

Imitation Learning for Human Pose Prediction

Borui Wang, Ehsan Adeli, Hsu-kuang Chiu, De-An Huang, Juan Carlos Niebles
 Stanford University

{wbr, eadeli, hkchiu, dahuang, jniebles}@cs.stanford.edu

Abstract

Modeling and prediction of human motion dynamics has long been a challenging problem in computer vision, and most existing methods rely on the end-to-end supervised training of various architectures of recurrent neural networks. Inspired by the recent success of deep reinforcement learning methods, in this paper we propose a new reinforcement learning formulation for the problem of human pose prediction, and develop an imitation learning algorithm for predicting future poses under this formulation through a combination of behavioral cloning and generative adversarial imitation learning. Our experiments show that our proposed method outperforms all existing state-of-the-art baseline models by large margins on the task of human pose prediction in both short-term predictions and long-term predictions, while also enjoying huge advantage in training speed.

1. Introduction

Modeling the dynamics of human motion and predicting human poses is an important and challenging problem in computer vision that has many useful applications in robotics, computer graphics, healthcare, public safety, etc. [15, 21, 22, 45, 46]. Previous work on this subject has mainly been focusing on designing different architectures of recurrent neural networks (RNNs) to model human motion dynamics and adopted a pure supervised-learning approach to train recurrent neural networks to predict future human poses in a sequence [8, 10, 20, 29].

These previous methods based on supervised training of RNN architectures face two main challenges: (1) due to the purely supervised nature of the training methods, the learned RNNs usually do not generalize well to unseen domains of the human motion space, which are very likely to appear during test time; (2) since the RNNs are required to generate the whole human pose prediction sequence all together, it is very difficult to keep a good balance between short-term and long-term prediction accuracies.

Recently, we have witnessed great success in the devel-

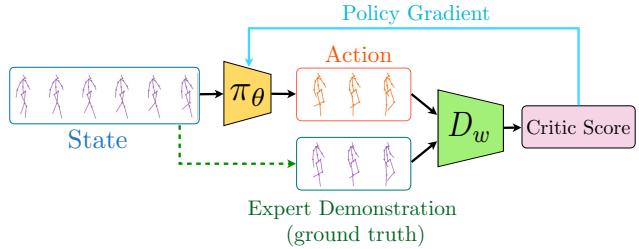


Figure 1: At the core of our imitation learning approach to human pose prediction is a Generative Adversarial Imitation Learning (GAIL) [16] process. With the critic function D_w and the policy generator π_θ , we alternate between updating D_w (D step) through comparing generated windows of predicted poses with ground truth, and updating π_θ (G step) through policy gradient over critic scores from D_w .

opment of deep reinforcement learning algorithms, which have achieved significantly enhanced state-of-the-art performance on many tasks in the areas of game, robotics and control [31, 38, 39]. These recent deep reinforcement learning algorithms, including Generative Adversarial Imitation Learning (GAIL) [16] and Deep Deterministic Policy Gradient [26], enjoy the advantages of generalizing well over unseen terrains and maintaining strong sequential correlation among local decisions across time. Therefore, in order to overcome the above limitations of the previous methods for human pose prediction, a natural question to ask is whether we could build a bridge between reinforcement learning and sequential modeling, such that we can harness the great power of deep reinforcement learning algorithms to help us better model human motion dynamics and predict human poses from sequential observations. In this paper, we propose a new modeling framework for human motion dynamics that transforms the task of predicting human poses into a reinforcement learning problem. The environment of this reinforcement learning problem cannot provide feedback signals as the learning agent interacts with it. All we have during the learning process is a training dataset consisting of trajectories of human poses recorded from real human motion. Therefore, we adopt an imitation learning [36, 55] approach and use the training dataset as

expert demonstrations for our prediction agent to imitate.

This imitation learning problem of predicting human poses has several key difficulties: (1) the action space is continuous-valued and very high-dimensional; (2) the state space is heterogeneous and encompasses very long sequences of historical observations with different lengths; (3) the expert demonstrations are performed by different human subjects and thus exhibit large variance across the underlying expert policies. To tackle this challenging imitation learning task, we extend the Generative Adversarial Imitation Learning (see Figure 1) framework with sequence-to-sequence architectures [42] and Deep Deterministic Policy Gradient [26] methods to train our prediction agent to make accurate predictions of human poses. We also use the efficient behavioral cloning [4] algorithm as pre-training to expedite the training process.

We evaluate the performance of our proposed imitation learning algorithm on the popular Human 3.6M dataset [18]. Our experiments demonstrate that our proposed algorithm outperforms all the previous methods for human pose prediction by large margins on both short-term and long-term predictions and sets the new state-of-the-art performance results on the Human 3.6M dataset. The experiments also show that our algorithm has huge advantage in speed and can be trained much more efficiently compared to previous algorithms.

To summarize, the main contributions of our work are: (1) We propose a new reinforcement learning formulation for the problem of human pose prediction that supports more accurate predictions over both short-term and long-term horizons; (2) We develop an imitation learning algorithm for human pose prediction based on this reinforcement learning formulation through a combination of behavioral cloning and generative adversarial imitation learning. Our algorithm combines the advantages from both learning frameworks and achieves a good balance between sample efficiency and policy generalizability; (3) We run extensive experiments on the challenging Human 3.6M dataset to evaluate the performance of our proposed method, and show that it outperforms all existing state-of-the-art baseline models by significant margins.

2. Related Work

Human Motion Prediction: Most of the previous work on video prediction predict future video sequences by reconstructing frames at the pixel level [28, 47], and predict dense trajectories [49], semantic labels [27, 48], or activity labels [3, 25, 41, 50] in the future. However, human motion dynamics is better captured by detailed joint locations (*i.e.*, pose) [1, 30, 35], and is often modeled by either *state transition models* [51, 52] or *recurrent neural networks* [11, 20, 29].

Recently, several works focused on forecasting human

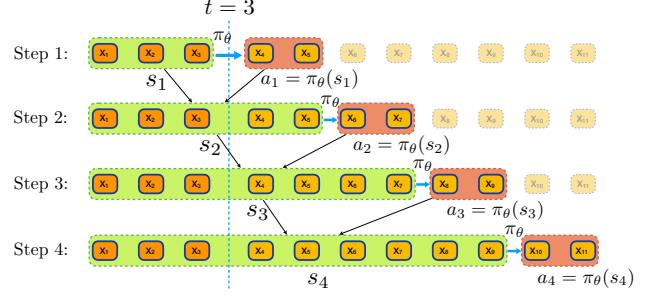


Figure 2: Illustration of progressive prediction and our reinforcement learning formulation of human pose prediction using an example pose prediction task with length of past observations $t = 3$, length of total future predictions $l = 8$ and size of step-wise prediction windows $m = 2$.

poses in videos [5, 7, 8, 20, 29, 50]. Chao *et al.* [7] proposed a 3D Pose Forecasting Network on static images; Barsoum *et al.* [5] took a probabilistic approach for pose prediction using Wasserstein GAN [2]; Walker *et al.* [50] proposed a variational autoencoder solution; Fragkiadaki *et al.* [10] proposed two architectures denoted by LSTM-3LR (3 layers of LSTM cells) and ERD (Encoder-Recurrent-Decoder); Yan *et al.* [54] and Zhao *et al.* [56] proposed methods for longer time prediction; Martinez *et al.* [29] used a carefully tailored RNN to learn human motion prediction; and Chiu *et al.* [8] proposed a multi-layer hierarchical RNN architecture (denoted by TP-RNN) to capture human dynamics. In contrast, instead of training a fully supervised model, in this work we introduce the unsupervised GAIL framework into our training process to enhance the generalizability of our learned prediction policy.

Reinforcement Learning for Prediction: Methods of reinforcement learning [26, 31, 32, 43] and imitation learning [16, 36] have been used for predicting future information in videos in different forms. For instance, DeepMimic [33] proposed a physics-based method for policy generation that is capable of tracking and motion capture. Other works such as R2P2 [34] and Tai *et al.* [44] used generative models with Wasserstein reformulation to perform navigation and forward path planning. Differently, in this work we reformulate sequential pose prediction into a reinforcement learning problem and train a prediction policy through imitation learning approaches.

3. Pose Prediction as Reinforcement Learning

In this section, we introduce how to transform human pose prediction into a reinforcement learning problem [43]. The core task of human pose prediction is the following: given a sequence of past pose observations $\{x_1, x_2, \dots, x_t\}$ of length t from a human subject, predict the future pose sequence $\{x_{t+1}, x_{t+2}, \dots, x_{t+l}\}$ up to length l . Most previous work adopts an end-to-end approach to train either a

single recurrent neural network or a pair of encoder-decoder RNNs to output the whole length- l prediction sequence $\{x_{t+1}, x_{t+2}, \dots, x_{t+l}\}$ after reading in the historical sequence $\{x_1, x_2, \dots, x_t\}$ [8, 20, 29]. In contrast, under our new reinforcement learning formulation, we evenly break the whole prediction sequence $\{x_{t+1}, x_{t+2}, \dots, x_{t+l}\}$ into multiple steps. In each step, we only make predictions over a small window of future poses conditioned on all observed and predicted pose information so far. This transformation turns our pose prediction task into a sequential decision-making problem, where reinforcement learning algorithms can naturally come into play.

More formally, suppose we divide the whole prediction sequence into K steps, then each step would correspond to a window of $m = \frac{l}{K}$ pose vectors (without loss of generality, assume that l is divisible by K). We now define a **Markov Decision Process** (MDP) [43] to model the generation of the pose prediction sequence. At each step i , where $i \in \{1, 2, \dots, K\}$, the state of the MDP is defined as the list of all previous pose vectors: $s_i = \{x_1, x_2, \dots, x_{t+(i-1)\times m}\}$ and the action of the MDP is defined as the next length- m window of pose vectors that the learning agent needs to predict: $a_i = \{x_{t+(i-1)\times m+1}, \dots, x_{t+i\times m}\}$. The transition dynamic of this MDP is deterministic, which means that at each step i , taking an action $a_i = \{x_{t+(i-1)\times m+1}, \dots, x_{t+i\times m}\}$ at state $s_i = \{x_1, x_2, \dots, x_{t+(i-1)\times m}\}$ would deterministically transition the MDP into the new state $s_{i+1} = \{x_1, x_2, \dots, x_{t+i\times m}\}$ at step $i + 1$. The new state s_{i+1} is formed by appending the action a_i to the end of the current state s_i . This MDP process of **progressive prediction** is illustrated in Figure 2.

Motivation: There are three key motivations behind this reinforcement learning formulation of the human pose prediction problem. First, through breaking the long prediction sequence into smaller pieces of pose windows, at each step the prediction agent only needs to focus on learning to predict a much shorter period of time into the future, which greatly reduces the difficulty of prediction for the agent. Second, the strong sequential correlation across different actions in a MDP guarantees that our agent’s prediction policy will not only focus on short-term prediction accuracy but also takes long-term prediction performance into consideration. Third, only through this formulation can we apply the unsupervised GAIL approach to enhance the generalizability of our pose prediction policy over unseen domains.

4. Imitation Learning for Predicting Human Pose Sequences

Now that we have formulated human pose prediction into a reinforcement learning problem, in this section, we develop an imitation learning method to train the prediction policy under this new reinforcement learning formulation.

4.1. Deterministic Policy

Under our formulation, the policy of our reinforcement learning agent would be a mapping from the space of all possible states \mathcal{S} to the space of all possible actions \mathcal{A} . This policy can be either deterministic or stochastic. Under our current setting of human pose prediction, stochastic policies [43] are not suitable because the dimensionality of the action space \mathcal{A} is very high. So we decide to use deterministic policies in our imitation learning algorithms for human pose prediction. Formally, the policy of our prediction agent is a function $a = \pi_\theta(s)$ that maps each possible state s in \mathcal{S} to a single action a in \mathcal{A} , where θ denotes the parameters of this policy mapping function.

4.2. Behavioral Cloning

A classical approach to imitation learning is behavioral cloning (BC) [4, 36], where a policy is trained using supervised learning methods to minimize the distance between its generated actions and the expert’s actions under the state distribution encountered by the expert. Let the expert policy function be π_E and the state distribution encountered by the expert be d_{π_E} , then a behavioral cloning algorithm tries to find an optimal policy π_{θ^*} such that:

$$\pi_{\theta^*} = \arg \min_{\pi_\theta \in \Pi} \mathbb{E}_{s \sim d_{\pi_E}} [l(\pi_\theta(s), \pi_E(s))], \quad (1)$$

where l is a loss function that measures the distance between two action vectors in the action space \mathcal{A} .

BC enjoys the advantage of being highly sample-efficient and fast to compute [37], since it aims to directly match the learning agent’s policy with the expert’s policy at every state that appears in the repository of expert-demonstrated trajectories. However, there are some caveats associated with BC. The first problem is the generalization issue — the agent’s policy may be overfitted to the expert’s demonstrations in the areas of the state space that the expert traversed, and thus may not generalize well to areas outside the expert’s experience. Second, behavioral cloning optimizes the agent’s policy at each individual state separately and ignores the sequential correlation across different steps in a trajectory, which tends to make the agent’s policy shortsighted. To tackle these problems, we introduce generative adversarial training into our algorithm and use an adapted version of GAIL to further optimize our agent’s policy.

4.3. Generative Adversarial Imitation Learning

Generative Adversarial Imitation Learning (GAIL) is a popular model-free imitation learning framework that was recently proposed by Ho and Ermon [16] and is inspired by the success of Generative Adversarial Networks (GAN) [13]. At its core, GAIL takes an inverse reinforcement learning [32] approach to the problem of imitation learning,

where it aims to first recover an estimate of the reward signals underlying the MDP from the expert’s demonstration and then learns to optimize the agent’s policy using the rewards signals that it recovered. Directly estimating reward functions from expert demonstration in high-dimensional and complex state-action space is intractable, so GAIL turns the inverse reinforcement learning problem into its equivalent dual problem of occupancy measure matching, where the agent seeks to match the distribution of state-action pairs (called occupancy measure) generated by its own policy to the distribution generated by the expert’s policy [16]. In its original formulation, GAIL employs a generative adversarial training process to iteratively minimize the Jensen-Shannon divergence between the two distributions, and its learning objective is:

$$\begin{aligned} \min_{\pi_\theta} \max_{D_w \in \mathcal{S} \times \mathcal{A} \rightarrow (0,1)} & (\mathbb{E}_{\pi_\theta}[\log(D_w(s, a))] \\ & + \mathbb{E}_{\pi_E}[\log(1 - D_w(s, a))] - \lambda H(\pi_\theta)), \end{aligned} \quad (2)$$

where D_w can be interpreted as a discriminative classifier trying to distinguish between state-action pairs generated from the agent’s policy and state-action pairs generated from the expert’s policy, and $H(\pi_\theta)$ denotes the discounted causal entropy of the policy π_θ . Then in analogy to the training of GANs, during actual implementation, a GAIL algorithm will alternate between a **D** step that updates the discriminator parameter w to maximize $\mathbb{E}_{\pi_\theta}[\log(D_w(s, a))] + \mathbb{E}_{\pi_E}[\log(1 - D_w(s, a))]$, and a **G** step that updates the parameter θ of the policy generator π_θ using policy gradient methods such as Trust Region Policy Optimization (TRPO) [38] or Proximal Policy Optimization (PPO) [39] in order to minimize $\mathbb{E}_{\pi_\theta}[\log(D_w(s, a))] + \lambda H(\pi_\theta)$, until the system reaches a saddle point (see Figure 1). Previous work shows that GAIL often tends to generalize better than BC and achieves superior performance on complex tasks.

However, in our human pose prediction scenario, there are two major difficulties that hinder the effective application of the original GAIL algorithm. First, the human motion dynamics is highly complex, and thus our randomly initialized policy generator will be vastly different from the expert’s policy at the beginning of GAIL training. This means that at the early stage of GAIL training, it is very likely that the policy generator manifold would have no non-negligible intersection with the expert manifold at all. This issue will let the Jensen-Shannon divergence to saturate quickly and cause the severe problem of vanishing gradients for the discriminator [2]. Second, the dimensionality of our action space \mathcal{A} is very high, which makes the sampling-based policy gradient reinforcement learning algorithms like TRPO and PPO no longer applicable in practice. Therefore, we adapt GAIL to a specific new form to fit the needs of our human pose prediction task, which we call WGAIL-div.

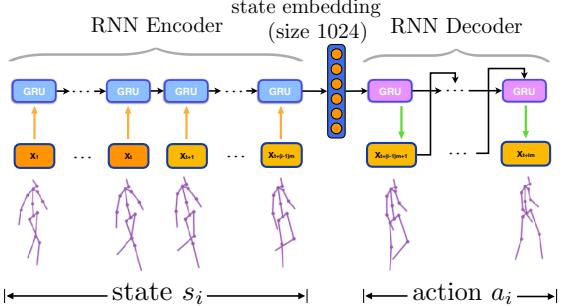


Figure 3: Seq2Seq architecture of our policy generator.

4.4. Adapted WGAIL-div

The problem of non-overlapping manifolds and vanishing gradients for the discriminator has long been studied for GANs [2], and several new GAN training frameworks based on Wasserstein distances between distributions has been proposed to overcome this problem. Some of the notable ones are Wasserstein GAN (WGAN) [2], Wasserstein GAN with gradient penalty (WGAN-gp) [14], and Wasserstein-divergence GAN (WGAN-div) [53]. This series of Wasserstein-based GANs changes the discriminator of GANs from a classifier that tries to distinguish between real and fake samples and outputs probability values between 0 and 1 into a critic function that directly outputs real numbers in $(-\infty, +\infty)$, and uses different methods to enforce the Lipschitz constraint on the critic function. In this work, we borrow the formulation of WGAN-div [53] into GAIL to construct an improved version of GAIL named **WGAIL-div**. More specifically, the objective function of our WGAIL-div algorithm is:

$$\begin{aligned} \min_{\pi_\theta} \max_{D_w \in \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}} & (\mathbb{E}_{\pi_\theta}[D_w(s, a)] - \mathbb{E}_{\pi_E}[D_w(s, a)] \\ & - k \mathbb{E}_{(\hat{s}, \hat{a}) \sim \mathbb{P}_u} [\|\nabla_{(\hat{s}, \hat{a})} D_w((\hat{s}, \hat{a}))\|^p]), \end{aligned} \quad (3)$$

where \mathbb{P}_u is the distribution obtained by sampling uniformly from the straight lines connecting points of state-action pairs generated by the policy generator and points of state-action pairs in expert demonstration in the state-action space $\mathcal{S} \times \mathcal{A}$, and k and p are hyperparameters that adjust the level of Lipschitz-constraint regularization. Note that in this WGAIL-div objective function we no longer have the causal entropy term $H(\pi_\theta)$ that appears in (2), since we use deterministic policies that have constant causal entropies.

In our implementation of WGAIL-div, we alternate between a **D** step that updates the discriminator parameter w to maximize $\mathbb{E}_{\pi_\theta}[D_w(s, a)] - \mathbb{E}_{\pi_E}[D_w(s, a)] - k \mathbb{E}_{(\hat{s}, \hat{a}) \sim \mathbb{P}_u} [\|\nabla_{(\hat{s}, \hat{a})} D_w((\hat{s}, \hat{a}))\|^p]$, and a **G** step that updates the parameter θ of the policy generator π_θ to minimize $\mathbb{E}_{\pi_\theta}[D_w(s, a)]$. As mentioned in the second problem facing GAIL above, for the **G** step, the classic sampling-based

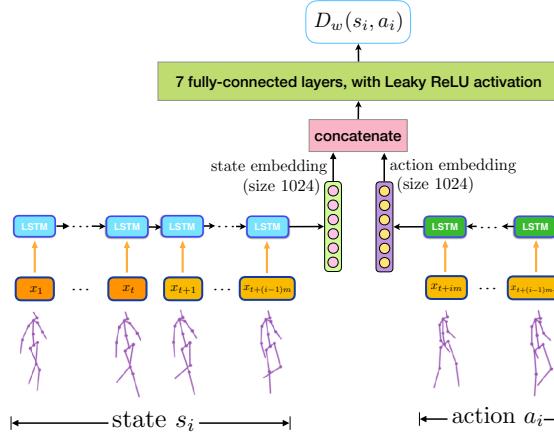


Figure 4: Architecture of our critic network in WGAIL-div.

approach taken by the original GAIL of using TRPO or PPO to optimize a parametrized stochastic policy fails in the high-dimensional action space \mathcal{A} . Therefore, inspired by [40] and [26], we use deterministic policy and substitute the TRPO step in G with a deterministic policy gradient step that directly use the gradient information from both the critic function D_w and the policy generating function π_θ to update θ in order to minimize the expected accumulated critic loss value incurred by the agent as it makes predictions according to its policy π_θ . More specifically, suppose that during the G step we execute our prediction agent’s current policy π_θ to generate a collection of N samples of state-action pairs $\{(s_i, a_i)\}_{i=1}^N$, then our current estimate of the gradient over the policy parameter θ would be:

$$\begin{aligned} & \nabla_\theta \mathbb{E}_{\pi_\theta}[D_w(s, a)] \\ & \approx \frac{1}{N} \sum_{i=1}^N \nabla_a D_w(s, a)|_{s=s_i, a=\pi_\theta(s_i)} \nabla_\theta \pi_\theta(s)|_{s=s_i} \end{aligned} \quad (4)$$

4.5. Model Architectures

Our imitation learning system has two components: a policy generator network π_θ and a critic network D_w .

Policy Generator Network: The policy generator network π_θ is at the core of our imitation learning system and is responsible for predicting future human poses by sequentially generating actions (small windows of consecutive pose vectors) from states (long sequences of previous pose observations and predictions). It is used in both behavioral cloning and WGAIL-div. At each prediction step i , π_θ needs to read in a length- $[t + (i - 1) \times m]$ sequence of state $s_i = \{x_1, x_2, \dots, x_{t+(i-1)m}\}$ and then outputs a length- m sequence as the action $a_i = \{x_{t+(i-1)m+1}, \dots, x_{t+i \times m}\}$. To achieve this functionality, we use the sequence-to-sequence (seq2seq) architecture [42] for our policy generator network. We construct a recurrent neural network (RNN) with a Gated Recurrent Unit (GRU) [9] cell of size

1024 as our encoder to read in the state sequence s_i , and construct another RNN with a different GRU cell of size 1024 as our decoder to generate the action sequence a_i . Similar to [29], we also add an extra layer of linear spatial decoder on top of each GRU to map its 1024-dimensional output vector down to a 54-dimensional vector (54 is the length of each pose vector in our dataset, see Section 5 for details). The model architecture of π_θ is shown in Figure 3. **Critic Network:** In our WGAIL-div algorithm, besides π_θ , we also need to construct a critic network D_w that assigns critic values to all state-action pairs (s_i, a_i) in $\mathcal{S} \times \mathcal{A}$. For this critic network, we build a RNN with a Long Short-Term Memory (LSTM) [17] cell of size 1024 to encode the state sequence $s_i = \{x_1, x_2, \dots, x_{t+(i-1)m}\}$ into a 1024-dimensional vector of state embedding $h(s_i)$, and build another RNN with a different LSTM cell of size 1024 to encode the action sequence $a_i = \{x_{t+(i-1)m+1}, \dots, x_{t+i \times m}\}$ into a 1024-dimensional vector of action embedding $u(a_i)$. We then concatenate $h(s_i)$ and $u(a_i)$ together into a 2048-dimensional vector embedding, which we feed into a 7-layer fully-connected neural network with the output size of each layer equals to 512, 256, 128, 64, 32, 16, 1, respectively and the activation functions being Leaky ReLU. This 7-layer fully-connected neural network outputs the final critic value for the state-action pair that D_w reads in. The model architecture of D_w is depicted in Figure 4.

4.6. Learning Algorithm

Our proposed imitation learning algorithm for human pose prediction utilizes two methods: BC and WGAIL-div. They both have their respective advantages and disadvantages, and their advantages and disadvantages make them highly complementary to each other. BC has fast training speed and high sample efficiency, but may suffer from bad generalization and error compounding. In contrast, WGAIL-div often produces policies that have better generalization property and pays more attention to the sequential relationship across different time steps, but can be slow and difficult to train during the beginning iterations when the freshly initialized agent policy is very far away from the expert policy. Then under our current setting of using imitation learning to predict human motion dynamics, these characteristics indicate that BC tends to perform better over short-term predictions while WGAIL-div tends to be superior over long-term predictions. Therefore, in order to have the best of both worlds and to let our learning algorithm excel at both short-term and long-term predictions, we first train our policy generator network π_θ using BC, and then use the trained parameters to initialize π_θ in the WGAIL-div procedure. On one hand, the initialization obtained from BC would help π_θ to quickly move to regions that are close to the expert policy, which greatly stabilizes and expedites

Algorithm 1 WGAIL-div for Human Pose Prediction

```

1: Initialize the policy generator network  $\pi_\theta$  using the parameter
    $\theta$  trained by BC (Algorithm 2 in Appendix A.1)
2: Randomly initialize the parameter  $w$  of the critic network  $D_w$ 
3: for iteration = 1, 2, ...,  $T$  do
   D Step:
4:   Randomly sample a batch of  $N_D$  length-( $t + l$ ) trajectories
      of human pose vectors from the training dataset  $\mathcal{E}$ 
5:   for  $j = 1, 2, \dots, N_D$  do
6:     Take the  $j$ -th sampled trajectory  $\{x_{j,1}, x_{j,2}, \dots, x_{j,t+l}\}$ 
7:     for  $i = 1, 2, \dots, K$  do
8:        $s_{j,i} = \{x_{j,1}, \dots, x_{j,t+(i-1)m}\}$ ,
9:        $a_{j,i} = \{x_{j,t+(i-1)m+1}, \dots, x_{j,t+m}\}$ 
10:      if  $i = 1$ , then  $\tilde{s}_{j,i} = s_{j,i}$ ,
        else  $\tilde{s}_{j,i} = \text{concatenate}[\tilde{s}_{j,i-1}, \tilde{a}_{j,i-1}]$ 
11:       $\tilde{a}_{j,i} = \pi_\theta(\tilde{s}_{j,i})$ 
12:      Sample  $\alpha$  from the uniform distribution:  $\alpha \sim U[0, 1]$ 
13:       $\hat{s}_{j,i} = \alpha s_{j,i} + (1 - \alpha) \tilde{s}_{j,i}$ ,
14:       $\hat{a}_{j,i} = \alpha a_{j,i} + (1 - \alpha) \tilde{a}_{j,i}$ 
15:    end for
16:   end for
17:   Take an Adam step on  $w$  to maximize the objective:
    $w \leftarrow \text{Adam} \left( \nabla_w \left[ \frac{1}{N_D \times K} \sum_{j,i} D_w(\tilde{s}_{j,i}, \tilde{a}_{j,i}) \right. \right.$ 
   
$$\left. \left. - D_w(s_{j,i}, a_{j,i}) - k \|\nabla_{(s_{j,i}, a_{j,i})} D_w((\hat{s}_{j,i}, \hat{a}_{j,i}))\|^p \right] \right)$$

   G Step:
18:   Randomly sample a batch of  $N_G$  length-( $t + l$ ) trajectories
      of human pose vectors from the training dataset  $\mathcal{E}$ 
19:   Generate the agent's and the expert's state-action pairs
       $\{(\tilde{s}_{j,i}, \tilde{a}_{j,i})\}$  and  $\{(s_{j,i}, a_{j,i})\}$  in the same way as in D.
20:   Take an Adam step on  $\theta$  to minimize the objective function:
    $\theta \leftarrow \text{Adam} \left( \frac{1}{N_G \times K} \sum_{j,i} \nabla_\theta D_w(s, a)|_{s=\tilde{s}_{j,i}, a=\pi_\theta(\tilde{s}_{j,i})} \right. \right.$ 
   
$$\left. \left. - \nabla_\theta \pi_\theta(s)|_{s=\tilde{s}_{j,i}} \right) \right)$$

21: end for

```

the training of WGAIL-div. On the other hand, WGAIL-div iterations can be viewed as further optimizing over the shortsighted policy generated from BC to make it generalize better to unfamiliar regions of the state space. As a result, our learning algorithm is consisted of two stages:

Stage 1: Behavioral Cloning. In the BC stage, we randomly sample expert trajectories from the training dataset \mathcal{E} , and update the parameter θ of the policy generator network π_θ in order to match its generated actions with the expert's actions over the states that appear in the sampled trajectories. Here, we use ℓ_1 -norm to measure the distance between two action vectors. Refer to Algorithm 2 in Appendix A.1 for the detailed steps of our BC algorithm.

Stage 2: WGAIL-div. In the WGAIL-div stage, our algorithm alternates between a **D** step that updates the critic function and a **G** step that optimizes the policy generator through deterministic policy gradients. The parameter w of the critic function is initialized randomly and the parameter θ of the policy generator is initialized using the trained

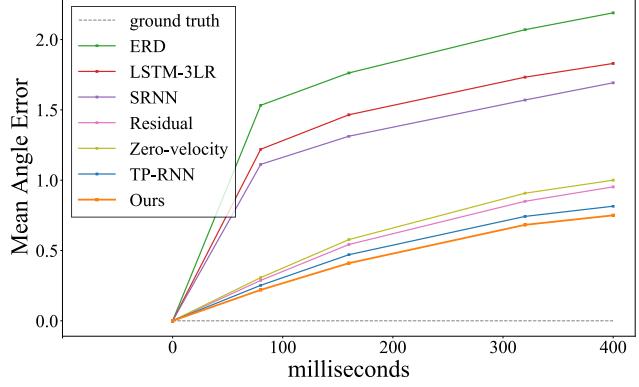


Figure 5: Plot of average mean angle error for different pose prediction models over *walking*, *eating*, *smoking* and *discussion* in the short-term prediction experiment.

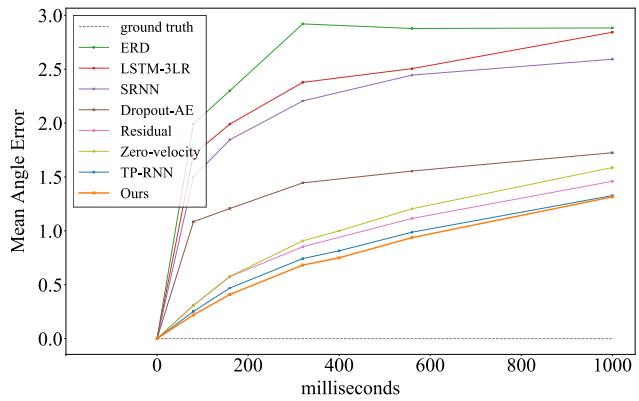


Figure 6: Plot of average mean angle error for different pose prediction models over *walking*, *eating*, *smoking* and *discussion* in the long-term prediction experiment.

parameter from Stage 1. Our WGAIL-div algorithm for human pose prediction is listed in Algorithm 1.

5. Experiments

To evaluate the performance of our proposed imitation learning approach on challenging real human pose prediction tasks and to have a thorough comparison with previous works, we run exhaustive experiments to test our algorithm using the popular Human 3.6M dataset [18] and compare our results with previous benchmarks.

The Human 3.6M dataset: The Human 3.6M dataset [18] is currently one of the largest publicly available motion capture dataset, and has been used by many previous papers on human pose prediction. This dataset includes video sequences recorded from 7 different actors performing 15 different categories of human activities, with each actor performing each activity in two different trials. The videos are recorded in 50Hz, and for a fair comparison, we follow previous papers [20, 29] to downsample the pose se-

Table 1: Mean Angle Error for the activities *walking*, *eating*, *smoking*, and *discussion* in the short-term prediction experiment. The best result in each column is highlighted with boldface.

milliseconds	Walking				Eating				Smoking				Discussion			
	80	160	320	400	80	160	320	400	80	160	320	400	80	160	320	400
ERD [10]	0.93	1.18	1.59	1.78	1.27	1.45	1.66	1.80	1.66	1.95	2.35	2.42	2.27	2.47	2.68	2.76
LSTM-3LR [10]	0.77	1.00	1.29	1.47	0.89	1.09	1.35	1.46	1.34	1.65	2.04	2.16	1.88	2.12	2.25	2.23
SRNN [20]	0.81	0.94	1.16	1.30	0.97	1.14	1.35	1.46	1.45	1.68	1.94	2.08	1.22	1.49	1.83	1.93
Residual [29]	0.28	0.49	0.72	0.81	0.23	0.39	0.62	0.76	0.33	0.61	1.05	1.15	0.31	0.68	1.01	1.09
Zero-velocity [29]	0.39	0.68	0.99	1.15	0.27	0.48	0.73	0.86	0.26	0.48	0.97	0.95	0.31	0.67	0.94	1.04
TP-RNN [8]	0.25	0.41	0.58	0.65	0.20	0.33	0.53	0.67	0.26	0.47	0.88	0.90	0.30	0.66	0.96	1.04
Ours	0.21	0.34	0.53	0.59	0.17	0.30	0.52	0.65	0.23	0.44	0.87	0.85	0.23	0.56	0.82	0.91

Table 2: Mean Angle Error for the activities *walking*, *eating*, *smoking*, and *discussion* in the long-term prediction experiment. The best result in each column is highlighted with boldface.

milliseconds	Walking					Eating					Smoking					Discussion				
	80	160	320	560	1000	80	160	320	560	1000	80	160	320	560	1000	80	160	320	560	1000
ERD [10]	1.30	1.56	1.84	2.00	2.38	1.66	1.93	2.88	2.36	2.41	2.34	2.74	3.73	3.68	3.82	2.67	2.97	3.23	3.47	2.92
LSTM-3LR [10]	1.18	1.50	1.67	1.81	2.20	1.36	1.79	2.29	2.49	2.82	2.05	2.34	3.10	3.24	3.42	2.25	2.33	2.45	2.48	2.93
SRNN [20]	1.08	1.34	1.60	1.90	2.13	1.35	1.71	2.12	2.28	2.58	1.90	2.30	2.90	3.21	3.23	1.67	2.03	2.20	2.39	2.43
Dropout-AE [12]	1.00	1.11	1.39	1.55	1.39	1.31	1.49	1.86	1.76	2.01	0.92	1.03	1.15	1.38	1.77	1.11	1.20	1.38	1.53	1.73
Residual [29]	0.32	0.54	0.72	0.86	0.96	0.25	0.42	0.64	0.94	1.30	0.33	0.60	1.01	1.23	1.83	0.34	0.74	1.04	1.43	1.75
Zero-velocity [29]	0.39	0.68	0.99	1.35	1.32	0.27	0.48	0.73	1.04	1.38	0.26	0.48	0.97	1.02	1.69	0.31	0.67	0.94	1.41	1.96
TP-RNN [8]	0.25	0.41	0.58	0.74	0.77	0.20	0.33	0.53	0.84	1.14	0.26	0.48	0.88	0.98	1.66	0.30	0.66	0.98	1.39	1.74
Ours	0.21	0.34	0.53	0.67	0.69	0.17	0.30	0.52	0.79	1.13	0.23	0.44	0.86	0.95	1.63	0.27	0.56	0.82	1.34	1.81

quence by 2. Each pose in this dataset is represented as an exponential map representation of 32 human joints in 3D, and after preprocessing of global translation and rotation as described in [8, 10, 20, 29], we adopt the following evaluation methods for our experiments: during both training and testing, we feed 2000ms (50 frames) of past pose vector sequence into the learning system, and the goal is to predict the next 1000ms (25 frames) of future pose vector sequence. We measure the Euclidean distance between predicted poses and ground truth poses in angle space as the evaluation metric of prediction error, and we report the average prediction error over 6 different timescales: 80, 160, 320, 400, 560, and 1000ms. Following previous works, we use Subjects 1, 6, 7, 8, 9, 11 as the training dataset and use Subject 5 as the testing dataset.

Baselines: We compare our experimental results with the following recent state-of-the-art methods for human pose prediction on the Human 3.6M dataset: ERD [10], LSTM-3LR [10], SRNN [20], Dropout-AutoEncoder [12], Residual [29], Zero-velocity [8, 29] and TP-RNN [8].

5.1. Results

The performance results published by previous papers are reported on slightly different prediction timescales (some report on short-term predictions over 400ms and the others report on long-term predictions over 1000ms) and activity categories (some only report on 4 different activities). To make a thorough comparison with all these previous methods, we report the prediction results of our model in Tables 1, 2 and 3, and plot the mean angle error curves in Figures 5 and 6.

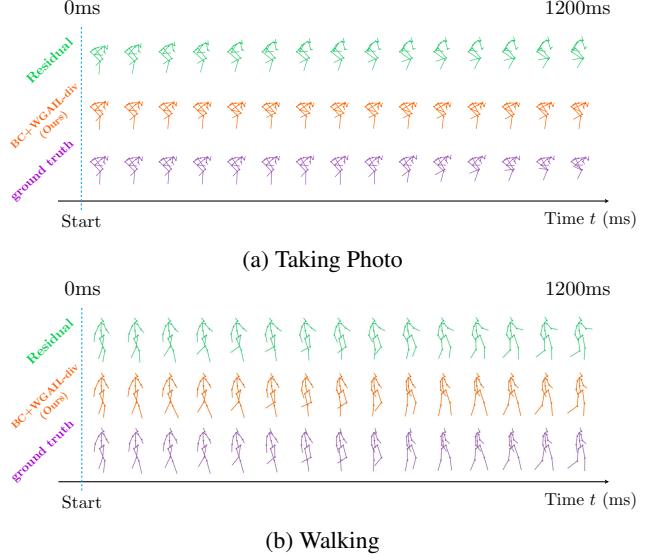


Figure 7: Visualization of long-term human pose prediction results for *taking photo* (top) and *walking* (bottom). The purple poses are ground truth, the green poses are predicted by the *Residual* [29] model and the orange poses are ours.

From the experimental results, we see that the prediction performance of our proposed imitation learning approach (BC+WGAIL-div) surpasses all the baseline models by significant margins on almost all 15 human activity categories across all different prediction timescales. Our prediction results set the new state-of-the-art performance for the task of predicting human motion on the Human 3.6M dataset over both short-term and long-term predictions.

Table 3: Mean Angle Error for the remaining 11 actions in Human 3.6M in the long-term prediction experiment. The best result in each column is highlighted with boldface.

	Purchases						Sitting						Sitting down						Taking photo					
millisec	80	160	320	400	560	1000	80	160	320	400	560	1000	80	160	320	400	560	1000	80	160	320	400	560	1000
Residual	0.60	0.86	1.24	1.30	1.58	2.26	0.44	0.74	1.19	1.40	1.57	2.03	0.51	0.93	1.44	1.65	1.94	2.55	0.33	0.65	0.97	1.09	1.19	1.47
TP-RNN	0.59	0.82	1.12	1.18	1.52	2.28	0.41	0.66	1.07	1.22	1.35	1.74	0.41	0.79	1.13	1.27	1.47	1.93	0.26	0.51	0.80	0.95	1.08	1.35
Ours	0.54	0.78	1.07	1.14	1.46	2.23	0.29	0.48	0.87	1.04	1.21	1.58	0.34	0.68	1.01	1.14	1.34	1.78	0.17	0.37	0.60	0.72	0.84	1.06
	Directions						Greeting						Talking on the phone						Posing					
millisec	80	160	320	400	560	1000	80	160	320	400	560	1000	80	160	320	400	560	1000	80	160	320	400	560	1000
Residual	0.44	0.69	0.83	0.94	1.03	1.49	0.53	0.88	1.29	1.45	1.72	1.89	0.61	1.12	1.57	1.74	1.59	1.92	0.47	0.87	1.49	1.76	1.96	2.35
TP-RNN	0.38	0.59	0.75	0.83	0.95	1.38	0.51	0.86	1.27	1.44	1.72	1.81	0.57	1.08	1.44	1.59	1.47	1.68	0.42	0.76	1.29	1.54	1.75	2.47
Ours	0.27	0.46	0.81	0.89	0.95	1.41	0.43	0.75	1.17	1.33	1.62	1.72	0.54	1.05	1.40	1.56	1.52	1.67	0.27	0.55	1.16	1.41	1.83	2.69
	Waiting						Walking dog						Walking together						Average over all 15 actions					
millisec	80	160	320	400	560	1000	80	160	320	400	560	1000	80	160	320	400	560	1000	80	160	320	400	560	1000
Residual	0.34	0.65	1.09	1.28	1.61	2.27	0.56	0.95	1.28	1.39	1.68	1.92	0.31	0.61	0.84	0.89	1.00	1.43	0.43	0.75	1.11	1.24	1.42	1.83
TP-RNN	0.30	0.60	1.09	1.31	1.71	2.46	0.53	0.93	1.24	1.38	1.73	1.98	0.23	0.47	0.67	0.71	0.78	1.28	0.37	0.66	0.99	1.11	1.30	1.71
Ours	0.25	0.53	0.96	1.19	1.59	2.39	0.46	0.80	1.12	1.31	1.65	1.86	0.19	0.41	0.61	0.64	0.67	1.15	0.31	0.57	0.90	1.02	1.23	1.65

Table 4: Ablation Study: Mean Angle Error of Behavioral Cloning (BC) alone, WGAIL-div alone, and WGAIL-div with Behavioral Cloning as pre-training (BC+WGAIL-div) averaged over the four activities *walking*, *eating*, *smoking*, and *discussion* in the human pose prediction experiment on the Human 3.6M dataset. The best result in each column is highlighted with boldface.

millisec	80	160	320	400	560	1000
BC	0.23	0.43	0.72	0.78	0.96	1.38
WGAIL-div	0.23	0.44	0.74	0.80	0.98	1.33
BC+WGAIL-div	0.21	0.41	0.69	0.75	0.94	1.32

We also plot visualization of the long-term human pose prediction results in Figure 7 for two representative human activities: *taking photo* and *walking*. From the visualization we can see that the predictions made by our imitation learning algorithm look very similar to the ground truth and are much better and more natural than the predictions made by the baseline model *Residual* [29]. See Appendix A.3 for more visualization of our prediction results.

Training Speed: During our experiments, we observed that the training speed of our proposed imitation learning method is much faster than previous methods. In our experiments using 4 NVIDIA Titan GPUs, on average, our BC + WGAIL-div algorithm finishes training within 20 minutes, while the baseline models *Residual* and *TP-RNN* finish training in more than 8 hours. There is at least a $\times 24$ speedup on training using our proposed method.

5.2. Ablation Study

Our proposed imitation learning algorithm is composed of two components: (1) a BC component (Algorithm 2 in Appendix A.1) as pretraining; and (2) a WGAIL-div component (Algorithm 1). In our ablation study, in order to test the importance of each component, we train our human pose prediction agent on the Human 3.6M dataset [19] using: (a)

BC alone; (b) WGAIL-div alone; and (c) WGAIL-div with BC as pretraining. The results of our ablation study experiments are summarized in Table 4. As we can see from Table 4, removing either BC or WGAIL-div from our imitation learning algorithm would worsen the prediction performance. Therefore, our ablation study shows that both components of our imitation learning algorithm are necessary in order to achieve optimal performance.

5.3. Discussion

In this work, all the pose predictions are solely based on past pose observations. In our future work, we plan to extend our work to tackle more sophisticated scenarios of human motion prediction. First, we plan to extend our WGAIL-div framework to multi-agent WGAIL-div in order to take into account other humans in the surroundings when performing group human motion prediction. Second, we plan to introduce latent variables into WGAIL-div in order to effectively model other factors such as environments, objects, activities, and intentions that might also be involved in human motion prediction.

6. Conclusion

In this work, we proposed a new reinforcement learning formulation of the human pose prediction problem, and developed an imitation learning algorithm for pose prediction under this new formulation. Our experiments showed that our proposed method can effectively learn to make accurate human pose predictions over both short terms and long terms with very fast training speed. Our work also has great potential to be generalized to other tasks of sequential modeling and time-series predictions in computer vision and machine learning, which will also be the direction of our future work.

Acknowledgments: We would like to thank Mindtree and Panasonic for their support.

References

- [1] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Monocular 3d pose estimation and tracking by detection. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 623–630. IEEE, 2010. 2
- [2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017. 2, 4
- [3] Mohammad M Arzani, Mahmood Fathy, Hamid Aghajan, Ahmad A Azirani, Kaamran Raahemifar, and Ehsan Adeli. Structured prediction with short/long-range dependencies for human activity recognition from depth skeleton data. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 560–567. IEEE, 2017. 2
- [4] Michael Bain and Claude Sammut. A framework for behavioural cloning. *Machine intelligence*, 15(15):103, 1999. 2, 3
- [5] Emad Barsoum, John Kender, and Zicheng Liu. Hp-gan: Probabilistic 3d human motion prediction via gan. *arXiv preprint arXiv:1711.09561*, 2017. 2
- [6] Chien-Yi Chang, De-An Huang, Danfei Xu, Ehsan Adeli, Li Fei-Fei, and Juan Carlos Niebles. Procedure planning in instructional videos. *arXiv preprint arXiv:1907.01172*, 2019.
- [7] Yu-Wei Chao, Jimei Yang, Brian Price, Scott Cohen, and Jia Deng. Forecasting human dynamics from static images. In *CVPR*, 2017. 2
- [8] Hsu-Kuang Chiu, Ehsan Adeli, Borui Wang, De-An Huang, and Juan Carlos Niebles. Action-agnostic human pose forecasting. *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 1, 2, 3, 7
- [9] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014. 5
- [10] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 4346–4354. IEEE, 2015. 1, 2, 7
- [11] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges. Learning human motion models for long-term predictions. *arXiv preprint arXiv:1704.02827*, 2017. 2
- [12] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges. Learning human motion models for long-term predictions. *arXiv preprint arXiv:1704.02827*, 2017. 7
- [13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 3
- [14] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777, 2017. 4
- [15] Ankur Gupta, Julieta Martinez, James J Little, and Robert J Woodham. 3d pose from motion for cross-view action recognition via non-linear circulant temporal encoding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2601–2608, 2014. 1
- [16] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in neural information processing systems*, pages 4565–4573, 2016. 1, 2, 3, 4
- [17] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 5
- [18] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2014. 2, 6
- [19] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2014. 8
- [20] Ashesh Jain, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. Structural-rnn: Deep learning on spatio-temporal graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5308–5317, 2016. 1, 2, 3, 6, 7
- [21] Hema Koppula and Ashutosh Saxena. Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation. In *International conference on machine learning*, pages 792–800, 2013. 1
- [22] Lucas Kovar, Michael Gleicher, and Frédéric Pighin. Motion graphs. In *ACM SIGGRAPH 2008 classes*, page 51. ACM, 2008. 1
- [23] Thanard Kurutach, Aviv Tamar, Ge Yang, Stuart J Russell, and Pieter Abbeel. Learning plannable representations with causal infogan. In *Advances in Neural Information Processing Systems*, pages 8733–8744, 2018.
- [24] Yunzhu Li, Jiaming Song, and Stefano Ermon. Infogail: Interpretable imitation learning from visual demonstrations. In *Advances in Neural Information Processing Systems*, pages 3812–3822, 2017.
- [25] 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. 2
- [26] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015. 1, 2, 5
- [27] Pauline Luc, Natalia Neverova, Camille Couprie, Jakob Verbeek, and Yann LeCun. Predicting deeper into the future of semantic segmentation. In *of: ICCV 2017-International Conference on Computer Vision*, page 10, 2017. 2
- [28] Reza Mahjourian, Martin Wicke, and Anelia Angelova. Geometry-based next frame prediction from monocular video. In *Intelligent Vehicles Symposium (IV), 2017 IEEE*, pages 1700–1707. IEEE, 2017. 2
- [29] Julieta Martinez, Michael J Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *CVPR*, 2017. 1, 2, 3, 5, 6, 7, 8
- [30] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *ACM Transactions on Graphics (TOG)*, 36(4):44, 2017.

- 2**

[31] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. **1, 2**

[32] Andrew Y Ng, Stuart J Russell, et al. Algorithms for inverse reinforcement learning. In *Icml*, volume 1, page 2, 2000. **2, 3**

[33] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions on Graphics (TOG)*, 37(4):143, 2018. **2**

[34] Nicholas Rhinehart, Kris M Kitani, and Paul Vernaza. R2p2: A reparameterized pushforward policy for diverse, precise generative path forecasting. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 772–788, 2018. **2**

[35] Gr  gory Rogez, Jonathan Rihan, Srikumar Ramalingam, Carlos Orrite, and Philip HS Torr. Randomized trees for human pose detection. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. **2**

[36] St  phane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635, 2011. **1, 2, 3**

[37] Claude Sammut. Behavioral cloning. *Encyclopedia of Machine Learning*, pages 93–97, 2010. **3**

[38] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897, 2015. **1, 4**

[39] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. **1, 4**

[40] David Silver, Guy Lever, Nicolas Heess, Thomas Degrif, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *ICML*, 2014. **5**

[41] Khurram Soomro, Haroon Idrees, and Mubarak Shah. Online localization and prediction of actions and interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. **2**

[42] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014. **2, 5**

[43] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018. **2, 3**

[44] Lei Tai, Jingwei Zhang, Ming Liu, and Wolfram Burgard. Socially compliant navigation through raw depth inputs with generative adversarial imitation learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1111–1117. IEEE, 2018. **2**

[45] Nikolaus F Troje. Decomposing biological motion: A framework for analysis and synthesis of human gait patterns. *Journal of vision*, 2(5):2–2, 2002. **1**

[46] Raquel Urtasun, David J Fleet, and Pascal Fua. 3d people tracking with gaussian process dynamical models. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 238–245. IEEE, 2006. **1**

[47] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. In *Advances In Neural Information Processing Systems*, pages 613–621, 2016. **2**

[48] Jacob Walker, Carl Doersch, Abhinav Gupta, and Martial Hebert. An uncertain future: Forecasting from static images using variational autoencoders. In *European Conference on Computer Vision*, pages 835–851. Springer, 2016. **2**

[49] Jacob Walker, Abhinav Gupta, and Martial Hebert. Dense optical flow prediction from a static image. In *Computer Vision (ICCV), 2015 IEEE International Conference on*, pages 2443–2451. IEEE, 2015. **2**

[50] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert. The pose knows: Video forecasting by generating pose futures. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3352–3361. IEEE, 2017. **2**

[51] Jack M Wang, David J Fleet, and Aaron Hertzmann. Gaussian process dynamical models for human motion. *IEEE transactions on pattern analysis and machine intelligence*, 30(2):283–298, 2008. **2**

[52] Di Wu and Ling Shao. Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 724–731, 2014. **2**

[53] Jiqing Wu, Zhiwu Huang, Janine Thoma, Dinesh Acharya, and Luc Van Gool. Wasserstein divergence for gans. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 653–668, 2018. **4**

[54] Xinchen Yan, Akash Rastogi, Ruben Villegas, Kalyan Sunkavalli, Eli Shechtman, Sunil Hadap, Ersin Yumer, and Honglak Lee. Mt-vae: Learning motion transformations to generate multimodal human dynamics. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 265–281, 2018. **2**

[55] Kuo-Hao Zeng, William B Shen, De-An Huang, Min Sun, and Juan Carlos Niebles. Visual forecasting by imitating dynamics in natural sequences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2999–3008, 2017. **1**

[56] Yi Zhou, Zimo Li, Shuangjiu Xiao, Chong He, Zeng Huang, and Hao Li. Auto-conditioned recurrent networks for extended complex human motion synthesis. In *ICLR*, 2018. **2**