This CVPR 2021 paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Hierarchical Video Prediction using Relational Layouts for Human-Object Interactions

Navaneeth Bodla[1] Gaurav Shrivastava[1] Rama Chellappa[2] Abhinav Shrivastava[1]
[1]University of Maryland, College Park [2]Johns Hopkins University

Abstract

Learning to model and predict how humans interact with objects while performing an action is challenging, and most of the existing video prediction models are ineffective in modeling complicated human-object interactions. Our work builds on hierarchical video prediction models, which disentangle the video generation process into two stages: predicting a high-level representation, such as pose sequence, and then learning a pose-to-pixels translation model for pixel generation. An action sequence for a human-object interaction task is typically very complicated, involving the evolution of pose, person's appearance, object locations, and object appearances over time. To this end, we propose a Hierarchical Video Prediction model using Relational Layouts. In the first stage, we learn to predict a sequence of layouts. A layout is a high-level representation of the video containing both pose and objects' information for every frame. The layout sequence is learned by modeling the relationships between the pose and objects using relational reasoning and recurrent neural networks. The layout sequence acts as a strong structure prior to the second stage that learns to map the layouts into pixel space. Experimental evaluation of our method on two datasets, UMD-HOI and Bimanual, shows significant improvements in standard video evaluation metrics such as LPIPS, PSNR, and SSIM. We also perform a detailed qualitative analysis of our model to demonstrate various generalizations.

1. Introduction

Video prediction is a challenging task of predicting future frames conditioned on one or more past frames. Videos in the real world are extremely complex. An everyday action, such as drinking a coffee, results from complicated interactions among various objects. For example, first, the person might reach the coffee pot and pour coffee into their cup. Next, they might start drinking from the cup while browsing their cell phone. Observe that this particular action involves interactions among various objects such as a coffee pot, cup, and cell phone. Each of the objects has its relative motion with respect to other objects and the person performing the action. While it is effortless for human beings to imagine such events, existing computer vision models often fail at these tasks.

Existing video prediction methods broadly fall into two categories: 1) models that directly predict the video in the pixel space, and 2) models that use hierarchical prediction. Hierarchical prediction methods are a preferred over directly predicting the video in pixel space as they learn a good intermediate representation which is then mapped to pixel space. Disentangling the prediction into simpler steps helps the models focus on smaller tasks and, hence, learn an improved frame prediction model. A natural choice of intermediate representation for videos is optical flow [1]. Similarly, for videos involving human actions, human-pose is typically used as an intermediate representation. Villegas et al. [2] and Walker et al. [3] have disentangled the video prediction by first predicting the pose sequence and then mapping the pose sequence to pixel space.

While pose is a great choice for videos involving human actions, pose alone is not sufficient to capture various dynamics in a Human-Object interaction sequence. For complex actions, such as human-object interactions where multiple objects evolve over time, pose alone does not fully capture the complex scene dynamics. Since the pose does not contain any information about the objects, the models fail to capture the object's motion and appearance faithfully. To mitigate this issue, we propose to learn a *layout sequence* as an intermediate representation. A layout sequence is a combination of pose and object sequences that not only captures the person's pose while performing an action, but also explicitly learns the locations of various objects at different times while the action is being performed. The naive way of learning these pose and object sequences independently is also not sufficient since the spatio-temporal evolution of an object is dependent on how the pose is evolving and vice versa. Hence, we propose a **Human-Object Relational Network (HORN)** to model these complex interactions among objects and poses.

Our key contributions for video prediction for human-

object interactions are: 1) we model the full-body motion for humans, 2) our intermediate representation captures both pose and object locations, and thus learns a better structure prior for the frame prediction stage. 3) our model can generate videos for novel interactions, e.g., it can generalize well to new people performing actions that were not part of the training set.

2. Related Work

Our work is closely related to video prediction, human-object interaction, and relational reasoning. We briefly review some related research in these areas.

Video Prediction. Early works in video prediction have focused on directly predicting the sequence of frames in pixel space [4–11]. Our work is different from these since we do not predict the frames in the pixel space directly. Instead, we use a hierarchical model that first generates a structure that acts as a prior to the next stage of generating pixels. Two-stage methods have been proposed to improve the video prediction models [2, 3, 12–16]. HVP [2] learns to predict a sequence of poses in the first stage and maps it to pixel space using a visual analogy in the second stage. For complicated videos such as human-object interaction, there is more than one object, and the person's pose evolving over time. Therefore, pose alone is not sufficient to capture good structure prior. We propose to learn a layout sequence, which is a combination of both pose and object sequences.

Human-Object Interaction. Modeling human-object interactions in images has been an active research topic in computer vision [17–22]. HoI-GAN [23] proposed a generative model using 3D convolutions for synthesizing human interactions with objects in videos. Similar to this, our work also aims to synthesize human-object interaction videos. However, it is different in the following aspects 1) we propose a two-stage method instead of synthesizing images directly in the pixel space. 2) We model interactions with more than one object, whereas HoI-GAN assumes interactions with only one object. 3) While HoI-GAN models egocentric videos predominantly involving hand motions, we model humans' full-body motion.

Relational Reasoning. Relational reasoning has been widely used to model the interactions among various entities for visual understanding tasks such as : object interactions in videos [24], object relationships in images [25], action prediction [26, 27], human-object interactions [28–30] human trajectory prediction [31], physical dynamics of object [32, 33] etc. To the best of our knowledge, this work is the first to use relational reasoning in the context of video prediction for real-world videos. Since our task involves modeling human-object interactions, which in-turn involves reasoning across entities such as humans and objects and objects and objects, relational reasoning is a natural choice for our first stage of layout generation.

3. Human-Object Interactions using Relational Layouts

Let x_0 be the first frame of a video sequence of a person performing an action a. Let $b_0 = \{b_0^0, b_0^1, \ldots, b_0^n\}$ be the set of objects in the first frame of the video. Our task is to predict how a person would perform action a by interacting with objects b_0. Formally, we would like to learn a mapping function $X_{1:T} = G(x_0, a, b_0)$ that takes the first frame, also called as guidance frame, x_0, action a and a set of objects b_0 in the guidance frame as inputs and learns to predict the subsequent frames $X_{1:T} = [x_1, x_2, \ldots, x_T]$.

We disentangle this process of generating the video $X_{1:T}$ conditioned on the guidance frame x_0 into two subtasks. In the first stage, we learn a coarser representation of the video, which is a sequence of layouts $L_{1:T}$. The layout sequence is a sequence of object locations and keypoints of the person performing the target action. In the second stage, we learn a video generator that takes the sequence of layouts and the guidance frame as inputs and predicts the video in the pixel space.

3.1. Stage 1 : Relational Layout generation

A layout sequence $L_{0:T}$ is a combination of pose and object sequences. Formally, let $P_{0:T} = [p_0, p_1, \ldots, p_T]$ be the sequence of poses, let $B^k_{0:T} = [b^k_0, b^k_1, \ldots, b^k_T]$ where $k = 1, 2, ..n$ be the sequence of k-th object. We assume that the number of objects, n remains constant through out the sequence. $L_{0:T} = [l_0, l_1, \ldots, l_T]$ is a sequence of layouts where $l_t = (p_t, b^1_t, b^2_t, \ldots, b^n_t)$. At each time step t layout l_t is a tuple of pose and objects.

Our Relational Layout Generator consists of three main building blocks: 1) a Human-Object sequence prediction model that learns to predict the sequence of poses and sequence of objects corresponding to the target action, 2) a Human-Object relational model that learns to reason about relations among various objects and poses and 3) the pose and object decoders that generate the final sequence of layouts.

Human-object sequence prediction. The pose and object features for every time step are learned recursively using recurrent networks. More than one object evolves over time in an action, and hence we model each object's sequence separately using an RNN. Since all the object sequences differ only in terms of the object category, we share the recurrent network parameters for all the object sequences. The pose and object features for the time step $t + 1$ are obtained by learning pose and object recurrent networks RNN_{θ_p} and

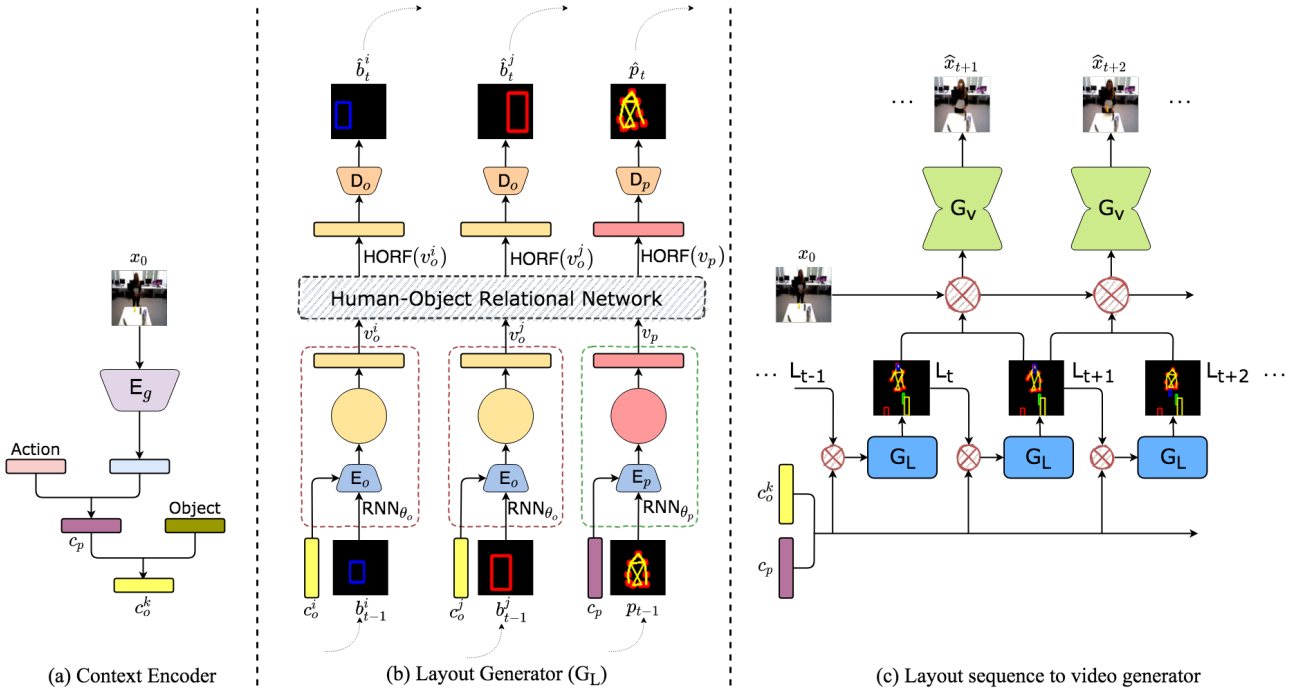


Figure 1: An overview of our approach. (a) Context encoder for encoding the first frame, action, and object information. (b) Layout generator for predicting the layout sequence using pose and object RNNs and relational network. (c) Video generator for mapping layouts to pixels.

RNN_{θ_o} as follows:

$$v_p = \text{RNN}_{\theta_p}(c_p, p_t), \qquad v_o^k = \text{RNN}_{\theta_o}(c_o, b_t^k) \quad (1)$$

where p_t is the pose and b_t^k is the location of the k-th object at time step t and θ_p, θ_o are the parameters of the RNNs for pose and object respectively. c_p and c_o^k are context features for pose and object RNNs that remain constant throughout the sequence prediction for every step. c_p and c_o^k are computed by learning the context mapping functions $c_p = f_{\theta_p}(x_0, a)$ and $c_o^k = f_{\theta_o}(x_0, a, y^k)$ where y^k is the object class for the k-th object.

Human-object relational reasoning. In a typical Human-Object interaction sequence, the pose and object sequences are highly correlated. Since the sequence prediction step learns a sequence of pose and object features independently, it cannot reason about the relations among them. Therefore to incorporate relational reasoning among various objects and human pose, we propose a Human-Object Relational Network (HORN). HORN takes the pose and object features from the sequence prediction step and enhances them by learning the complicated Human-Object relationships. HORN learns two kinds of Human-Object relations: 1) Object-Object and 2) Human-Object.

Given a set of object features $\mathcal{O} = \{v_o^1, v_o^2, \ldots, v_o^n\}$ we define pairwise relations between the object feautres as $g_\theta(v_o^i, v_o^j)$ where $g_\theta(.,.)$ is a feature extraction function parameterized by θ and v_o^i, v_o^j are the object features, which are the outputs of the sequence prediction step. The pairwise relational features are further aggregrated to obtain object relational features (ORF) for every object as:

$$\text{ORF}(v_o^i) = g_\phi(g_\theta(v_o^i, v_o^j) : v_o^j \in \mathcal{O}) \quad (2)$$

g_ϕ is the aggregation function parameterized by ϕ that learns to aggregate all pairs for object features.

Object relational features are further encouraged to learn Human-Object Relational Features (HORF) by learning their relationships with pose features. To do this, we learn $h_\theta(v_p, \text{ORF}(v_o^i))$ where $h_\theta(v_p, .)$ is a feature extraction function parameterized by θ, v_p is the pose feature and $\text{ORF}(v_o^i)$ is the object relational feature of the i-th object. Similarly, the pose feature v_p is encouraged to learn the relationships with the objects. This is done by learning a function $h_\phi(v_p, \text{ORF}(v_o^i))$ where $h_\phi(v_p, .)$ is a feature extraction function parameterized by ϕ. Therefore,

$$\begin{aligned} \text{HORF}(v_o^i) &= h_\theta(v_p, \text{ORF}(v_o^i))), \\ \text{HORF}(v_p) &= h_\phi(v_p, \text{ORF}(v_o^i))). \end{aligned} \quad (3)$$

where $v_o^i \in \mathcal{O}$

Pose and object decoders. The HORN outputs are enhanced pose and object feature vectors obtained by learning the relationships among them. These enhanced pose

and object feature vectors are passed through their respective pose and object decoders D_p and D_o to predict the pose $\widehat{p}_{t+1}$ and objects $\widehat{b}^k_{t+1}$ for the next step. This process is recursively continued to generate the layout sequence $\widehat{L}_{1:N}$. The overall layout generator (G_L) is learned by minimizing the layout loss $\mathcal{L}_{\text{Layout}} = \mathcal{L}_{\text{pose}} + \mathcal{L}_{\text{object}}$ as:

$$\mathcal{L}_{\text{pose}} = \text{BCE}(p_{1:T}, \widehat{p}_{1:T}),\ \mathcal{L}_{\text{object}} = \sum_{k=1,..,n} \text{MSE}(b^k_{1:T}, \widehat{b}^k_{1:T}) \tag{4}$$

3.2. Stage 2 : Layout sequence to video generation

Once the layout sequence is predicted, the next step is to learn to map them to video. The layout sequence acts as a structure prior that helps the generator to synthesize temporally coherent videos with high fidelity. The video generator needs to learn to generate frames with the same context as the guidance frame while respecting the input layout. While the guidance frame provides contextual information such as background, the appearance of the objects in place, and the person's appearance performing the action, the layout offers information about the pose and location of objects. The video generator learns to extract this contextual information from the guidance frame, propagates it throughout the video, and maintains the pose and object locations.

The video generator model (G_v) is an encoder-decoder style convolutional neural network that takes a layout as input and generates a frame. To maintain temporal consistency across the predicted frames, along with the current layout, G_v takes the previous layout and the guidance frame as inputs to generate the current frame. To generate the frame at time step t, G_v takes l_t, l_{t-1} and x_0 as inputs. The video generator is learned by minimizing the $\mathcal{L}_1$ loss between the generated and ground truth video as follows:

$$\widehat{x}_t = \quad G_v(l_t, l_{t-1}, x_0), \quad \mathcal{L}_{\text{pix}} = \sum_{t=1,...,T} |\widehat{x}_t - x_t| \tag{5}$$

Since the input to the video generator is just the first frame, $\mathcal{L}_1$ loss alone is not sufficient to generate high fidelity videos. With $\mathcal{L}_1$ loss alone, the videos tend to be blurry and not guaranteed to synthesize objects faithfully. This problem gets aggravated when the objects are too small with significant motion in very few frames. Hence, to further improve the generated videos' visual quality, we learn the video generator by training it with feature loss and discriminator loss functions. We use the same feature loss ($\mathcal{L}_{\text{feat}}$) as proposed in HVP [2]. For improving the realism of the generated videos, we propose three discriminators that operate at various granularity levels: 1) an object discriminator to enhance objects' appearance, 2) a frame discriminator to make frames look more realistic, and 3) a video discriminator to improve the spatio-temporal quality of the generated video in pixel space.

Conditional Object Discriminator. To obtain a perceptually good appearance for the objects, we train G_v against an object discriminator. The generated video frames must have the following two properties: 1) the objects' appearance in every frame needs to match with that of the guidance frame, and 2) the objects must belong to the same class as the guidance frame. Hence the conditional object discriminator D_{obj} takes cropped objects from guidance frame, cropped objects from generated (and real) frames, and the corresponding object classes as inputs. D_{obj} is trained to distinguish a fake triplet from a real triplet, and the generator G_v is encouraged to synthesize the objects that can fool the discriminator. G_v and D_{obj} are trained by minimizing the loss function:

$$\mathcal{L}_{\text{obj}} = [\log D_{\text{obj}}(x^c_t; y, x^c_0)] + [\log(1 - D_{\text{obj}}(\widehat{x}^c_t; y, x^c_0))]. \tag{6}$$

where x^c_t is the cropped object from real video and $\widehat{x}^c_t$ is the cropped object from the predicted video at time t.

Conditional Frame Discriminator. To ensure that the video generator accurately learns the person's appearance, we train it against a frame discriminator D_f. D_f takes the guidance frame and each of the subsequent frames as inputs and learns to distinguish it from a fake pair. The generator is trained to synthesize frames that fool the discriminator and hence learns to transfer the person's appearance from the guidance frame to the entire video. G_v and D_f are trained by minimizing the loss function:

$$\mathcal{L}_{\text{frame}} = [\log D_f(x_t; x_0)] + [\log(1 - D_f(\widehat{x}_t; x_0))]. \tag{7}$$

Conditional Video Discriminator. To obtain temporally coherent predicted videos, we train the video generator G_v against a video discriminator D_v. D_v is also designed to ensure that the generated frames are coherent with the input layout. To do so, D_v takes the layout sequence and real video as inputs and learns to distinguish with the fake pair. G_v and D_v are trained by minimizing the loss function:

$$\mathcal{L}_{\text{video}} = [\log D_v(X_{1:T}, L_{1:T})] + [\log(1 - D_v(\widehat{X}_{1:T}, L_{1:T}))]. \tag{8}$$

3.3. Architecture and training details

Inputs. In our model, the guidance frame is a tensor of size $128 \times 128 \times 3$. The action vector is a one-hot embedding of size $1 \times d_a$ and the object vectors are one-hot embeddings of size $1 \times d_o$ where d_a and d_o are number of action classes and object classes respectively. For the pose sequence, pose at every time step is represented as a 2D map of N_{kp} keypoints. That is, p_t is a tensor of size $128 \times 128 \times N_{kp}$. For the object sequence of every object, b^k_t is represented using object class and bounding box information. b^k_t is a tensor of size $1 \times (d_o + 4)$ which is obtained

Table 1: Quantitative comparision of our method with baselines on Bimanual dataset. FID and I-DTW scores are lower the better and mAP is higher the better.

Metric	FID(↓)	I-DTW(↓)	mAP(↑)
MoCoGAN	181.65	14.97	7.83
HoI-GAN	214.67	13.29	8.77
HVP	69.92	11.32	57.52
HORN (ours)	**35.82**	**10.52**	**59.04**

by concatenating one-hot vector of the object class and the bounding box, x, y, dx, dy.

Layout generator. We use convolutional encoders to encode frames and poses in the layout generator. The guidance frame encoder E_g, the pose encoder E_p and the pose decoder D_p are convolutional networks with 4×4 convolutions and stride 2. All convolutional layers are followed by batch normalization and ReLU. For pose and object sequences, we use two layer LSTMs with hidden size of 128. Note that all the object sequences share the same LSTM. The HORN is implemented using a 2 layer MLP as mentioned in Palm et al. [34]

Video generator and discriminators. Video generator is a Pix2PixHD [35] architecture with 2 local enhancers and 2 global blocks. The input to the generator is a tensor of size $128 \times 128 \times 3 + 2 * (N_{kp} + d_o)$. This is obtained by concatenating the guidance frame, pose maps, and object maps of the current and previous frames.

The frame discriminator and object discriminators are 2D convolutional networks, and the video discriminator is a 3D convolutional network with spectral normalization. For the object discriminator, the objects are cropped and resized to a fixed size of 32×32. We train the Model using $Adam$ optimizer with a learning rate of 0.0002 and batch size 24. The layout generation and pixel generation stages are trained independently. Additional training details, such as data augmentation and preprocessing, are provided in the supplementary material. In all our experiments, the videos are of length 17 such that given the first frame, the next 16 frames are predicted by the model.

4. Experiments

We present the qualitative and quantitative evaluation of our approach on two datasets: 1) UMD-HOI [36] and 2) Bimanual [37]. Both these datasets contain the videos where a performer approaches a set of objects and performs a task such as drinking, speaking on a telephone, cooking with bowls, etc. The main advantages of these datasets are: 1) the performer is fully visible in the video, 2) the actions performed are complex enough to involve full-body motion, and 3) An action may contain interaction with multiple objects. The UMD-HOI dataset has videos with single object interactions mainly done with either right or left hand. The Bimanual dataset has videos with more complicated interactions with multiple objects involving both hands. Therefore, these datasets are better suited to evaluate various components in our method, such as the effectiveness of pose sequences, object sequences, and their interactions.

UMD-HOI dataset. UMD-HOI dataset has a total of 64 videos with actions performed by ten subjects performing six interactions with four objects. The dataset is divided into train and test splits. The train split contains 50 videos, and the test split contains 14 videos. To evaluate generalization to new subjects, a random subject is held out of training. The training and test videos are chosen randomly.

Bimanual dataset. The Bimanual dataset contains 540 videos with actions performed by six subjects. It has an overall twelve objects and nine tasks. The tasks in this dataset involve interactions with multiple objects over time. For example, "cooking with bowls" action involves picking up a whisk, mixing in the bowl, poring from another bowl, etc. We selected 6 tasks and computed the tracklets for each object in the video using two heuristics : 1) IoU and 2) similarity score based on the color histogram. The dataset is divided into training and testing splits. The training has 300 videos, and testing has 120. The data is split so that two random actions performed by every subject are held out from training and are included in the test set.

We compare our method with the following baselines :

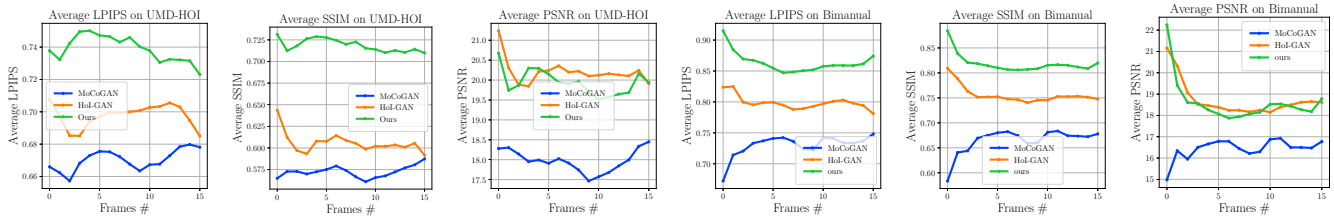
- im2vid MoCoGAN [38]: MoCoGAN is a widely used video prediction model that synthesizes videos from random noise. We use an image to video prediction version of the MoCoGAN as mentioned in [38] (section 4.3) to compare our method. It is a stronger baseline and closer to our approach. We modify it to take the first frame and action as inputs to predict the video.

- im2vid HOI-GAN [23]: Like MoCoGAN, we modify HOI-GAN to image to the video prediction model. For that, we make two modifications to the original model: 1) We remove noise as the input, and 2) the generator is trained with an additional L_1 loss function.

4.1. Quantitave Analysis

Perceptual similarity measures. To quantitatively evaluate the effectiveness of our approach, we compare our method with the baselines using the following standard video evaluation metrics: 1) LPIPS [39], 2) SSIM [40] and 3) PSNR. The results are shown in Figure 2.

Our method significantly outperforms the baselines on LPIPS and SSIM metrics for both the UMD-HoI and bimanual datasets. On the PSNR metric, our approach has a similar performance as that of HoI-GAN. SSIM, PSNR, and LPIPS metrics are limited in accurately measuring the



(a) LPIPS (b) SSIM (c) PSNR (d) LPIPS (e) SSIM (f) PSNR

Figure 2: Comparison of our method with baselines on UMD-HOI (a-c) and Bimanual (d-f) datasets

Figure 3: Limitations of pose only intermediate representation as compared to our layout representation.

performance of video prediction methods. These are frame-based metrics that assume that the predicted video is aligned temporally with the ground truth video, which is not valid. For example, an actor could be performing an action slowly in the predicted videos compared to the ground truth, which leads to phase shift. Standard video evaluation metrics fail to capture these properties. Therefore we further evaluate our model on the bimanual dataset using three additional metrics: 1) FID score [41], 2) Inception-DTW score, and 3) mAP of the objects detected by a FasterRCNN based detector in the generated frames. FID score is more widely accepted and a standard metric for quantitatively evaluating an image generation model's effectiveness. We use this metric to assess the quality of the generated frames. While the FID score measures the overall frame quality, it doesn't consider the generated videos' temporal quality. For estimating this, we propose a pre-trained Inception network-based Dynamic Time Warping metric (I-DTW). Like FID and Inception scores [42], I-DTW relies on features from a pre-trained inception network to measure the temporal video quality. Since inception based metrics have been used widely for evaluating the image generation models, we extend it to evaluate video generation models. To compute the I-DTW score, we extract the normalized Inception features for the generated video sequence and the ground truth sequence and calculate the DTW distance between these two sequences in the feature space. Lower the I-DTW score, better the generated video since it is closer to the ground truth in the feature space learned by the inception network. Our method significantly outperforms the baselines on these metrics, as shown in Table 1.

4.2. Ablations

We perform a detailed ablation analysis to measure the usefulness of various components of our model.

Importance of layout generation. The layout generation step is essential to our model for generating a good video, and we argue that pose alone is not sufficient. Figure 3 shows the importance of object locations and poses in predicting a video. For this experiment, we use our layout generator's output to train a pose only to frame mapping, as described in HVP [2]. We compare this visually against our model's output. Note that for the pose only model, while the posture changes, the objects more or less remain in the same position as the guidance frame. In Figure 3, a red bowl is an object of interaction that changes its location over time. However, in the pose only model, the red bowl doesn't move at all.

Table 2: Quantitative evaluation of the importance of various modules in our model. Each row is a model trained with different loss functions indicated in the columns.

	$\mathcal{L}_{pix}$	$\mathcal{L}_{feat}$	Frame GAN	Object GAN	Video GAN	FID(↓)	I-DTW(↓)	mAP(↑)
HORN Models (ours)	✓	✓	-	-	-	49.05	12.93	53.54
	✓	✓	✓	-	-	40.44	10.75	57.21
	✓	✓	✓	✓	-	38.30	10.64	58.67
	✓	✓	✓	✓	✓	**35.83**	**10.52**	**59.04**

Table 3: Results on Bimanual dataset. **++** implies **improved baselines** for HoI-GAN and MoCoGAN respectively. Ab1, Ab2 and Ab3 are architecture ablations of our model without relational reasoning, without prior layouts and conditioning on previous generated frame instead of the first frame respectively. SSIM, PSNR and LPIPs are average values.

	FID	mAP	I-DTW	SSIM	PSNR	LPIPS
HoI-GAN++	210.42	7.28	13.25	0.74	18.47	0.78
MoCoGAN++	181.29	7.69	14.92	0.67	16.81	0.73
Ab1	65.73	55.46	11.38	0.80	17.99	0.83
Ab2	42.97	58.28	**10.19**	**0.81**	**18.60**	**0.86**
Ab3	140.80	34.47	14.69	0.67	15.75	0.69
Ours	**35.82**	**59.04**	10.52	**0.81**	**18.61**	**0.86**

Similarly, for the second example of "screwing a harddrive", the objects are not predicted by the pose only model. In contrast, our model can capture both pose and objects evolution over time. To quantitatively evaluate the layout generation step, we compare our method with HVP by computing the FID, I-DTW, and mAP metrics, as shown in Table 1. Our method outperforms HVP on all the three metrics that confirm that pose alone is not sufficient to generate high-quality videos. The proposed way of predicting the entire layout sequence is vital for generating high-quality videos.

Importance of various loss functions. The success of our layout-to-frame generation step is due to the proposed discriminator losses such as Frame-GAN, Object-GAN, and Video-GAN. We evaluate each of these loss functions' usefulness and contribution in Table 2 by training the models in different configurations. First, we train the model without any discriminator losses (no GAN). For the next two models, we train one with Frame-GAN and the other with Frame and Object GANs. Finally, we train a model with all the loss functions. We observe from Table 2 that Frame GAN significantly improves the frames' visual quality, which helps synthesize better objects (we can see that mAP increases by 4% with the introduction of frame GAN). Object and Video GANs have moderate contributions where Object GAN improves the mAP by 1.5%, and video gan improves it by 0.4%.

Architecture ablation analysis. We evaluate the effectiveness of the proposed architecture by training models in different configurations such as without relational reasoning (Ab1), without prior layout for video generation (Ab2) and conditioning the layout generator on previous generated frame instead of the first frame (Ab3). We also compare these models with improved baselines where they are trained by augmenting the input with corresponding pose and object information in the channel dimension. The results of these experiments are shown in Table 3. We do not observe any significant improvements in the baselines and our proposed method still outperforms these. Our method shows improvements in FID and mAP without degradation in other metrics compared to not using prior layouts.

4.3. Qualitative Analysis

This section presents a detailed qualitative analysis of our model and its generalizations.

We visualize the layout sequence and the corresponding video predicted by our model in Figure 4. Figure 4(a) shows a few examples of generated object sequences. Observe that our model learns to predict the sequences with a slight phase shift with respect to the ground truth. Therefore, our model is not merely trying to mimic the ground truths but is learning to generate new sequences.

Figure 4(b) is an example of "cooking" that is performed by interacting with a bowl and a whisk. The first two rows correspond to the ground truth sequence, and the next two rows correspond to sequences predicted by our model. Observe that in the predicted layout sequence, as the bowl moves up (red box), the pose of the person changes such that the wrist gets very close to the bowl, and simultaneously in the pixel space, the bowl starts to move up. This shows that our model predicts the correct layout sequence, which is aligned temporally with the generated video.

In this experiment, the performer's test set actions have no overlap with the same performer's training set actions. Therefore, in the testing phase, the performer's target action is a novel action for this performer as it was not part of the training set. Our qualitative results show that the model can successfully generate the video by learning to transfer the action to the test set performer. This validates that our model can generalize well to new performers.

4.4. Visualization on UMD-HOI Dataset

Figure 5, shows a few examples of the generated videos using the UMD-HoI dataset. Figure 5(a) is an example of a "speaking in a telephone action" performed by an out of sample performer. The model has never seen this performer during training. We observe that our model is able to generalize well to new people while faithfully generating the target video. Figure 5(b, c) shows examples of generated videos by changing an object's target action. In these ex-

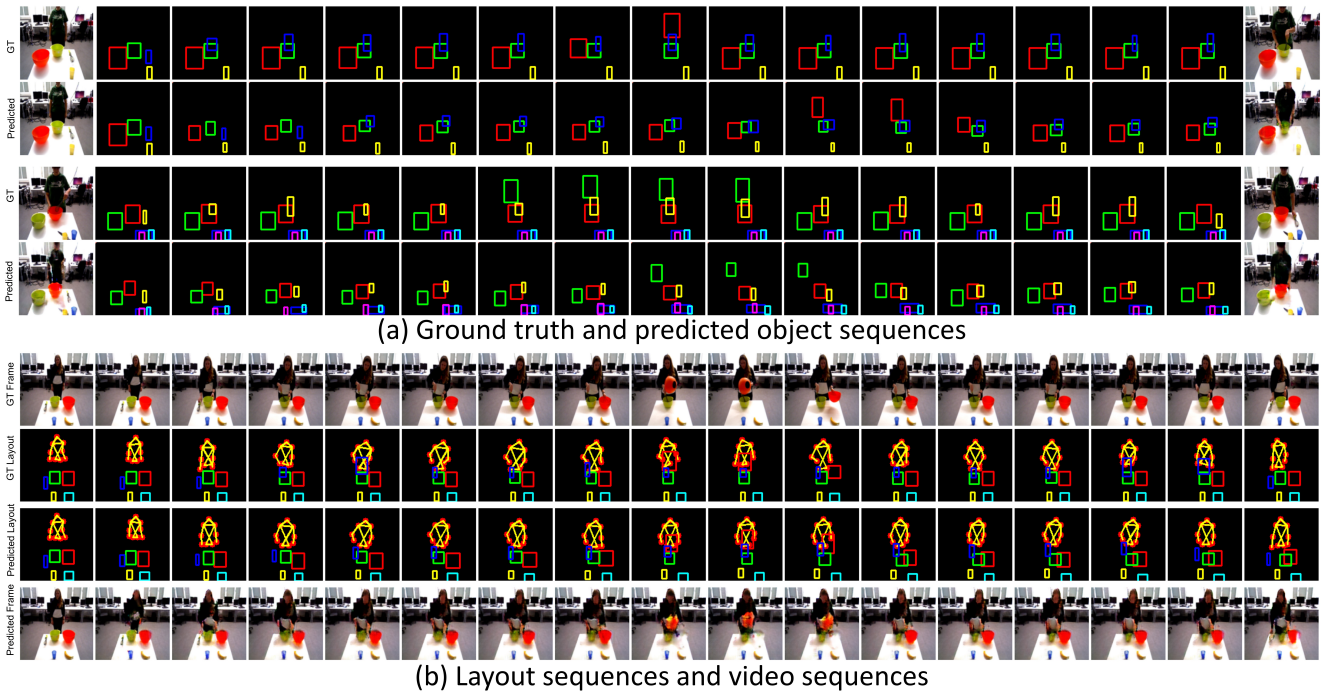


Figure 4: Qualitative results on Bimanual dataset.

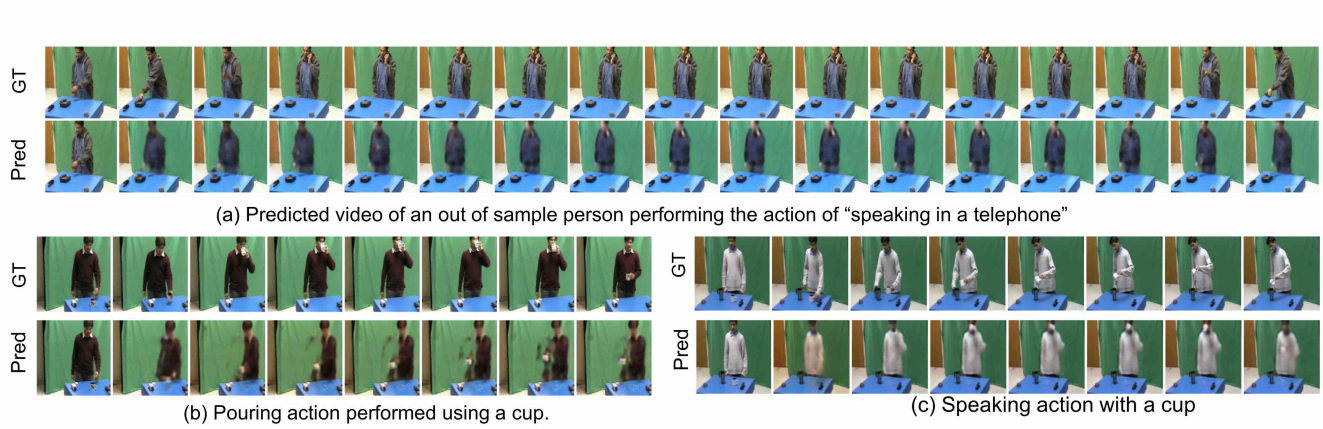


Figure 5: Qualitative results and generalization study on UMD-HOI dataset

amples we ask the model to perform: 1) pouring from a cup and 2) speaking with a cup actions. We observe that the model is able to reasonably generate the videos of new target actions with the objects.

5. Conclusion

We presented a novel hierarchical video prediction model for human-object interactions by disentangling the video prediction into two stages. The first stage involves predicting a layout sequence, which is a combination of pose and object sequences. Since the spatio-temporal motion of poses and objects is highly correlated, a relational reasoning module is learned to model their interactions. Finally, in the second stage, a layout sequence to video mapping is learned to generate high-fidelity videos. We presented an extensive evaluation of our method to show that the model can learn novel interactions with objects.

Acknowledgements. This work was supported by the DARPA SAIL-ON program via ARO contract no. W911NF2020009 and IARPA via contract no. D17PC00345. The views and conclusions are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation.

References

[1] Yijun Li, Chen Fang, Jimei Yang, Zhaowen Wang, Xin Lu, and Ming-Hsuan Yang. Flow-grounded spatial-temporal video prediction from still images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 600–615, 2018. 1

[2] Ruben Villegas, Jimei Yang, Yuliang Zou, Sungryull Sohn, Xunyu Lin, and Honglak Lee. Learning to generate long-term future via hierarchical prediction. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3560–3569. JMLR. org, 2017. 1, 2, 4, 6

[3] Jacob Walker, Kenneth Marino, Abhinav Gupta, and Martial Hebert. The pose knows: Video forecasting by generating pose futures. In *Proceedings of the IEEE international conference on computer vision*, pages 3332–3341, 2017. 1, 2

[4] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852, 2015. 2

[5] Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. In *Advances in neural information processing systems*, pages 2863–2871, 2015.

[6] Michael Mathieu, Camille Couprie, and Yann LeCun. Deep multi-scale video prediction beyond mean square error. *arXiv preprint arXiv:1511.05440*, 2015.

[7] Nal Kalchbrenner, Aäron van den Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu. Video pixel networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1771–1779. JMLR. org, 2017.

[8] T Xue, J Wu, K Bouman, and B Freeman. Probabilistic modeling of future frames from a single image. *Advances in Neural Information Processing Systems (NIPS)*, 2, 2016.

[9] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing motion and content for natural video sequence prediction. *arXiv preprint arXiv:1706.08033*, 2017.

[10] Emily L Denton et al. Unsupervised learning of disentangled representations from video. In *Advances in neural information processing systems*, pages 4414–4423, 2017.

[11] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. In *Advances in neural information processing systems*, pages 64–72, 2016. 2

[12] Nevan Wichers, Ruben Villegas, Dumitru Erhan, and Honglak Lee. Hierarchical long-term video prediction without supervision. *arXiv preprint arXiv:1806.04768*, 2018. 2

[13] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris Metaxas. Learning to forecast and refine residual motion for image-to-video generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 387–403, 2018.

[14] Haoye Cai, Chunyan Bai, Yu-Wing Tai, and Chi-Keung Tang. Deep video generation, prediction and completion of human action sequences. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 366–382, 2018.

[15] Jun-Ting Hsieh, Bingbin Liu, De-An Huang, Li F Fei-Fei, and Juan Carlos Niebles. Learning to decompose and disentangle representations for video prediction. In *Advances in Neural Information Processing Systems*, pages 517–526, 2018.

[16] Ceyuan Yang, Zhe Wang, Xinge Zhu, Chen Huang, Jianping Shi, and Dahua Lin. Pose guided human video generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018. 2

[17] Bangpeng Yao, Xiaoye Jiang, Aditya Khosla, Andy Lai Lin, Leonidas Guibas, and Li Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *2011 International Conference on Computer Vision*, pages 1331–1338. IEEE, 2011. 2

[18] Vincent Delaitre, Josef Sivic, and Ivan Laptev. Learning person-object interactions for action recognition in still images. In *Advances in neural information processing systems*, pages 1503–1511, 2011.

[19] Abhinav Gupta and Larry S Davis. Objects in action: An approach for combining action understanding and object perception. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

[20] Abhinav Gupta, Aniruddha Kembhavi, and Larry S Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(10):1775–1789, 2009.

[21] He Wang, Sören Pirk, Ersin Yumer, Vladimir G Kim, Ozan Sener, Srinath Sridhar, and Leonidas J Guibas. Learning a generative model for multi-step human-object interactions from videos. In *Computer Graphics Forum*, volume 38, pages 367–378. Wiley Online Library, 2019.

[22] Weidong Yin, Ziwei Liu, and Leonid Sigal. Person-in-context synthesis with compositional structural space. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2827–2836, 2021. 2

[23] Megha Nawhal, Mengyao Zhai, Andreas Lehrmann, and Leonid Sigal. Zero-shot generation of human-object interaction videos. *arXiv preprint arXiv:1912.02401*, 2019. 2, 5

[24] Fabien Baradel, Natalia Neverova, Christian Wolf, Julien Mille, and Greg Mori. Object level visual reasoning in videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 105–121, 2018. 2

[25] Bo Dai, Yuqi Zhang, and Dahua Lin. Detecting visual relationships with deep relational networks. In *Proceedings of the IEEE conference on computer vision and Pattern recognition*, pages 3076–3086, 2017. 2

[26] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Kevin Murphy, Rahul Sukthankar, and Cordelia Schmid. Actor-centric relation network. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 318–334, 2018. 2

[27] Chen Sun, Abhinav Shrivastava, Carl Vondrick, Rahul Sukthankar, Kevin Murphy, and Cordelia Schmid. Relational action forecasting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 273–283, 2019. 2

[28] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *2018 ieee winter conference on applications of computer vision*

(wacv), pages 381–389. IEEE, 2018. 2

[29] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8359–8367, 2018.

[30] Bangpeng Yao and Li Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 17–24. IEEE, 2010. 2

[31] Alexandre Alahi, Kratarth Goel, Vignesh Ramanathan, Alexandre Robicquet, Li Fei-Fei, and Silvio Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 961–971, 2016. 2

[32] Michael B Chang, Tomer Ullman, Antonio Torralba, and Joshua B Tenenbaum. A compositional object-based approach to learning physical dynamics. *arXiv preprint arXiv:1612.00341*, 2016. 2

[33] Sjoerd Van Steenkiste, Michael Chang, Klaus Greff, and Jürgen Schmidhuber. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. *arXiv preprint arXiv:1802.10353*, 2018. 2

[34] Rasmus Palm, Ulrich Paquet, and Ole Winther. Recurrent relational networks. In *Advances in Neural Information Processing Systems*, pages 3368–3378, 2018. 5

[35] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 5

[36] Abhinav Gupta, Aniruddha Kembhavi, and Larry S Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(10):1775–1789, 2009. 5

[37] Christian R. G. Dreher, Mirko Wächter, and Tamim Asfour. Learning object-action relations from bimanual human demonstration using graph networks. *IEEE Robotics and Automation Letters (RA-L)*, 5(1):187–194, 2020. 5

[38] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. Mocogan: Decomposing motion and content for video generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1526–1535, 2018. 5

[39] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 5

[40] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 5

[41] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637, 2017. 6

[42] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016. 6