# ELMA: Energy-Based Learning for Multi-Agent Activity Forecasting

**Yu-Ke Li[1], Pin Wang[1], Li-Xiong Chen[2], Zheng Wang[3]*, Ching-Yao Chan[1]***

[1]California PATH, UC Berkeley,
[2]Department of Engineering Science, University of Oxford,
[3]School of Computer Science, Wuhan University,

## Abstract

This paper describes an energy-based learning method that predicts the activities of multiple agents simultaneously. It aims to forecast both upcoming actions and paths of all agents in a scene based on their past activities, which can be jointly formulated by a probabilistic model over time. Learning this model is challenging because: 1) it has a large number of time-dependent variables that must scale with the forecast horizon and the number of agents; 2) distribution functions have to contain multiple modes in order to capture the spatio-temporal complexities of each agent's activities. To address these challenges, we put forth a novel Energy-based Learning approach for Multi-Agent activity forecasting (ELMA) to estimate this complex model via maximum log-likelihood estimation. Specifically, by sampling from a sequence of factorized marginalized multi-modal distributions, ELMA generates the possible future actions efficiently. Moreover, by graph-based representations, ELMA also explicitly resolves the spatio-temporal dependencies of all agents' activities in a single pass. Our experiments on two large-scale datasets prove that ELMA outperforms recent leading studies by an obvious margin.

## Introduction

Being able to predict an agent's upcoming behavior is important for intelligent systems to understand the physical world. For instance, autonomous vehicles have to share the road reasonably with nearby pedestrians in real-time under all circumstances. This means the system must be able to foresee which direction an agent is moving to (*i.e.*, path) and what the agent is going to do (*i.e.*, action) as Fig. 1 depicts.

Existing work performs path prediction and action prediction in an agent-independent manner. Be the model a variational autoencoder (VAE) (Walker et al. 2017), generative adversarial networks (GAN) (Gupta et al. 2018; Chen, Bao, and Kong 2020) or GLOW (Guan et al. 2020), variables are partitioned into independent subsets with an assumption of a relatively simple distribution. Specifically, for conditional distributions over which future variables are dependent on past observations, a set of additional latent variables that explain the generation of multi-model joint distribution must rely on good guesses. Hence, modeling complex multi-model distributions accurately is essential for performance.

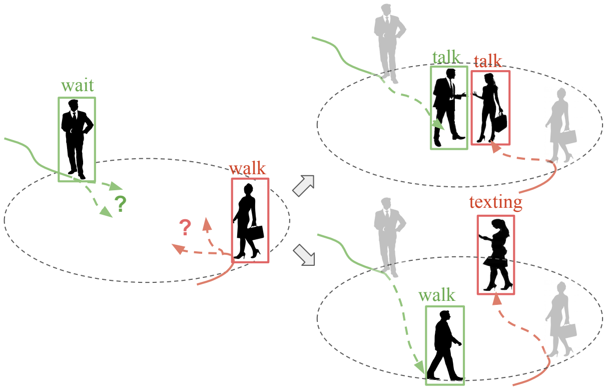

Figure 1: An example of multi-agent activity forecasting. Given the observed activities (in solid bounding boxes and lines) of two agents, we want to forecast the possibility of their future actions (in dashed bounding boxes) and locations (in dashed lines). Two outcomes of multi-agent interactions are displayed, and it can be readily seen that one agent's activities also depend on the activities of the other.

We believe, however, that actions and paths of all agents are preferably addressed in a unified framework derived by a novel concept that we term as multi-agent activity forecasting. We introduce this concept because apparent across-agent dependencies exist. On the one hand, an agent has to adjust his/her reasonable route instantaneously depending on the past actions taken by others in the surrounding; on the other hand, new routes may impose new constraints on all other agents' next moves. Given that an agent's activities has very high variance in nature, the underlying joint distribution of both actions and paths is highly complex and likely to be multi-modal. In other word, existing methods, cannot be applied to address this new task directly due to the escalated complexity of the underlying joint distribution.

In order to make effective predictions out of complex distributions, we introduce an Energy-based Learning solution for Multi-Agent activity forecasting, ELMA, that approximates complex distributions implicitly with learnable energy functions (LeCun et al. 2006). In contrast to VAE and GAN, ELMA covers multimodal distributions without complying

*These authors contributed equally.

with a simple and parametric latent distribution multi-step training. Instead, we need to design an energy function to suit the needs for activity forecasting. In particular, in a multi-agent scenario, the activities of co-occurring agents exhibit strong dependencies not only spatially but also temporally. For Example, an agent may slow down to avoid collision if another agent suddenly gets in his/her way. Hence, in contrast to (Liang et al. 2019; Malla, Dariush, and Choi 2020), ELMA builds a sequence of graph neural networks (GNN) to analyze these spatio-temporal dependencies in a unified framework (Battaglia et al. 2018; Xu et al. 2019), where each node in the graph represents a specific agent, and activities are encoded in edge weights. Furthermore, ELMA is trained via maximum likelihood estimation, the solution to which is obtained using contrastive divergence. Additionally, ELMA forecasts future activities through iterative sampling. Since the resultant activity prediction in each roll-out associates with a relevant mode, ELMA is able to capture the uncertain nature of the future.

In summary,we introduce a novel energy-based learning solution for activity forecasting. Our main contributions are as follows:

- We extend the concept of activity forecasting to a multi-agent scenario, with which a unified task that simultaneously predicts path and action is defined.
- We design a novel energy-based learner, ELMA, to tackle this task. To the best of our knowledge, it is the first to apply energy-based learners for multi-agent activity forecasting.
- We describe a sampling procedure integrating GNN and contrastive divergence that learns the model effectively.
- We evaluate our method using challenging large-scale video datasets. The results show that our solution outperforms SOTA methods by an obvious margin.

The remainder of this paper is organized as follows: related works are discussed in Section . Section  describes the design of ELMA and explains how it is learned via MLE. Section  provides our analysis on the experimental results. Section  concludes this paper.

## Related Work

The relevant literature has accumulated several efforts to tackle down the challenges of future forecasting. A theme pertinent to our topic is to predict the action features (Koppula and Saxena 2016; Vondrick, Pirsiavash, and Torralba 2016; Walker et al. 2017; Zeng et al. 2017; Li 2018; He et al. 2018; Yuan and Kitani 2020a; Sun et al. 2019; Zeng et al. 2020; Epstein, Chen, and Vondrick 2020; Li, Wang, and Chan 2021; Epstein, Chen, and Vondrick 2021; Li et al. 2021; Xu et al. 2021). Most of these frameworks obtain the upcoming action features based on a deep neural network (Goodfellow, Bengio, and Courville 2016). Another line of studies attempted to forecast the agent path features (Alahi et al. 2016; Li 2017, 2018; Luo, Yang, and Urtasun 2018; Gupta et al. 2018; Li 2019; Tang and Salakhutdinov 2019; Ivanovic and Pavone 2019; Yang et al. 2020; Yuan and Kitani 2020b; Zhang et al. 2020; Sun, Jiang, and Lu 2020; Malla, Dariush, and Choi 2020; Salzmann et al. 2020; Mangalam et al. 2020).

In this paper, rather than solely forecasting action features or path features, we aim to foresee them together – the activities in other words – for each individual agent in the scene. The authors of (Guan et al. 2020) predicted the activity under a one-person scenario, which leaves open the question of its applicability to the multi-agent scenes. (Chen, Bao, and Kong 2020) presented an upcoming group-level action. Nevertheless, this method overlooked the fact that individual action may deviate from the entire group. To the best of our knowledge, the closest work to our study in the objective aspect is Next (Liang et al. 2019). It designs a multi-task pipeline with a two-stream network to predict agent action nd path features, respectively. The main shortcoming of Next is that the future's uncertainty is ignored. This lead to that Next cannot predict the multiple plausible futures, which is one key requirement of multi-agent activity forecasting. Another denominator is that Next proceeds without modeling the spatiotemporal dependencies among agents. Such a deficiency makes that Next treats each individual independently, and fails to meet the definition of multi-agent activity forecasting (see details in section ).

ELMA has the following vital distinctions from previous works: 1. Concerning the studies on only forecasting action or path features, we sufficiently consider the mutual dependence across these two factors; 2. With respect to Next, ELMA is able to cover the multimodal futures; 3. Unlike (Liang et al. 2019; Guan et al. 2020), ELMA suits the multi-agent scene better given that we can jointly reason the spatiotemporal dependencies.

Recently, the authors of (Xie, Zhu, and Nian Wu 2017; Gao et al. 2018; Du, Li, and Mordatch 2020; Ho, Jain, and Abbeel 2020; Gao et al. 2021; Suhail et al. 2021) have attempted to leverage energy-based model to solve some vision tasks, such as image synthesizing and object detection. Our work conducts the *first* research exploring multi-agent activity forecasting based upon the energy functions.

## Energy-Based Learning for Multi-Agent Activity Forecasting

Formally, given the observed action features and the path features $\mathrm{h}_t = \{a_1, a_2, ...a_t, x_1, x_2...x_t\}$ , ELAM predicts $\{a_{t+1}, a_{t+2}, ...a_{t+K}\}$ together with $\{x_{t+1}, x_{t+2}, ...x_{t+K}\}$ using conditional distribution:

$$p(a_{t+K}, x_{t+K}, \ldots, a_{t+1}, x_{t+1}|\mathrm{h}_t) \tag{1}$$

The spatio-temporal dependencies of both types of features among all agents in the scene are encoded by a series of fully connected graphs and learned by Graph Neural Networks (GNNs). In particular, these GNNs representing the joint distribution of all features over time are trained in an energy-based learning framework, and their samples generated using Markov Chain Monte Carlo with Langevin dynamics.

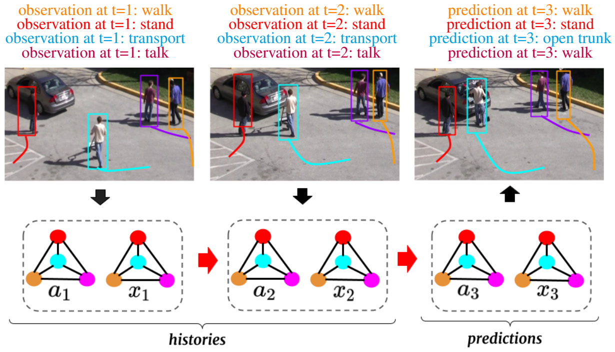

Figure 2: An example of graph representation of ELMA. Each node represents an agent and the edge connecting two nodes represents their pair-wise interaction. $a_1$ and $x_1$ denote actions and paths observed at time instant $t = 1$, respectively. A graph instance describes the scene at a specific time instant. The graph structure encodes the spatial information of the scene, and a sequence of the graphs encodes its temporal information.In this example, ELMA predicts the activities of four agents at $t = 3$ based on the observations made at $t = 1$ and $t = 2$.

## Problem Formulation

The one-step forward conditional distribution in Eq. 1 can be further factorized as follows:

$$\begin{aligned} &p(x_{t+2}, a_{t+2}, x_{t+1}, a_{t+1}|\mathrm{h}_t) \\ =&p(x_{t+2}, a_{t+2}|\mathrm{h}_{t+1})p(x_{t+1}, a_{t+1}|\mathrm{h}_t) \\ =&p(x_{t+2}|a_{t+2}, \mathrm{h}_{t+1})p(a_{t+2}|\mathrm{h}_{t+1})p(x_{t+1}, a_{t+1}|\mathrm{h}_t), \end{aligned} \tag{2}$$

where $p(\mathrm{h}_{t+1}) = p(x_{t+1}, a_{t+1}, \mathrm{h}_t)$ holds for density functions involving $t$ and $t+1$. Accordingly, Eq. 1 can be rewritten as:

$$\prod_{k=0}^{K} p(x_{t+k}|a_{t+k}, \mathrm{h}_{t+k-1}) \prod_{k=0}^{K} p(a_{t+k}|\mathrm{h}_{t+k-1}), \tag{3}$$

which is initialized by $p(a_0|h_0) = p(a_0)$. Fig. 3 visualizes this dependency and our goal is to determine a this conditional joint distribution through Maximum Likelihood Estimation (MLE).

However, direct optimization over the likelihood function of Eq.2 is essentially intractable (Goodfellow et al. 2014) due to its structural complexity. Hence, alternatively, we propose to approximate it using Boltzmann distributions in terms of model parameters $\theta$. So, Eq.3 can be further unfolded as a product of:

$$\frac{1}{Z_x(\theta)} exp\{\sum_k E_x(\mathrm{h}_{t+k-1}, a_{t+k}, x_{t+k}; \theta)\} \tag{4}$$

and

$$\frac{1}{Z_a(\theta)} exp\{\sum_k E_a(\mathrm{h}_{t+k-1}, a_{t+k}; \theta)\} \tag{5}$$

where $Z_x(\theta)$ and $Z_a(\theta)$ are the two partition functions parameterized separately. The corresponding log-likelihood

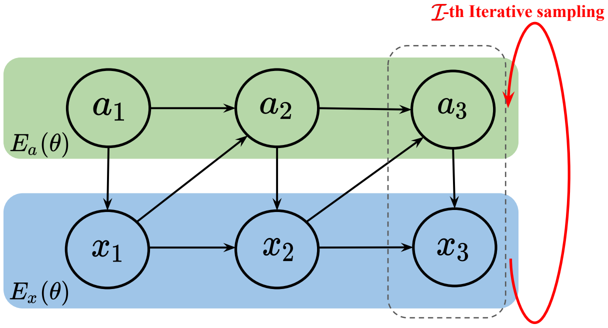

Figure 3: A schematic diagram that illustrates the operation described in Eq. 12. Following the scenario in Figure 2, ELMA realizes a sample future sequence at each iteration. The outcomes from the final iteration are treated as $\{a_1, a_2, a_3\}$ and $\{x_1, x_2, x_3\}$ for Eq. 11.

function can be written as:

$$\begin{aligned} l(\theta) =& \sum_k E_x(\mathrm{h}_{t+k-1}, a_{t+k}, x_{t+k}; \theta) - logZ_x(\theta) \\ &+ \sum_k E_a(\mathrm{h}_{t+k-1}, a_{t+k}; \theta) - logZ_a(\theta) \\ =& \sum_k (E_x(\theta) + E_a(\theta)) - logZ_x(\theta) - logZ_a(\theta), \end{aligned} \tag{6}$$

where the energy functions $E_x(\theta)$ and $E_a(\theta)$ are parameterized and learned through GNNs.

## GNN-based Energy Functions

The graph representations enable us to infer the spatio-temporal dependencies via relational reasoning (Battaglia et al. 2018; Xu et al. 2019). We deploy two separate garph neural networks for this purpose:

$$\begin{aligned} H_t^a &= \mathrm{GNN}(a_t, H_{t-1}^a; \theta) \\ H_t^x &= \mathrm{GNN}(a_t, x_t, H_{t-1}^x; \theta) \end{aligned} \tag{7}$$

where $\mathrm{GNN}(\cdot)$ updates and concatenates its hidden state variable $H_t$ that encodes all historical information about $h_t$. In particular, Eq. 7 is implemented using AGC-LSTM (Si et al. 2019) which handles both spatial and temporal variations are handled simultaneously. At each time instant, the spatial component is processed via graph convolutions, whereas the temporal component is addressed by the LSTM.

Specifically, $a_t$,$x_t$,$H_{t-1}^x$ and $H_{t-1}^a$ are encoded into a series of fully connected graph structural data with a pair of functions to evaluate $E_x^t(\theta)$ and $E_a^t(\theta)$ separately. For instance, to evaluate $E_a^t(\theta)$, function $f_{edge}^a$, implemented by a pre-trained bi-linear layer, is deployed to process edge attributes, and its output is taken by $f_{node}^a$, that processes node attributes in the form a pre-trained multi-layer perceptrons (MLP). For each edge linking agent $i$ and $j$ we have:

$$\{e_t^a\}_{i,j} = f_{edge}^a(\{a_t\}_i, \{a_t\}_j), \tag{8}$$

and the attributes $\{v_t^a\}_i$ of node $i$ at $t$ can be formulated as:

$$\{v_t^a\}_i = \mathrm{ELU}\big(f_{node}^a\big(\frac{1}{\mid N_i \mid} \sum_{j \in N_i} \{e_t^a\}_{i,j}\big)\big) \tag{9}$$

where $N_i$ indicates the neighbors of agent $i$ in the graph. Averaging the edge attributes of node $i$ ensures that $v_t^a$ is permutation-invariant. The evaluation of $E_x^t(\theta)$ can be carried out in a similar manner with $\{e_t^x\}_{i,j}$ and $\{v_t^x\}_i$ correspondingly.

Furthermore, $H_{t+k}$ is fed to a MLP to obtain the final energy measure:

$$\begin{aligned} E_a^k(\theta) &= \| \sigma(W_{t+k,a}^T \cdot H_{t+k}^a) \|_1 \\ E_x^k(\theta) &= \| \sigma(W_{t+k,x}^T \cdot H_{t+k}^x) \|_1 \end{aligned} \tag{10}$$

Two stacked AG-CLSTM layers with 512 channels are leveraged to calculate Eq. 10. In practice, we consider building our graph with 50 nodes for the experiments. The virtual nodes with zero-padded attributes are used if the labeled persons are less than 50. Statistically, we found that the maximum numbers of labeled persons in all the video sequences are less than 50 agents for the datasets exploited to conduct experiments. Both $W_{t+k,a}^T$ and $W_{t+k,x}^T$ transform the 50×512 dimensional vector to a positive scalar.

## Training

Eq. 6 can be optimized in contrastive divergence manner (LeCun et al. 2006; Hinton 2002). Namely, we update the gradients $E_\theta$ by:

$$\nabla l(\theta_x, \theta_a) = \sum_k \left(\frac{\partial E_x(\theta)}{\partial \theta} + \frac{\partial E_a(\theta)}{\partial \theta}\right) \tag{11}$$

where

$$\begin{aligned} \frac{\partial E_x(\theta)}{\partial \theta} =& \mathbb{E}\left(\frac{\partial E_x(\mathrm{h}'_{t+k-1}, a'_{t+k}, x'_{t+k}; \theta)}{\partial \theta}\right) - \\ & \mathbb{E}\left(\frac{\partial E_x(\mathrm{h}_{t+k-1}, a_{t+k}, x_{t+k}; \theta)}{\partial \theta}\right) \end{aligned}$$

where $\mathrm{h}'_{t+k-1}$,$a'_{t+k}$, and $x_{t+k}$ denote the ground truth included in the training data and $\mathrm{h}_{t+k-1}$,$a_{t+k}$, and$x_{t+k}$ are the samples drawn from the distribution with $\theta$ of current iteration. Specifically, ELMA resorts to the gradient-based iterative MCMC method – Langevin dynamics (Welling and Teh 2011; Song and Ermon 2019) to simulate both forthcoming action features and paths. To sample an future action sequence $a$=$\{a_{t+1}, \dots, a_{t+K}\}$ and path sequence $x$=$\{x_{t+1}, \dots, x_{t+K}\}$ at iteration $\mathcal{I}$, we perform the following:

$$\begin{aligned} a^{\mathcal{I}} &= a^{\mathcal{I}-1} + \delta \epsilon^{\mathcal{I}} - \frac{\delta^2}{2} \nabla_{a^{\mathcal{I}}} E_a(\theta_a) \\ x^{\mathcal{I}} &= x^{\mathcal{I}-1} + \delta \Psi^{\mathcal{I}} - \frac{\delta^2}{2} \nabla_{x_t^{\mathcal{I}}} E_x(\theta_x) \end{aligned} \tag{12}$$

where $\delta$ denotes the step size of Langevin dynamics, and $\epsilon$ and $\Psi$ denote the additive noise that are independently sampled from the standard Gaussian distribution. After the final iteration, $a_{t+k}^{\mathcal{I}}$ and $x_{t+k}^{\mathcal{I}}$ are taken by Eq. 11 as an operation of contrastive divergence (Hinton 2002) to update $\theta$. Fig. 3 displays the pipeline of our energy functions.

It can be readily observed from Eq. 12 that predictions are drawn using the energies $E_\theta$ with their gradients being evaluated with respect to the features obtained from the previous step. We initialize $a^0$ and $x^0$ by drawing *i.i.d.* samples from $\mathcal{U}(0, 1)$ up to $t$. In practice, a replay buffer stores the predictions from the final iteration and uses them to start the second epoch as (Tieleman and Hinton 2009) suggested for faster convergence.

Likewise, $\frac{\partial E_a(\theta)}{\partial \theta}$ can be updated in a similar manner to the one described by Eq. 12.

## Activity Forecasting

Eq. 12 allows us to directly generate a sample sequence $\{a_{t+1}, x_{t+1}, \dots, a_{t+K}, x_{t+K}\}$ in one pass from $\mathrm{h}_t$ using a trained model whose parameters $\theta_a$ and $\theta_x$ are fixed. With sufficient number of samples, we expect all modes of the conditional distribution in Eq. 3 to be visited.

The simulated sequence $a$ and $x$ are expected to approach a particular mode of Eq. 1 and form a valid forecast for activities. In our implementation, a pre-trained MLP takes $x$ from the final sampling iteration to produce future agent paths in the form of a set of 2D coordinates, and a pre-trained classifier samples $a$ from the final iteration to assign on-hot digits as action labels.

# Experiment

Two large-scale datasets, Activities in Extended Videos (ActEV/VIRAT) (Awad et al. 2018) benchmark and TITAN (Malla, Dariush, and Choi 2020), are used to assess the performance of ELMA. They contain video sequences with annotations of both agent actions and locations provided for each frame. Actions are labeled with corresponding bounding boxes around a person or object in the each scene. They are the ground truth in our evaluation. Specifically, ActEV/VIRAT consists of 455 video clips captured in 12 scenes, which make up recordings of more than 12 hours. The videos resolution is 1920×1080. Twenty-nine categories of human actions are defined, such as ‶Transport″and ‶Interaction″. It also provides a set of completely labeled paths of moving pedestrians/agents during their entire appearances in the scene. TITAN contains 400 videos for training, 200 videos for validation, and 100 videos for test. All sequences are filmed from a moving camera, with each video clip of resolution 1920×1200 lasting 10-20 seconds. TITAN has eight action categories, 48 classes in total that describes either a person or a vehicle. The full trajectories of all agents are annotated.

## Experimental Setup

To make a fair comparison on ActEV/VIRAT, we follow the experimental protocols opted in Next (Liang et al. 2019). Ground truth bounding boxes annotating people and objects throughout the entire video sequence are used. $a_t$ is obtained by a pre-trained feature pyramid network (Lin et al. 2017) on ImageNet (Russakovsky et al. 2015) that extracts action features inside the bounding boxes. For each agent, We observe his/her past 8 steps and forecast the activities of the subsequent 12 steps. In other words, the first 3.2 seconds of a video is used to predict the content of up to 4.8 seconds of its remaining portion. We follow (Liang et al. 2019) to

| Influence of varying $k$ on ActEV/VIRAT benchmark | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Inception score ↑ | AM score ↓ | max mAP ↑ | ASD ↑ | FSD ↑ | NLL ↓ | minADE ↓ | minFDE ↓ |
| ELMA-10 | 7.96 | 0.87 | 25.50 | 29.64 | 44.98 | 6.14 | 15.94 | 32.73 |
| ELMA-20 | 8.83 | 0.81 | 25.92 | 44.93 | 52.59 | 5.77 | 14.80 | 30.65 |
| **ELMA-50** | 9.45 | **0.74** | 26.62 | 57.22 | 64.75 | 5.25 | 14.19 | **30.37** |
| ELMA-100 | **9.49** | 0.74 | **27.41** | **61.35** | **66.90** | **4.88** | **13.94** | 28.42 |
| Influence of varying $k$ on TITAN dataset | | | | | | | | |
| | Inception score ↑ | AM score ↓ | max mAP ↑ | ASD ↑ | FSD ↑ | NLL ↓ | minADE ↓ | minFDE ↓ |
| ELMA-10 | 7.75 | 1.12 | 43.14 | 43.29 | 54.22 | 8.25 | 9.40 | 18.09 |
| ELMA-20 | 8.27 | 0.98 | 44.69 | 52.40 | 64.95 | 7.76 | 8.87 | 17.47 |
| **ELMA-50** | **9.82** | 0.91 | 46.26 | 65.27 | 76.00 | 7.04 | **9.21** | 16.53 |
| ELMA-100 | 9.80 | **0.88** | **46.42** | **70.71** | **79.02** | **7.01** | 9.21 | **16.48** |

Table 1: The results generated by ELMA running different number of sampling steps $\mathcal{I}$. (↑) means that the higher the score the better, whereas (↓) means the opposite. There are significant improvements of ELMA-50 and ELMA-100 against ELMA-10 and ELMA-20 on all criteria, suggesting that in general more iterations always lead to improved performance. However, the comparison between ELMA-100 and ELMA-50 suggests that this improvement tends to be marginal and is expected to diminish eventually. Furthermore, since a larger $\mathcal{I}$ results in heavier computational burden, We thus select the outcomes from ELMA-50 as our *main results* to balance the trade-off between performance and computational cost.

solely obtain the upcoming $K$=12 action categories with a pre-trained classifier. To experiment on TITAN, similar to (Malla, Dariush, and Choi 2020), we extend the corresponding values to $t$=10 and $K$=20, and an identical fine-tuned single-stream I3D network (Carreira and Zisserman 2017) is deployed (Malla, Dariush, and Choi 2020) to obtain $a_t$. Also, we apply a pre-trained multi-head classification scheme to TITAN.

In our experiments, RMSProp optimizer (Goodfellow, Bengio, and Courville 2016) are employed with the learning rate initialized at $8 \times 10^{-5}$. Our implementation uses PyTorch. The experiments are executed on four Nvidia GeForce TITAN XPs, with 48 GB of memory in total.

## Criteria and Baselines

**Criteria:** We are primarily interested in verifying the performances of our algorithm concerning the diversity and quality aspects. Evaluating the diversity essentially quantifies how well the multimodality of Eq. 1 is covered. In particular, we include the inception score (Salimans et al. 2016), average self distance (ASD) and final self distance (FSD) (Yuan and Kitani 2020b) validate the variety part. In the meantime, it is not a trivial task to justify the quality – if the extensive range of outputs includes the true future. We particularly consider the maximum mean average precision (max mAP) of action predictions from all roll-outs, the minimum values of ADE (minADE) and FDE (minFDE) for this point. In addition, The AM score (Zhou et al. 2018) measures the confidence over assigned action categories, while the negative log-likelihood (NLL) determining the fit of ground truth paths to the estimated distribution.

Higher amounts of inception score, ASD, FSD and max mAP indicate preferable performances, while lower values of AM score, NLL, minADE, and minFDE suggest better outputs.

**Baselines:** As per comparing approaches, we assess our ELMA versus a leading study on activity forecasting Next (Liang et al. 2019), which achieves cutting-edge results on the ActEV/VIRAT benchmark. The TITAN framework (Malla, Dariush, and Choi 2020), which performs best on the TITAN dataset, is selected as well. We also study the efficacy of our proposal through analyses against the following baselines:

1. ELMA-$\mathcal{I}$: $\mathcal{I}$ is set to 10, 20, 50 and 100 to verify the effects of different configurations on iterative sampling steps. We note that the training is unstable and hard to converge when $\mathcal{I}$ <10. Additionally, the computational cost spikes if $\mathcal{I}$ >100. Therefore, we neither test for $\mathcal{I}$ <10 nor $\mathcal{I}$ >100.
2. ELMA-*a* & ELMA-*x*: ELMA-*a* shares the similar idea of Next (Sun et al. 2019), which conditions the action predictions upon the path predictions. ELMA-*x* simply reverses this settings.
3. ELMA-S: To highlight the merits of achieving our objective via Eq. 2, we remove the setting of forecasting paths conditioning on action predictions to set a ELMA-S baseline. This baseline shares a similar idea of multi-task learning with Next (Liang et al. 2019) and hypothesizes the independence between upcoming actions and paths. The rest of the framework remains intact.
4. ELMA-RNN: We term another baseline as ELMA-RNN to examine the impacts of our spatiotemporal dependencies handling mechanism. ELMA-RNN assigns one recurrent neural network (RNN) (Goodfellow, Bengio, and Courville 2016) per agent while disregarding the graph structure. This leads to foreseeing individual activity separately.
5. cVAE & cGAN: We justify the advantages of our energy-based generating process over cVAE and the cGAN frameworks conditioning on the forecasting context.
6. ELMA-NE: To verify the advantages of the energy functions, we drop the energy functions and build a ELMA-NE baseline. ELMA-NE has an identical backbone with

| Quantitative results on ActEV/VIRAT benchmark | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | Inception score ↑ | AM score ↓ | max mAP ↑ | ASD ↑ | FSD ↑ | NLL ↓ | minADE ↓ | minFDE ↓ |
| Next (Liang et al. 2019) | 1.46 | 1.31 | 19.20 | – | – | 7.16 | 17.99 | 37.24 |
| ELMA-NE | 1.73 | 1.24 | 19.84 | – | – | 6.89 | 16.70 | 36.16 |
| cGAN | 2.08 | 1.02 | 20.71 | 5.92 | 7.49 | 6.75 | 15.57 | 34.90 |
| cVAE | 4.81 | 0.94 | 22.04 | 21.64 | 33.55 | 6.22 | 15.20 | 33.98 |
| ELMA-RNN ($\mathcal{I} = 50$) | 6.27 | 1.53 | 19.19 | 23.47 | 38.38 | 7.29 | 18.82 | 37.75 |
| ELMA-S ($\mathcal{I} = 50$) | 6.40 | 1.13 | 20.91 | 27.80 | 42.61 | 6.94 | 16.06 | 36.42 |
| ELMA-*x* ($\mathcal{I} = 50$) | 7.39 | 1.07 | 22.47 | 42.01 | 51.22 | 5.95 | 14.91 | 31.97 |
| ELMA-*a* ($\mathcal{I} = 50$) | 8.21 | 0.95 | 24.23 | 36.60 | 46.08 | 6.28 | 15.22 | 32.44 |
| **ELMA-50(ours)** | **9.45** | **0.74** | **26.62** | **57.22** | **64.75** | **5.25** | **14.19** | **30.37** |
| Quantitative results on TITAN dataset | | | | | | | | |
| | Inception score ↑ | AM score ↓ | max mAP ↑ | ASD ↑ | FSD ↑ | NLL ↓ | minADE ↓ | minFDE ↓ |
| Next (Liang et al. 2019) | 1.74 | 1.98 | 32.62 | – | – | 8.85 | 13.39 | 24.74 |
| TITAN (Malla, Dariush, and Choi 2020) | – | – | – | – | – | – | 11.32 | 19.53 |
| ELMA-NE | 2.07 | 1.86 | 33.25 | – | – | 8.74 | 11.17 | 19.04 |
| cGAN | 2.30 | 1.71 | 34.59 | 4.48 | 6.90 | 8.59 | 10.94 | 18.29 |
| cVAE | 4.78 | 1.55 | 39.81 | 21.29 | 31.75 | 8.14 | 10.85 | 17.66 |
| ELMA-RNN ($\mathcal{I} = 50$) | 6.02 | 2.06 | 31.53 | 37.91 | 43.86 | 9.06 | 14.62 | 26.57 |
| ELMA-S ($\mathcal{I} = 50$) | 6.36 | 1.73 | 33.03 | 40.05 | 49.14 | 8.70 | 11.07 | 18.81 |
| ELMA-*x* ($\mathcal{I} = 50$) | 7.07 | 1.40 | 41.08 | 55.95 | 62.60 | 8.82 | 9.98 | 17.49 |
| ELMA-*a* ($\mathcal{I} = 50$) | 7.40 | 1.31 | 41.90 | 52.74 | 60.91 | 8.96 | 10.05 | 18.08 |
| **ELMA-50 (ours)** | **9.82** | **0.91** | **46.26** | **65.27** | **76.00** | **7.04** | **9.21** | **16.53** |

Table 2: The quantitative evaluations or ELMA on ActEV and TITAN datasets. (↑) means that the higher the score the better, whereas (↓) means the opposite. The improvements of ELMA-50 comparing with ELMA-*a*, ELMA-*x* and ELMA-S demonstrate the efficacy of our model (Eq. 3). Overall, ELMA-50 provides a better performance when compared with ELMA-NE, cGAN, cVAE, Next (Liang et al. 2019) and TITAN (Guan et al. 2020), which speaks for the effectiveness of energy-based models in capturing the uncertainties of future activities. We attribute ELMA's advantage to its capability of addressing the spatio-temporal dependencies in a multi-agent environment.

the full model, but is trained with minimizing L2 loss for path predictions, and optimizing cross entropy for action predictions.

## Benchmark Results

We carry out our experiments by generating 20 roll-outs of each video sequence on both datasets. Table 1 and Table 2 summarize the quantitative results on the ActEV/VIRAT and the TITAN datasets. In order to analyze the superiorities of EL in detail, we explore the following facets:

**Influence of varying $\mathcal{I}$:** We investigate the impacts of choosing different sampling steps $\mathcal{I}$ of ELMA, and report the results in Table 1. The significant improvements of ELMA-50 and ELMA-100 against ELMA-10 and ELMA-20 on all criteria manifest the benefits of more iterations. However, ELMA-100 only makes marginal improvements versus ELMA-50. This corroborates that $\mathcal{I}$=50 results in ELMA reaching a steady state on both datasets. We thus select the outcomes from ELMA-50 as our *main results* to balance the trade-off between the performance and computational cost. Also, ELMA-S, ELMA-*a*, ELMA-*x* and ELMA-RNN baselines consider 50 iterations throughout the experiments.

**Objective modeling:** Our proposal forecasts future activities upon the basis of formulating our objective in Eq. 3, rather than gaining action predictions independent of path predictions. We can observe that ELMA-50 incurs remarkable advantages with respect to the ELMA-*a*, ELMA-*x* and ELMA-S baselines in terms of both diversity and quality, as shown in Table 2. These findings overwhelmingly demonstrate the preference of our method over the ELMA-*a*, ELMA-*x* and ELMA-S for properly learning Eq. 1. These outcomes can be traced back to the more reasonable objective modeling of ELMA with respect to ELMA-*a*, ELMA-*x* and ELMA-S.

**Understanding the uncertain nature of the future:** The proposed ELMA-50 drastically advances, by far, the cVAE and cGAN baselines on all metrics. These comparisons meet our expectations that ELMA-50 can better explore the diversity of future activities. The primary reason can be traced back to the superior capability of the ELMA-50 approach in directly uncovering different modes of the learned distribution. During the experiments, the cVAE and cGAN baselines just attain predictions with limited variety compared to the proposed ELMA-50. We believe this stems from: 1. The assumptions on the learnt distribution lead that cVAE can neither align with the real data distribution nor sufficiently capture the future's uncertainty as our proposal does; 2. The outcomes of our ELMA-50 benefits from only training the energy functions as opposed to cGAN that has different networks.

Moreover, ELMA-NE and Next (Liang et al. 2019) do not achieve satisfactory results because they fail to consider the future's uncertainty.

**Spatiotemporal dependencies processing:** The overall amounts of ELMA-50 considerably exceed those of the ELMA-RNN baseline and Next. This provides the evidence to support the necessity of explicitly taking the spatiotemporal dependencies into account for multi-agent activity forecasting. In fact, even our ELMA-S baseline outdoes the

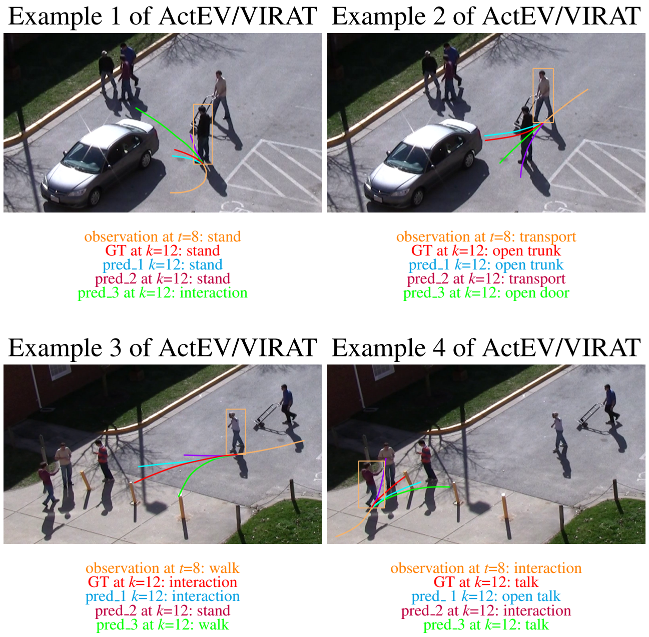
Example 1 of ActEV/VIRAT
Example 2 of ActEV/VIRAT
observation at $t$=8: stand
GT at $k$=12: stand
pred_1 $k$=12: stand
pred_2 at $k$=12: stand
pred_3 at $k$=12: interaction
observation at $t$=8: transport
GT at $k$=12: open trunk
pred_1 $k$=12: open trunk
pred_2 at $k$=12: transport
pred_3 at $k$=12: open door
Example 3 of ActEV/VIRAT
Example 4 of ActEV/VIRAT
observation at $t$=8: walk
GT at $k$=12: interaction
pred_1 $k$=12: interaction
pred_2 at $k$=12: stand
pred_3 at $k$=12: walk
observation at $t$=8: interaction
GT at $k$=12: talk
pred_ 1 $k$=12: open talk
pred_2 at $k$=12: interaction
pred_3 at $k$=12: talk

Figure 4: A visualization of the qualitative results on the ActEV/VIRAT dataset. For clearer visualization, we isolate an agent and his/her multiple plausible future activities per example. Each example renders a yellow line for the historical path, a red line for ground truth, a cyan line for the path predictions with lowest minADE scores, and the rest for randomly sampled path predictions. We depict the top-1 classification of action predictions at $k$=12 are painted by matched colors with path predictions.

ELMA-RNN baseline and Next.

Furthermore, the outcomes in Table 2 favor the ELMA-NE baseline over Next and TITAN framework (Malla, Dariush, and Choi 2020). This is mainly due to that jointly handling spatiotemporal dependencies suggests a better strategy than treating them separately.

### Visual Results

Fig. 4 $\sim$ 5 visualize the example outputs from ELMA-50. We highlight the path prediction with the lowest minADE, two other path predictions that are randomly selected from all rollouts, and their associated top-1 classifications on action predictions in each example. Notably, our ELMA-50 forecasts the activities of all agents in the scene simultaneously from $t$+1 to $t + k$. We display the predictions of an individual agent in every example to obtain clearer visualizations. The ELMA-50 method can evidently forecast diverse forthcoming activities as well as generate the true future. For instance, in example 2 of Fig. 4, our approach uncovers diverse possibilities such as that the agent would open the door or keep transporting. Also, ELMA-50 successfully foresees that the agent will open the trunk as ground truth. Furthermore, we can observe the impact of spatiotemporal dependencies on individual activities through visual demonstrations. Example 1 of TITAN in Fig. 5 presents that predicting the agent loading is impossible without capturing the relationships between the person and the car.

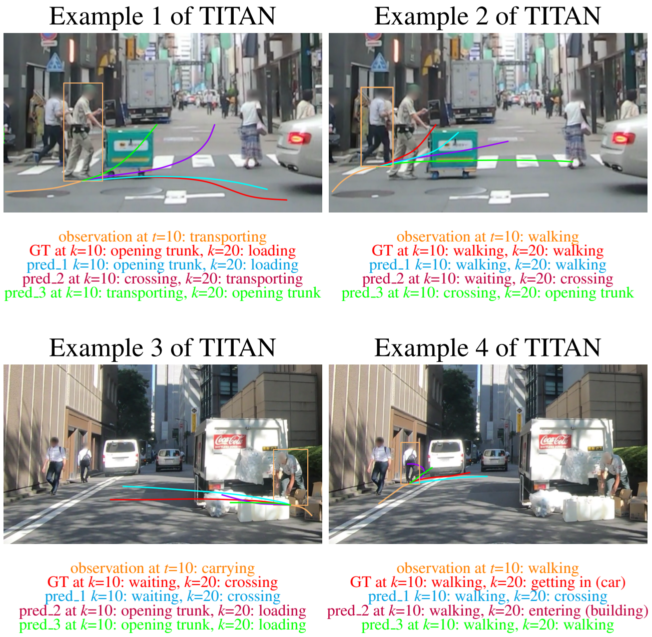
Example 1 of TITAN
Example 2 of TITAN
observation at $t$=10: transporting
GT at $k$=10: opening trunk, $k$=20: loading
pred_1 $k$=10: opening trunk, $k$=20: loading
pred_2 at $k$=10: crossing, $k$=20: transporting
pred_3 at $k$=10: transporting, $k$=20: opening trunk
observation at $t$=10: walking
GT at $k$=10: walking, $k$=20: walking
pred_1 $k$=10: walking, $k$=20: walking
pred_2 at $k$=10: waiting, $k$=20: crossing
pred_3 at $k$=10: crossing, $k$=20: opening trunk
Example 3 of TITAN
Example 4 of TITAN
observation at $t$=10: carrying
GT at $k$=10: waiting, $k$=20: crossing
pred_1 $k$=10: waiting, $k$=20: crossing
pred_2 at $k$=10: opening trunk, $k$=20: loading
pred_3 at $k$=10: opening trunk, $k$=20: loading
observation at $t$=10: walking
GT at $k$=10: walking, $k$=20: getting in (car)
pred_1 $k$=10: walking, $k$=20: crossing
pred_2 at $k$=10: walking, $k$=20: entering (building)
pred_3 at $k$=10: walking, $k$=20: walking

Figure 5: The visual results on dataset TITAN. In order to obtain a clearer visualization, we separately illustrate an agent and his/her predictions in each example. The yellow line presents the historical paths, the red lines denote the ground truths, Cyan lines pertain to the path predictions with the lowest minADE scores, and the rest are randomly sampled path predictions. We showcase the top-1 classifications of action predictions at $k$=10 and $k$=20 are painted are painted by matched colors with path predictions.

Example 4 in Fig. 5 shows an imperfect case. ELMA-50 fails to forecast that the agent will enter the vehicle, given that their interactions might be too subtle.

## Conclusion

In this paper, we propose an energy-based learning framework, named ELMA, that forecasts the activities of multiple agents simultaneously. ELMA differs from the majority of existing work in that it establishes a unified framework in which the actions and paths of multiple agents are analyzed in an integral formulation. Furthermore, we introduce a pair of energy functions to model multi-agent activities probabilistic-ally. This formulation allows us to grasp the uncertain nature of the activities in future without the inclusion of additional latent variables as what VAE or GLOW do. Using GNN, ELMA resolves the spatial and temporal dependencies effectively. The experiment results justify that ELMA makes better predictions in terms of both diversity and quality concerning prior works and other generative baselines.

We believe that ELMA can benefit future studies on various real-world applications. One possible direction would be to apply our framework to enable a self-navigating robot or an autonomous vehicle to make more informed decisions on selecting an optimal path after sensing agents' activities in a dynamic environment. This will be a part of our future work.

## References

Alahi, A.; Goel, K.; Ramanathan, V.; Robicquet, A.; Fei-Fei, L.; and Savarese, S. 2016. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 961–971.

Awad, G.; Butt, A.; Curtis, K.; Lee, Y.; Fiscus, J.; Godil, A.; Joy, D.; Delgado, A.; Smeaton, A.; Graham, Y.; et al. 2018. TRECVID 2018: Benchmarking Video Activity Detection, Video Captioning and Matching, Video Storytelling Linking and Video Search. In *Proceedings of TRECVID 2018*.

Battaglia, P.; Hamrick, J. B. C.; Bapst, V.; Sanchez, A.; Zambaldi, V.; Malinowski, M.; Tacchetti, A.; Raposo, D.; Santoro, A.; Faulkner, R.; Gulcehre, C.; Song, F.; Ballard, A.; Gilmer, J.; Dahl, G. E.; Vaswani, A.; Allen, K.; Nash, C.; Langston, V. J.; Dyer, C.; Heess, N.; Wierstra, D.; Kohli, P.; Botvinick, M.; Vinyals, O.; Li, Y.; and Pascanu, R. 2018. Relational inductive biases, deep learning, and graph networks. *https://arxiv.org/pdf/1806.01261.pdf*.

Carreira, J.; and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6299–6308.

Chen, J.; Bao, W.; and Kong, Y. 2020. Group Activity Prediction with Sequential Relational Anticipation Model. In *European Conference on Computer Vision*, 581–597. Springer.

Du, Y.; Li, S.; and Mordatch, I. 2020. Compositional Visual Generation with Energy Based Models. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M. F.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 6637–6647. Curran Associates, Inc.

Epstein, D.; Chen, B.; and Vondrick, C. 2020. Oops! predicting unintentional action in video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 919–929.

Epstein, D.; Chen, B.; and Vondrick, C. 2021. Learning the Predictability of the Future. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Gao, R.; Lu, Y.; Zhou, J.; Zhu, S.-C.; and Wu, Y. N. 2018. Learning generative convnets via multi-grid modeling and sampling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9155–9164.

Gao, R.; Song, Y.; Poole, B.; Wu, Y. N.; and Kingma, D. P. 2021. Learning Energy-Based Models by Diffusion Recovery Likelihood. In *International Conference on Learning Representations*.

Goodfellow, I.; Bengio, Y.; and Courville, A. 2016. *Deep Learning*. MIT Press.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, 2672–2680.

Guan, J.; Yuan, Y.; Kitani, K. M.; and Rhinehart, N. 2020. Generative hybrid representations for activity forecasting with no-regret learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 173–182.

Gupta, A.; Johnson, J.; Fei-Fei, L.; Savarese, S.; and Alahi, A. 2018. Social GAN: Socially Acceptable Trajectories With Generative Adversarial Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

He, J.; Lehrmann, A.; Marino, J.; Mori, G.; and Sigal, L. 2018. Probabilistic video generation using holistic attribute control. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 452–467.

Hinton, G. E. 2002. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8): 1771–1800.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising Diffusion Probabilistic Models. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M. F.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 6840–6851. Curran Associates, Inc.

Ivanovic, B.; and Pavone, M. 2019. The Trajectron: Probabilistic Multi-Agent Trajectory Modeling With Dynamic Spatiotemporal Graphs. In *The IEEE International Conference on Computer Vision (ICCV)*.

Koppula, H. S.; and Saxena, A. 2016. Anticipating human activities using object affordances for reactive robotic response. *IEEE transactions on pattern analysis and machine intelligence*, 38(1): 14–29.

LeCun, Y.; Chopra, S.; Hadsell, R.; Ranzato, M.; and Huang, F. 2006. A tutorial on energy-based learning. *Predicting structured data*, 1(0).

Li, Y. 2017. Pedestrian Path Forecasting in Crowd: A Deep Spatio-Temporal Perspective. In *Proceedings of the 2017 ACM on Multimedia Conference*, MM '17, 235–243. New York, NY, USA: ACM. ISBN 978-1-4503-4906-2.

Li, Y. 2018. A Deep Spatiotemporal Perspective for Understanding Crowd Behavior. *IEEE Transactions on Multimedia*, 20(12): 3289–3297.

Li, Y. 2018. Video Forecasting with Forward-Backward-Net: Delving Deeper into Spatiotemporal Consistency. In *2018 ACM Multimedia Conference on Multimedia Conference*, 211–219. ACM.

Li, Y. 2019. Which Way Are You Going? Imitative Decision Learning for Path Forecasting in Dynamic Scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 294–303.

Li, Y.; Wang, P.; and Chan, C.-Y. 2021. RESTEP into the Future: Relational Spatio-Temporal Learning for Multi-Person Action Forecasting. *IEEE Transactions on Multimedia*, 1–1.

Li, Y.-K.; Wang, P.; Ye, M.; and Chan, C.-Y. 2021. *Imitative Learning for Multi-Person Action Forecasting*, 451–459. Association for Computing Machinery.

Liang, J.; Jiang, L.; Niebles, J. C.; Hauptmann, A. G.; and Fei-Fei, L. 2019. Peeking into the future: Predicting future person activities and locations in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5725–5734.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, 2117–2125.

Luo, W.; Yang, B.; and Urtasun, R. 2018. Fast and Furious: Real Time End-to-End 3D Detection, Tracking and Motion Forecasting With a Single Convolutional Net. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 3569–3577.

Malla, S.; Dariush, B.; and Choi, C. 2020. TITAN: Future Forecast using Action Priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 11186–11196.

Mangalam, K.; Girase, H.; Agarwal, S.; Lee, K.-H.; Adeli, E.; Malik, J.; and Gaidon, A. 2020. It is not the journey but the destination: Endpoint conditioned trajectory prediction. In *European Conference on Computer Vision*, 759–776. Springer.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; and Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3): 211–252.

Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; and Chen, X. 2016. Improved Techniques for Training Gans. In *Advances in neural information processing systems*, 2234–2242.

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

Si, C.; Chen, W.; Wang, W.; Wang, L.; and Tan, T. 2019. An attention enhanced graph convolutional lstm network for skeleton-based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1227–1236.

Song, Y.; and Ermon, S. 2019. Generative Modeling by Estimating Gradients of the Data Distribution. In *Proceedings of the 33rd Annual Conference on Neural Information Processing Systems*.

Suhail, M.; Mittal, A.; Siddiquie, B.; Broaddus, C.; Eledath, J.; Medioni, G.; and Sigal, L. 2021. Energy-Based Learning for Scene Graph Generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Sun, C.; Shrivastava, A.; Vondrick, C.; Sukthankar, R.; Murphy, K.; and Schmid, C. 2019. Relational Action Forecasting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 273–283.

Sun, J.; Jiang, Q.; and Lu, C. 2020. Recursive Social Behavior Graph for Trajectory Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 660–669.

Tang, C.; and Salakhutdinov, R. R. 2019. Multiple futures prediction. In *Advances in Neural Information Processing Systems*, 15424–15434.

Tieleman, T.; and Hinton, G. 2009. Using fast weights to improve persistent contrastive divergence. In *Proceedings of the 26th annual international conference on machine learning*, 1033–1040.

Vondrick, C.; Pirsiavash, H.; and Torralba, A. 2016. Anticipating visual representations from unlabeled video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 98–106.

Walker, J.; Marino, K.; Gupta, A.; and Hebert, M. 2017. The pose knows: Video forecasting by generating pose futures. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 3352–3361. IEEE.

Welling, M.; and Teh, Y. W. 2011. Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, 681–688.

Xie, J.; Zhu, S.-C.; and Nian Wu, Y. 2017. Synthesizing dynamic patterns by spatial-temporal generative convnet. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 7093–7101.

Xu, D.; Hu, R.; Xiong, Z.; Wang, Z.; Luo, L.; and Li, D. 2021. *Trajectory is Not Enough: Hidden Following Detection*, 5373–5381. New York, NY, USA: Association for Computing Machinery. ISBN 9781450386517.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations (ICLR)*.

Yang, F.; Chen, L.; Zhou, F.; Gao, Y.; and Cao, W. 2020. Relational State-Space Model for Stochastic Multi-Object Systems. In *International Conference on Learning Representations*.

Yuan, Y.; and Kitani, K. 2020a. Dlow: Diversifying latent flows for diverse human motion prediction. In *European Conference on Computer Vision*, 346–364. Springer.

Yuan, Y.; and Kitani, K. M. 2020b. Diverse Trajectory Forecasting with Determinantal Point Processes. In *International Conference on Learning Representations*.

Zeng, K.-H.; Mottaghi, R.; Weihs, L.; and Farhadi, A. 2020. Visual Reaction: Learning to Play Catch with Your Drone. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11573–11582.

Zeng, K.-H.; Shen, W. B.; Huang, D.-A.; Sun, M.; and Niebles, J. C. 2017. Visual Forecasting by Imitating Dynamics in Natural Sequences. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 3018–3027. IEEE.

Zhang, Z.; Gao, J.; Mao, J.; Liu, Y.; Anguelov, D.; and Li, C. 2020. STINet: Spatio-Temporal-Interactive Network for Pedestrian Detection and Trajectory Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11346–11355.

Zhou, Z.; Cai, H.; Rong, S.; Song, Y.; Ren, K.; Zhang, W.; Wang, J.; and Yu, Y. 2018. Activation Maximization Generative Adversarial Nets. In *International Conference on Learning Representations (ICLR)*.