Reciprocal Learning Networks for Human Trajectory Prediction

Hao Sun, Zhiqun Zhao, and Zhihai He
University of Missouri
{hshq7,zzhv7,hezhi}@mail.missouri.edu

Abstract

We observe that the human trajectory is not only forward predictable, but also backward predictable. Both forward and backward trajectories follow the same social norms and obey the same physical constraints with the only difference in their time directions. Based on this unique property, we develop a new approach, called reciprocal learning, for human trajectory prediction. Two networks, forward and backward prediction networks, are tightly coupled, satisfying the reciprocal constraint, which allows them to be jointly learned. Based on this constraint, we borrow the concept of adversarial attacks of deep neural networks, which iteratively modifies the input of the network to match the given or forced network output, and develop a new method for network prediction, called reciprocal attack for matched prediction. It further improves the prediction accuracy. Our experimental results on benchmark datasets demonstrate that our new method outperforms the state-of-the-art methods for human trajectory prediction.

1. Introduction

Human motion trajectories and motion patterns are governed by human perception, behavioral reasoning, common sense rules, social conventions, and interactions with others and the surrounding environment. Human can effectively predict short-term body motion of others and respond accordingly. The ability for a machine to learn these rules and use them to understand and predict human motion in complex environments is highly valuable with a wide range of applications in social robots, intelligent systems, and smart environments [22, 24]. The central research question of human trajectory prediction is: *given observed motion trajectories of human, can we predict their future trajectories within a short period of time, for example, 5 seconds, in the future?*

Predicting human motion and modeling their common sense behaviors are a very challenging task [2]. An efficient algorithm for human trajectory prediction needs to accomplish the following tasks: (1) *obeying physical constraints of the environment.* To walk on a feasible terrain and avoid obstacles or other physical constraints, we need to analyze the local and global spatial information surrounding the person and pay attention to important elements in the environment. (2) *Anticipating movements of other persons or vehicles and their social behaviors.* Some trajectories are physically possible but socially unacceptable. Human motions are governed by social norms, such as yielding right-of-way or respecting personal space. (3) *Finding multiple feasible paths.* There are often multiple choices of motion trajectories to reach the destination. This uncertainty poses significant challenges for accurate human trajectory prediction.

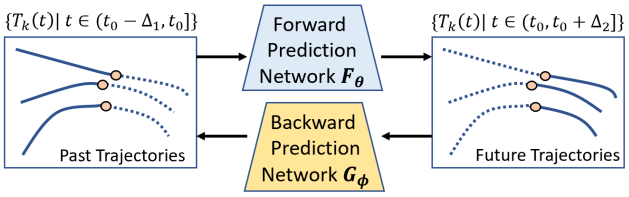

Figure 1. Illustration of our idea of reciprocal learning for human trajectory prediction.

Recently, a number of methods based on deep neural networks have been developed for human trajectory prediction [2, 20]. Earlier methods have been focused on learning dynamic patterns of moving agents (human and vehicles) [2] and modeling the semantics of the navigation environment [17]. Methods have been developed to model human-human interactions [14], understand social acceptability [4, 1, 20], and model the joint influence of all agents in the scene [12]. Efforts have also been taken to predict multiple feasible paths of human [1, 20, 35].

In this work, we propose to explore the unique characteristics of human trajectories and develop a new approach, called *reciprocal learning* for human trajectory prediction. As illustrated in Figure 1, we observe that the human trajectory is not only forward predictable, but also backward predictable. Imagine that the time is reversed and the person is traveling backwards. As discussed in the above, the for-

ward moving trajectories follow the social norm and obey the environmental constraints. So do the backward moving trajectories since the only difference between them is that the time is reversed. From the training data, we can train two different prediction networks, the forward prediction network $\mathbf{F}_\theta$ and the backward prediction network $\mathbf{G}_\phi$. These two networks are tightly coupled together, satisfying a reciprocal constraint. For example, using the forward network, we can predict the future trajectory $\mathbf{Y} = \mathbf{F}_\theta(\mathbf{X})$ from the observed or past trajectory $\mathbf{X}$. If the prediction $\mathbf{Y}$ is accurate, then $\mathbf{G}_\phi(\mathbf{Y})$ must be approximately equal to $\mathbf{X}$.

Based on this observation and the unique reciprocal constraint, we develop a new approach called *reciprocal network learning* for accurate and robust prediction of human trajectories. We introduce the reciprocal prediction loss and establish an iterative procedure for training these two tightly coupled networks. We borrow the concept of the adversarial attacks of deep neural networks which iteratively modifies the input of the network to match a given target or forced network output. We integrate the reciprocal constraint with the adversarial attack method to develop a new matched prediction method for human trajectory prediction. Our experimental results on benchmark datasets demonstrate that our new method outperforms the state-of-the-art methods for human trajectory prediction.

The rest of the paper is organized as follows. Section 2 reviews related work on human trajectory prediction. The proposed reciprocal network learning and matched prediction are presented in Section 3. Section 4 presents the experimental results, performance comparisons, and ablation studies. Section 5 summarizes our major contributions and concludes the paper.

2. Related Work

In this section, we review related work, including human-human models and human-scene models for human trajectory prediction, adversarial attacks, and cycle consistency.

(1) Human-human models for trajectory prediction. A number of methods have been developed in the literature to model human social interactions and behaviors in crowded scenes, such as people attempting to avoid walking into each other. Helbing and Molnar [14] introduced the Social Force Model to characterize social interactions among people in crowded scenes using coupled Langevin equations. In recent methods based on LSTM (Long Short Term Memory) [1], social pooling was introduced to share features and hidden representations between different agents. The key idea is to merge hidden states of nearby pedestrians to make each trajectory aware of its neighbourhood. [5] found out that groups of people moving coherently in one direction should be excluded from the above pooling mechanism. [12] used a Generative Adversarial Network (GAN) to discriminate between multiple feasible paths. This model is able to capture different movement styles but does not differentiate between structured and unstructured environments. [34] predicted human trajectories using a spatio-temporal graph to model both position evolution and interactions between pedestrians.

(2) Human-scene models for trajectory prediction. Another set of methods for human trajectory prediction has focused on learning the effects of physical environments. For example, human tend to walk along the sidewalk, around a tree or other physical obstacles. Sadeghian *et al.* [29] considered both traveled areas and semantic context to predict social and context-aware positions using a GAN (Generative Adversarial Network). [21] extracted multiple visual features, including each person's body keypoints and the scene semantic map to predict human behavior and model interaction with the surrounding environment. [4] has studied attractions towards static objects, such as artworks, which deflect straight paths in several scenarios such as museums. [2] proposed a Bayesian framework to predict unobserved paths from previously observed motions and to transfer learned motion patterns to new scenes. In [8], the dynamics and semantics for long-term trajectory predictions have been studied. Scene-LSTM [23] divided the static scene into grids and predicted pedestrian's location using LSTM. The CAR-Net method [30] integrated past observations with bird's eye view images and analyzed them using a two-levels attention mechanism.

(3) Adversarial attacks. As one of our major contributions, we explore adversarial attacks for network prediction based on reciprocal constraints. The goal of adversarial attacks is to add small noises on input examples to make them mis-classified by the network. One of the first successful methods to generate adversarial examples is the fast gradient sign method (FGSM) [11]. Kurakin *et al.* [18] proposed a variant of FGSM called I-FGSM which iteratively applies the FGSM update with a small step size. Note that both FGSM and I-FGSM aim to minimize the Chebyshev distance between the inputs and the generated adversarial examples. Optimization-based methods [32, 25, 7] have also been developed for generating adversarial samples. Our work borrows the idea from FGSM to perform adversarial attacks as a post-processing step on our predicted future trajectories to minimize the self-consistency loss, as explained in Section 3.5.

(4) Cycle consistency learning. Using transitivity as a way to regularize structured data has been studied. For example, in visual tracking, [16, 31] developed a forward-backward consistency constrain. In language processing, [6, 13, 33] studied human and machine translators to verify and improve translations based on back translation and reconciliation mechanisms. Cycle consistency has also been explored in motion analysis [37], 3D shape matching

[15], dense semantic alignment [40, 39], depth estimation [10, 36, 38], and image-to-image translation [3, 41]. CycleGAN [41] introduces a cycle consistence constraint for learning a mapping to translate an image from the source domain into the target domain. In this work, we explore the unique characteristics of human trajectories and develop the new approach of reciprocal learning. Our idea is related to the cycle consistency but is quite unique. We introduce the reciprocal loss and design two tightly coupled prediction networks, the forward and backward prediction networks, which are jointly learned based on the reciprocal constraint.

3. Reciprocal Networks for Human Trajectory Prediction

In this section, we present our reciprocal network learning method for human trajectory prediction.

3.1. Problem Formulation

We follow the standard formulation of trajectory forecasting problem in the literature [34, 21]. With observed trajectories of all moving agents in the scene, including persons and vehicles, the task is to predict the moving trajectories of all agents for the next period of time in the near future. Specifically, let $\mathbf{X} = [X_1, X_2, \cdots, X_N]$ be the trajectories of all human in the scene. Our task is to predict the future trajectories of all human $\hat{\mathbf{Y}} = [\hat{Y}_1, \hat{Y}_2, \cdots, \hat{Y}_N]$ simultaneously. The input trajectory of human n is given by $X_n = (x_n^t, y_n^t)$ for time steps $t = 1, 2, \cdots, T_o$. The ground truth of future trajectory is given by $Y_n = (x_n^t, y_n^t)$ for time step $t = T_o + 1, \cdots, T_p$.

3.2. Method Overview

As illustrated in Figure 1, in reciprocal learning, we are learning two coupling networks, the forward prediction network $\mathbf{F}_\theta$ which predicts the future trajectories $\mathbf{Y} = \mathbf{F}_\theta(\mathbf{X})$ from the past data $\mathbf{X}$, and the backward prediction network $\mathbf{G}_\phi$ which predicts the past trajectories $\mathbf{X} = \mathbf{G}_\phi(\mathbf{Y})$ from the future data $\mathbf{Y}$. It should be noted that, during training, both the past and future data are available. If both networks are well trained, then we should have following two reciprocal consistency constraints:

$$\mathbf{X} \approx \mathbf{G}_\phi(\mathbf{F}_\theta(\mathbf{X})), \quad (1)$$

$$\mathbf{Y} \approx \mathbf{F}_\theta(\mathbf{G}_\phi(\mathbf{Y})). \quad (2)$$

These two networks are able to help each other to improve the learning and prediction performance. Specifically, if the backward prediction network $\mathbf{G}_\phi$ is trained, we can use the reciprocal constraint (1) to double check the accuracy of the forward prediction network $\mathbf{F}_\theta$ and improve its performance during training. Likewise, if the forward prediction network $\mathbf{F}_\theta$ is trained, we can use (2) to improve the training performance of the backward prediction network $\mathbf{G}_\phi$. This results in a tightly coupled iterative learning and performance improvement process between these two prediction networks. Once the forward and backward networks are successfully trained using the reciprocal learning approach, we develop a new network inference method called *reciprocal attack for matched prediction*. It borrows the concept of adversarial attacks of deep neural networks where the input is iteratively modified such that the network output matches a given target [11].

Our proposed idea echoes some thoughts in CycleGAN [41] which presents an approach for learning a mapping to translate an image from a source domain to a target domain. They also learn an inverse mapping and introduce the cycle consistence constraint. Our approach is significantly different from this CycleGAN method. We design two tightly coupled prediction networks, the forward and backward prediction networks, which are jointly learned based on the reciprocal constraint. For the testing part, our approach introduces a new reciprocal attack method for matched prediction of human trajectory.

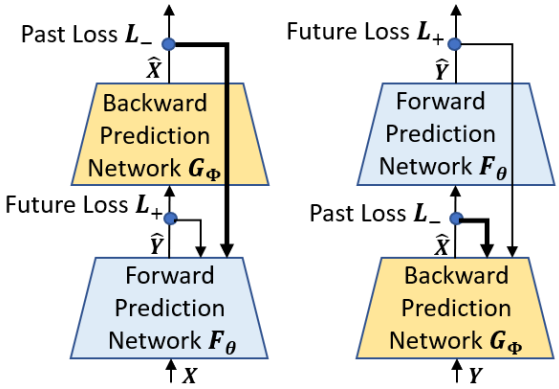

Figure 2. The training process of reciprocal learning.

3.3. Reciprocal Network Training

To successfully train the forward and backward prediction networks, we define two loss functions, J_- and J_+, to measure the prediction accuracy of the past and future trajectories. One reasonable choice will be the L_2 norm between the original trajectory and its prediction. These two loss functions will be updated alternatively and combined to guide the training of each of these two networks, as illustrated in Figure 2. For example, when training the forward prediction network $\mathbf{F}_\theta$, the loss function used in existing literature is the prediction error of the future trajectory L_+. In reciprocal training, we first pre-train the backward prediction network $\mathbf{G}_\phi$ using the training data with all trajectories reversed in time. We then use this network to map the prediction result of $\mathbf{F}_\theta$, $\hat{\mathbf{Y}} = \mathbf{F}_\theta(\mathbf{X})$, back to the past

trajectory, which is given by

$$\hat{\mathbf{X}} = \mathbf{G}_\phi(\hat{\mathbf{Y}}) = \mathbf{G}_\phi(\mathbf{F}_\theta(\mathbf{X})). \tag{3}$$

The past trajectory loss is then given by $L_- = ||\mathbf{X} - \hat{\mathbf{X}}||_2$. We refer to this loss as *reciprocal loss*. It will be combined with L_+ to form the loss function for the forward prediction network $\mathbf{F}_\theta$:

$$\begin{aligned} J_+[\theta] &= \lambda \cdot L_+ + (1-\lambda) \cdot L_- \\ &= \lambda \cdot ||\mathbf{Y} - \mathbf{F}_\theta(\mathbf{X})||_2 \\ &+ (1-\lambda) \cdot ||\mathbf{X} - \mathbf{G}_\phi(\mathbf{F}_\theta(\mathbf{X}))||_2. \end{aligned} \tag{4}$$

Similarly, we can derive the loss function for the backward prediction network $\mathbf{G}_\phi$:

$$\begin{aligned} J_-[\phi] &= \lambda \cdot L_- + (1-\lambda) \cdot L_+ \\ &= \lambda \cdot ||\mathbf{X} - \mathbf{G}_\phi(\mathbf{Y})||_2 \\ &+ (1-\lambda) \cdot ||\mathbf{Y} - \mathbf{F}_\theta(\mathbf{G}_\phi(\mathbf{Y}))||_2. \end{aligned} \tag{5}$$

In reciprocal training, we first pre-train the forward and backward prediction networks independently. Then, these two networks are jointly trained in an iterative manner based on the reciprocal constraint.

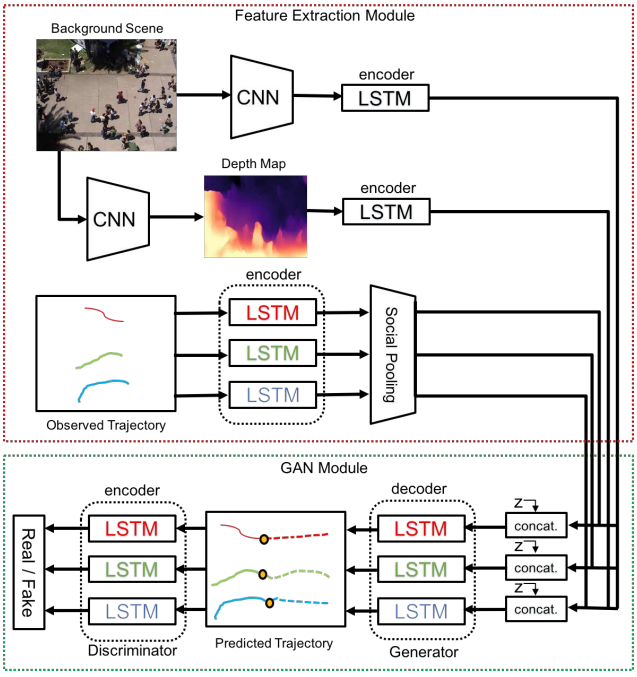

Figure 3. Our prediction network has two key components: (1) Feature Extraction Module and (2) LSTM-based GAN module.

3.4. Constructing the Forward and Backward Prediction Networks

Both the forward and backward networks share the same network structure. In the following, we use the forward prediction network $\mathbf{F}_\theta$ as an example to explain our network design. As illustrated in Figure 3, we adopt the existing Social-GAN in [12] as our baseline prediction network. Our model consists of two key components: (1) a feature extraction module and (2) an LSTM (Long Short Term Memory)-based GAN (Generative Adversarial Network) module.

3.4.1 Feature Extractor

Our feature extractor module has three major components to be explained in the following. Specifically, we first use the LSTM encoder to capture the temporal pattern and dependency within each trajectory of human n and encode them into a high-dimensional feature $\mathbf{F}_h^t(n)$. To capture the joint influence of all surrounding human movements on the prediction of the target human n, we borrow the idea from [12] to build a social pooling module which extracts the joint social feature $\mathbf{F}_s^t(n)$ of all human in the scene to encode the human-human interactions. The relative distance values between the target person and others are calculated. These distance vectors are concatenated with the hidden state in the LSTM network for each person and then embedded by an MLP and followed by a Max-Pooling function to form the joint feature. A maximum number of moving human in the scene is set, whose default value is 0 if the corresponding agent does not exist at the current time.

As recognized in [35, 29], the environmental context affects the decision of the human in planning its next step of movement. Features of the current scene can be incorporated into the reasoning process. Similar to prior work [29], we use VGGNet-19 [28] pre-trained on the ImageNet [28] to extract the visual feature of background scene I^t, which is then fed into an LSTM encoder to compute the hidden state tensor $\mathbf{F}_v^t$.

As a unique feature of our proposed method, we propose to also incorporate the 3D scene depth map into the reasoning process, which also improves the prediction accuracy of human trajectory. This is because the human motion occurs in the original 3D environment. Therefore, its natural behavior and motion patterns are better represented by its 3D trajectory, instead of the 2D image coordinates. For example, the trajectory of a person walking near the camera is much different from that of a person walking far away from the camera due to the camera perspective transform. To address this issue, we propose to infer a depth image from a single image using existing depth estimation method [9]. We use their pre-trained model to perform monocular depth estimation and obtain the depth map M_d^t of scene I^t, then use an LSTM to encode it into a depth feature $\mathbf{F}_d^t$.

3.4.2 LSTM-based GAN For Trajectory Prediction

Inspired by previous work [12, 29], in this paper we use an LSTM-based Generative Adversarial Network (GAN) module to generate human's future path as illustrated in Figure

3. The generator is constructed by a decoder LSTM. Similar to the conditional GAN [24], a white noise vector $\mathbf{Z}$ is sampled from a multivariate normal distribution. Then, a merge layer is used in our proposed network which concatenates all encoded features mentioned above with the noise vector $\mathbf{Z}$. We take this as the input to the LSTM decoder to generate the candidate future paths for each human. The discriminator is built with an LSTM encoder which takes the input as randomly chosen trajectory from either ground truth or predicted trajectories and classifies them as "real" or "fake". Generally speaking, the discriminator classifies the trajectories which are not accurate as "fake" and forces the generator to generate more realistic and feasible trajectories.

Within the framework of our reciprocal learning for human trajectory prediction, let $G^\theta : \mathbf{X} \to \mathbf{Y}$ and $G^\phi : \mathbf{Y} \to \mathbf{X}$ be the generators of the forward prediction network $\mathbf{F}_\theta$ and the backward prediction network $\mathbf{G}_\phi$, respectively. D^θ is the discriminator for $\mathbf{F}_\theta$. Its input $\mathbf{Y}'$ is randomly selected from either ground truth $\mathbf{Y}$ or the predicted future trajectory $\hat{\mathbf{Y}}$. Similarly, for D^ϕ is discriminator for $\mathbf{G}_\phi$. To train $\mathbf{F}_\theta$ and $\mathbf{G}_\phi$, we combine the adversarial loss with the forward prediction loss $J_+[\theta]$ and the backward prediction loss $J_-[\phi]$ in Eqs. (4) and (5) together to construct the overall loss function for $\mathbf{F}_\theta$ and $\mathbf{G}_\phi$, respectively:

$$\mathcal{L}_\theta = L^\theta_{GAN} + J_+[\theta], \quad \mathcal{L}_\phi = L^\phi_{GAN} + J_-[\phi], \tag{6}$$

where adversarial losses L^θ_{GAN} and L^ϕ_{GAN} are defined as:

$$\begin{aligned} L^\theta_{GAN} &= \min_G \max_D \mathbb{E}_{\mathbf{Y}' \sim p(\mathbf{Y}, \hat{\mathbf{Y}})}[\log D(\mathbf{Y}')] \\ &+ \mathbb{E}_{\mathbf{X} \sim p(\mathbf{X}), \mathbf{Z} \sim p(\mathbf{Z})}[\log(1 - D(G(\mathbf{X}, \mathbf{Z})))], \end{aligned} \tag{7}$$

$$\begin{aligned} L^\phi_{GAN} &= \min_G \max_D \mathbb{E}_{\mathbf{X}' \sim p(\mathbf{X}, \hat{\mathbf{X}})}[\log D(\mathbf{X}')] \\ &+ \mathbb{E}_{\mathbf{Y} \sim p(\mathbf{Y}), \mathbf{Z} \sim p(\mathbf{Z})}[\log(1 - D(G(\mathbf{Y}, \mathbf{Z})))]. \end{aligned} \tag{8}$$

3.5. Reciprocal Attack for Matched Prediction of Human Trajectories

Once the forward and backward networks are successfully trained with the above loss functions based on the reciprocal learning approach, we are ready to perform prediction of the human trajectories. By taking advantage of the reciprocal property of the forward and backward networks, we develop a new network inference method called *reciprocal attack for matched prediction* as a post-processing step to further improve the prediction accuracy by making full use of the current observation.

As illustrated in Figure 4, $\mathbf{F}_\theta$ is our trained network for human trajectory prediction. With the past trajectories $\mathbf{X}$ as input, it predicts the future trajectories $\hat{\mathbf{Y}} = \mathbf{F}_\theta(\mathbf{X})$. During network testing or actual prediction, we do not know the ground truth of the future trajectory. How do we know

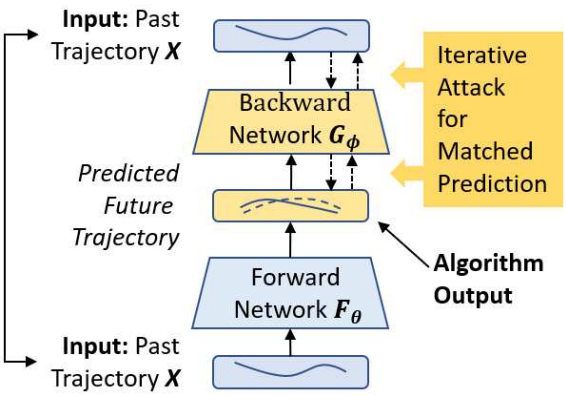

Figure 4. Illustration of the proposed attack method.

if this prediction $\hat{\mathbf{Y}}$ is accurate or not? How can we further improve its accuracy? Fortunately, in our reciprocal learning framework, we have another network, the backward prediction network $\mathbf{G}_\phi$, which can be used to map the estimated $\hat{\mathbf{Y}}$ back to the known input $\mathbf{X}$. Our reasoning is that, if $\hat{\mathbf{Y}}$ is accurate, then its backward prediction $\hat{\mathbf{X}} = \mathbf{G}_\phi(\hat{\mathbf{Y}}) = \mathbf{G}_\phi(\mathbf{F}_\theta(\mathbf{X}))$ should match the original input $\mathbf{X}$. When the prediction $\hat{\mathbf{Y}}$ is not accurate, we can modify the prediction such that the above matching error is minimized. This leads to the following optimization problem:

$$\hat{\mathbf{Y}}^* = \arg \min_{\tilde{\mathbf{Y}} = \hat{\mathbf{Y}} + \Delta(t)} ||\mathbf{X} - \mathbf{G}_\Phi(\tilde{\mathbf{Y}})||_2. \tag{9}$$

Here, $\Delta(t)$ is the small perturbation or modification added to the existing prediction result $\hat{\mathbf{Y}}$. The above optimization procedure aims to find the best modification $\hat{\mathbf{Y}}^* = \hat{\mathbf{Y}} + \Delta(t)$ to minimize the matching error.

This optimization problem can be solved by adversarial attack methods recently studied in the literature of deep neural network attack and defense. In this work, we propose to borrow the idea from the Fast Gradient Sign method (FGSM) developed by Goodfellow *et al.* [11] to perform adversarial attacks. Essentially, it is the same error back propagation procedure as network training. The only difference is that network training modifies the network weights based on error gradients. However, the adversarial attack does not modify the network weights, it propagates the error all the way to the input layer to modify the original input image to minimize the loss.

This approach uses the sign of the gradient at each pixel to determine its direction of change in its pixel value. In our case, we remove the sign function and directly use the gradient to update the input trajectory. With the matching error of human trajectories $E = ||\mathbf{X} - \mathbf{G}_\Phi(\tilde{\mathbf{Y}})||_2$, we can perform multiple iterations of the modified FGSM attack on the prediction $\hat{\mathbf{Y}}$ such that the matching error is minimized.

At iteration m, the attacked trajectory (input) is given by

$$\hat{\mathbf{Y}}^m = \hat{\mathbf{Y}}^{m-1} + \epsilon \cdot \nabla_{\hat{\mathbf{Y}}} E(\mathbf{X}, \hat{\mathbf{Y}}^{m-1}), \tag{10}$$

with $\hat{\mathbf{Y}}^0 = \hat{\mathbf{Y}}$. ϵ is the magnitude of attacks [11]. Intuitively, the updated trajectory $\hat{\mathbf{Y}}^m$ will minimize E. We then perform an exponential average of $\{\hat{\mathbf{Y}}^m\}$ to obtain the improved prediction

$$\hat{\mathbf{Y}}^* = \left[\sum_{m=1}^{M} e^{\alpha \cdot m} \cdot \hat{\mathbf{Y}}^m\right] / \sum_{m=1}^{M} e^{\alpha \cdot m}, \tag{11}$$

where M is the total iterations and α is a constant to control the relative weights between these different iterations of attacks. Its value is chosen based on heuristic studies. In our experiments, we set $\alpha = 0.1$.

4. Experimental Results

In this section, we present our experimental results, performance comparison with state-of-the-art methods, and ablation studies.

4.1. Benchmark Datasets

The comparison and ablation experiments are performed on ETH [27] and UCY [19] datasets which contain real world human trajectories and various natural human-human interaction circumstance. In total, 5 sub-datasets, ETH, HOTEL, UNIV, ZARA1 and ZARA2, are included in these two datasets. Each set contains bird-view images and 2D locations of each person. In total there are 1536 persons in these five sets of data. They contain challenging situations, including human collision avoidance, human crossing each other and group behaviors [29].

4.2. Implementation Details

Our GAN model is constructed using the LSTM for the encoder and decoder. The generator and discriminator are trained iteratively with the Adam optimizer. We choose the batch size of 64 and the initial learning rate of 0.001. The whole model is trained for 200 epochs. The trajectories are embedded using a single layer MLP with dimension of 16. The encoder and decoder for the generator use an LSTM with the hidden state's dimension of 32. In the LSTM encoder for the discriminator, the hidden state's dimension is 48. The maximum number of human surrounded with the target human is set as 32. This value is chosen since in all datasets, none of them has more than 32 human in any frame. For the depth map extraction, we use the pre-trained model "monodepth2" from [9] and the depth feature is embedded using a single layer MLP with an embedding dimension of 16. The weight for our loss function is $\lambda = 0.5$. We perform the reciprocal attack for 20 iterations, the perturbation ϵ is set as -0.05.

4.3. Evaluation Metrics and Methods

We use the same error metrics in [1, 26] for performance evaluations. (1) Average Displacement Error (ADE) is the average L_2 distance between the ground truth and our prediction over all predicted time steps from $T_o + 1$ to T_p. (2) Final Displacement Error (FDE) is the Euclidean distance between the predicted final destination and the true final destination at end of the prediction period T_p. They are defined as:

$$\text{ADE} = \frac{\sum_{i \in \Psi} \sum_{t=T_o+1}^{T_p} \sqrt{((\hat{x}_t^i, \hat{y}_t^i) - (x_t^i, y_t^i))^2}}{|\Psi| \cdot T_p}, \tag{12}$$

$$\text{FDE} = \frac{\sum_{i \in \Psi} \sqrt{((\hat{x}_{T_p}^i, \hat{y}_{T_p}^i) - (x_{T_p}^i, y_{T_p}^i))^2}}{|\Psi|}, \tag{13}$$

where $(\hat{x}_t^i, \hat{y}_t^i)$ and (x_t^i, y_t^i) are the predicted and ground truth coordinates for human i at time t, Ψ is the set of human and $|\Psi|$ is the total number of human in the test set.

Following previous papers [1, 12, 29], we use the similar leave-one-out evaluation methodology. Four datasets are used for training and the remaining one is used for testing. Given the human trajectory for the past 8 time steps (3.2 seconds), our model predicts the future trajectory for next 12 time steps (4.8 seconds).

4.4. Comparison with Existing Methods

We compare our method with the following state-of-the-art methods: (1) *Linear*: This method applies a linear regression to estimate linear parameters by minimizing the least square error [12]. (2) *LSTM*: This is the baseline model for the LSTM method, which does not consider any human-human interaction or background scene information. (3) *S-LSTM* [1]: This method models each human by an LSTM and proposes a social pooling mechanism with the hidden states of human within a certain grid at each time step. (4) *S-GAN* [12]: This is one of the first GAN-based methods. During the pooling stage, all human in the scene are considered. *S-GAN* and *S-GAN-P* are different only in whether the pooling mechanism is applied or not. The method chooses the best trajectory from 20 network predictions as the final test result. (5) *SoPhie* [29]: This work implements the so-called physical constrain described by background scene features. Also the attention mechanism is introduced in this GAN-based method. (6) *Next* [21]: This method implements a multiple feature pooling LSTM-based predictor. In the test part, besides using a single model, the paper follows [12] to train 20 different models using random initialization. They reported both "single model" and "20 outputs" evaluation results in the paper. In our comparison, we select the best results from these two parts.

Table 1. Comparisons of different methods on ETH (Column 2 and 3) and UCY (Column 4-6) datasets on the task of predicting 12 future time steps, given the previous 8 time steps. Error metrics reported are ADE / FDE in meter scale.

Method	ETH	HOTEL	UNIV	ZARA1	ZARA2	Avg
Linear	1.33 / 2.94	0.39 / 0.72	0.82 / 1.59	0.62 / 1.21	0.77 / 1.48	0.79 / 1.59
LSTM	1.09 / 2.14	0.86 / 1.91	0.61 / 1.31	0.41 / 0.88	0.52 / 1.11	0.70 / 1.52
S-LSTM[1]	1.09 / 2.35	0.79 / 1.76	0.67 / 1.40	0.47 / 1.00	0.56 / 1.17	0.72 / 1.54
S-GAN[12]	0.81 / 1.52	0.72 /1.61	0.60 / 1.26	0.34/ 0.69	0.42 / 0.84	0.58 / 1.18
S-GAN-P[12]	0.87 / 1.62	0.67 / 1.37	0.76 / 1.52	0.35 / 0.68	0.42 / 0.84	0.61 / 1.21
SoPhie[29]	0.70 / 1.43	0.76 / 1.67	0.54 / 1.24	0.30 / 0.63	0.38 / 0.78	0.54 / 1.15
Next[21]	0.73 / 1.65	**0.30 / 0.59**	0.60 / 1.27	0.38 / 0.81	0.31 / 0.68	0.46 / 1.00
Ours	**0.69 / 1.24**	0.43 / 0.87	**0.53 / 1.17**	**0.28 / 0.61**	**0.28 / 0.59**	**0.44 / 0.90**

4.5. Quantitative Results

Table 1 shows the comparison results of our method against existing methods on performance metrics ADE and FDE in meter scale. We follow the prior work [12] to choose the best prediction among multiple samples in L_2 norm for quantitative evaluation. We can see that our method outperforms all other methods except the Hotel dataset against the *Next* method. The *Linear* model generally performs the worst. It can only predict the straight trajectory and suffers from degraded performance in complicated human-human and human-environment interactions. The *LSTM* approach performs better than *Linear* method since it can handle more complicated trajectories. *S-LSTM* also outperforms the *Linear* model since it uses the social pooling mechanism, but it performs worse than *LSTM*. According to [12], the *S-LSTM* [1] is trained on a synthetic dataset and fine-tuned on the real dataset to improve the accuracy.

To evaluate the performance of our method in predicting feasible paths in crowded scenes, we follow the procedure in previous papers [29] to report a new evaluation metric which is the percentage of *near-collisions* among humans. A collision is defined when the euclidean distance between two human is smaller than 0.1m. We compute the average percentage of human near-collision in each frame of ETH and UCY datasets. The comparison results against the *Linear* , *S-GAN* and *SoPhie* methods are shown in Table 2. We can see that our method outperforms these three methods on the ETH, HOTEL, and ZARA2 datasets, producing less human collisions in the future time. For the other two datasets, UNIV and ZARA1, *S-GAN* and *SoPhie* are slightly better than ours. However, they suffer from significant performance degradation on other datasets. Overall, the experimental results demonstrate that our method can predict better physical and socially acceptable paths when compared to these existing methods.

4.6. Ablation Studies

To systematically evaluate our method and study the contribution of each algorithm component, we perform a number of ablation experiments. Our algorithm has three major new components, the reciprocal learning, the incorporation of 3D depth map features, and the reciprocal attacks for matched prediction. In the first row of Table 3, we list the ADE and FDE results for our method (full algorithm). The second row shows the results for our method without reciprocal training. The third row shows results without depth map features. The last row shows results without reciprocal attacks for prediction. We can clearly see that each algorithm component is contributing to the overall performance.

With the reciprocal consistency constraint, during training, our model forces the backward predicted trajectory to be consistent with the observed past trajectory, thus the predicted future trajectory which is the input of the backward network will be forced to be closer to the ground truth. Results show the benefit of the depth feature since it can help the model to better understand human behavior and the background scene context. The reciprocal attack mechanism modifies the predicted trajectory in an iterative manner to match the original trajectory with the backward prediction network.

Table 2. Average percentage of colliding human for each scene in ETH and UCY datasets. A human collision is defined and detected as the Euclidean distance between two human is less than 0.1m [29]. The first column represents the ground truth.

	GT	Linear	S-GAN	SoPhie	Ours
ETH	0.000	3.137	2.509	1.757	**1.512**
HOTEL	0.092	1.568	1.752	1.936	**1.547**
UNIV	0.124	1.242	**0.559**	0.621	0.563
ZARA1	0.000	3.776	1.749	**1.027**	1.094
ZARA2	0.732	3.631	2.020	1.464	**1.252**
Avg	0.189	2.670	1.717	1.361	**1.194**

4.7. Qualitative Results

Figure 5 shows successful and failure examples of our predicted trajectories from ETH, HOTEL, UNIV, ZARA1 and ZARA2 datasets in each row. Following prior work S-GAN [12], we show the best predicted trajectory among 20

Table 3. Ablation experiments of our full algorithm without different components. Error metrics reported are ADE / FDE in meter scale.

Method	ETH	HOTEL	UNIV	ZARA1	ZARA2
Our Method (Full Algorithm)	0.69 / 1.24	0.43 / 0.87	0.53 / 1.17	0.28 / 0.61	0.28 / 0.59
- Without Reciprocal Learning	0.73 / 1.31	0.49 / 0.97	0.60 / 1.22	0.38 / 0.73	0.36 / 0.70
- Without Depth Features	0.71 / 1.30	0.43 / 0.88	0.56 / 1.19	0.31 / 0.63	0.31 / 0.62
- Without Reciprocal Attacks	0.70 / 1.26	0.45 / 0.90	0.55 / 1.18	0.32 / 0.65	0.30 / 0.61

model outputs in the figure. The first two columns show scenarios that our proposed method is able to correctly predict the future path. According to the background scene, we can see that our method can ensure that each human path follows the physical constrains of the scene, such as walking around obstacles, *e.g.* trees, and staying on sidewalks. Our method also shows the decent prediction results under human-human interactions circumstance. When persons walk in a crowded road, they can avoid each other when they merge from various directions and then walk towards a common direction.

The last column in Figure 5 shows some failure cases which have relatively large error rates. For example, we see human slowing down or even stops for a while, or human taking a straight path rather than making a detour around the obstacles. Nevertheless, in most case, our method still can predict the plausible path, even though the predicted path is not quite same as the ground truth. For example, for the first, third and fifth cases in the last column, in our prediction paths, the target human are trying to walk around another human or the tree in the road, which are quite reasonable in practice.

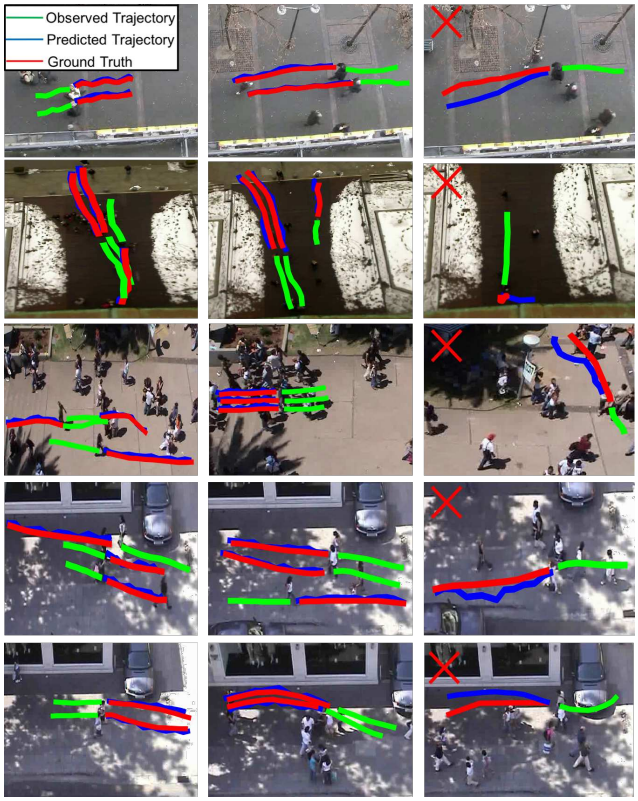

Figure 5. Illustration of our method predicting trajectories of future 12 time steps, given the observed trajectories of past 8 time steps. The results for ETH, HOTEL, UNIV and ZARA1 and ZARA2 are shown in rows 1 to 5, respectively. We show examples where our model successfully predicts the trajectories with small errors in first two columns. The last column shows some failure cases. Note that, we cropped and resized the original image for better visualization.

5. Conclusion and Major Contributions

In this paper, we have explored the unique characteristics of human trajectories and developed a new approach, reciprocal network learning, for human trajectory prediction. Extensive experimental results have demonstrated our approach achieves the state-of-art performance on public benchmark datasets.

The **major contributions** of this work can be summarized as follows. (1) We have established a forward and backward prediction network structure for human trajectory prediction, which satisfies the reciprocal prediction constraints. (2) Based on this constraint, we have developed a reciprocal learning approach to jointly train these two prediction networks in an collaborative and iterative manner. (3) Once the network is successfully trained, we have developed a new approach for network inference by integrating the concept of adversarial attacks with the reciprocal constraint. It is able to iteratively refine the predicted trajectory by the forward network such that the reciprocal constraint is satisfied. (4) Our ablation studies have shown that the proposed new approach is very effective with significant contributions to the overall performance of our method, which outperforms other state-of-the-art methods in the literature.

Acknowledgement

This work was supported in part by National Science Foundation under grants 1647213 and 1646065. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016.

[2] Lamberto Ballan, Francesco Castaldo, Alexandre Alahi, Francesco Palmieri, and Silvio Savarese. Knowledge transfer for scene-specific motion prediction. In *European Conference on Computer Vision*, pages 697–713. Springer, 2016.

[3] Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh. Recycle-gan: Unsupervised video retargeting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 119–135, 2018.

[4] Federico Bartoli, Giuseppe Lisanti, Lamberto Ballan, and Alberto Del Bimbo. Context-aware trajectory prediction. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 1941–1946. IEEE, 2018.

[5] Niccoló Bisagno, Bo Zhang, and Nicola Conci. Group lstm: Group trajectory prediction in crowded scenarios. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[6] Richard W Brislin. Back-translation for cross-cultural research. *Journal of cross-cultural psychology*, 1(3):185–216, 1970.

[7] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017.

[8] Pasquale Coscia, Francesco Castaldo, Francesco AN Palmieri, Alexandre Alahi, Silvio Savarese, and Lamberto Ballan. Long-term path prediction in urban scenarios using circular distributions. *Image and Vision Computing*, 69:81–91, 2018.

[9] Clément Godard, Oisin Mac Aodha, Michael Firman, and Gabriel J Brostow. Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3828–3838, 2019.

[10] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 270–279, 2017.

[11] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[12] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi. Social gan: Socially acceptable trajectories with generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2255–2264, 2018.

[13] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*, pages 820–828, 2016.

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

[15] Qi-Xing Huang and Leonidas Guibas. Consistent shape maps via semidefinite programming. In *Computer Graphics Forum*, volume 32, pages 177–186. Wiley Online Library, 2013.

[16] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Forward-backward error: Automatic detection of tracking failures. In *2010 20th International Conference on Pattern Recognition*, pages 2756–2759. IEEE, 2010.

[17] Kris M Kitani, Brian D Ziebart, and J Andrew. Bagnell, and martial hebert. activity forecasting. In *European Conference on Computer Vision. Springer*, volume 59, page 88, 2012.

[18] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[19] Laura Leal-Taixé, Michele Fenzi, Alina Kuznetsova, Bodo Rosenhahn, and Silvio Savarese. Learning an image-based motion context for multiple people tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3542–3549, 2014.

[20] Namhoon Lee, Wongun Choi, Paul Vernaza, Christopher B Choy, Philip HS Torr, and Manmohan Chandraker. Desire: Distant future prediction in dynamic scenes with interacting agents. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 336–345, 2017.

[21] Junwei Liang, Lu Jiang, Juan Carlos Niebles, Alexander G Hauptmann, and Li Fei-Fei. Peeking into the future: Predicting future person activities and locations in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5725–5734, 2019.

[22] Matthias Luber, Johannes A Stork, Gian Diego Tipaldi, and Kai O Arras. People tracking with human motion predictions from social forces. In *2010 IEEE International Conference on Robotics and Automation*, pages 464–469. IEEE, 2010.

[23] Huynh Manh and Gita Alaghband. Scene-lstm: A model for human trajectory prediction. *arXiv preprint arXiv:1808.04018*, 2018.

[24] Ramin Mehran, Alexis Oyama, and Mubarak Shah. Abnormal crowd behavior detection using social force model. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 935–942. IEEE, 2009.

[25] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016.

[26] Stefano Pellegrini, Andreas Ess, Konrad Schindler, and Luc Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *2009 IEEE 12th International Conference on Computer Vision*, pages 261–268. IEEE, 2009.

[27] Stefano Pellegrini, Andreas Ess, and Luc Van Gool. Improving data association by joint modeling of pedestrian trajectories and groupings. In *European conference on computer vision*, pages 452–465. Springer, 2010.

[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large

scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[29] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1349–1358, 2019.

[30] Amir Sadeghian, Ferdinand Legros, Maxime Voisin, Ricky Vesel, Alexandre Alahi, and Silvio Savarese. Car-net: Clairvoyant attentive recurrent network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 151–167, 2018.

[31] Narayanan Sundaram, Thomas Brox, and Kurt Keutzer. Dense point trajectories by gpu-accelerated large displacement optical flow. In *European conference on computer vision*, pages 438–451. Springer, 2010.

[32] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[33] Mark Twain. *The jumping frog: in English, then in French, then clawed back into a civilized language once more by patient, unremunerated toil*. Courier Corporation, 1971.

[34] Anirudh Vemula, Katharina Muelling, and Jean Oh. Social attention: Modeling attention in human crowds. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–7. IEEE, 2018.

[35] Hao Xue, Du Q Huynh, and Mark Reynolds. Ss-lstm: A hierarchical lstm model for pedestrian trajectory prediction. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1186–1194. IEEE, 2018.

[36] Zhichao Yin and Jianping Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1983–1992, 2018.

[37] Christopher Zach, Manfred Klopschitz, and Marc Pollefeys. Disambiguating visual relations using loop constraints. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1426–1433. IEEE, 2010.

[38] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1851–1858, 2017.

[39] Tinghui Zhou, Yong Jae Lee, Stella X Yu, and Alyosha A Efros. Flowweb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1191–1200, 2015.

[40] Tinghui Zhou, Philipp Krahenbuhl, Mathieu Aubry, Qixing Huang, and Alexei A Efros. Learning dense correspondence via 3d-guided cycle consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 117–126, 2016.

[41] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.