.

Goal Estimation Module. The goal module $\mathcal{G}$ is a U-Net [31] that predicts a probability distribution of plausible final positions (i.e. goals) and intermediate waypoints distributions for each input trajectory. Similar to [12], observed trajectories $\mathbf{u}_p \in \mathbb{R}^{K_p\times 2}$ are spatially encoded into 2D Gaussian heatmaps by mapping each u_k coordinates into a 2D Gaussian heatmap of shape $H \times W$ centered at (x_k, y_k) with a pre-defined standard deviation σ, resulting in an tensor $\mathbf{U}_p \in \mathbb{R}^{K_p\times H\times W}$. The same applies to the target trajectory $\mathbf{u}_f \in \mathbb{R}^{K_f\times 2}$ that is encoded into Gaussian heatmaps $\mathbf{U}_f \in \mathbb{R}^{K_f\times H\times W}$. Other than the 2D encoded trajectory, similar to [10] we employ a pre-trained U-Net model $\mathcal{Q}$ to extract the 2D semantic maps of the scene to

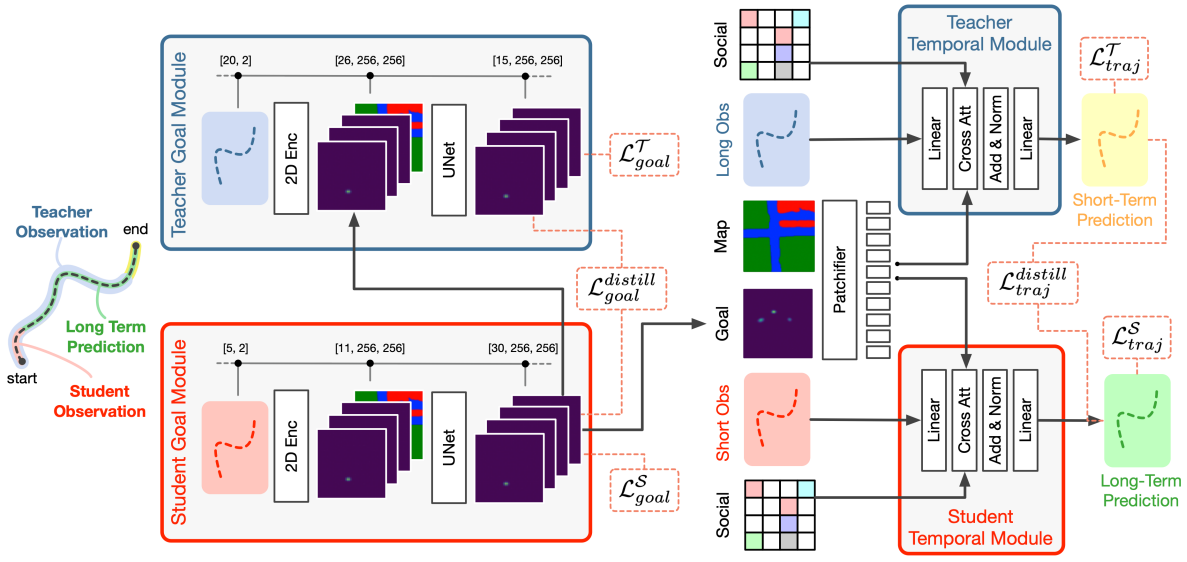

Fig. 2: **Detailed overview of the Di-Long Model Components.** Di-Long is composed by a student and the teacher both having a goal module and a temporal module. The student processes short observations and predicts long predictions, the teacher observes long trajectories and predicts short ones. The goal modules processes 2D encoded sequences and semantic maps, producing goal and waypoints heatmaps. The temporal modules, given the observed trajectory, the goals, the semantic maps, and the social information, predict the future trajectory. The distillation is done both in the goal and in the temporal modules. See Sec. III for more details.

exploit the physical contextual information useful for the goal module. Specifically, given the scene image $\mathcal{I}$, the pre-trained model provides the segmented one-hot encoded maps $\mathbf{M} \in \mathbb{R}^{C\times H\times W}$ where the set of semantic categories are related to different physical categories of the scene (such as "road", "tree", "terrain", etc.). We subsequently concatenate the past encoded trajectory $\mathbf{U}_p$ and the estimated semantic maps $\mathbf{M}$ along the channel dimension, and we feed the obtained tensor $\mathbf{x} \in \mathbb{R}^{(K_p+C)\times H\times W}$ to the goal module to produce the goal heatmaps $\mathbf{z} = \mathcal{G}(\mathbf{x}) \in \mathbb{R}^{K_f\times H\times W}$. In the following, we summarize the computation of the goal module:

$$\begin{aligned} \mathbf{M} &= \mathcal{Q}(\mathcal{I}), \qquad \mathbf{U}_p = Gauss2D(\mathbf{u}_p) \\ \mathbf{x} &= Cat(\mathbf{U}_p, \mathbf{M}), \qquad \mathbf{z} = \mathcal{G}(\mathbf{x}) \end{aligned} \tag{1}$$

The channel $\mathbf{z}[c] \in \mathbb{R}^{H\times W}$ is a sigmoid-activated probability heatmap related to the specific trajectory forecasting at the time step $c \in [0, K_f - 1]$, and for $c = K_f - 1$ it predicts the final destination (a.k.a. goal) heatmap, whereas for other values of $c = \bar{c}$ it computes an intermediate waypoint distribution in between the last observed point and the goal. Following [10], [12], to reduce the future uncertainty, we sample multiple goals and waypoints from the corresponding heatmap channels of $\mathbf{z}$, and we recompute the Gaussian encoded maps that are used by the temporal backbone module. In our investigation, similar to [12] we make use of the goal $\mathbf{z}[K_f - 1]$ and a single waypoint $\mathbf{z}[\bar{c}]$ at $\bar{c} = K_f//2$, and we subsequently channel-wise concatenate them into $\hat{\mathbf{z}} \in \mathbb{R}^{2\times H\times W}$.

Temporal Module. Similar to [12], our temporal module T is an autoregressive transformer that processes the past observed coordinates $\mathbf{u}_p \in \mathbb{R}^{K_p\times 2}$ and predicts the future trajectory $\mathbf{v} \in \mathbb{R}^{K_f\times 2}$ related to the ground truth $\mathbf{u}_f \in \mathbb{R}^{K_f\times 2}$. However, in contrast to [12], we employ a transformer decoder-only (GPT [14] style) that uses cross-attention modules to be conditioned by several input modalities such as goal and waypoints estimated maps, scene semantic maps, and social information from other agents in the scene. More specifically, the predicted semantic map of the scene $\mathcal{Q}(\mathcal{I}) = \mathbf{M} \in \mathbb{R}^{C\times H\times W}$ is channel-wise concatenated to the estimated sampled maps $\hat{\mathbf{z}}$ (containing the goal $\mathbf{z}[K_f]$ and the waypoints $\mathbf{z}[\bar{c}]$) and together are patchified, linearly projected, and fed to the autoregressive transformer decoder. Inspired by [15], in our temporal module, we employ an agent-aware cross-attention between the past observed trajectory and the concatenated multi-modal information (semantic map, goals, and waypoints). This attention mechanism preserves the notion of agent identity and learns the intra and inter-dependencies among all the agents [15]. In the following, we summarize the temporal module forward computation:

$$\begin{aligned} \hat{\mathbf{u}}_{\mathcal{G}} &= Proj(Patch(Cat(\hat{\mathbf{z}}, \mathbf{M}))) \\ \mathbf{v} &= T(\mathbf{u}_p, \hat{\mathbf{u}}_{\mathcal{G}}) \end{aligned} \tag{2}$$

where $Proj(\cdot)$ is a linear projection layer and $Patch(\cdot)$ is the standard patchification operation of the vision transformer [33] that tokenizes a 2D tensor into tokens.

Method	Social	Semantic Map	Goal	Distill.	ADE	FDE	KDE-NLL	Miss Rate
Social-GAN [3]	✓	✗	✗	✗	38.57	84.61	-	-
PECNet [26]	✓	✗	✓	✗	20.25	32.95	-	-
R-PECNet [26], [10]	✓	✗	✓	✗	341.80	1702.64	-	-
Social-STGCNN† [32]	✓	✗	✗	✗	22.48±0.14	33.61±0.29	13.67±0.01	0.58±0.01
Agentformer† [15]	✓	✗	✗	✗	36.26±0.15	46.36±0.34	19.53±0.01	0.63±0.01
Trajectron++† [21]	✓	✓	✗	✗	24.16±0.13	36.86±0.33	15.45±0.02	0.65±0.01
Goal-SAR† [12]	✗	✗	✓	✗	18.53±0.10	26.59±0.37	9.80±0.01	0.34±0.01
Y-Net* [10]	✗	✓	✓	✗	15.19±0.13	23.17±0.31	10.17±0.01	0.32±0.02
Di-Long (Ours)	✓	✓	✓	✗	17.79±0.13	23.33±0.42	9.73±0.02	0.36±0.01
Di-Long (Ours)	✓	✓	✓	✓	**14.89±0.15**	**19.85±0.28**	**9.52±0.01**	**0.24±0.01**

TABLE I: **Intersection Drone Dataset (inD) Long-term Results.** Results are reported in terms of the best ADE and FDE among 20 predicted generated samples, in meter (lower is better). The KDE-NLL is calculated by averaging over 20 predictions. Miss Rate is calculated over 20 predictions with a center distance threshold of 20 meter. Bold and underlined numbers indicate the best and second-best from previous works. **Social**, **Semantic Map**, **Goal** indicate additional input information (other than temporal) used by the models. **Distill.** is not a modality but an enhancement to the model. † *means that we re-trained the model as the authors do not provide the results in the long-term setting;* $*$ *means that we re-trained the model in order to align the data pre-processing w.r.t. all the other previous works.*

Method	Social	Semantic Map	Goal	Distill.	ADE	FDE	KDE-NLL	Miss Rate
Social-GAN [3]	✓	✗	✗	✗	155.32	307.88	-	-
PECNet [26]	✓	✗	✓	✗	72.22	118.13	-	-
R-PECNet [26], [10]	✓	✗	✓	✗	261.27	750.42	-	-
Social-STGCNN† [32]	✓	✗	✗	✗	69.89±0.09	112.73±0.14	13.24±0.01	0.37±0.01
Agentformer† [15]	✓	✗	✗	✗	66.43±0.10	102.60±0.16	12.13±0.01	0.35±0.01
Trajectron++† [21]	✓	✓	✗	✗	71.51±0.13	130.41±0.13	14.68±0.01	0.55±0.01
Goal-SAR† [12]	✗	✗	✓	✗	53.08±0.04	79.26±0.10	11.53±0.01	0.31±0.01
Y-Net* [10]	✗	✓	✓	✗	**46.00±0.23**	77.45±0.85	11.23±0.01	**0.26±0.01**
Di-Long (Ours)	✓	✓	✓	✗	48.92±0.09	76.31±0.18	10.93±0.01	0.34±0.01
Di-Long (Ours)	✓	✓	✓	✓	48.21±0.09	**72.41±0.21**	**10.85±0.01**	0.28±0.01

TABLE II: **Stanford Drone Dataset (SDD) Long-term Results.** Results are reported as the best ADE and FDE among 20 predicted samples in pixels (lower is better). The KDE-NLL is calculated by averaging over 20 predictions. Miss Rate is calculated over 20 predictions with a center distance threshold of 80 pixels. † *and* $*$ *should be interpreted as previously reported in Table I.*

C. Short-to-Long Term Knowledge Distillation

In our framework, we create two instances of the same model for the teacher $\mathcal{T}$ and the student $\mathcal{S}$ where both have the previously described goal and temporal modules. However, we let the two networks process different observations with different duration (longer for the teacher and shorter for the student), and we distill the information of the teacher to guide the student's predictions both for the goal and temporal modules. More specifically, the teacher $\mathcal{T}$ observes trajectories of length $t_p + t_a$ seconds with $0 \leq t_a < t_f$ while the student $\mathcal{S}$ processes trajectories of duration t_p. We define t_a as the anchor time, and as t_a increases, the teacher solves a less uncertain task since its observation grows and the length of its prediction $t_f - t_a$ becomes smaller and more deterministic. In the following, we consider $t_a = K_f//2$ as investigated in Sec. IV-C. $\mathcal{T}$ is only used at training time, and during inference, only $\mathcal{S}$ is considered.

Goal Module Distillation. Trajectory forecasting is heavily dependent on the predicted goal [10], [12] as a better goal estimation leads to a better-predicted trajectory closer to the ground truth. For this reason, we focused on the improvement of our goal module $\mathcal{G}$ by following a knowledge distillation strategy. Following Eq. 1, the student goal module processes $\mathbf{x}^{\mathcal{S}} \in \mathbb{R}^{(K_p+C)\times H\times W}$ producing $\mathbf{z}^{\mathcal{S}} \in \mathbb{R}^{K_f\times H\times W}$, whereas the teacher processes $\mathbf{x}^{\mathcal{T}} \in \mathbb{R}^{(K_p+K_a+C)\times H\times W}$ producing $\mathbf{z}^{\mathcal{T}} \in \mathbb{R}^{(K_f-K_a)\times H\times W}$, with $K_a = t_a \cdot f_s$. The teacher and student goal modules are then trained by optimizing their corresponding losses as defined in the following:

$$\begin{aligned}\mathcal{L}^{\mathcal{S}}_{goal} &= \frac{1}{K_f}\sum_{k=0}^{K_f-1} BCE\left(\mathbf{z}^{\mathcal{S}}[k], \mathbf{U}_f[k]\right)\\ \mathcal{L}^{\mathcal{T}}_{goal} &= \frac{1}{K_f-K_a}\sum_{k=0}^{K_f-K_a-1} BCE\left(\mathbf{z}^{\mathcal{T}}[k], \mathbf{U}_f[K_a+k]\right)\end{aligned} \tag{3}$$

where $BCE(p,t)$ is the binary cross-entropy loss between a generic prediction p and target t.

In our investigation, the teacher and the student are trained from scratch together in an online fashion. Interestingly, during training, we found it beneficial to distill information from the teacher in an *implicit* manner without using the usual explicit distillation loss [18], [28]. Specifically, we let the teacher see the standard observation $\mathbf{x}^{\mathcal{T}}$ as well as an augmented version of it $\mathbf{x}^{\mathcal{TS}}$ by replacing the last portion of the observation (from K_p up to $K_p + K_a$) with the time-related student's prediction counterpart. More specifically:

$$\begin{aligned}\mathbf{x}^{\mathcal{TS}}[0:K_p] &= \mathbf{x}^{\mathcal{T}}[0:K_p],\\ \mathbf{x}^{\mathcal{TS}}[K_p:K_p+K_a] &= \mathbf{z}^{\mathcal{S}}[0:K_a].\end{aligned} \tag{4}$$

In this way, the teacher provides a stronger regularization for the student with the guidance of a teacher who is *informed* of the current student's prediction state, and the student is updated with the gradients that pass through the teacher's predictions. Given the two views $\mathbf{x}^{\mathcal{T}}$ and $\mathbf{x}^{\mathcal{TS}}$ the teacher goal module produces respectively $\mathbf{z}^{\mathcal{T}}$ and $\mathbf{z}^{\mathcal{TS}}$, and we

Method	Distill.	ADE	FDE	KDE-NLL	MR
Social-GAN [3]	✗	27.23	41.44	-	-
CF-VAE [34]	✗	12.60	22.30	-	-
P2T [35]	✗	12.58	22.07	-	-
SimAug [36]	✗	10.27	19.71	-	-
PECNet [26]	✗	9.96	15.88	-	-
Social-STGCNN† [32]	✗	18.88±0.01	33.14±0.01	9.32±0.01	0.56±0.01
Agentformer† [15]	✗	10.25±0.01	16.51±0.03	9.56±0.01	0.37±0.01
Trajectron++† [21]	✗	10.29±0.01	15.98±0.02	8.10±0.01	0.32±0.01
Goal-SAR* [12]	✗	7.98±0.01	12.21±0.03	7.93±0.01	0.23±0.01
Y-Net* [10]	✗	8.25±0.01	12.10±0.02	8.22±0.01	0.25±0.01
Di-Long (Ours)	✗	**7.43±0.01**	12.13±0.01	8.05±0.01	**0.20±0.01**
Di-Long (Ours)	✓	**7.43±0.01**	**12.07±0.01**	**7.85±0.01**	**0.20±0.01**

TABLE III: **Short-term results on the SDD.** Miss Rate (MR) is calculated over 20 predictions with a center distance threshold of 14 pixels. $*$ means that we re-trained the models in order to align the data pre-processing w.r.t. all the other previous works.

Method	Distill.	ADE	FDE	KDE-NLL	MR
Social-GAN [3]	✗	0.48	0.99	-	-
ST-GAT [17]	✗	0.48	1.00	-	-
AC-VRNN [30]	✗	0.42	0.80	-	-
Social-STGCNN† [32]	✗	0.59±0.01	0.96±0.01	7.97±0.01	0.61±0.01
Agentformer† [15]	✗	0.57±0.01	0.87±0.01	6.86±0.01	0.53±0.01
Trajectron++† [21]	✗	0.62±0.01	0.98±0.01	8.13±0.01	0.64±0.01
Goar-SAR* [12]	✗	0.44±0.01	0.70±0.01	5.47±0.01	0.49±0.01
Y-Net* [10]	✗	0.55±0.01	0.93±0.01	7.20±0.01	0.60±0.01
Di-Long (Ours)	✗	0.39±0.01	0.61±0.01	5.95±0.01	0.27±0.01
Di-Long (Ours)	✓	**0.37±0.01**	**0.59±0.01**	**5.32±0.01**	**0.25±0.01**

TABLE IV: **Short-term results on the inD.** Miss Rate (MR) is calculated over 20 predictions with a center distance threshold of 5 meter. $*$ means that we re-trained the models in order to align the data pre-processing w.r.t. all the other previous works.

define the distillation loss as:

$$\mathcal{L}_{goal}^{distill} = \frac{1}{K_f - K_a} \sum_{k=0}^{K_f - K_a - 1} BCE\left(\mathbf{z}^{\mathcal{TS}}[k], \mathbf{U}_f[k]\right) \quad (5)$$

Finally, the total loss for training the goal modules in our knowledge distillation setting results as in the following:

$$\mathcal{L}_{goal} = \mathcal{L}_{goal}^{\mathcal{S}} + \mathcal{L}_{goal}^{\mathcal{T}} + \mathcal{L}_{goal}^{distill}. \quad (6)$$

Temporal Module Distillation. Similarly to the goal modules, we employ a knowledge distillation strategy from the temporal module of the teacher $\mathcal{T}$ to the temporal module of the student $\mathcal{S}$ in order to provide the student useful guidance from the teacher that processes a longer observation and solves a less uncertain and easy task. Following Eq. 2, the student temporal module takes as input $\mathbf{u}_p^{\mathcal{S}} \in \mathbb{R}^{K_p \times 2}$ as well as $\hat{\mathbf{z}}$ and $\mathbf{M}$ and produces $\mathbf{v}^{\mathcal{S}} \in \mathbb{R}^{K_p \times 2}$. Similarly, the teacher processed $\mathbf{u}_p^{\mathcal{T}} \in \mathbb{R}^{(K_p+K_a)\times 2}$, $\hat{\mathbf{z}}$ and $\mathbf{M}$, and produces $\mathbf{v}^{\mathcal{T}} \in \mathbb{R}^{(K_f-K_a)\times 2}$. In our investigation, for the temporal module, we use a more standard knowledge distillation mechanism by employing an explicit loss term on the teacher and student predictions. In the following, we summarize all the losses we designed for training the teacher and student temporal modules:

$$\begin{aligned}
\mathcal{L}_{traj}^{\mathcal{S}} &= \frac{1}{K_f} \sum_{k=0}^{K_f-1} \left\| \mathbf{v}^{\mathcal{S}}[k] - \mathbf{u}_f[k] \right\|_2^2 \\
\mathcal{L}_{traj}^{\mathcal{T}} &= \frac{1}{K_f - K_a} \sum_{k=0}^{K_f-K_a-1} \left\| \mathbf{v}^{\mathcal{T}}[k] - \mathbf{u}_f[K_a + k] \right\|_2^2 \\
\mathcal{L}_{traj}^{distill} &= \frac{1}{K_f - K_a} \sum_{k=0}^{K_f-K_a-1} \left\| \mathbf{v}^{\mathcal{S}}[K_a + k] - \mathbf{v}^{T}[k] \right\|_2^2 .
\end{aligned} \quad (7)$$

Subsequently, we define the total loss related to the temporal modules as:

$$\mathcal{L}_{traj} = \mathcal{L}_{traj}^{\mathcal{S}} + \mathcal{L}_{traj}^{\mathcal{T}} + \mathcal{L}_{traj}^{distill}. \quad (8)$$

Finally, we combine the goal and the temporal losses, as well as their distillation loss terms together, and the total loss results:

$$\mathcal{L} = \mathcal{L}_{goal} + \lambda \mathcal{L}_{traj} \quad (9)$$

where $\lambda \in \mathbb{R}$ is a mixing hyper-parameter that balances the importance of the goal estimation and the temporal module prediction.

IV. Experiments

A. Experimental Setting

Long and Short-Term Setting. We follow the long-term forecasting setting of previous works [10], [12] where the observation is $t_p = 5$ sec and the future target trajectory has duration is $t_f = 30$ sec, whereas for the short-time setting we follow the well-established protocol in which $t_p = 3.2$ sec and $t_f = 4.8$ sec (e.g. [1], [3]).

Datasets. We used two popular datasets for trajectory prediction: Intersection Drone Dataset (inD) [37] and Stanford Drone Dataset (SDD) [38].

The inD dataset contains 10 hours of recordings of 4 different intersections in an urban environment, where pedestrians interact with cars to reach their destinations. The dataset has sample rate 25 fps and we follow the same pre-processing of [20] to prepare the dataset for the long and short-term settings, such as resampling the sequences at f_s fps ($f_s = 1$ fps in the long-term scenario, and $f_s = 2.5$ for the short-term ones), consider only pedestrian tracks, remove short sequences, and use sliding window without overlap to split long trajectories. We follow the standard dataset splits and evaluation setting of [10].

SDD contains $11,000$ unique pedestrian tracks across 20 top-down scenes from the Stanford campus in a bird's eye view captured with a drone. We follow the dataset splits and pre-processing from [26], where the initial tracks at 25 fps are resampled at $f_s = 1$ fps for the long-term setting, and at $f_s = 2.5$ fps for the short-term setting. For both datasets we obtained sequences of length $K_p = 5$, $K_f = 30$ for the long-term and $K_p = 8$, $K_f = 12$ for the short-term scenarios.

Metrics. For the quantitative evaluation, we consider two standard error metrics, namely the Average Displacement Error (ADE) and the Final Displacement Error (FDE). The

ADE is calculated by measuring the average ℓ_2 distance between the predicted and ground truth trajectories, while for the FDE, only the final positions are considered. Following previous works [10], [12] that provided the analysis in a stochastic setting, we reported the best-of-K ADE and FDE over $K = 20$ generated predictions for each input trajectory.

Additionally, to measure the correctness of the output predicted distribution, we consider the Kernel Density Estimate Negative Log Likelihood (KDE-NLL) metric introduced by [39], where the output pdf is first estimated at each prediction step, and subsequently used to compute the average log-likelihood. A trajectory forecast is considered as a miss if the FDE between the ground truth and the predicted trajectory is above a center distance threshold. We calculate Miss Rate [40] over 20 predictions and empirically choose the center distance threshold values for different settings for different datasets. For inD long-term, the threshold is 20 meters; for inD short-term, it is 5 meters. On the other hand, for SDD long-term, the threshold is 80 pixels; for SDD short-term, it is 14 pixels. As for the short-term setting, the predicted trajectories deviate less from the ground truth compared to the long-term setting, we choose a lower threshold value for calculating Miss Rate in the short-term than long-term setting. Compared to inD, SDD contains more complex scenarios and hence higher FDE values are observed in SDD. That is why we define tighter thresholds for trajectory forecasting on inD, compared to SDD.

B. Experimental Results

Intersection Drone Dataset. In Table I, we reported our results on the Intersection Drone Dataset in the long-term setting. In this scenario, our Di-Long model outperforms Goal-SAR and Y-Net by a considerable margin in terms of ADE (-0.3 w.r.t. Y-Net), FDE (-3.32 w.r.t. Y-Net), KDE-NLL (-0.28 w.r.t. Goal-SAR) and Miss Rate (-0.3 w.r.t. Y-Net). Most importantly, we show that our distillation framework is beneficial for our Di-Long model as, without enabling it, our model has higher ADE, and comparable FDE, KDE-NLL, and Miss Rate w.r.t. the second best previous models (underlined results in the table). In Table IV we reported the results for the same dataset for the short-term setting. Also in this case, our model outperforms previous works in all the metrics, however, the distillation improvement is smaller, probably due to the short-term setting where the future uncertainty that can be reduced by distillation by a teacher is less prominent.

Stanford Drone Dataset. In Table II, we reported our results on the Stanford Drone Dataset in long-term settings. Similar to the inD dataset, our model shows impressive performances in terms of FDE (-5.04), which suggests that our goal module combined with our distillation strategy (where the distillation gain is -3.9 FDE) is crucial to improve the final prediction. We also improve the KDE-NLL measure (-0.38) w.r.t. the second best model (results underlined in the table), however, we slightly degrade the ADE and Miss Rate performances w.r.t. Y-Net. In Table III, we show our results on SDD in the short-term settings. Similarly to the inD short-term results, the distillation gain is smaller compared to the long-term setting, however, we confirm to improve the results in terms of ADE, FDE and KDE-NLL w.r.t. previous works.

C. Ablation Studies

Ablation of the Di-Long Components. We investigate the important components of our Di-Long model by starting with a plain architecture with fewer capabilities (without processing semantic maps, with no social information, not using the goal and waypoints, and without distillation) and adding one component at a time. This investigation is reported in Fig 3. The ablation is conducted on the inD dataset on the long-term prediction setting, and we report the ADE (left) and FDE metrics (right). We compare our increasingly improving Di-Long model with Goal-SAR and Y-Net. As shown in the table, the use of the semantic maps, the goal and waypoint, and the social information are important, but clearly, our distillation strategy (in different forms) makes a big difference and improves the ADE and FDE, surpassing previous works.

Increasing the Long-Term Time Horizon. In Fig. 4, we studied the prediction capability of the Di-Long model on longer time horizons than $t_f = 30$ sec, and compared to Goal-SAR and Y-Net. Specifically, we increase the prediction length starting from $t_f = 10$ sec up to $t_f = 60$ sec. As shown in the figure, our Di-Long model consistently outperforms Y-Net and Goal-SAR, and interestingly, at shorter time horizons, the performances of the models are comparable, however, when the prediction length increases, the Di-Long model maintains a low degradation, outperforming previous works in terms of ADE and FDE, without manifesting sudden degradation after a critical time horizon.

Optimal Teacher Observation Length. In Fig. 5, we investigate on the long-term setting on the inD dataset, the impact of the length of the teacher observation $t_p + t_a$ where t_p is the student observation duration, and t_a is the anchor time that is additionally seen by the teacher during training. As shown in the figure, spanning $t_p + t_a$ from 5 to 35 the results of the student network behave differently: at the extremes, when the teacher observation is too short or too long, the final results in terms of ADE and FDE are not optimal, and at $t_p + t_a = 20$ sec, there is a minimum in both metrics, suggesting that setting the anchor time to $t_a = 15$ leads to best final results. This value of the anchor interestingly coincides with the time step where we extracted the waypoint from the goal module.

D. Technical Details

The scene $\mathcal{I}$ can have different dimensions in the considered datasets, and for this reason, following [10], [12], we resize them to $H = W = 256$, preserving the scale-ratio. The considered classes for the semantic segmentation maps $\mathbf{M}$ are $C = \{$`pavement, terrain,`

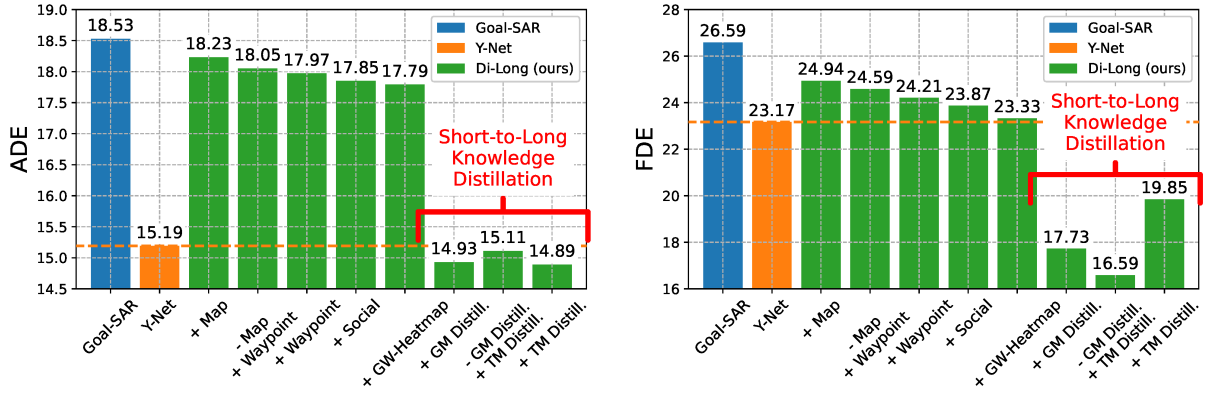

Fig. 3: **Ablation Study on the Di-Long Components.** In **GW-Heatmap**, a 2D Gaussian heatmap of the goal/waypoint (GW) is appended to the semantic map, patchified, projected, and passed as the control input of the transformer decoder. **GM Distill** corresponds to goal module distillation, while **TM Distill** corresponds to temporal module distillation, respectively.

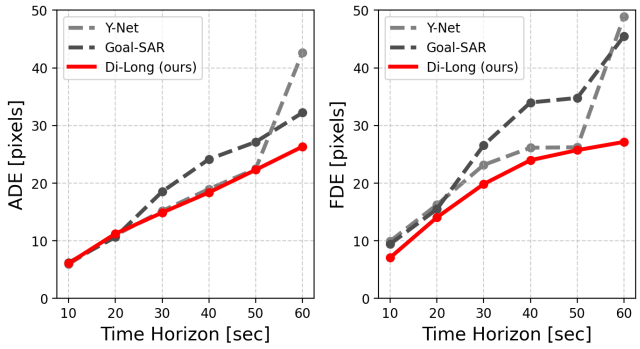

Fig. 4: **Performance Across Longer Time Horizons.** These results are obtained on the inD dataset.

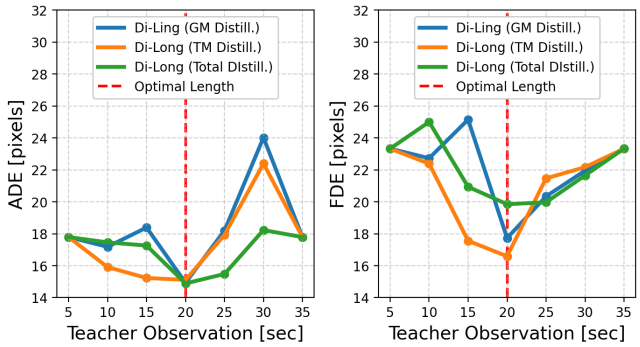

Fig. 5: **Optimal Teacher Observation Length.** These results are obtained on the inD dataset.

structure, tree, road, not defined}. The U-Nets in the student and the teacher goal estimator have 5 down-sampling and up-sampling residual blocks, with respectively $[32, 32, 64, 64, 64]$ and $[64, 64, 64, 32, 32]$ channels for the encoder and decoder, as in [10]. Estimated goals are sampled from $\mathbf{z}$ and re-encoded into $\hat{\mathbf{z}}$, and for sampling, we use the Test-Time-Sampling-Trick `TTST` proposed in [10]. We implemented our codebase using PyTorch, and we trained the Di-Long model on a single GeForce RTX 2080 Ti GPU for 500 epochs (that corresponds to approximately 21 hours of training) using the Adam [41] optimizer with a batch size of 8 and a constant learning rate set to $1e{-}4$. The segmentation network $\mathcal{Q}$ is frozen and initialized with the pre-trained weights from ImageNet-1k, similar to [10].

E. Qualitative Results

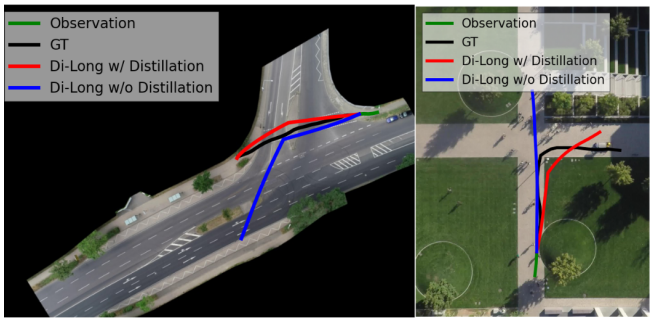

Fig. 6: **Qualitative Results.** Observed trajectories are depicted in green; ground truth (GT) predictions in black; Di-Long without knowledge distillation in blue; Di-Long "full model" predictions in red. The left figure shows a long-term prediction example from inD (scene 00), whereas the right one is for long-term prediction in SDD (Hyang0 scene).

In Fig. 6 (*left*), we report an example where the Di-Long w/o Knowledge Distillation tends to strictly follow the road semantics. Still, the final goal is far from the ground truth, whereas due to goal module distillation, the full model predicts the final goal very close to the ground truth. In Fig. 6 (*right*), we show another example in which the Di-Long full model clearly matches the ground truth behavior.

V. Conclusions

In this work, we introduced Di-Long, which is a simple, scene and socially compliant auto-regressive transformer-based architecture empowered with a novel short-to-long term knowledge distillation scheme for trajectory prediction. Di-Long achieves state-of-the-art performances in long-term as well as in short-term prediction on both the Intersection Drone (inD) and Stanford Drone (SDD) datasets, thus proving the effectiveness of the proposed solution.

REFERENCES

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[2] L. Ballan, F. Castaldo, A. Alahi, F. A. N. Palmieri, and S. Savarese, "Knowledge transfer for scene-specific motion prediction," in *European Conference on Computer Vision (ECCV)*, 2016.

[3] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GAN: Socially Acceptable Trajectories with Generative Adversarial Networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[4] F. Marchetti, F. Becattini, L. Seidenari, and A. Del Bimbo, "MANTRA: Memory Augmented Networks for Multiple Trajectory Prediction," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[5] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: a survey," *The International Journal of Robotics Research (IJRR)*, vol. 39, pp. 895–935, 2019.

[6] M. Monforte, A. Arriandiaga, A. Glover, and C. Bartolozzi, "Where and when: event-based spatiotemporal trajectory prediction from the icub's point-of-view." in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

[7] A. Cui, S. Casas, K. Wong, S. Suo, and R. Urtasun, "Gorela: Go relative for viewpoint-invariant motion forecasting," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[8] G. Camporese, P. Coscia, A. Furnari, G. M. Farinella, and L. Ballan, "Knowledge distillation for action anticipation via label smoothing," in *IAPR International Conference on Pattern Recognition (ICPR)*, 2020.

[9] N. Osman, G. Camporese, and L. Ballan, "TAMformer: Multi-Modal Transformer with Learned Attention Mask for Early Intent Prediction," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

[10] K. Mangalam, Y. An, H. Girase, and J. Malik, "From goals, waypoints & paths to long term human trajectory forecasting," in *IEEE/CVF Int'l Conference on Computer Vision (ICCV)*, 2021.

[11] H. Girase, H. Gang, S. Malla, J. Li, A. Kanehara, K. Mangalam, and C. Choi, "LOKI: Long Term and Key Intentions for Trajectory Prediction," in *IEEE/CVF Int'l Conference on Computer Vision (ICCV)*, 2021.

[12] L. F. Chiara, P. Coscia, S. Das, S. Calderara, R. Cucchiara, and L. Ballan, "Goal-driven self-attentive recurrent networks for trajectory prediction," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2022.

[13] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NIPS)*, 2017.

[14] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[15] Y. Yuan, X. Weng, Y. Ou, and K. Kitani, "AgentFormer: Agent-Aware Transformers for Socio-Temporal Multi-Agent Forecasting," in *IEEE/CVF Int'l Conference on Computer Vision (ICCV)*, 2021.

[16] L. Franco, L. Placidi, F. Giuliari, I. Hasan, M. Cristani, and F. Galasso, "Under the hood of transformer networks for trajectory forecasting," *Pattern Recognition*, vol. 138, p. 109372, 2023.

[17] Y. Huang, H. Bi, Z. Li, T. Mao, and Z. Wang, "STGAT: Modeling Spatial-Temporal Interactions for Human Trajectory Prediction," *IEEE/CVF Int'l Conference on Computer Vision (ICCV)*, 2019.

[18] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Proc. of the NeurIPS Deep Learning and Representation Learning Workshop*, 2015.

[19] P. Zhang, W. Ouyang, P. Zhang, J. Xue, and N. Zheng, "SR-LSTM: State Refinement for LSTM Towards Pedestrian Trajectory Prediction," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[20] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, "Spatio-temporal graph transformer networks for pedestrian trajectory prediction," in *European Conference on Computer Vision (ECCV)*, 2020.

[21] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control," in *European Conference on Computer Vision (ECCV)*, 2020.

[22] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. S. Torr, and M. Chandraker, "DESIRE: Distant Future Prediction in Dynamic Scenes with Interacting Agents," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[23] A. Sadeghian, V. Kosaraju, A. R. Sadeghian, N. Hirose, and S. Savarese, "SoPhie: An Attentive GAN for Predicting Paths Compliant to Social and Physical Constraints," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[24] S. V. Albrecht, C. Brewitt, J. Wilhelm, B. Gyevnar, F. Eiras, M. Dobre, and S. Ramamoorthy, "Interpretable goal-based prediction and planning for autonomous driving," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[25] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, C. Li, and D. Anguelov, "TNT: Target-driven trajectory prediction," in *International Conference on Robot Learning (CoRL)*, 2020.

[26] K. Mangalam, H. Girase, S. Agarwal, K.-H. Lee, E. Adeli, J. Malik, and A. Gaidon, "It is not the journey but the destination: Endpoint conditioned trajectory prediction," *European Conference on Computer Vision (ECCV)*, 2020.

[27] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "GOHOME: Graph-Oriented Heatmap Output for future Motion Estimation," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[28] A. Monti, A. Porrello, S. Calderara, P. Coscia, L. Ballan, and R. Cucchiara, "How many observations are enough? knowledge distillation for trajectory forecasting," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[29] Y. Wang, P. Zhang, L. Bai, and J. Xue, "Enhancing mapless trajectory prediction through knowledge distillation," *arXiv preprint arXiv:2306.14177*, 2023.

[30] A. Bertugli, S. Calderara, P. Coscia, L. Ballan, and R. Cucchiara, "AC-VRNN: Attentive Conditional-VRNN for Multi-Future Trajectory Prediction," *Computer Vision and Image Understanding (CVIU)*, vol. 210, p. 103245, 2021.

[31] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

[32] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, "Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *International Conference on Learning Representations (ICLR)*, 2021.

[34] A. Bhattacharyya, M. Hanselmann, M. Fritz, B. Schiele, and C. N. Straehle, "Conditional flow variational autoencoders for structured sequence prediction," *Proc. of the NeurIPS Bayesian Deep Learning Workshop*, 2019.

[35] N. Deo and M. M. Trivedi, "Trajectory forecasts in unknown environments conditioned on grid-based plans," *ArXiv*, vol. abs/2001.00735, 2020.

[36] J. Liang, L. Jiang, and A. Hauptmann, "SimAug: Learning Robust Representations from 3D Simulation for Pedestrian Trajectory Prediction in Unseen Cameras," *European Conference on Computer Vision (ECCV)*, 2020.

[37] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein, "The inD Dataset: A Drone Dataset of Naturalistic Road User Trajectories at German Intersections," in *IEEE Intelligent Vehicles Symposium (IV)*, 2019.

[38] A. Robicquet, A. Sadeghian, A. Alahi, and S. Savarese, "Learning social etiquette: Human trajectory understanding in crowded scenes," in *European Conference on Computer Vision (ECCV)*, 2016.

[39] B. Ivanovic and M. Pavone, "The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs," *IEEE/CVF Int'l Conference on Computer Vision (ICCV)*, 2018.

[40] Y. Xu, L. Chambon, E. Zablocki, M. Chen, A. Alahi, M. Cord, and P. Perez, "Towards motion forecasting with real-world perception inputs: Are end-to-end approaches competitive?" in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations (ICLR)*, 2015.