Pattern Recognition 133 (2023) 109030

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

CSR: Cascade Conditional Variational Auto Encoder with Socially-aware Regression for Pedestrian Trajectory Prediction

Hao Zhou[a,d], Dongchun Ren[b], Xu Yang[d], Mingyu Fan[b,c,*], Hai Huang[a,*]

[a] National Key Laboratory of Science and Technology of Underwater Vehicle, Harbin Engineering University, Harbin, China
[b] Research Center for Autonomous Vehicles, Meituan, Beijing, China
[c] College of Computer Science, Wenzhou University, Wenzhou, China
[d] State Key Laboratory of Management and Control for Complex System, Institute of Automation, Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Article history:
Received 18 November 2021
Revised 27 July 2022
Accepted 4 September 2022
Available online 7 September 2022

2021 MSC:
11-01
99-00

Keywords:
Pedestrian trajectory prediction
Socially-aware model
Conditional variational autoencoder (CVAE)

ABSTRACT

Pedestrian trajectory prediction is a key technology in many real applications such as video surveillance, social robot navigation, and autonomous driving, and significant progress has been made in this research topic. However, there remain two limitations of previous studies. First, the losses of the last time steps are heavier weighted than that of the beginning time steps in the objective function at the learning stage, causing the prediction errors generated at the beginning to accumulate to large errors at the last time steps at the inference stage. Second, the prediction results of multiple pedestrians in the prediction horizon might be socially incompatible with the interactions modeled by past trajectories. To overcome these limitations, this work proposes a novel trajectory prediction method called CSR, which consists of a cascaded conditional variational autoencoder (CVAE) module and a socially-aware regression module. The CVAE module estimates the future trajectories in a cascaded sequential manner. Specifically, each CVAE concatenates the past trajectories and the predicted location points so far as the input and predicts the adjacent location at the following time step. The socially-aware regression module generates offsets from the estimated future trajectories to produce the corrected predictions, which are more reasonable and accurate than the estimated trajectories. Experiments results demonstrate that the proposed method exhibits significant improvements over state-of-the-art methods on the Stanford Drone Dataset (SDD) and the ETH/UCY dataset of approximately 38.0% and 22.2%, respectively. The code is available at https://github.com/zhouhao94/CSR.

1. Introduction

The prediction of future pedestrian trajectories according to their past trajectories is a key issue in many applications such as autonomous driving [1,2], robotic systems [3], and surveillance system [4]. For example, in the autonomous driving scenario, an accurate prediction of pedestrian trajectories is required for the vehicle to plan a safe and effective trajectory; otherwise, a car accident may occur.

One of the key challenges in pedestrian trajectory prediction is the high degree of uncertainty caused by two factors, *i.e.*, the inherently multimodal attribute, and the complex social interactions among pedestrians. In Fig. 1, the prediction errors of PECNet [5] and the proposed CSR method are visualized using average displacement error (ADE) between the ground truth and predictions. This figure reveals that the degree of motion uncertainty increases over the prediction time horizon. This is expected because the motion patterns and social interactions of pedestrians change over time; thus, the predicted points at the last time steps might be quite erratic.

Recently, pioneering algorithms have been proposed in the pedestrian trajectory prediction community, which generally use one shared model to predict future locations at different time steps. However, at the learning stage, using a shared model to optimize losses of all time steps means exerting more emphasis (larger weights) on the losses of the last time steps that dominate the objective function. The unbalanced optimization tendency can cause the large errors generated at the beginning to accumulate to large errors at the last time steps. On the other hand, previous works use the past trajectories to model the social interactions among pedestrians. However, the interaction features extracted from the past trajectories can only represent the social interactions in the past but cannot be used to examine the social reasonability in the future. Humans can predict the motion of the surrounding

* Corresponding authors.
E-mail addresses: fanmingyu@wzu.edu.cn (M. Fan), haihus@163.com (H. Huang).

https://doi.org/10.1016/j.patcog.2022.109030
0031-3203/

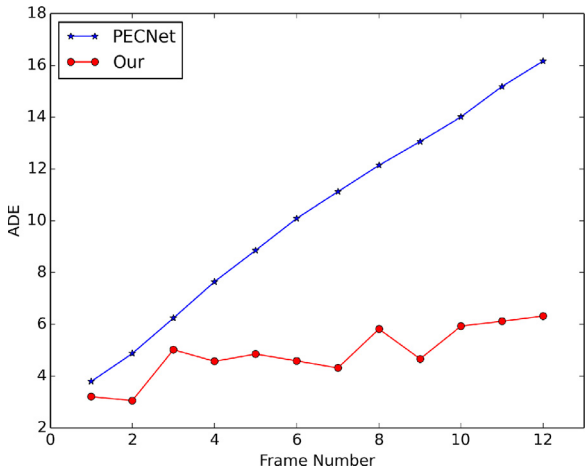

Fig. 1. The variations of average displacement error (ADE) *versus* time frame for the PECNet and the proposed CSR on the SDD dataset.

pedestrians and then adjust their motion plan by considering the prediction result. Consequently, the interactions among pedestrians based on the predicted trajectories should also be considered for improving the prediction accuracy.

This work proposes a novel trajectory prediction method, named CSR, which consists of a cascaded CVAE module and a socially-aware regression module. The detailed structure of CSR is illustrated in Fig. 2. In the cascaded CVAE module, unshared CVAEs are used to sequentially predict the future locations at different time steps using the updated past trajectories as the inputs. Each updated past trajectory is obtained by concatenating the past trajectory and the predicted points so far. It is clear that the proposed method adopts the auto-regressive strategy. Moreover, our cascaded CVAE module balances the losses of different time steps in the objective function and respectively minimizes the errors using unshared, independent CVAEs. This is a significant advantage over other methods because the larger errors of long-term prediction and smaller errors of short-term prediction does not need to reconcile with each other in the minimization to achieve the suboptimal results of every time step.

Besides, the socially-aware regression module is proposed to extract the interaction features from both the past trajectories and the predicted trajectories and then use them to refine the final predictions. Specifically, this module first extracts features of both the past trajectories and the predicted trajectories by two encoders E_{opast} and $E_{pfuture}$. Then, the two features are concatenated and fed to a social attention mechanism to extract the global interaction feature. Finally, the decoder $D_{offsets}$ decodes the interaction features to produce offsets to refine the future trajectories predicted by the cascaded CVAE module. The ADE of the proposed CSR method is also visualized in Fig. 1, from which it is evident that the prediction error increases slowly over time when compared to the latest state-of-the-art method PECNet.

The main contributions of this paper are summarized as follows:

- By analyzing the objective function for trajectory prediction, we argue that the proposed method could balance the losses of different time steps using unshared prediction models, which is advantageous over the classic prediction methods using a shared prediction model for all time steps. Our prediction models are able to respectively minimize the prediction errors at different time steps, achieve lower prediction errors, and thus make more accurate predictions using longer updated past trajectories.
- The proposed socially-aware regression module extracts interaction features from both the past trajectories and the predicted trajectories. Via this strategy, the compatibility of the interaction coding and the predicted trajectories is concerned for correcting the crude predictions. Thus, the regression module can generate offsets to improve the predicted future trajectories.
- Extensive experiments have been conducted on two benchmark trajectory prediction datasets, the Stanford Drone Dataset (SDD) [6] and the ETH&UCY dataset [7,8]. The results indicate that the proposed CSR method surpasses the latest state-of-the-art methods by a large margin.

The remainder of this paper is organized as follows. Some closely related works on trajectory prediction are discussed in Section 2. The proposed CSR method is introduced in Section 3. In Section 4, experiments on 2 benchmark pedestrian trajectory prediction datasets are provided. Some concluding remarks are given in Section 5.

2. Related works

2.1. Direct prediction methods

The direct strategy involves the utilization of one model that is capable of predicting the entire future trajectory in one shot. Most trajectory prediction methods are characterized by direct prediction. For example, early works adopted the Bayesian network method [9], the Gaussian process regression method [10], and the kinematic method [11] to directly predict the future sequence. However, these methods did not consider the interactions between agents and usually failed in crowded scenarios.

Recently, with the development of deep learning, convolutional neural networks (CNNs) and multilayer perceptrons (MLPs) have been used in trajectory prediction. For example, GRIP [12] uses a temporal graph to model the interactions and a temporal CNN to predict the future sequences. VectorNet [13] utilizes self-attention to aggregate interactions and the MLP to predict future sequences. Social-STGCN [14] and AST-GNN [15] capture interactions using a graph attention (GAT) and predict future sequences using a temporal CNN. Conv2D [16] is a novel 2D convolutional model specifically designed for pedestrian trajectory prediction, and has been found to outperform recurrent models and achieves new state-of-the-art results on the ETH [7] and TrajNet [17] datasets.

The direct prediction strategy has been proven simple and efficient. However, it uses only the past trajectory for interaction feature extraction and ignores the compatibility between the interaction coding and the predicted trajectories. Meanwhile, the previous methods use only one shared prediction model for all the future time steps. Large and small errors are mixed up in a single objective function, which is suboptimal in view of optimization. On the other hand, the proposed method considered the compatibility between the interaction coding and the predicted trajectories for correcting the crude prediction results. Large and small prediction errors are decoupled in our loss function by using independent prediction models for different time steps.

2.2. Recursive prediction methods

The recursive strategy involves the reuse of a one-step prediction model multiple times, in which the predictions of the previous time steps are used for the prediction of the following time step. RNN models including long short-term memory (LSTM) [18] and gated recurrent unit (GRU) [19] have been widely adopted in the recursive prediction methods. For example, S-LSTM [20] uses LSTM to extract the past trajectories, and then aggregates the interactions of different pedestrians in the hidden state using a social

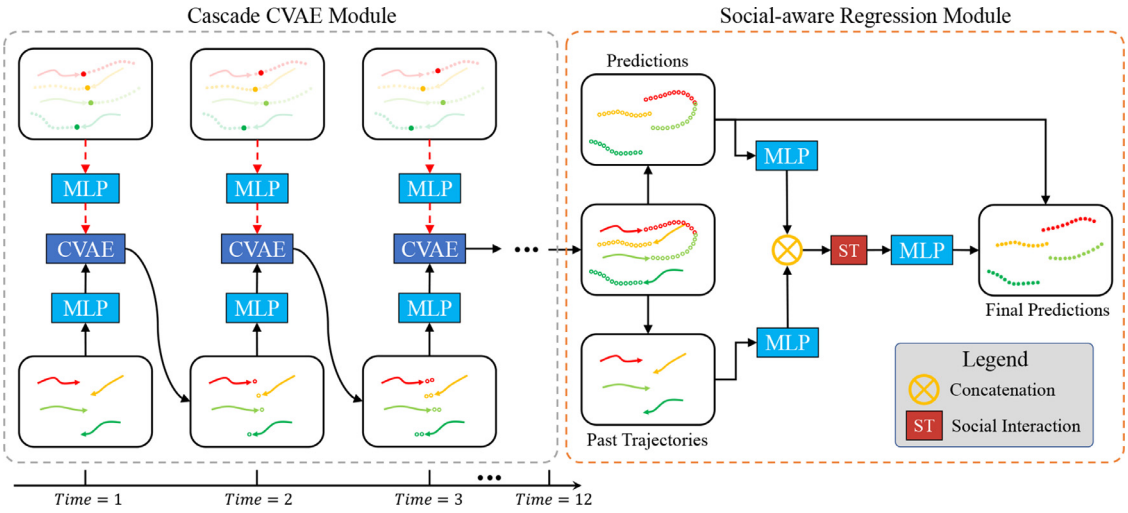

Fig. 2. The overview of the CSR model. First, the cascaded CVAE module estimates the future trajectory in a sequential manner. Each CVAE concatenates the past trajectory and the predicted location points so far as the input and predicts the location at the following time step. Then, the socially-aware regression module extracts interaction features from both the predicted trajectories and the past trajectories to refine the preliminary predictions.

pooling mechanism. CF-LSTM [21] improves LSTM by employing a cascaded feature that can simultaneously capture location and velocity information. S-GAN [22] extends S-LSTM by using GAN to generate multimodal predictions. SoPhie [23] and PIF [24] use a CNN to extract scene features and LSTM to extract motion features, and then adopt LSTM to predict scene-compliant trajectories. SR-LSTM [25] enables the utilization of the current intention of neighbors through a message passing framework. The social-affinity LSTM model [26] learns general human movement patterns and the Social Affinity Map using LSTM. The Social Affinity Map connects neighbors with a weight matrix that learns the social dependencies between correlated pedestrians. MI-LSTM [27] uses LSTM to extract the features of a cyclist and his/her neighbors, and adopts the focal attention mechanism to focus on more relevant features of the neighbors. Trajectron++ [28] introduces a modular graph-structured recurrent model that can integrate agent dynamics and heterogeneous data into trajectory prediction. EvolveGraph [29] proposes a dynamic mechanism to adaptively evolve the latent interaction graph to model explicit interaction among interactive agents.

The recursive prediction methods have shown promising performance in the literature. However, the importance and contribution of the motion information at different time steps are lost in the cell states and hidden states. Moreover, small prediction errors amplify significantly over time because of their loop structure for prediction. Comparably, the proposed CVAE module extracts the motion feature from an updated past trajectory consisting of the past trajectory and the predicted points so far in an adaptively weighted manner. Besides, the accumulated prediction errors are reduced by independently minimizing the prediction error at each time step.

2.3. Target-conditioned methods

The target-conditioned strategy first predicts the target locations and then generates the future trajectories conditioned on the targets. For example, PECNet [5] first predicts distant trajectory endpoints using a CVAE and then generates multimodal predictions conditioned on the endpoints. TNT [30] improves PECNet by introducing a scoring and selection stage that ranks and selects multimodal predictions using their likelihood scores. Y-net [31] further improves PECNet by iteratively predicting intermediate waypoints and trajectories. LOKI [32] proposes a novel large-scale dataset and a new model to tackle joint trajectory and intention prediction for traffic agents. MSN [33] first uses agents' end-point plannings and their interaction context as the basis for the behavior classification and then different style channels are used to give a series of predictions with significant style differences in parallel.

Recently, the target-conditioned methods achieve new state-of-the-art performances in trajectory prediction. They reduce the overall uncertainty by predicting a long-term target point and then predicting the future trajectory conditioned on the target point. The key to the excellent performance of target-conditioned methods is the accurate prediction of target points first because different target points will lead to the generation of completely different future trajectories. On the other hand, our method predicts the adjacent location point rather than the distant target point. The objective function is minimized sequentially at every time step to reduce the accumulated uncertainty.

PECNet is a representative target-conditioned prediction method. The proposed CSR reuses some same network components as PECNet, such as the CVAE network and the non-local pooling layer. Even though, the proposed method is inspired differently from PECNet in many aspects. Firstly, it uses unshared CVAEs to sequentially predict the location points instead of one shared CVAE as in PECNet to predict the whole future trajectories. By this strategy, the errors in the loss function are decoupled and further minimized at different times. Secondly, instead of using the original past trajectory as input, the proposed CSR uses an updated past trajectory which consists of the past trajectory and the predicted points so far as the input. In this way, CSR can explicitly extract motion features from longer past trajectories and thus decrease the prediction errors sequentially. For the non-local pooling layer, we extract interaction features from both the past trajectories and the predicted trajectories rather than only the past trajectories, which means the compatibility between the interaction coding and the predicted trajectories is concerned. Moreover, instead of using the interaction features to generate future trajectories, we use the interaction features to produce the offsets to correct the crude predictions.

3. Methodology

Pedestrian trajectory prediction is a task to predict the future trajectories of pedestrians based on their past trajectories. First, it

is assumed that videos of pedestrians walking are preprocessed to obtain the spatial coordinates. For example, $p_i^t = \left(x_i^t, y_i^t\right)$ represents the coordinate of the i-th pedestrian at the t-th time step. Then, the coordinates of each pedestrian in the scene are divided into the past and future trajectories according to two default values, namely the observation horizon τ and the prediction horizon δ. The definitions of the past trajectory $\mathbf{X}_i$ and the future trajectory $\mathbf{Y}_i$ are as follows:

$$\mathbf{X}_i = \left\{\left(x_i^t, y_i^t\right) | t = 1, 2, \ldots, \tau\right\}, \tag{1}$$

$$\mathbf{Y}_i = \left\{\left(x_i^t, y_i^t\right) | t = \tau+1, \tau+2, \ldots, \tau+\delta\right\}. \tag{2}$$

Finally, given the past trajectories $\{\mathbf{X}_i | i = 1, 2, \ldots, N\}$ of pedestrians in scene, the goal is to generate future predictions $\left\{\hat{\mathbf{Y}}_i | i = 1, 2, \ldots, N\right\}$ that are as close as the ground truth future trajectories $\{\mathbf{Y}_i | i = 1, 2, \ldots, N\}$, where N represents the number of pedestrians in a scene. The definition of predicted $\hat{\mathbf{Y}}_i$ is presented as follows:

$$\hat{\mathbf{Y}}_i = \left\{\left(\hat{x}_i^t, \hat{y}_i^t\right) | t = \tau+1, \tau+2, \ldots, \tau+\delta\right\}. \tag{3}$$

To generate practical and accurate predictions, a pedestrian trajectory prediction method that consists of a cascaded CVAE module and a socially-aware regression module is proposed, as shown in Fig. 2. Because the direct prediction of a long-range future trajectory $\mathbf{Y}_i$ has a high degree of uncertainty, the cascaded CVAE module is first designed to predict the future points in a sequential manner using the updated past trajectories as the input. The prediction error is reduced by sequentially optimizing the predicted location at the following time step. Meanwhile, the learning ability is improved because there is a CVAE model for the prediction at each time step. Then, considering that the CVAE is unable to model social interactions, a socially-aware regression module is proposed to further refine the predicted future trajectories using social interaction features from both the past trajectories and predicted trajectories.

3.1. Discussion

Recently, with the progress of deep learning, different strategies, including direct prediction strategy (Section 2.1), recursive prediction strategy (Section 2.2), and target-conditioned strategy (Section 2.3), have been proposed in trajectory prediction community. Commonly, in the objective function of these methods, the prediction errors of all time steps are minimized simultaneously using one shared prediction model, which can be roughly written as follows (in the mean square manner for example):

$$\sum_{t=\tau+1}^{\tau+\delta} \left\| p^t - \hat{p}^t \right\|_2^2, \tag{4}$$

where p^t and $\hat{p}^t$ respectively denote the ground truth and predicted location at time step t, τ and δ respectively are the observation horizon and the prediction horizon. Let $l_t = \left\| p^t - \hat{p}^t \right\|_2$, we get an arithmetic mean-geometric mean (AM-GM) inequality such that:

$$\sum_{t=\tau+1}^{\tau+\delta} l_t^2 \geq \delta \left(\prod_{\tau+1}^{\tau+\delta} l_t \right)^{\frac{1}{\delta}} \tag{5}$$

The equality holds if and only if $\{l_t\}_{t=\tau+1}^{\tau+\delta}$ are equal.

The inequality (5) reveals that the objective function (4) achieves the minimization when all errors are of the same value. However, it is impossible because the trajectory prediction model is shared for all time steps and $l_t \leq l_{t+1}$ for $\forall t$ (the prediction error increases over time as visualized in Figure 1). As a result, the objective function exerts different weights on l_t of different time steps, *i.e.*, the weights for larger l_t at the last time steps are relatively heavier than the weights for smaller l_t at the beginning time steps. Therefore, the objective function (4) of a shared prediction model is suboptimal in view of reducing the accumulated errors over time. Inspired by this, a cascaded CVAE module is proposed to adopt a more optimal solution, as presented in the next subsection.

3.2. Cascaded CVAE module

The prediction of future trajectories is inherently ambiguous and has a high degree of uncertainty. And, it is a suboptimal solution that a shared prediction model is used to learn a deterministic function f that directly maps the past trajectories $\{\mathbf{X}_i | i = 1, 2, \ldots, N\}$ to the future trajectories $\{\mathbf{Y}_i | i = 1, 2, \ldots, N\}$, as described in the fourth paragraph of Introduction. Therefore, a cascaded CVAE module, as illustrated in Fig. 2 and dummyTXdummy-Fig. 4b, is proposed to use unshared CVAEs to decompose the prediction of future trajectory into the prediction of the future locations at different time steps. In this way, the errors in the objective function (4) is decoupled and minimized independently using unshared prediction models at different time steps, which is a more optimal solution in view of reducing the accumulated errors over time.

The detailed structure of each cascaded CVAE is illustrated in Fig. 3, which produces the distribution of future location conditioned on the updated past trajectory by introducing a stochastic latent variable. The training of each CVAE requires two inputs, namely the updated past trajectory $\widetilde{\mathbf{X}}_i^t$ and the ground truth coordinates $p_i^t = \left(x_i^t, y_i^t\right)$ of pedestrian i at future time step t. The updated past trajectory consists of the past trajectory and the predicted future points by time step t using the concatenation operation, such that:

$$\widetilde{\mathbf{X}}_i^t = Concat\left(\mathbf{x}_i, \left\{\left(\hat{x}_i^s, \hat{y}_i^s\right) | s = \tau+1, \tau+2, \ldots, t-1\right\}\right). \tag{6}$$

In order to avoid the possible bias between training and inference, the predicted future points are used to build the updated past trajectories in both the training and testing stages. The updated past trajectory $\widetilde{\mathbf{X}}_i^t$ and the ground truth future point p_i^t are first encoded using two MLP encoders E_{upast} and E_{point}, as follows:

$$\mathbf{f}_{upast} = E_{upast}\left(\widetilde{\mathbf{X}}_i^t\right), \tag{7}$$

$$\mathbf{f}_{point} = E_{point}\left(p_i^t\right). \tag{8}$$

Then, the two feature representations $\mathbf{f}_{upast}$ and $\mathbf{f}_{point}$ are concatenated together and fed to the MLP encoder E_{latent} to generate parameters $\left(\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t\right)$ of the latent distribution, such that:

$$\left(\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t\right) = E_{latent}\left(Concat\left(\mathbf{f}_{upast}, \mathbf{f}_{point}\right)\right). \tag{9}$$

Finally, the latent variable $\mathbf{z}_i^t$ is randomly sampled from distribution $\mathcal{N}\left(\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t\right)$, and is concatenated with the feature $\mathbf{f}_{upast}$ and yield prediction $\hat{p}_i^t = \left(\hat{x}_i^t, \hat{y}_i^t\right)$ using the MLP decoder D_{latent}, as follows:

$$\hat{p}_i^t = D_{latent}\left(Concat\left(\mathbf{z}_i^t, \mathbf{f}_{upast}\right)\right). \tag{10}$$

In the inference stage, because the ground truth point p_i^t is unavailable, the latent variable $\mathbf{z}_i^t$ is randomly sampled from a prior distribution $\mathcal{N}(0, \mathbf{I})$, and concatenated with the feature $\mathbf{f}_{upast}$, and the prediction $\hat{p}_i^t$ is then generated using the trained decoder D_{latent}.

For each pedestrian i, there are δ future time steps. Thus, δ CVAEs with the same structure are sequentially operated to generate the predictions of δ future points $\hat{p}_i^{\tau+1}, \hat{p}_i^{\tau+2}, \ldots, \hat{p}_i^{\tau+\delta}$. It

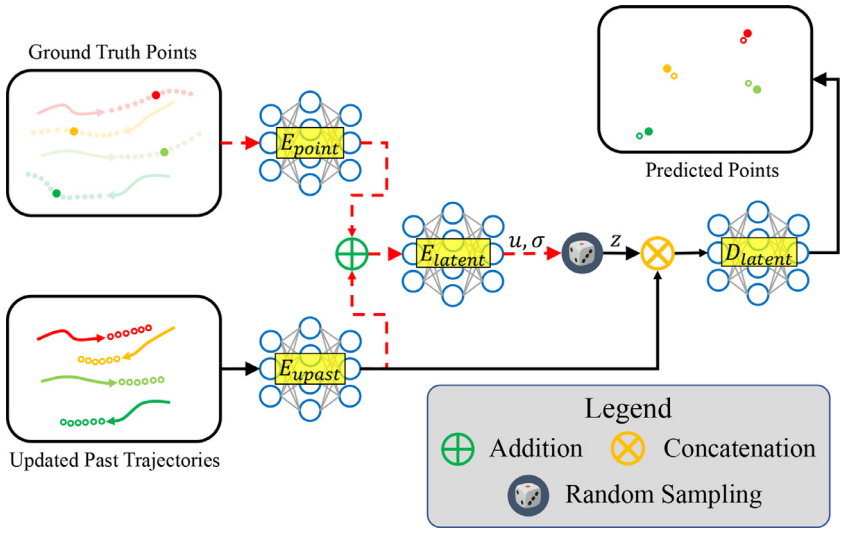

Fig. 3. The illustration of the cascaded CVAE unit. The CVAE uses the updated past trajectory and ground truth future point to train a model for multimodal future point prediction. The dots and circles respectively represent the predicted future points and the ground truth future points. Red dotted lines denote the layers utilized only during training.

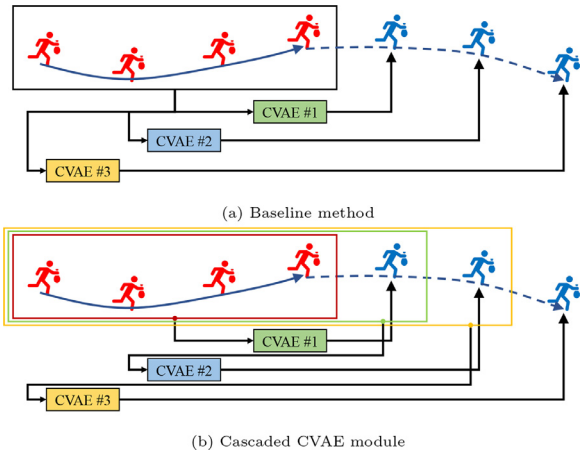

Fig. 4. The illustration of the baseline method and the cascaded CVAE module. The baseline method uses the original past trajectories to simultaneously predict δ future points. The cascaded CVAE module predicts δ future points using updated past trajectories in a sequential manner.

should be noted that model parameters for the CVAE, including E_{upast}, E_{point}, E_{latent} and D_{latent}, are not shared because the length of the updated past trajectory of each CVAE is different, and the CVAEs are used to predict future points at different future time steps.

3.3. Socially-aware regression module

The CVAE cannot model interactions among different pedestrians; thus, the future trajectories predicted by the cascaded CVAE module may be socially unreasonable and inaccurate. Most previous works extract interaction features from past trajectories. However, the interaction features extracted from the past trajectories can only represent the social interactions that happened in the future but cannot examine the social reasonability in the future. Instead of extracting the interaction features from only the past trajectories, a socially-aware regression module is proposed to extract the interaction features from both the past and predicted future trajectories. Besides, different from extracting the interaction features to implicitly assist in future trajectory prediction, the proposed module explicitly generates offsets to refine the predicted future trajectories.

The detailed structure of the socially-aware regression module is illustrated in Fig. 2. To generate appropriate offsets, two MLP encoders E_{opast} and $E_{pfuture}$ are first used to extract features from the original past trajectory and the predicted future trajectory separately as follows:

$$\mathbf{f}^i_{opast} = E_{opast}(\mathbf{X}_i), \tag{11}$$

$$\mathbf{f}^i_{pfuture} = E_{pfuture}(\hat{\mathbf{Y}}_i). \tag{12}$$

Then, the two feature representations $\mathbf{f}^i_{opast}$ and $\mathbf{f}^i_{pfuture}$ are concatenated to form a global motion feature of the i-th pedestrian:

$$\mathbf{f}^i_{global} = Concat(\mathbf{f}^i_{opast}, \mathbf{f}^i_{pfuture}). \tag{13}$$

The global features of all pedestrians in the same scene are passed into a social interaction layer SE to produce interaction features, such that:

$$[\mathbf{f}^1_{inter}, \mathbf{f}^2_{inter}, \ldots, \mathbf{f}^B_{inter}] = SE([\mathbf{f}^1_{global}, \mathbf{f}^2_{global}, \ldots, \mathbf{f}^B_{global}]), \tag{14}$$

where B is the number of pedestrians in the current scene and $\mathbf{f}^i_{inter}$ is extracted interaction feature of the i-th pedestrian. Here, the social non-local pooling layer proposed by [5] is adopted as the SE layer. Note that if there is only one pedestrian in a scene, the interaction feature $\mathbf{f}^i_{inter}$ of that pedestrian will remain the same as its global motion feature $\mathbf{f}^i_{global}$. Finally, the MLP decoder $D_{offsets}$ takes the extracted interaction feature $\mathbf{f}^i_{inter}$ to estimate the offsets of the predicted future trajectories:

$$\Delta\hat{\mathbf{Y}}_i = D_{offsets}(\mathbf{f}_{st}), \tag{15}$$

where $\Delta\hat{\mathbf{Y}}_i = \left\{\left(\Delta\hat{x}^t_i, \Delta\hat{y}^t_i\right) | \forall t \in \{\tau+1, \tau+2, \ldots, \tau+\delta\}\right\}$. Using the estimated offsets, the prediction is refined as $\hat{\mathbf{Y}}_i \doteq \hat{\mathbf{Y}}_i + \Delta\hat{\mathbf{Y}}_i$.

3.4. Training the model

To train the δ CVAEs in the cascaded CVAE module, two loss terms, namely the average point loss $\mathcal{L}_{AP}$ and the Kullback-Leibler (KL) divergence loss $\mathcal{L}_{KLD}$, are used. The definitions of $\mathcal{L}_{AP}$ and $\mathcal{L}_{KLD}$ are as follows:

$$\mathcal{L}_{AP} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=\tau+1}^{\tau+\delta}\left\|p^t_i - \hat{p}^t_i\right\|^2, \tag{16}$$

$$\mathcal{L}_{KLD} = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=\tau+1}^{\tau+\delta} D_{KL}\left(\mathcal{N}\left(\boldsymbol{\mu}_i^t, \boldsymbol{\sigma}_i^t\right)||\mathcal{N}(0, \mathbf{I})\right). \tag{17}$$

The KL divergence loss is a regularization loss that measures the distance between the sampling distribution at the test stage and the sampling distribution of the latent variable learned at the training stage and is used to train the CVAEs. The average point loss measures the displacement error between the predicted future points and the ground truth future points and is used to train E_{upath}, E_{point}, E_{latent}, and D_{latent}. In addition, a regression loss $\mathcal{L}_R$ is used to train the socially-aware regression module. The definition of $\mathcal{L}_R$ is:

$$\mathcal{L}_R = \frac{1}{N}\sum_{i=1}^{N}\sum_{t=\tau+1}^{\tau+\delta} \left\| p_i^t - \hat{p}_i^t - \Delta\hat{p}_i^t \right\|, \tag{18}$$

where $\Delta p_i^t = \left(\Delta\hat{x}_i^t, \Delta\hat{y}_i^t\right)$. The regression loss measures the difference between the predicted offsets and the real offsets, and is used to train E_{opast}, $E_{pfuture}$, and $D_{offsets}$. Finally, the total loss $\mathcal{L}_T$ is defined as follows:

$$\mathcal{L}_T = \mathcal{L}_{KLD} + \mathcal{L}_{AP} + \mathcal{L}_R. \tag{19}$$

This multi-task loss is used to train the entire model in an end-to-end manner.

4. Experiments

4.1. Datasets

Two pedestrian trajectory prediction datasets, namely the Stanford Drone Dataset (SDD) and the ETH/UCY dataset, were used to validate the proposed method. The trajectories in both datasets were sampled at a frame rate of 2.5Hz; the former 3.2 seconds of the trajectories (eight frames) are used as the inputs, and the latter 4.8 seconds (twelve frames) are used as the future trajectories to be predicted.

The SDD [6] is an aerial-view pedestrian trajectory prediction dataset captured by a drone. The dataset contains eight scenes in a college campus. There are total 11,000 pedestrians with 185,000 interactions between agents and 40,000 interactions between agents and scenes [6]. The standard method for the division of the training and test sets used in [22,23,34] was adopted in the experiments.

The ETH/UCY [7,8] is a dataset group that contains two datasets, *i.e.* ETH and UCY datasets. The ETH and UCY datasets contain five scenes, namely ETH, HOTEL, ZARA1, ZARA2, and UNIV. In these scenes, there are a total of 1536 unique pedestrians and thousands of nonlinear trajectories. To ensure fair comparisons, the same leave-one-out strategy adopted in previous studies [20,22,23,35] was used to train and evaluate the proposed model.

4.2. Implementation details

The proposed CSR is implemented in PyTorch [36] installed on a desktop computer that runs the Ubuntu 16.04 operating system and contains an NVIDIA 1080TI GPU. The Adam algorithm [37] with a learning rate of $3e^{-4}$ is used as the optimizer to train the CSR model. In each experiment, the model is trained for 600 epochs with a batch size of 512.

Network Configuration: The sub-networks of the two proposed modules consists of MLP and ReLU activation functions. The detailed network architectures of these sub-networks are presented in Table 1. The network configuration for both of the datasets is the same.

Table 1
The detailed architectures of all sub-networks.

Name	Network Architecture
E_{upast}	$2t \times 512 \times 256 \times 16 (\tau \leq t \leq \tau+\delta-1)$
E_{point}	$2 \times 8 \times 16 \times 16$
E_{latent}	$32 \times 8 \times 50 \times 32$
D_{latent}	$32 \times 1024 \times 512 \times 1024 \times 2$
E_{opast}	$16 \times 512 \times 256 \times 16$
$E_{pfuture}$	$24 \times 512 \times 256 \times 16$
$D_{offsets}$	$32 \times 1024 \times 512 \times 1024 \times 24$

Evaluation Metrics: Average displacement error (ADE) and final displacement error (FDE) are two commonly used evaluation metrics in trajectory prediction. The definitions of ADE and FDE are presented as follows:

$$ADE = \frac{\sum_{t=\tau+1}^{\tau+\delta}\left\| p_i^t - \hat{p}_i^t \right\|}{\delta}, \tag{20}$$

$$FDE = \left\| p_i^{\tau+\delta} - \hat{p}_i^{\tau+\delta} \right\|. \tag{21}$$

ADE measures the mean Euclidean distance between the predicted future points and the ground truth future points, and FDE measures the Euclidean distance between the predicted final points and the ground truth final points. In this paper, minimum ADE (ADE_k) and minimum FDE (FDE_k) over k randomly-sampled trajectories are employed to evaluate our method.

The proposed method is compared with the following state-of-the-art baselines:

- S-LSTM [20] Each pedestrian is modeled using an LSTM, and the interactions are extracted by pooling the hidden states among neighbors.
- DESIRE [38]. An inverse optimal control (IOC)-based trajectory planning method is used to refine the predicted trajectories.
- S-GAN [22]. This method improves S-LSTM by introducing a GAN to generate multimodal prediction results.
- SoPhie [23]. This method improves GAN-based methods by applying attention to model social relationships and physical constraints.
- PIF [24]. Visual features about human behavioral information and interaction with their surroundings are extracted for trajectory prediction.
- SR-LSTM [25]. This paper utilizes the current intention of neighbors, and jointly and iteratively refines the current states of all pedestrians.
- P2TIRL [34]. A maximum entropy inverse reinforcement learning strategy is introduced to learn a grid-based trajectory prediction method.
- SimAug [39]. This paper utilizes multi-view 3D simulation data to learn robust representations for trajectory prediction.
- CF-VAE [40]. A conditional normalizing flow-based prior is proposed to improve the VAE for the generation of effective predictions.
- CGNS [41]. Conditional variational divergence minimization and conditional latent space learning are used to predict the future trajectory.
- CF-LSTM [21]. A feature-cascaded method that can simultaneously capture location and velocity information is proposed to improve LSTM.
- Trajectron++ [28]. This method presents a modular graph-structured recurrent model that can integrate agent dynamics and heterogeneous data.
- PECNet [5]. This method first predicts distant trajectory endpoints and then generates the future predictions conditioned on endpoints.

Table 2
The comparison of the proposed CSR model with several baseline methods and previous state-of-the-art methods (labeled by ⋆) on the SDD dataset. (a) Results of the best of 20 samples. (b) Results of the best of 5 samples. The best results are marked in bold.

Methods	S-GAN	CF-VAE	P2TIRL	SimAug	PECNet	Y-net⋆	CSR
ADE_{20}	27.23	12.60	12.58	10.27	9.96	7.85	**4.87**
FDE_{20}	41.44	22.30	22.07	19.71	15.88	11.85	**6.32**

(a) Best of 20 samples

Methods	DESIRE	TNT	PECNet	Y-net⋆	CSR
ADE_5	19.25	12.79	12.23	11.49	**8.38**
FDE_5	34.05	29.58	21.16	20.23	**13.43**

(b) Best of 5 samples

Table 3
The comparison of the proposed CSR model with several baseline methods and previous state-of-the-art methods (labeled by ⋆) on the ETH/UCY dataset. Results ADE_{20}/ FDE_{20} are tested using best of 20 samples. The best results are marked in bold.

Scenes	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
S-LSTM	0.70/1.40	0.37/0.73	0.60/1.32	0.49/1.15	0.39/0.89	0.51/1.10
S-GAN	0.81/1.52	0.72/1.61	0.60/1.26	0.34/0.69	0.42/0.84	0.58/1.18
Sophie	0.70/1.43	0.76/1.67	0.54/1.24	0.30/0.63	0.38/0.78	0.54/1.15
CGNS	0.62/1.40	0.70/0.93	0.48/1.22	0.32/0.59	0.35/0.71	0.49/0.97
PECNet	0.54/0.87	0.18/0.24	0.35/0.60	0.22/0.39	0.17/0.30	0.29/0.48
Trajectron+	0.39/0.83	0.12/0.21	**0.20**/0.44	0.15/0.33	0.11/0.25	0.19/0.41
Y-net⋆	**0.28**/**0.33**	0.10/0.14	0.24/0.41	0.17/0.27	0.13/0.22	0.18/0.27
CSR	**0.28** /0.53	**0.07**/**0.08**	0.24/**0.35**	**0.07**/**0.09**	**0.05**/**0.09**	**0.14**/**0.23**

- TNT [30]. This method improves PECNet using a scoring and selection stage that ranks and selects multimodal predictions with likelihood scores.
- Y-net [31]. This method improves PECNet and TNT by iteratively predicting intermediate waypoints and trajectories.

4.3. Quantitative analysis

The proposed CSR model was compared with the aforementioned baseline methods on the SDD and ETH/UCY datasets via the ADE and FDE metrics.

4.3.1. Comparison with VAE-based methods

The VAE is widely used to generate the latent variables of trajectory distributions, and many VAE-based prediction methods have been proposed. Unlike these methods, the proposed CSR method uses multiple CVAEs to predict the future points, in a cascaded manner. Several VAE-based methods, including DESIRE, CF-VAE, and CGNS, were compared in this study. As reported in Table 2, on the SDD, CSR respectively outperformed CF-VAE and DESIRE by 61.3%/71.7% and 56.5%/60.6% in terms of the ADE/FDE metrics. Moreover, as reported in Table 3, CSR outperformed CGNS by 71.4%/76.3% on the ETH/UCY. This comparison validates that the proposed CSR, which utilizes CVAEs in a cascaded manner, significantly outperforms the previous VAE-based methods.

4.3.2. Comparison with RNN-based methods

The proposed CSR predicts the future trajectory in a sequential manner, which is similar to RNN-based methods. However, RNN-based methods do not explicitly encode the complete predicted points as the past trajectories to refine the prediction result. To validate the updated past trajectory used in the proposed model, CSR was compared with several RNN-based methods, including DESIRE, CF-VAE, CGNS, and Trajectron++. As shown in Table 2, on the SDD, the proposed CSR exhibited respective improvements over DESIRE and CF-VAE of 56.5%/60.6% and 61.3%/71.7% in terms of the ADE/FDE metrics. Moreover, as shown in Table 3, the proposed CSR exhibited respective improvements over CGNS and Trajectron++ of 71.4%/76.3% and 26.3%/52.1% on the ETH/UCY dataset. This comparison verifies that the proposed method, which explicitly encodes the whole predicted trajectories as the past trajectories, achieves superior performance as compared to RNN-based methods.

4.3.3. Comparison with target-conditioned methods

Target-conditioned trajectory prediction methods have recently become popular. These methods first predict the target point and then generate future predictions conditioned on the predicted target. The performance of this type of method is easily affected by the predicted target; however, target prediction is inherently a difficult problem. The proposed CSR was compared with three target-conditioned methods, namely PECNet, TNT, and Y-net. As presented in Table 2, CSR exhibited respective ADE/FDE improvements over TNT, PECNet, and Y-net (best of 5 samples) of 34.5%/54.6%, 31.5%/36.5%, and 27.1%/33.6% on the SDD. Moreover, as illustrated in Table 3, the proposed CSR respectively outperformed PECNet and Y-net by 51.7%/52.1% and 22.2%/14.8% on the ETH/UCY dataset. This comparison demonstrates that the proposed CSR performs better than target-conditioned trajectory prediction methods.

4.3.4. Comparison with state-of-the-art methods

Tables 2 and 3 demonstrate that Y-net achieved the best performance among all the selected baselines on the SDD and ETH/UCY. Compared to Y-net, the proposed CSR exhibited improvements in the ADE_{20}/FDE_{20} metrics of 38.0%/46.7% on the SDD and 22.2%/14.8% on the ETH/UCY dataset, thereby achieving new state-of-the-art performance.

4.3.5. Inference speed comparisons

The comparisons of the inference speed between the proposed CSR and the publicly available models are listed in Table 4. As can be seen, PECNet shows the fastest inference speed with 0.092 seconds per inference step. On the other hand, the inference speed of CSR is 0.185 seconds per inference step, which is 2 $\times$ slower than PECNet. This experiment validates that the cascaded CVAEs of CSR indeed increase the computation cost, but does not slow the inference speed too much even when it is compared with the fastest prediction model, PECNet. This result can be attributed to two reasons. First, trajectory prediction models do not use dense inputs, such as images, and have a low requirement for the computing power of GPU. Second, the unshared CVAEs which consist of four fully connected layers are lightweight modules.

Table 4
Comparisons with several baseline methods on inference speed on the ETH/UCY dataset. All models are benchmarked using an Nvidia GTX 1080 Ti GPU. The text in blue shows how many times our model is slower than others. The best result is marked in bold.

Method	Inference time
S-LSTM	1.179 (x0.16)
SR-LSTM	0.158 (x1.17)
S-GAN	0.097 (x1.91)
PIF	0.115 (x1.61)
Trajectron+	0.102 (x1.81)
PECNet	**0.092** (x2.01)
CSR	0.185

4.4. Ablation study

This experiment is designed to validate the effectiveness of the proposed cascaded CVAE module and socially-aware regression module. To validate these modules, a baseline method that consists of δ CVAEs is introduced for comparison. Note that CVAEs in the baseline method has the same structure as CVAE used in PECNet. Different from CVAE used in PECNet which uses past trajectory as input to predict endpoint, δ CVAEs in the baseline method use original past trajectory as input to respectively predict δ trajectory points at different future time steps, as illustrated in Fig. 4a.

4.4.1. Cascaded CVAE module

The proposed cascaded CVAE module predicts future trajectory in a cascaded manner, in which each CVAE sequentially predicts a future point using an updated past trajectory that consists of the original past trajectory and the predicted future points before the current time step. To prove the effectiveness of the cascaded CVAE module, it is compared with a baseline method in which δ CVAEs that use the original past trajectory as the input are respectively used to predict δ future points. As reported in Tables 5 and 6, the cascaded CVAE module outperforms the baseline by 37.0%/54.7% on the ETH/UCY and 47.1%/49.9% on the SDD, in terms of the ADE_{20}/FDE_{20} metrics. This demonstrates that the cascaded CVAE module can extract useful information from the predicted future points, and thus improves the prediction performance of the future trajectory.

4.4.2. Socially-aware regression module

The future trajectory predicted by the cascaded CVAE module may be impractical because the CVAE cannot model interactions among pedestrians. Thus, the socially-aware regression module was introduced to refine the final predictions via social interaction. To prove its effectiveness, the model is respectively trained with and without it. As presented in Tables 5 and 6, the socially-aware regression module can further boost the performance of the cascaded CVAE module by 17.6%/4.2% on the ETH/UCY and 20.4%/19.7% on the SDD, in terms of the ADE_{20}/FDE_{20} metrics, which indicates that this module can extract interaction features from the predicted trajectories to refine the final predictions. Besides, the predictions of the model trained with or without the socially-aware regression module are visualized in Fig. 5. The left image of each scenario visualizes the predicted distributions without using the socially-aware regression module. While the right image of each scenario visualizes the predicted distributions using the socially-aware regression module. Figs. 5a, 5b and 5c reveal that, compared to the less coordinated trajectories predicted by the cascade CVAE module, more socially reasonable trajectories are refined by the socially-aware regression module. Fig. 5d demonstrates that the collision problem in trajectory prediction also can be improved by the socially-aware regression module.

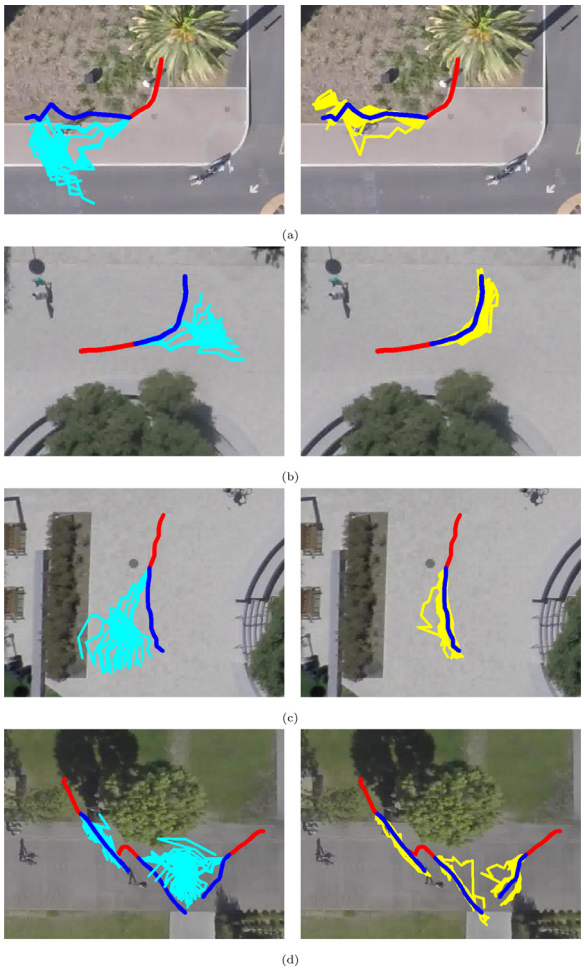

Fig. 5. Visualization of the trajectories predicted by the cascaded CVAE module with or without the socially-aware regression module on four different scenarios. The red and blue lines respectively represent the ground truth of past and future trajectories. The cyan lines in the left image of each scenario denotes predicted distributions without using the socially-aware regression module. While the yellow lines in the right image of each scenario denotes predicted distributions using the socially-aware regression module. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.5. Qualitative analysis

Qualitative results of CSR and PECNet on the SDD are visualized in Fig. 6. Figs. 6a and 6b present the predictions of linear trajectories. The two methods can both produce trajectories that are very close to the ground truth. It is expected because the prediction of linear trajectories is easy. Fig. 6c and 6d show predictions of crooked trajectories. In this case, the predictions of CSR are more accurate than that of the PECNet. These results demonstrate that the proposed CSR has better performance in predicting nonlinear trajectories.

Figs. 6e and 6f respectively visualize two easy interaction samples, *i.e.* turning together and going straight together. Both PECNet and CSR can generate trajectories that are close to the ground truth because the interactions between pedestrians are simple. Fig. 6g and 6h visualize predictions of two complex interactions, crossing and collision avoidance. The predictions of CSR are socially compliant and more reasonable than the results of PECNet. These results

Table 5
Ablation study of different modules of the proposed method on the ETH/UCY dataset. *BL* represents the introduced baseline method. The detailed structure of BL is introduced in the first paragraph of Section 4.4. *CCM* denotes the proposed cascaded CVAE module. *SRM* denotes the proposed socially-aware regression module. The best results are marked in bold.

Modules			Performance					
BL	*CCM*	*SRM*	ETH	HOTEL	UNIV	ZARA1	ZARA2	AVG
✓			0.48/0.94	0.19/0.35	0.30/0.61	0.22/0.42	0.16/0.32	0.27/0.53
	✓		0.33/0.57	0.08/0.10	0.28/0.36	**0.09**/0.10	0.07/0.09	0.17/0.24
	✓	✓	**0.28**/**0.53**	**0.07**/**0.08**	**0.24**/**0.35**	**0.07**/**0.09**	**0.05**/**0.09**	**0.14**/**0.23**

Table 6
Ablation study of different modules of the proposed method on the SDD dataset. *BL* represents the introduced baseline method. The detailed structure of BL is introduced in the first paragraph of Section 4.4. *CCM* denotes the proposed cascaded CVAE module. *SRM* denotes the proposed socially-aware regression module. The best results are marked in bold.

Modules			Performance
BL	*CCM*	*SRM*	
✓			11.56/15.71
	✓		6.12/7.87
	✓	✓	**4.87**/**6.32**

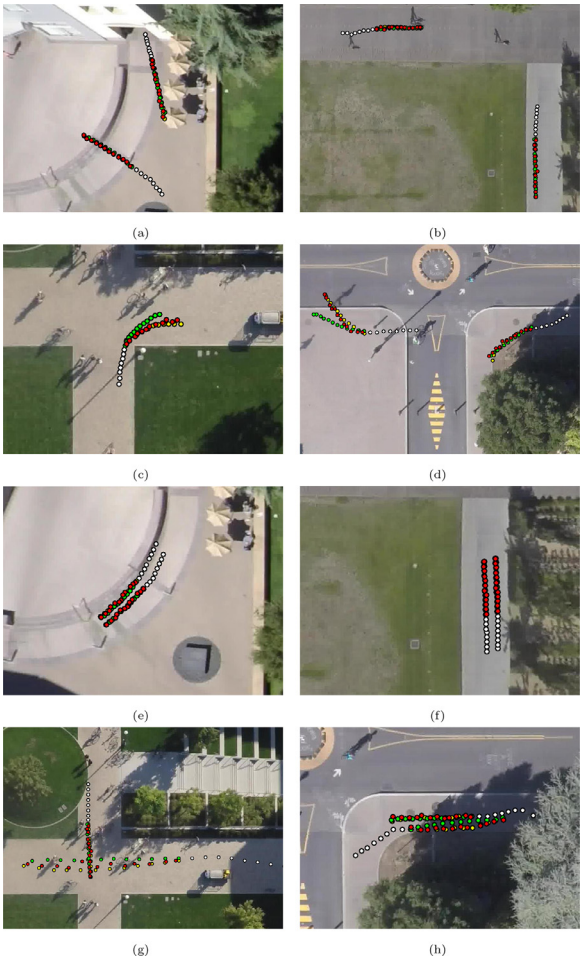

Fig. 6. Qualitative results of CSR in SDD dataset. The visualized trajectories are the best predictions sampled from 20 trials. The white dots, yellow dots, green dots, and red dots respectively represent the past trajectories, ground truth future trajectories, predictions of PECNet, and predictions of the proposed CSR. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

demonstrate that the proposed CSR can effectively model the complex interactions between pedestrians.

5. Conclusion

In this work, a novel cascaded CVAE with socially-aware regression (CSR) method is proposed for pedestrian trajectory prediction. The cascaded CVAE module predicts the future trajectories in a sequential manner using the updated past trajectory. And the socially-aware regression module extracts interaction features from both the past trajectories and the predicted trajectories to refine the preliminary predictions. Extensive experiments on the SDD and ETH/UCY datasets demonstrate the effectiveness of the proposed method. Furthermore, the experiment results also indicate that the proposed CSR outperforms other state-of-the-art methods by a large margin and achieves a new state-of-the-art performance.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work is supported partly by the National Natural Science Foundation of China (Nos. U21A20490, 61633009, 61973301, 61972020, 61772373, 51579053, and U1613213), partly by Beijing Nova Program (No. Z201100006820046), and partly by Meituan Open R&D Fund.

References

[1] J. Hong, B. Sapp, J. Philbin, Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.
[2] P. Raksincharoensak, T. Hasegawa, M. Nagai, Motion planning and control of autonomous driving intelligence system based on risk potential optimization framework, Int. J. Automot. Eng. 7 (AVEC14) (2016) 53–60.
[3] Y. Luo, P. Cai, A. Bera, D. Hsu, W.S. Lee, D. Manocha, Porca: modeling and planning for autonomous driving among many pedestrians, IEEE Rob. Autom. Lett. 3 (4) (2018) 3418–3425.
[4] H. Xue, D. Huynh, M. Reynolds, Location-velocity attention for pedestrian trajectory prediction, in: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), 2019, pp. 2038–2047, doi:10.1109/WACV.2019.00221.
[5] K. Mangalam, H. Girase, S. Agarwal, K.-H. Lee, E. Adeli, J. Malik, A. Gaidon, It is not the journey but the destination: Endpoint conditioned trajectory prediction, in: European Conference on Computer Vision, Springer, 2020, pp. 759–776.
[6] A. Robicquet, A. Sadeghian, A. Alahi, S. Savarese, Learning social etiquette: Human trajectory understanding in crowded scenes, in: European conference on computer vision, Springer, 2016, pp. 549–565.
[7] S. Pellegrini, A. Ess, K. Schindler, L. Van Gool, You'll never walk alone: Modeling social behavior for multi-target tracking, in: 2009 IEEE 12th International Conference on Computer Vision, IEEE, 2009, pp. 261–268.
[8] A. Lerner, Y. Chrysanthou, D. Lischinski, Crowds by example, in: Computer graphics forum, volume 26, Wiley Online Library, 2007, pp. 655–664.
[9] S. Lefèvre, C. Laugier, J. Ibañez Guzmán, Exploiting map information for driver intention estimation at road intersections, in: 2011 ieee intelligent vehicles symposium (iv), IEEE, 2011, pp. 583–588.

[10] C.E. Ramussen, C. Williams, Gaussian processes for machine learning (adaptive computation and machine learning), 2006, (????).
[11] R. Toledo-Moreo, M.A. Zamora-Izquierdo, Imm-based lane-change prediction in highways with low-cost gps/ins, IEEE Trans. Intell. Transp. Syst. 10 (1) (2009) 180–185.
[12] X. Li, X. Ying, M.C. Chuah, Grip: Graph-based interaction-aware trajectory prediction, in: 2019 IEEE Intelligent Transportation Systems Conference (ITSC), IEEE, 2019, pp. 3960–3966.
[13] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, C. Schmid, Vectornet: Encoding hd maps and agent dynamics from vectorized representation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11525–11533.
[14] A. Mohamed, K. Qian, M. Elhoseiny, C. Claudel, Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 14424–14432.
[15] H. Zhou, D. Ren, H. Xia, M. Fan, X. Yang, H. Huang, Ast-gnn: an attention-based spatio-temporal graph neural network for interaction-aware pedestrian trajectory prediction, Neurocomputing 445 (2021) 298–308.
[16] S. Zamboni, Z.T. Kefato, S. Girdzijauskas, C. Norén, L. Dal Col, Pedestrian trajectory prediction with convolutional neural networks, Pattern Recognit. 121 (2022) 108252, doi:10.1016/j.patcog.2021.108252. https://www.sciencedirect.com/science/article/pii/S0031320321004325.
[17] A. Sadeghian, V. Kosaraju, A. Gupta, S. Savarese, A. Alahi, Trajnet: towards a benchmark for human trajectory prediction, arXiv preprint (2018).
[18] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.
[19] J. Chung, C. Gulcehre, K. Cho, Y. Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, NIPS 2014 Workshop on Deep Learning, December 2014, 2014.
[20] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, 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.
[21] Y. Xu, J. Yang, S. Du, Cf-lstm: Cascaded feature-based long short-term networks for predicting pedestrian trajectory, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 2020, pp. 12541–12548.
[22] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, 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.
[23] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, S. Savarese, Sophie: An attentive gan for predicting paths compliant to social and physical constraints, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 1349–1358.
[24] J. Liang, L. Jiang, J.C. Niebles, A.G. Hauptmann, L. Fei-Fei, Peeking into the future: Predicting future person activities and locations in videos, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5725–5734.
[25] P. Zhang, J. Xue, P. Zhang, N. Zheng, W. Ouyang, Social-aware pedestrian trajectory prediction via states refinement LSTM, IEEE Trans. Pattern Anal. Mach. Intell. (2020), doi:10.1109/TPAMI.2020.3038217. 1–1
[26] Z. Pei, X. Qi, Y. Zhang, M. Ma, Y.-H. Yang, Human trajectory prediction in crowded scene using social-affinity long short-term memory, Pattern Recognit. 93 (2019) 273–282.
[27] Z. Huang, J. Wang, L. Pi, X. Song, L. Yang, Lstm based trajectory prediction model for cyclist utilizing multiple interactions with environment, Pattern Recognit. 112 (2021) 107800, doi:10.1016/j.patcog.2020.107800. https://www.sciencedirect.com/science/article/pii/S0031320320306038.
[28] T. Salzmann, B. Ivanovic, P. Chakravarty, M. Pavone, Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16, Springer, 2020, pp. 683–700.
[29] J. Li, F. Yang, M. Tomizuka, C. Choi, Evolvegraph: multi-agent trajectory prediction with dynamic relational reasoning, Adv. Neural Inf. Process. Syst. 33 (2020) 19783–19794.
[30] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, et al., Tnt: target-driven trajectory prediction, arXiv preprint arXiv:2008.08294 (2020).
[31] K. Mangalam, Y. An, H. Girase, J. Malik, From goals, waypoints & paths to long term human trajectory forecasting, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 15233–15242.
[32] H. Girase, H. Gang, S. Malla, J. Li, A. Kanehara, K. Mangalam, C. Choi, Loki: Long term and key intentions for trajectory prediction, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 9803–9812.
[33] C. Wong, B. Xia, Q. Peng, X. You, Msn: multi-style network for trajectory prediction, arXiv preprint arXiv:2107.00932 (2021).
[34] N. Deo, M.M. Trivedi, Trajectory forecasts in unknown environments conditioned on grid-based plans, arXiv preprint arXiv:2001.00735 (2020).
[35] V. Kosaraju, A. Sadeghian, R. Martín-Martín, I. Reid, H. Rezatofighi, S. Savarese, Social-biGAT: multimodal trajectory forecasting using bicycle-GAN and graph attention networks, Adv. Neural Inf. Process. Syst. 32 (2019) 137–146.
[36] 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, Adv. Neural Inf. Process. Syst. 32 (2019) 8026–8037.
[37] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, in: International Conference on Learning Representations (ICLR), 2015.
[38] N. Lee, W. Choi, P. Vernaza, C.B. Choy, P.H.S. Torr, M. Chandraker, Desire: Distant future prediction in dynamic scenes with interacting agents, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 336–345.
[39] J. Liang, L. Jiang, A. Hauptmann, Simaug: Learning robust representations from simulation for trajectory prediction, in: European Conference on Computer Vision, Springer, 2020, pp. 275–292.
[40] A. Bhattacharyya, M. Hanselmann, M. Fritz, B. Schiele, C.-N. Straehle, Conditional flow variational autoencoders for structured sequence prediction, 4th Workshop on Bayesian Deep Learning, bayesiandeeplearning. org, 2019.
[41] J. Li, H. Ma, M. Tomizuka, Conditional generative neural system for probabilistic trajectory prediction, in: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2019, pp. 6150–6156.

Hao Zhou received the B.S. and M.S. degrees from Harbin Engineering University, Harbin, China, in 2016 and 2019, respectively. He is currently pursuing an Ph.D. degree at National Key Laboratory of Science and Technology on Underwater Vehicle, Harbin Engineering University. His current research interests include object detection and trajectory prediction.

Dongchun Ren received the B.S. and Ph.D. degrees from Nankai University and Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2008 and 2014, respectively. He is currently a senior Engineer with Beijing Sankuai Online Technology Co.,Ltd. His current research interests include computer vision and autonomous driving.

Xu Yang received the B.S. and Ph.D. degrees from China Ocean University and Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2009 and 2014, respectively. He is currently an Associate Professor with Institute of Automation, Chinese Academy of Science. His interests include image processing and graph matching.

Mingyu Fan received the B.S. and Ph.D. degrees from Minzu University of China and Academy of Mathematics and System Science, Chinese Academy of Science, Beijing, China, in 2006 and 2011, respectively. He is currently a Professor with Wenzhou University. His current research interests include machine learning and autonomous driving.

Hai Huang received the B.S. and Ph.D. degrees from Harbin Institute of Technology, Harbin, China, in 2001 and 2008, respectively. He is currently a Professor with National Key Laboratory of Science and Technology of Underwater Vehicle, Harbin Engineering University. His current research interests include underwater vehicle and autonomous operation.