

Lifelong Vehicle Trajectory Prediction Framework Based on Generative Replay

Peng Bao*, Zonghai Chen*, Jikai Wang*, Deyun Dai* and Hao Zhao*

*Department of Automation, University of Science and Technology of China, Hefei, PR China

Abstract—Accurate trajectory prediction of vehicles is essential for reliable autonomous driving. To maintain consistent performance as a vehicle driving around different cities, it is crucial to adapt to changing traffic circumstances and achieve lifelong trajectory prediction model. To realize it, catastrophic forgetting is a main problem to be addressed. In this paper, a divergence measurement method based on conditional Kullback-Leibler divergence is proposed first to evaluate spatiotemporal dependency difference among varied driving circumstances. Then based on generative replay, a novel lifelong vehicle trajectory prediction framework is developed. The framework consists of a conditional generation model and a vehicle trajectory prediction model. The conditional generation model is a generative adversarial network conditioned on position configuration of vehicles. After learning and merging trajectory distribution of vehicles across different cities, the generation model replays trajectories with prior samplings as inputs, which alleviates catastrophic forgetting. The vehicle trajectory prediction model is trained by the replayed trajectories and achieves consistent prediction performance on visited cities. A lifelong experiment setup is established on four open datasets including five tasks. Spatiotemporal dependency divergence is calculated for different tasks. Even though these divergence, the proposed framework exhibits lifelong learning ability and achieves consistent performance on all tasks.

Index Terms—Gaussian mixture model, Conditional KL divergence, Conditional GAN, Generative replay, Lifelong trajectory prediction.

I. INTRODUCTION

IN autonomous driving, an ability to predict surrounding vehicles' future trajectories accurately is a key to make appropriate decision. Therefore, many research works contribute on vehicle kinematics and interactive modelling [1]. Recent works also adopt data-driven approaches [2]–[8]. Benefitted from these ingenious works, precision of vehicle trajectory prediction on various open datasets has been promoted significantly.

However, in real application, an intelligent vehicle equipped with an autonomous driving system is supposed to visit varied road sections, cities or even countries. To guide the vehicle safely, the system is required to adapt to heterogeneous distribution of surrounding vehicles' motion and interaction pattern and predict their future trajectories accurately. For this purpose, the system needs to learn new knowledge about emerging traffic environments continuously without forgetting old ones. In addition, with limited storage resource, the system can not afford to store large amount of trajectory data. Continuous learning with limited storage resource to achieve

good performance in all processed tasks is also called lifelong learning. Unfortunately, most existing vehicle trajectory prediction models are trained and tested specifically for each dataset, which fails to accommodate to other different datasets if applied directly.

A crucial problem impeding lifelong learning is catastrophic forgetting. In different traffic circumstances, interaction pattern among vehicles varies, which can be interpreted as spatiotemporal dependency divergence. The divergence makes a prediction model trained in old circumstances performs poorly in a new different circumstance. If trained in the new circumstances, the model would fit spatiotemporal dependency in the new one while forget that in the old circumstances, which makes it perform poorly in the old circumstances in turn. In summary, to perform consistently in all circumstances, the prediction model is required to learn new knowledges while not forgetting old ones.

Therefore, to measure spatiotemporal dependency divergence that arises from varied traffic circumstances, Mixture Density Networks (MDNs) are introduced to estimate conditional probability density function (CPDF) and conditional Kullback-Leibler divergence (CKLD) is computed through Monte-Carlo (MC) sampling. Then to realize lifelong trajectory prediction, a new framework based on conditional generative replay is proposed. The framework consists of two models, a generative memory and a task solver. The generative memory is designed to address catastrophic forgetting, a key challenge in lifelong learning. Through adversarial learning and distributions merging across different traffic circumstances, the generative memory replays trajectories conditioned on spatial configuration of vehicles. The replayed trajectories is used as an input to train a trajectory prediction model. As the generative memory is capable of generating trajectories within the same distribution of all recorded data, the prediction model achieves consistent performance on all processed tasks.

In summary, our contributions include:

- An innovative spatiotemporal dependency divergence measurement method is developed for two trajectory datasets.
- A novel research topic is revealed. Lifelong vehicle trajectory prediction is first proposed to promote adaptation and performance on heterogeneous tasks for autonomous driving system.
- Based on generative replay, an initiate lifelong learning framework is introduced for vehicle trajectory prediction.
- Experimental results on four different datasets have demonstrated that the proposed framework can achieve

lifelong learning and realize satisfactory prediction performance on all processed tasks.

II. RELATED WORK

Vehicle trajectory prediction is a longstanding research topic and is becoming more important as the development of autonomous driving. Kinematics models [9], [10] are established to predict surrounding vehicles' future trajectories, which performs well in short horizon. For a longer prediction horizon over 3 seconds, vehicles' intentions are important and have to be clarified first. Prototype classification [11] and intention recognition [12] are two research directions. Through collecting and clustering of tremendous trajectories in advance, prototype classification methods match online history trajectories with that in database. Multimodel future trajectories can be generated once matched. Intention recognition methods classify drivers' potential maneuvers into limited classes. Combining heuristic information including road topology, traffic signal and vehicle turn signal, classification models such as Support Vector Machines (SVMs) [13] and Hidden Markov Models (HMMs) [14] are introduced to recognize surrounding drivers' intentions. However, drivers' behaviours are affected by various factors and highly personalized [15], which makes it hard to be recognized accurately in real application. Besides, a vehicle's trajectory is not determined by its driver's intention only but affected interactively.

To make an accurate prediction, it's necessary to consider vehicles' motion as a dynamic interactive system and model interactive impact among vehicles. Pairwise interactive modelling is of high complexity. Deo Nachiket [16] simplified interactive factor as a cost function that penalizes vehicle collision to filter out collide future trajectory pairs. Data-driven approaches are another practical methods and have been digged extensively. In these approaches, vehicles' sequential features are learned through Long-Short Term Memory Networks (LSTMs). To learn interactive factor, Convolutional Neural Networks (CNNs) or max-pooling module is introduced. Deo [17] divided certain surrounding area into grid cells and use CNNs to model spatial relationship among vehicles. Li [4] builded an adjacent matrix of surrounding vehicles where each elements represents pairwise proximity. A CNN is utilized to learn interactive factor. Messaoud [18] discretized traffic environment into 3D grid cells and a Relational Recurrent Neural Network (RRNN) is used to predict future trajectory. Gupta [19] utilized a pooling module to extract dense interactive features among vehicles. More recent works urge to merge data-driven and knowledge-driven approaches into a unified neural network [6], [7].

However, although prediction precision is improved continually, generalization and adaptation of proposed models still remain an open problem. Majority of research works validate proposed methods on one open dataset only. Some works validate on more datasets but train and test models for each dataset individually, which is not consistent with real application. Pushing forward to real application, a novel prediction model is urged to be proposed to cope with various traffic circumstances. Two questions arise naturally. How to measure

difference among heterogeneous circumstances? And how to fulfill consistent vehicle trajectory prediction performance on emerging circumstances and visited ones, which is also called lifelong learning?

Divergence measurement of heterogeneous traffic circumstances is an open problem. To measure trajectory similarity, various methods are proposed, such as euclidean distance (ED) [20], dynamic time warping (DTW) [21], longest common subsequence (LCSS) [22], merge distance (MD) [23], and spatiotemporal locality in-between polylines (STLIP) distance [24], etc. Su [25] made a survey on 15 widely used trajectory distance measures in the literature. It can be deduced that ED is suitable for vehicle trajectories distance measurement that have the same total length and sample frequency. However, these similarity measurement methods consider one trajectory with another trajectory each time, while we aim to measure differences of traffic circumstances where dynamic number of trajectories are presented. In different traffic circumstances, vehicles' interaction pattern changes and the way affecting future motion varies. In other words, spatiotemporal dependency between future and past motion differs, which is essentially a conditional probability density function (CPDF) alteration problem. Therefore, a more reasonable method is to estimate distance between two unknown CPDF with empirical samples only.

Estimation distance of two unknown CPDF with samples only is challenging. As a commonly used probability divergence measurement method, KL divergence can not work without analytic CPDF. The Donsker-Varadhan variational formula [26], [27] can be utilized to estimate CKLD empirically. However, it suffers from convergence problem for large divergence between two CPDFs, which is usually the case in real data. K nearest neighbor [28] is another approach to approximate CKLD. It requires distance calculation of condition data for two datasets, which is not committed for traffic circumstances with dynamic number of vehicles. As a conditional extension of traditional Maximum Mean Discrepancy (MMD) [29], conditional MMD (CMMD) [30] is proposed to measure distance between two CPDFs. Similar with MMD, CMMD measures embedding probabilities distance in reproducing kernel Hilbert space (RKHS). In MMD, a PDF is mapped into a point in RKHS, while in CMMD a CPDF is a family of points with different conditions. Therefore, CMMD is averaged for distances with different condition, which implies samples conditioned on a same condition. For two traffic circumstances in our work, condition data can not guaranteed to be the same. In fact, CMMD is usually used as training loss function [31], [32] of neural networks where predicted and real value can be obtained on the same condition. Another method to measure PDF distance empirically is optimal transport. Esteban [33] proposes a data-driven conditional optimal transport (COT) method. The COT represents empirical CPDF distance computation as a optimal transport problem constrained by CPDF alignment. CKLD is utilized to interpret the constraint and then converted into KL divergence between joint distributions through chain rule [34]. By using Donsker-Varadhan variational formula and Lagrange multiplier, the constrained COT is relaxed into a minimax opti-

mization problem that can be optimized empirically. However, the COT prones to local minimum and the minimax game is hard to converge. Moreover, for high dimension data as in our work, it is difficult to identify local minimum.

Lifelong learning aims to solve a series of tasks incrementally [35]. When addressing a new task, small amount or none data of old tasks are stored. After the final task is presented, all tasks should be solved by one task model with good performance. Key to lifelong learning is avoiding catastrophic forgetting of old tasks' knowledge when updating the task model to solve a new task. From an aspect of model training, approaches to mitigate catastrophic forgetting are classified into three categories, architectural, regularization and rehearsal strategies. Architectural strategies train different models or subnetworks for incoming new tasks. A selector is used to choose an appropriate model or subnetwork for a task. Typical research works includes Progressive Neural Network (PNN) [36], Incremental Learning through Deep Adaptation (DAN) [37], Copy Weight with Re-init (CWR) [38], etc. These methods preserve performances of old tasks while conflicting with storage limitation of lifelong learning. Regularization strategies extend loss functions with additional term to retain performances of old tasks. Learning without Forgetting (LwF) [39] and Elastic Weight Consolidation (EWC) [40] are two representative methods. LwF proposes to use outputs of old models as soft targets to substitute data of old tasks, which is reported to suffer a buildup drop in old tasks' performance as the task sequence grows longer [41]. EwC evaluates importance of parameters for old tasks and adds a penalty to changes when training on new tasks, which pays more attention to preserving the knowledge on old tasks but prevents the model from achieving competitive performance on new tasks [42]. Rehearsal strategies generally use an external memory to store part of old data [43], [44] or patterns [45]. As storage is limited and Generative Adversarial Networks (GANs) develop, generative replay [46] is proposed as a memory of previous data and its feasibility has been validate on several works [47]–[51]. Although quality of the generation model is a bottleneck, many works [47], [50], [52]–[55] have proved that an elaborately designed generation model practically outperforms mainstream lifelong methods such as EWC, LwF, MAS [56], PathNet [57], and iCaRL [44] *et al.*

In this paper, CPDF distance between two traffic circumstances are calculated first to reveal spatiotemporal dependency divergence. Then a generative replay based lifelong trajectory prediction framework is proposed to enhance generalization and adaptation over different traffic environment. As a key of alleviate catastrophic forgetting, a generation model is realized through a novel conditional GAN (CGAN), which is called Recurrent Regression GAN (R2GAN). Through merging different generation models trained on different tasks, the generation model finally learns all knowledge involved in processed tasks. Eventually, a trajectory prediction model trained on generated data performs well on all tasks. A task chain including five tasks that stem from four open datasets is used as lifelong setup. Experiments on the task chain demonstrate effectiveness of proposed framework.

The rest of this paper is organized as follows. In section 3, a mathematic formulation for lifelong trajectory prediction is addressed. Divergence between two traffic circumstances is measured first in section 4. Then generative replay based lifelong prediction framework is introduced in section 5 in detail. In section 6, evaluation experiments are performed to evaluate quantitatively. Finally, conclusion and future work are introduced in section 6.

III. PROBLEM FORMULATION

Formally, lifelong vehicle trajectory prediction is characterized by a set of tasks $D = \{d_1, d_2, \dots, d_n\}$ to be learned by a parameterized model. In this work, with unsupervised learning nature of trajectory prediction, task data $d_i \in D$ have training samples $X_{1:t}^i$ where $t = t_h + t_f$. Target vehicle and surrounding vehicles' trajectories lasting for t_h are regarded as history information to train the parameterized model to predict future t_f trajectory of the target vehicle. In a traffic circumstance involving n_d vehicles, spatiotemporal dependency is formulated as a CPDF $p(Y|X)$ where Y represents future t_f trajectory of the target vehicle and X represents history t_h trajectories of all vehicles. Samples are drawn *i.i.d* from an unknown distribution $X_{1:t}^i \in P_{d_i}$ associated with task d_i . Distribution P_{d_i} can be different from each other for different $i \in \{1, 2, \dots, n\}$. In lifelong learning, task data d_i are observed sequentially and when the next data d_{i+1} arrive, data d_i are abandoned completely or only kept partly in a limited storage. Ultimately, the prediction model can predict accurately in all n tasks after observing all task data.

IV. DIVERGENCE MEASUREMENT OF DIFFERENT TRAFFIC CIRCUMSTANCES

As an effective divergence measurement method, KL divergence is extended to CKLD to measure spatiotemporal dependency difference of two traffic circumstances, which is formulated as

$$\begin{aligned} CKLD(p_1(Y|X) || p_2(Y|X)) \\ = \int p_1(X) \int \log \left(\frac{p_1(Y|X)}{p_2(Y|X)} \right) p_1(Y|X) dY dX. \end{aligned} \quad (1)$$

The CKLD can not be computed without analytic formulation of $p(Y|X)$. Therefore, parameters of GMMs are estimated by a MDN to approximate $p(Y|X)$ first and then MC sampling can be performed to calculate CKLD.

A. Dimension normalization for dynamic traffic circumstances

In a dynamic traffic circumstances involving n_d vehicles, condition X should be represented as $X = (x_{1:t_h}^1, x_{1:t_h}^2, \dots, x_{1:t_h}^{n_d})$ where $x_{1:t_h}^i$ represents sequential coordinate of vehicle i that lasts t_h , which possesses dynamic dimension. To facilitate model learning, a fixed dimension is preferred. Notice that target vehicle's future motion is affected by limited number of neighboring vehicles, it is reasonable to consider n_v closest vehicles only, which is also a common practice in trajectory prediction research [4], [58]. To represent interactive relationship between considered vehicles, a Laplacian matrix is calculated. Being different from usual

3D case [59], a 2D Laplacian matrix is calculated through weighting on time dimension. Then eigen vectors corresponding to the biggest k eigen values are concatenated with target vehicle's history trajectory, which forms a condition vector $X = (x_{1:t_h}^e, v_1, \dots, v_k)$ with fixed dimension $d_X = 2t_h + kn_v$ where the superscript e represents the target vehicle. The Laplacian matrix is calculated through

$$\begin{aligned} L &= D - A, \\ A &= (a_{ij})_{n_v \times n_v}, \\ D &= (d_{ij})_{n_v \times n_v}, \\ a_{ij} &= \exp\left(-\sum_{k=1}^{t_h} w_k d(x_k^i, x_k^j) / \sum_{k=1}^{t_h} w_k\right), \quad (2) \\ w_k &= \lambda^{t_h - k}, k = 1, \dots, t_h, \\ d_{ij} &= \begin{cases} \sum_{j=1}^{n_v} a_{ij}, & i = j \\ 0, & i \neq j \end{cases}, \end{aligned}$$

where $d(x_k^i, x_k^j)$ is ED between vehicle i and j at time k and λ is a decay parameter.

B. Estimation on GMMs based on a MDN

To calculate CKLD between two CPDFs, GMMs are introduced to approximate CPDF as $p(Y|X) = \sum_{i=1}^m \alpha_i(X) \phi_i(Y|X)$, where m is number of gaussian distribution hypothesis and $\phi_i(Y|X) = \exp\left(-\frac{\|Y - \mu_i(X)\|^2}{2\sigma_i(X)^2}\right) / (2\pi)^{d_X/2} \sigma_i(X)^{d_X}$. For $i = 1, \dots, m$, mixing coefficient $\alpha_i(X)$, mean $\mu_i(X)$, and variance $\sigma_i(X)$ are estimated through MDN [60], [61]. As shown in Fig.1, a Multi-Layer Perceptron (MLP) is applied for input X to obtain a feature encoding Z . Then three seprative Fully Connected (FC) layers are utilized to calculate parameters of GMMs. To enforce $\sum_{i=1}^m \alpha_i(x) = 1$, a softmax function is applied $\alpha_i(X) = \exp(FC(Z)_i) / \sum_{j=1}^m \exp(FC(Z)_j)$, where $FC(\bullet)$ represents a FC layer and the subscript i and j represents vector component. Means are unconstrained. Variances $\sigma_i(X)$ should be positive. A softplus function is applied hence $\sigma_i(X) = \log(1 + \exp(FC(Z)_i))$. Training loss function for MDN is $L_{mdn} = -\log(Y|X)$.

C. Calculation of CKLD through Monte-Carlo sampling

After GMMs are estimated for each condition X , CKLD can be computed. As in (1), for every sample condition $X_i, i = 1, \dots, n_1$ on $p_1(X)$, KLD can be calculated as

$$\begin{aligned} KLD(p_1(Y|X_i) || p_2(Y|X_i)) \\ = \int \log\left(\frac{p_1(Y|X_i)}{p_2(Y|X_i)}\right) p_1(Y|X_i) dY. \end{aligned} \quad (3)$$

Although KL divergence between two GMMs is not analytically tractable, some techniques are developed to estimate effectively. Hershey [62] compared 5 methods and concluded that MC sampling reaches clearly the best accuracy. Suppose

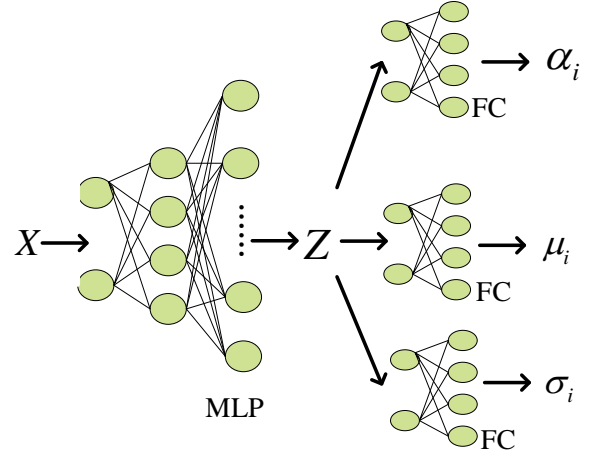


Fig. 1. MDN architecture.

Algorithm 1 CKLD between two traffic circumstances

Require: Sample pairs $(X_{1i}, Y_{1i}) \sim p_1(X, Y), i = 1, \dots, n_1$ and $(X_{2i}, Y_{2i}) \sim p_2(X, Y), i = 1, \dots, n_2$.
Ensure: $CKLD(p_1(Y|X) || p_2(Y|X))$.
 Calculate Laplacian matrix according to (2) and normalize condition to uniform dimension.
for $i = 1, \dots, n_1$ **do**
 Fit a MDN with (X_{1i}, Y_{1i}) and loss function L_{mdn} .
end for
for $i = 1, \dots, n_2$ **do**
 Fit a MDN with (X_{2i}, Y_{2i}) and loss function L_{mdn} .
end for
 $CKLD \leftarrow 0$
for $i = 1, \dots, n_1$ **do**
 Sampling $Y_j \sim p_1(Y|X_i), j = 1, \dots, n_{mc}$
 Calculate KLD according to (4).
 $CKLD \leftarrow CKLD + KLD(p_1(Y|X_i) || p_2(Y|X_i))$.
end for
 $CKLD \leftarrow CKLD / n_1$.
return CKLD

samples $Y_j, j = 1, \dots, n_{mc}$ are sampled from $p_1(Y|X_i)$, then CKLD can be calculated as

$$\begin{aligned} KLD(p_1(Y|X_i) || p_2(Y|X_i)) \\ = \sum_{j=1}^{n_{mc}} (\log p_1(Y_j|X_i) - \log p_2(Y_j|X_i)) / n_{mc}, \quad (4) \\ CKLD(p_1(Y|X) || p_2(Y|X)) \\ = \sum_{i=1}^{n_1} KLD(p_1(Y|X_i) || p_2(Y|X_i)) / n_1. \end{aligned}$$

The complete CKLD computation flow is summarized in Algorithm 1.

V. GENERATIVE REPLAY BASED LIFELONG TRAJECTORY PREDICTION

Based on our previous research on trajectory generation [63], a novel trajectory generation model trained by the standard GAN [64] loss is proposed to memorize data distribution

of tasks. With a vehicle trajectory prediction model, a lifelong vehicle trajectory prediction framework is realized.

A. Generator Conditioned on Relative Position Configuration

In a vehicle trajectory prediction task, we need to predict a target vehicle's future t_f trajectory according to its t_h history trajectory and its neighboring vehicles' histories. To facilitate generation process, a full prediction scenario is required to be generated that consists of target vehicle's and its neighboring vehicles' trajectories lasting for $t = t_h + t_f$ horizon.

Being different from traditional GANs that model generation procedure as mapping from a prior probability distribution to a target one, we are inspired by Quant GAN [65] and model prediction scenario generation as mapping between stochastic processes. Gaussian process with RBF kernel is selected as prior stochastic process. For single vehicle, m Gaussian process samplings are obtained, which constitutes $(N_{1:t}^1, N_{1:t}^2, \dots, N_{1:t}^{mn})$ for total n interactive vehicles. Conditional GANs are easier to train and make generated samples more controllable. Therefore, multiple vehicles' spatial configuration condition (C_1, C_2, \dots, C_n) is utilized as conditional inputs for our generator, where C_i represents condition for vehicle i . Spatial configuration condition (C_1, C_2, \dots, C_n) and input sampling $(N_{1:t}^1, N_{1:t}^2, \dots, N_{1:t}^{mn})$ are encoded by two MLPs individually first. To map sequential feature of Gaussian process into target data, a bidirectional GRU is utilized, where initial hidden states h_{gf0} and h_{gb0} are set by encoded position condition [66], [67]. In the bidirectional GRU, forward code $E_{f1:t}^j$ of agent j and backward code $E_{b1:t}^j$ are averaged to form sequential code $E_{1:t}^j$. A MLP is attached later to encode spatial relations. A fully connected (FC) layer is used later with tanh activation function to output spatiotemporal data $(\tilde{X}_{1:t}^1, \tilde{X}_{1:t}^2, \dots, \tilde{X}_{1:t}^n)$ of n agents. Complete framework is shown in Fig.2.

Let $G(\bullet)$ represents a GRU and $M(\bullet)$ a MLP unit, the generation procedure can be formulated as:

$$\begin{aligned} E_{f1:t}^j &= G_{gf} \left(\begin{array}{c} M_{g0} (N_{1:t}^{mj-m}, N_{1:t}^{mj-m+1}, \dots, N_{1:t}^{mj}) \\ h_{gf0} = M_{g1} (C_j) \end{array} \right), \\ E_{b1:t}^j &= G_{gb} \left(\begin{array}{c} M_{g0} (N_{1:t}^{mj-m}, N_{1:t}^{mj-m+1}, \dots, N_{1:t}^{mj}) \\ h_{gb0} = M_{g2} (C_j) \end{array} \right), \\ E_{1:t}^j &= M_{g3} \left((E_{f1:t}^j + E_{b1:t}^j) / 2 \right), \\ E_{1:t}^{i,j} &= E_{1:t}^i - E_{1:t}^j, \\ \tilde{X}_{1:t}^j &= M_{g5} \left(E_{1:t}^j, \left(\sum_{i=1, i \neq j}^n M_{g4} (E_{1:t}^{i,j}) \right) / (n-1) \right), \end{aligned} \quad (5)$$

for agent $i, j = 1, 2, \dots, n$.

B. MLP Based Regression Discriminator

As generated prediction scenario data are used for vehicle trajectory prediction task, it is of significant importance to maintain sequential dependences in generated samples. For this, a regression discriminator is first proposed by us to distinguish multiple agents' real data from generated one. The regression discriminator learns to model joint distribution of

inputs and outputs in a prediction task. Specifically, distribution of target vehicle's history data, target vehicle's future data and neighboring vehicles' history data are taken as inputs of the regression discriminator and modelled jointly. The regression discriminator outputs a classification probability that indicates degree of true.

Architecture of a regression discriminator is shown in Fig.3. For vehicle $j = 1, 2, \dots, n$, trajectory data $\tilde{X}_{1:t}^j$ are pre-processed through

$$\begin{aligned} \hat{X}_{1:t}^j &= \tilde{X}_{1:t}^j - \tilde{X}_{t_h}^e, \\ X_{1:t}^j &= \hat{X}_{1:t}^j / \left\| \hat{X}_{1:t}^1, \dots, \hat{X}_{1:t}^n \right\|_{\infty}. \end{aligned} \quad (6)$$

After centering and normalisation pre-processing, target vehicle data are separated from neighboring agents' data. A MLP is applied to the target vehicle to encode its history $X_{1:t_h}^e$ and future data $X_{t_h+1:t_h+t_f}^e$ into E_{rd}^h and E_{rd}^f specifically. Relative difference between the target vehicle and neighboring vehicles are calculated and encoded into E_{rd}^{nei} through a MLP with a mean pooling layer, which is invariant to neighboring vehicles' sequence. All codes are concatenated and encoded by two FC layers to get a feature vector F_{rd} . Finally, another FC layer is applied to the feature vector to obtain classification probability L_{rd} .

Computation workflow can be formalized as:

$$\begin{aligned} E_{rd}^h &= M_{rd0} (X_{1:t_h}^e), \\ E_{rd}^f &= M_{rd0} (X_{t_h+1:t_h+t_f}^e), \\ E_{rd}^{nei} &= \left(\sum_{i=1}^n M_{rd1} (X_{1:t_h}^{e,i \neq e}) \right) / (n-1), \\ F_{rd} &= FC_{rd0} (E_{rd}^h, E_{rd}^f, E_{rd}^{nei}), \\ L_{rd} &= FC_{rd1} (F_{rd}). \end{aligned} \quad (7)$$

C. Evolution of Generation Model

In a lifelong task chain $D = \{d_1, \dots, d_n\}$, generation models are required to be merged to a long-term model when a new task d_{i+1} arrives. In general, there are two fusion methods. As illustrated in Fig.4, one method [46], [48] merges generation model G_i trained by task d_i with the new task d_{i+1} . Generated samples from long-term model G_i and real samples drawn from d_{i+1} are combined as real samples to train a new generation model G_{i+1} , which we call Longterm-Data-Merge(LDM) method. Another method [47], [53] trains a temporal generation model G_{i+1}^t for the new task d_{i+1} . Generated samples from long-term model G_i and temporal model G_{i+1}^t are combined as real samples to train a new long-term model G_{i+1} , which we call Longterm-Temporal-Merge(LTM) method. Although LDM performs better than LTM method intuitively, they are both applied to our lifelong task and compared.

D. Task model for vehicle trajectory prediction

To perform vehicle trajectory prediction task with generated prediction scenario, a prediction model is proposed. As with mainstream trajectory prediction methods, history information

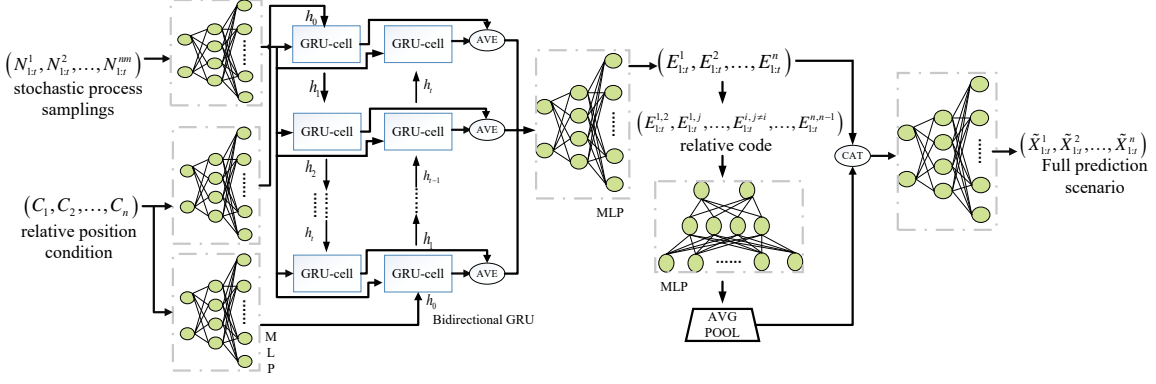


Fig. 2. Proposed generator framework.

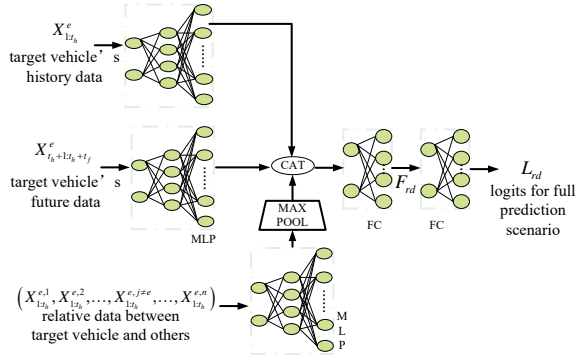


Fig. 3. Proposed regression discriminator framework.

and interactive relationship between target vehicle and neighboring vehicles are utilized to predict future trajectory of the target vehicle. Overall architecture is shown in Fig.5. First, target vehicle's trajectory $X^e_{1:t_h}$ is separated from neighboring vehicles and encoded by a LSTM layer. Then, difference between target vehicle and others are encoded by a MLP and a mean pooling layer. These two parts are concatenated and encoded by a MLP further. A LSTM layer and a MLP is used to output predicted trajectories $\tilde{X}^e_{t_h+1:t_h+t_f}$.

Let $LSTM(\bullet)$ represents a LSTM unit, then for a prediction workflow, we have

$$\begin{aligned} E_r^h &= M_{r2} \left(LSTM \left(M_{r0} \left(X^e_{1:t_h} \right) \right) \right), \\ E_{rd}^{nei} &= M_{r2} \left(\sum_{i=1}^n M_{r1} \left(X^e_{1:t_h} \right) \right), \\ \tilde{X}^e_{t_h+1:t_h+t_f} &= M_{r3} \left(LSTM \left(E_r^h, E_{rd}^{nei} \right) \right). \end{aligned} \quad (8)$$

VI. EXPERIMENTS AND ANALYSES

All experiments are realized via Pytorch [68]. Running environment is Ubuntu 16.04, Intel Core i9-9900X CPU, GeForce GTX 1080 Ti, and 64GB RAM. All code including CKLD calculation and lifelong learning experiments and pre-processed data are available online¹.

A. Dataset and generation model setup

To construct a lifelong task chain, five sub-datasets recorded in different locations are selected from four open datasets. Some traffic circumstances are illustrated in Fig.6.

- NGSIM dataset [69]. The NGSIM dataset contains two sub-datasets, US101 dataset and I801 dataset that are recorded on southbound US 101 and eastbound I-80 specifically. As tremendous vehicle trajectories are recorded, it's time consuming to learn them all. Therefore, 7:50 a.m. to 8:05 a.m. trajectory records in US101 dataset are selected and named with *US101-1*. To simulate a case happened in real application where drives visit same place at different time period, 8:05 a.m. to 8:15 a.m trajectory records in US101 dataset are selected as d_5 and named with *US101-2*. Without losing generality, we keep full prediction scenarios that contains 4 and 5 vehicles only to ease the learning burden furtherly and over 188k items still remains. I801 dataset are refined as the same way and over 129k items remains. The selected datasets are separated into training, validation and testing dataset by 7:1:2.
- HighD dataset [70]. The highD dataset is a new dataset of naturalistic vehicle trajectories recorded at six different locations on Germany highways, which results in sixty recordings. Considering learning burden, the 20th recording is selected and pre-processed, which results in 88k items left for training. It is noted that as trajectories are recorded in 10HZ in NGSIM while 25HZ in HighD dataset, we aim to generate trajectories in 5HZ. As recording items is not comparable with NGSIM dataset in scale, full prediction scenarios that contains 2 to 9 vehicles are kept. For ease of representation, this dataset is called *highd20*.
- Interaction dataset [71]. The interaction dataset contains naturalistic motions of various traffic participants in a variety of highly interactive driving scenarios from different countries. Trajectories in DR_CHN_Merging_ZS map is a lane merging dataset in China urban area. To be comparable with other datasets in scale, five trajectories records in the map are merged into a dataset named *inter5d*, which consists of 126k items.

¹<https://github.com/CliffBao/GRTP>

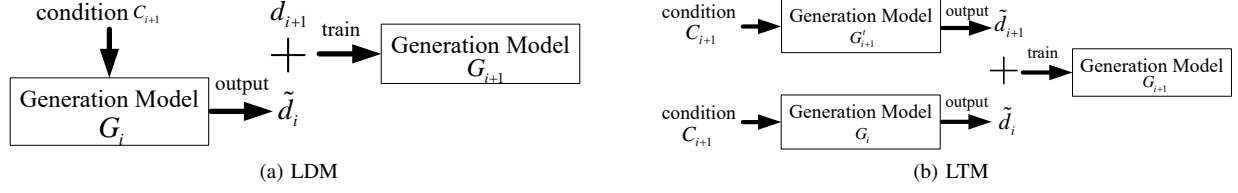


Fig. 4. Two methods of generation model evolving.

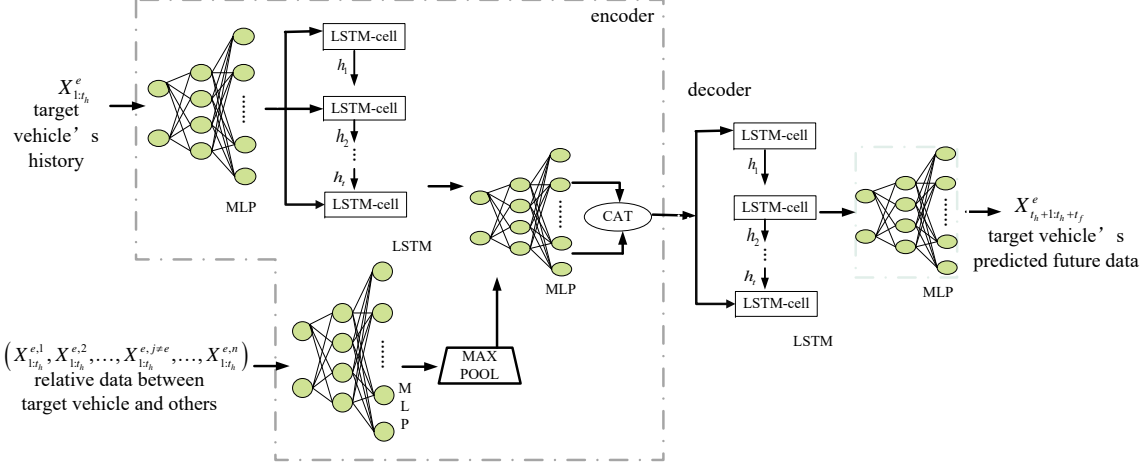


Fig. 5. Proposed vehicle trajectory prediction model.

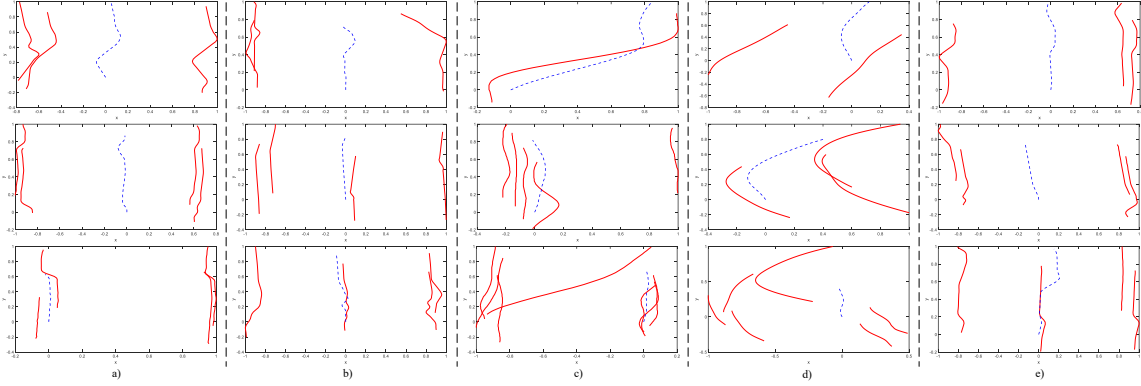


Fig. 6. Illustration of traffic circumstances in a) *US101-1*, b) *i801*, c) *highd20*, d) *inter5d*, and e) *US101-2*. Blue dash line indicates target vehicle. Red lines indicate neighboring vehicles.

Therefore, a lifelong task chain is formed by above five datasets

$$\begin{aligned} D &= \{d_1, d_2, d_3, d_4, d_5\} \\ &= \{US101-1, i801, highd20, inter5d, US101-2\}. \end{aligned} \quad (9)$$

B. Divergence between datasets

To measure CKLD between two datasets, we fix maximum vehicle number to $n_v = 5$ and only top $k = 3$ eigen vectors are extracted. Gaussian hypothesis number of GMMs is set into $m = 20$. The MDN is optimized through adam [72] optimizer with learning rate $\gamma_m = 0.0004$ and batch size $b_m = 4096$. CKLD results of pairwise datasets are presented in Table I. In Table I, CKLD between d_1 and d_5 is the closest, which is

TABLE I
CKLD BETWEEN DATASETS

CKLD	dataset 2				
	d_1	d_2	d_3	d_4	d_5
dataset 1					
d_1	0	22.14	503.22	98.43	16.72
d_2	29.00	0	756.03	98.57	24.44
d_3	88.46	53.87	0	142.26	121.42
d_4	75.05	63.54	652.00	0	62.50
d_5	19.57	17.09	511.17	93.49	0

expected as they are collected in the same highway at different time period. Divergence between d_3 and other datasets are quite large because only d_3 is collected in Germany. The same goes for d_4 .

TABLE II
RMSE(M) COMPARISON OF PROPOSED TASK MODEL
AND OTHER COMMON METHODS ON COMPLETE
NGSIM DATASET.

PH(s)	1	2	3	4	5
CV	0.73	1.78	3.13	4.78	6.68
GAIL-GRU	0.69	1.51	2.55	3.65	4.71
S-LSTM	0.65	1.31	2.16	3.25	4.55
CS-LSTM	0.61	1.27	2.09	3.10	4.37
Ours	0.55	1.28	2.18	3.30	4.64

C. Validation of trajectory prediction model

We compare proposed the task model performance with other benchmark trajectory prediction models.

- Constant Velocity(CV). A constant velocity Kalman filter reported in [17].
- GAIL-GRU. A generative adversarial imitation learning model described in [73].
- LSTM with fully connected social pooling (S-LSTM). This uses the fully connected social pooling described in [74] and generates a unimodal output distribution.
- LSTM with convolutional social pooling (CS-LSTM). This uses convolutional social pooling and generates a unimodal output distribution [17]. As designing an extraordinary prediction model is not our main goal, we use results reported in [17].

To present a fair comparison, full NGSIM dataset is used to demonstrate validity of proposed prediction model, which is different from lifelong setups. RMSE performance for varied prediction horizon (PH) is given out in Table II. Suppose BS batch size predicted future trajectories $\tilde{X}_t^i, i = 1, 2, \dots, BS$ at time t are calculated and real future trajectories $X_t^i, i = 1, 2, \dots, BS$ are available, then RMSE at time t is

$$RMSE(t) = \sqrt{\sum_{i=1}^{BS} \|X_t^i - \tilde{X}_t^i\|^2 / BS}. \quad (10)$$

From Table II, it can be concluded that the task model possess similar prediction ability with mainstream methods. As we are not aiming to significantly improve prediction accuracy on single dataset but instead improve generality and lifelong prediction ability over multiple tasks, the performance of proposed prediction model may not compatible with state-of-the-art methods.

D. Lifelong trajectory prediction

In R2GAN, a generator takes inputs as several Gaussian process samplings and label conditions of trajectories that indicate relative position to a target vehicle. For example, -1 indicates a vehicle is located on the left lane of the target vehicle at the beginning. The generator outputs corresponding vehicle trajectory snippets lasting for 8 seconds, i.e. 41 steps. For a regression discriminator, real or generated prediction scenario are classified as real or fake. R2GAN is trained by adam [72] optimizer with learning rate $\gamma = 0.0001$. Non-saturating GAN loss function is applied as in [75]. To demonstrate lifelong prediction ability of proposed framework, three other methods are realized and compared with our approach.

- Generative replay based trajectory prediction (GRTP). This is the proposed lifelong trajectory prediction model based on generative replay. Resulted from two fusion methods LDM and LTM, the GRTP is furtherly classified into GRTP-D and GRTP-T specifically.
- Joint training (JT). Joint training violates essential storage limitation and assumes that all data are available. This is regarded as the best possible performance over any lifelong learning methods.
- Fixed model (FM). A trajectory model trained by task d_1 will not be adjusted anymore and will be applied to new tasks directly.
- Fine tuning (FT). A trajectory model is trained while new task data d_i are available. This is a possible choice but is expected to forget everything about old tasks. From some perspective, research works on trajectory prediction model design and optimization can be categorized into this method, although they did not test the model trained on new dataset on old ones.

The RMSE plots through the full lifelong task chain is illustrated in Fig.7, where local RMSE around t_f is zoomed in by a mini plot. As the lifelong task chain proceeds from d_1 to d_5 , prediction performance on future 5 seconds horizon is validated on more tasks. Exact numeric result after addressing d_5 is given out in Table III. From experiment results, we can see that

- it is obvious FT forgets old knowledge while attaining new knowledge. Indeed, FT is a common practice when new task arrives and performs well if divergence between old and new task is small. From CKLD computation result, divergence between d_1 , d_2 , and d_5 is relatively small. Therefore, FT performs well on these three tasks after d_5 . On the contrary, CKLD between d_5 and d_3 , d_4 is relatively large, which results in poor performance on d_3 and d_4 after tuning on task d_5 . The same consequence of applying FT to new task is also remarkable in Fig.7.
- FM is trained on d_1 only. As a result, good performance on d_1 and d_5 is attainable after d_5 while large RMSE is observed on other tasks.
- The proposed GRTP-T and GRTP-D perform well consistently over all tasks and possess close RMSE to JT. As JT stores all task data and can be considered as the best possible performance in lifelong task chain, we can conclude that GRTP mitigates catastrophic forgetting and realizes lifelong learning whether with LDM or LTM fusion method.
- Although intuitively thinking, GRTP-D will outperforms GRTP-T as it merges longterm generation model with new data directly and avoids training a new generation model on new data, which avoids distribution learning bias introduced by the temporal generation model. However, it can be observed in Table III and Fig.7 that no significant performance gap exists between them. This observation demonstrates that minor or even no distribution bias is introduced by our proposed R2GAN, which validates effectiveness of proposed R2GAN and model merging method implicitly.

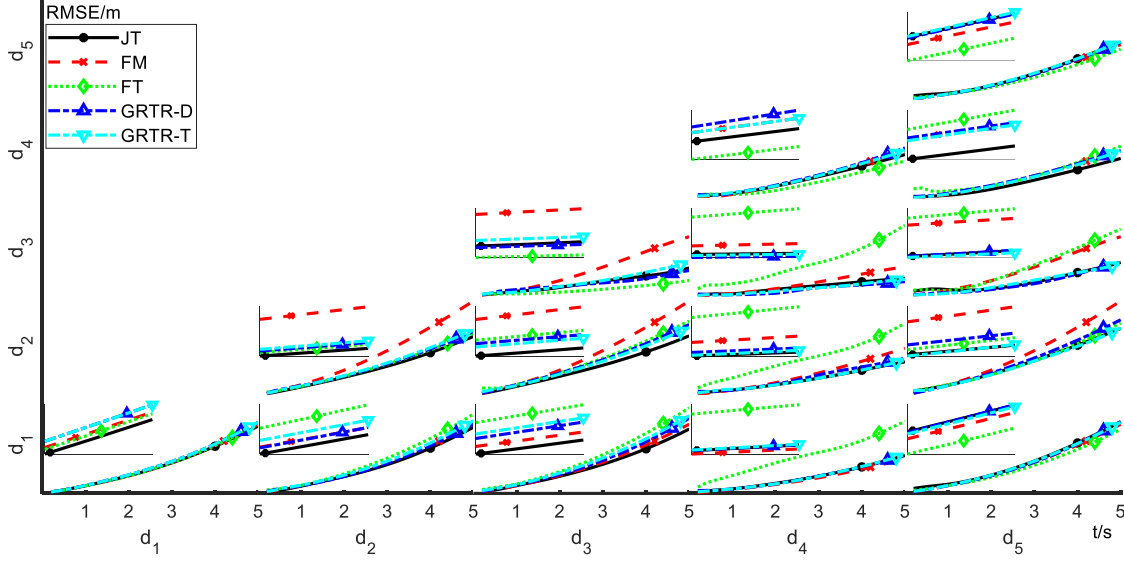


Fig. 7. RMSE for full lifelong task chain.

TABLE III
RMSE(M) COMPARISON OF DIFFERENT MODELS AFTER
FINISHING LIFELONG TASK CHAIN.

	PH(s)	1	2	3	4	5
d_1	JT	0.93	1.86	3.30	5.14	7.30
	FM	0.79	1.82	3.18	4.94	7.06
	FT	0.79	1.63	2.86	4.49	6.53
	GRTP-D	0.80	1.80	3.19	5.06	7.32
	GRTP-T	0.81	1.86	3.29	5.08	7.25
d_2	JT	1.03	2.15	3.53	5.09	6.80
	FM	1.08	2.56	4.54	6.91	9.57
	FT	1.09	2.09	3.48	5.26	7.34
	GRTP-D	1.00	2.25	3.79	5.61	7.67
	GRTP-T	0.99	2.10	3.49	4.97	6.85
d_3	JT	1.46	1.66	2.97	4.94	7.01
	FM	1.36	3.30	6.06	9.24	12.25
	FT	1.32	3.21	6.49	10.01	13.81
	GRTP-D	0.81	1.47	2.68	4.66	7.07
	GRTP-T	0.70	1.88	3.41	4.93	6.66
d_4	JT	0.36	0.96	1.89	2.95	4.09
	FM	0.44	1.28	2.34	3.55	4.86
	FT	0.79	1.32	2.30	3.69	5.33
	GRTP-D	0.57	1.35	2.33	3.56	4.89
	GRTP-T	0.47	1.25	2.25	3.46	4.82
d_5	JT	0.71	1.48	2.70	4.21	5.92
	FM	0.65	1.52	2.65	4.07	5.67
	FT	0.62	1.31	2.32	3.63	5.20
	GRTP-D	0.62	1.47	2.64	4.15	5.90
	GRTP-T	0.63	1.54	2.75	4.24	5.96

VII. CONCLUSION

Maintaining consistent performance on vehicle trajectory prediction over different traffic circumstances is of significant importance for safe driving. Lifelong trajectory prediction is first addressed by us. Key problem hiding lifelong learning is catastrophic forgetting which arises from existence of spatiotemporal dependency divergence. To analysis the divergence between different traffic circumstances, CKLD is calculated based on GMMs approximation and MC sampling. Then a R2GAN is developed to generate dynamic number of vehicles

in traffic circumstances, which guarantees inherent spatiotemporal dependency through a novel regression discriminator. Two methods are applied to construct a lifelong R2GAN model, LTM and LDM. LTM merges generated trajectories from long-term R2GAN and temporal R2GAN to train a new long-term generation model. LDM takes trajectories generated from long-term R2GAN and sampled from real dataset as real data to update the long-term generation model. Different spatiotemporal dependency are remembered by the long-term generation model that capable of generating samples from all processed tasks, thus mitigating catastrophic forgetting problem. Both merging method are validated through a constructed lifelong task chain and fulfill lifelong trajectory prediction task with consistent performance.

In this work, four datasets are applied to verify effectiveness of proposed lifelong framework. In future work, more diversified traffic circumstances can be introduced to improve lifelong prediction performance.

VIII. ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (Grant No. 91848111, No. 62103393).

REFERENCES

- [1] S. Qiao, N. Han, J. Wang, R. H. Li, L. A. Gutierrez, and X. Wu, "Predicting long-term trajectories of connected vehicles via the prefix-projection technique," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2017.
- [2] L. Hou, L. Xin, S. E. Li, B. Cheng, and W. Wang, "Interactive trajectory prediction of surrounding road users for autonomous driving using structural-lstm network," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–11, 2019.
- [3] J. Li, H. Ma, and M. Tomizuka, "Interaction-aware multi-agent tracking and probabilistic behavior prediction via adversarial learning," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 6658–6664.
- [4] X. Li, X. Ying, and M. C. Chuah, "Grip: Graph-based interaction-aware trajectory prediction," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 3960–3966.

- [5] R. Chandra, U. Bhattacharya, A. Bera, and D. Manocha, "Traffic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8483–8492.
- [6] A. Dulian and J. C. Murray, "Multi-modal anticipation of stochastic trajectories in a dynamic environment with conditional variational autoencoders," *arXiv preprint arXiv:2103.03912*, 2021.
- [7] M. Bahari, I. Nejjar, and A. Alahi, "Injecting knowledge in data-driven vehicle trajectory predictors," *arXiv preprint arXiv:2103.04854*, 2021.
- [8] A. Zyner, S. Worrall, and E. Nebot, "Naturalistic driver intention and path prediction using recurrent neural networks," *IEEE transactions on intelligent transportation systems*, vol. 21, no. 4, pp. 1584–1594, 2019.
- [9] S. Ammoun and F. Nashashibi, "Real time trajectory prediction for collision risk estimation between vehicles," in *2009 IEEE 5th International Conference on Intelligent Computer Communication and Processing*. IEEE, 2009, pp. 417–422.
- [10] P. Lytrivis, G. Thomaidis, and A. Amditis, "Cooperative path prediction in vehicular environments," in *2008 11th International IEEE Conference on Intelligent Transportation Systems*. IEEE, 2008, pp. 803–808.
- [11] Q. Tran and J. Firl, "Online maneuver recognition and multimodal trajectory prediction for intersection assistance using non-parametric regression," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE, 2014, pp. 918–923.
- [12] G. S. Aoude, V. R. Desara, L. H. Stephens, and J. P. How, "Behavior classification algorithms at intersections and validation using naturalistic data," in *2011 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2011, pp. 601–606.
- [13] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," in *2013 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2013, pp. 797–802.
- [14] T. Streubel and K. H. Hoffmann, "Prediction of driver intended path at intersections," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*. IEEE, 2014, pp. 134–139.
- [15] L. Xu, J. Hu, H. Jiang, and W. Meng, "Establishing style-oriented driver models by imitating human driving behaviors," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2522–2530, 2015.
- [16] N. Deo, A. Rangesh, and M. M. Trivedi, "How would surround vehicles move? a unified framework for maneuver classification and motion prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 2, pp. 129–140, 2018.
- [17] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 1468–1476.
- [18] K. Messaoud, I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi, "Relational recurrent neural networks for vehicle trajectory prediction," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 1813–1818.
- [19] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.
- [20] A. C. Sanderson and A. K. Wong, "Pattern trajectory analysis of nonstationary multivariate data," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 10, no. 7, pp. 384–392, 1980.
- [21] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007.
- [22] M. T. Robinson, "The temporal development of collision cascades in the binary-collision approximation," *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, vol. 48, no. 1–4, pp. 408–413, 1990.
- [23] A. Ismail and A. Vigneron, "A new trajectory similarity measure for gps data," in *Proceedings of the 6th ACM SIGSPATIAL International Workshop on GeoStreaming*, 2015, pp. 19–22.
- [24] N. Pelekis, I. Kopanakis, G. Marketos, I. Ntoutsi, G. Andrienko, and Y. Theodoridis, "Similarity search in trajectory databases," in *14th International Symposium on Temporal Representation and Reasoning (TIME'07)*. IEEE, 2007, pp. 129–140.
- [25] H. Su, S. Liu, B. Zheng, X. Zhou, and K. Zheng, "A survey of trajectory distance measures and performance evaluation," *The VLDB Journal*, vol. 29, no. 1, pp. 3–32, 2020.
- [26] M. D. Donsker and S. R. S. Varadhan, "Asymptotic evaluation of certain markov process expectations for large time. iv," *Communications on Pure and Applied Mathematics*, vol. 36, no. 2, pp. 183–212, 1983. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160360204>
- [27] M. I. Belghazi, A. Baratin, S. Rajeshwar, S. Ozair, Y. Bengio, A. Courville, and D. Hjelm, "Mutual information neural estimation," in *International Conference on Machine Learning*. PMLR, 2018, pp. 531–540.
- [28] Q. Wang, S. R. Kulkarni, and S. Verdú, "A nearest-neighbor approach to estimating divergence between continuous random vectors," in *2006 IEEE International Symposium on Information Theory*. IEEE, 2006, pp. 242–246.
- [29] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola, "A kernel method for the two-sample-problem," *Advances in neural information processing systems*, vol. 19, pp. 513–520, 2006.
- [30] L. Song, K. Fukumizu, and A. Gretton, "Kernel embeddings of conditional distributions: A unified kernel framework for nonparametric inference in graphical models," *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 98–111, 2013.
- [31] Y. Ren, J. Zhu, J. Li, and Y. Luo, "Conditional generative moment-matching networks," *Advances in Neural Information Processing Systems*, vol. 29, pp. 2928–2936, 2016.
- [32] C.-X. Ren, P. Ge, D.-Q. Dai, and H. Yan, "Learning kernel for conditional moment-matching discrepancy-based image classification," *IEEE transactions on cybernetics*, 2019.
- [33] E. G. Tabak, G. Trigila, and W. Zhao, "Data driven conditional optimal transport," *Machine Learning*, pp. 1–21, 2021.
- [34] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.
- [35] M. Delange, R. Aljundi, M. Masana, S. Parisot, X. Jia, A. Leonardis, G. Slabaugh, and T. Tuytelaars, "A continual learning survey: Defying forgetting in classification tasks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
- [36] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *arXiv preprint arXiv:1606.04671*, 2016.
- [37] A. Rosenfeld and J. K. Tsotsos, "Incremental learning through deep adaptation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 3, pp. 651–663, 2018.
- [38] V. Lomonaco and D. Maltoni, "Core50: a new dataset and benchmark for continuous object recognition," in *Conference on Robot Learning*. PMLR, 2017, pp. 17–26.
- [39] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 12, pp. 2935–2947, 2017.
- [40] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska et al., "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
- [41] R. Aljundi, P. Chakravarty, and T. Tuytelaars, "Expert gate: Lifelong learning with a network of experts," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3366–3375.
- [42] X. Yao, T. Huang, C. Wu, R.-X. Zhang, and L. Sun, "Adversarial feature alignment: Avoid catastrophic forgetting in incremental task lifelong learning," *Neural computation*, vol. 31, no. 11, pp. 2266–2291, 2019.
- [43] S. Hou, X. Pan, C. C. Loy, Z. Wang, and D. Lin, "Lifelong learning via progressive distillation and retrospection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 437–452.
- [44] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010.
- [45] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," *arXiv preprint arXiv:1706.08840*, 2017.
- [46] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," *arXiv preprint arXiv:1705.08690*, 2017.
- [47] X. Su, S. Guo, T. Tan, and F. Chen, "Generative memory for lifelong learning," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 6, pp. 1884–1898, 2019.
- [48] T. Lesort, H. Caselles-Dupré, M. Garcia-Ortiz, A. Stoian, and D. Filliat, "Generative models from the perspective of continual learning," in *2019 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2019, pp. 1–8.
- [49] T. Lesort, A. Gepperth, A. Stoian, and D. Filliat, "Marginal replay vs conditional replay for continual learning," in *International Conference on Artificial Neural Networks*. Springer, 2019, pp. 466–480.
- [50] X. Liu, C. Wu, M. Menta, L. Herranz, B. Raducanu, A. D. Bagdanov, S. Jui, and J. v. de Weijer, "Generative feature replay for class-incremental learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 226–227.

- [51] H. Li, W. Dong, and B.-G. Hu, "Incremental concept learning via online generative memory recall," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.
- [52] G. M. Van de Ven and A. S. Tolias, "Generative replay with feedback connections as a general strategy for continual learning," *arXiv preprint arXiv:1809.10635*, 2018.
- [53] N. Kamra, U. Gupta, and Y. Liu, "Deep generative dual memory network for continual learning," *arXiv preprint arXiv:1710.10368*, 2017.
- [54] C. Wu, L. Herranz, X. Liu, Y. Wang, J. Van de Weijer, and B. Raducanu, "Memory replay gans: learning to generate images from new categories without forgetting," *arXiv preprint arXiv:1809.02058*, 2018.
- [55] Y. Xiang, Y. Fu, P. Ji, and H. Huang, "Incremental learning using conditional adversarial networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6619–6628.
- [56] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, and T. Tuytelaars, "Memory aware synapses: Learning what (not) to forget," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 139–154.
- [57] C. Fernando, D. Banarse, C. Blundell, Y. Zwols, D. Ha, A. A. Rusu, A. Pritzel, and D. Wierstra, "Pathnet: Evolution channels gradient descent in super neural networks," *arXiv preprint arXiv:1701.08734*, 2017.
- [58] D. Lee, Y. Gu, J. Hoang, and M. Marchetti-Bowick, "Joint interaction and trajectory prediction for autonomous driving using graph neural networks," *arXiv preprint arXiv:1912.07882*, 2019.
- [59] R. Chandra, T. Guan, S. Panuganti, T. Mittal, U. Bhattacharya, A. Bera, and D. Manocha, "Forecasting trajectory and behavior of road-agents using spectral clustering in graph-lstms," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4882–4890, 2020.
- [60] C. M. Bishop, "Mixture density networks," 1994.
- [61] J. Rothfuss, F. Ferreira, S. Walther, and M. Ulrich, "Conditional density estimation with neural networks: Best practices and benchmarks," *arXiv preprint arXiv:1903.00954*, 2019.
- [62] J. R. Hershey and P. A. Olsen, "Approximating the kullback leibler divergence between gaussian mixture models," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, vol. 4. IEEE, 2007, pp. IV–317.
- [63] P. Bao, Z. Chen, J. Wang, and D. Dai, "Multiple agents' spatiotemporal data generation based on recurrent regression dual discriminator gan," *Neurocomputing*, vol. 468, pp. 370–383, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0925231221015344>
- [64] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," *arXiv preprint arXiv:1406.2661*, 2014.
- [65] M. Wiese, R. Knobloch, R. Korn, and P. Kretschmer, "Quant gans: Deep generation of financial time series," *Quantitative Finance*, vol. 20, no. 9, pp. 1419–1440, 2020.
- [66] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3128–3137.
- [67] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.
- [68] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, pp. 8026–8037, 2019.
- [69] V. Alexiadis, J. Colyar, J. Halkias, R. Hranac, and G. McHale, "The next generation simulation program," *Institute of Transportation Engineers. ITE Journal*, vol. 74, no. 8, p. 22, 2004.
- [70] R. Krajewski, J. Bock, L. Kloecker, and L. Eckstein, "The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2118–2125.
- [71] W. Zhan, L. Sun, D. Wang, H. Shi, A. Clausse, M. Naumann, J. Kümmerle, H. Königshof, C. Stiller, A. de La Fortelle, and M. Tomizuka, "INTERACTION Dataset: An INTERnational, Adversarial and Cooperative moTION Dataset in Interactive Driving Scenarios with Semantic Maps," *arXiv:1910.03088 [cs, eess]*, 2019.
- [72] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [73] A. Kuefler, J. Morton, T. Wheeler, and M. Kochenderfer, "Imitating driver behavior with generative adversarial networks," in *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2017, pp. 204–211.
- [74] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 961–971.
- [75] I. Goodfellow, "Nips 2016 tutorial: Generative adversarial networks," *arXiv preprint arXiv:1701.00160*, 2016.

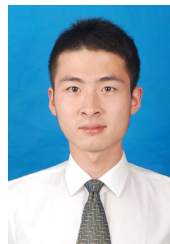


Peng Bao Peng Bao received the B.S. degree from the University of Science and Technology of China (USTC) in 2014. He is now a PhD candidate in the Department of Automation, USTC. His research interests include intelligent information processing, mobile robots SLAM and deep learning.



Zonghai Chen Zonghai Chen was born in Anhui, China, in 1963. He received the B.S. degree in automation and the M.E. degree in control theory and control engineering from the University of Science and Technology of China (USTC), Hefei, China, in 1988 and 1991, respectively. He has been a Professor with the Department of Automation, USTC, since 1998. His research interests include modeling and control of complex systems, intelligent robotic and information processing, energy management technologies for electric vehicles, and smart microgrids.

Prof. Chen is a member of the Robotics Technical Committee and Modelling, Identification and Signal Processing Technical Committee of the International Federation of Automation Control (IFAC). He was a recipient of special allowances from the State Council of PR China.



Jikai Wang JiKai Wang received the B.S. degree from the University of Yanshan in 2014. He is now a post doctor in the Department of Automation, University of Science and Technology of China (USTC), China. His research interests include knowledge representation, intelligent information processing, robotics, visual SLAM, and machine learning.



Deyun Dai Deyun Dai received the B.S. degree from Harbin Engineering University, in 2016. She is now a PhD candidate in the Department of Automation, USTC. Her research interests include computer vision, environment perception in autonomous driving scenarios.



Hao Zhao Hao Zhao received the B.S. and M.S. degree from the Southwest University of Science and Technology in 2014 and 2017. He is now a PhD candidate in the Department of Automation, University of Science and Technology of China (USTC). His research interests include object recognition and detection, scene perception and knowledge representation.