6 Supplementary

This section contains the supplementary information supporting the content in the main paper. We also provide a video (see `1586_video.mp4` in the supplementary material) showing all the qualitative samples together.

- Qualitative evaluation and analysis of HOI-GAN to supplement Section 4.3.
- Qualitative evaluation of baselines: samples generated using baselines to supplement Section 4.3.
- Additional quantitative evaluation of our model to supplement Section 4.3.
- Details of preprocessing and data splits for each dataset to supplement Section 4.1.
- Implementation details of our model to supplement Section 3.

6.1 Qualitative Evaluation and Analysis of HOI-GAN

Note: Please open the video file `1586_video.mp4` in a suitable video player to see the samples together.

Qualitative Evaluation (GS1). We present samples generated using our HOI-GAN in generaton scenario 1 (GS1). In GS1 setting, the target context image and the target action-object composition are unseen during training. Thus, the context image is from the test set (obtained in zero-shot compositional setting) and the object mask in the context image corresponds to the target object. As shown in Figure 6, our model is able to create objects and enact the prescribed action on the object in the given context. Figure 6 also shows the real videos from the test set corresponding to the given compositions and context frame. The results clearly demonstrate that our model is able to generate realistic videos depicting the given action-object in the given context. The visual appearance of objects and actions (hand movements) are somewhat different in the generated videos compared to the corresponding real video because the model had to generalize based on other depictions of the object and action that were seen separately in training. Nevertheless, the results show that the generated video is also a realistic depiction of the target composition showing the target action on the target object in the target context.

Qualitative Evaluation (GS2). We present samples generated using our HOI-GAN in generation scenario 2 (GS2). In GS2 setting, the target context background is seen during training while the target action-object composition is unseen. Specifically, the context image is from a video in the training set and the object mask in the context image corresponds to an object different than the target object. Also, the action corresponding to the context image may or may not be same as the target action. As such, the background may or may not be fully amenable for the target action-object composition. As shown in Figure 7, our model is able to create the required objects and enact the prescribed actions on the objects in the given context background. Moreover, our model is also able to modify the background as and when needed based on the target composition

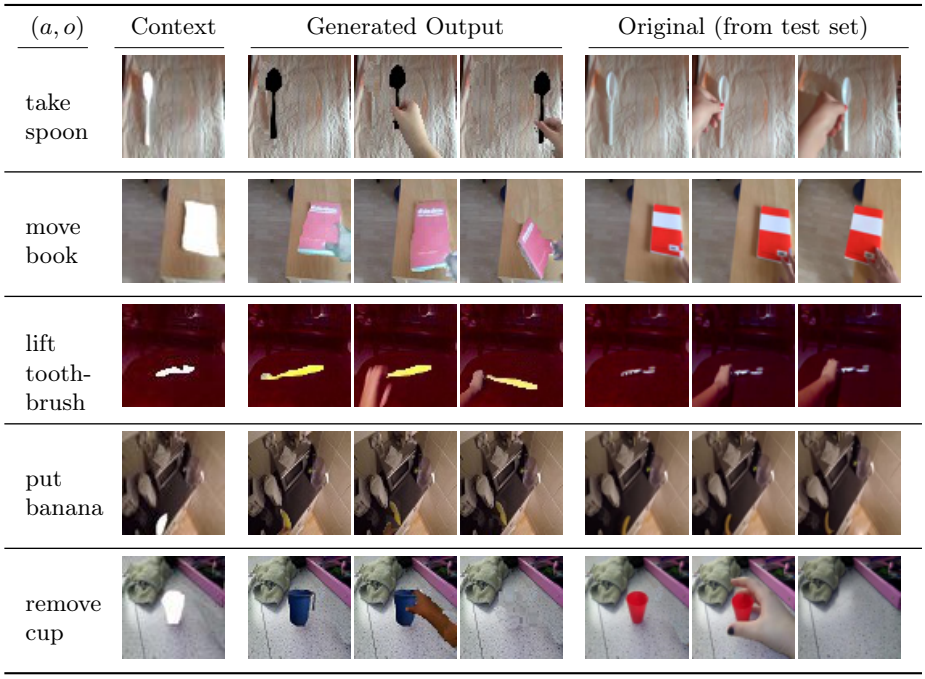


Fig. 6. Qualitative Evaluation (GS1). Samples generated using our model in Generation Scenario 1, *i.e.*, both the target context image and the target action-object (a, o) composition are unseen during training. We provide 3 frames of the generated output and 3 frames of the original video (same context, action, object) from the test set for comparison – please refer to the video `1586_video.mp4` to see the corresponding video samples.

to be generated. The results clearly demonstrate that our model is able to generate realistic videos depicting the given action-object in the given context. In particular, the *move book* sample provides a comparison with its corresponding sample of *move book* in the GS1 setting (see Figure 6). In the GS2 setting seen here, the mask in the context frame corresponds to a handbag. The model is able to align the orientation of the book with the provided mask of the handbag and fit the object *book* in the mask. In contrast, the size of *book* with respect to the mask in this case is different from that seen in the GS1 example.

In addition to showing the diversity in generated samples, we also generate videos corresponding to various sets of compositions with the same target context, same target action and different target objects. Samples in Figure 8 indicate our model is able to synthesize videos with the same action in the same context being performed on multiple objects differently. For instance, hand(s) appear from different directions and look different. Our model is also able to scale the objects appropriately based on the mask (see *lift handbag* in Figure 8).

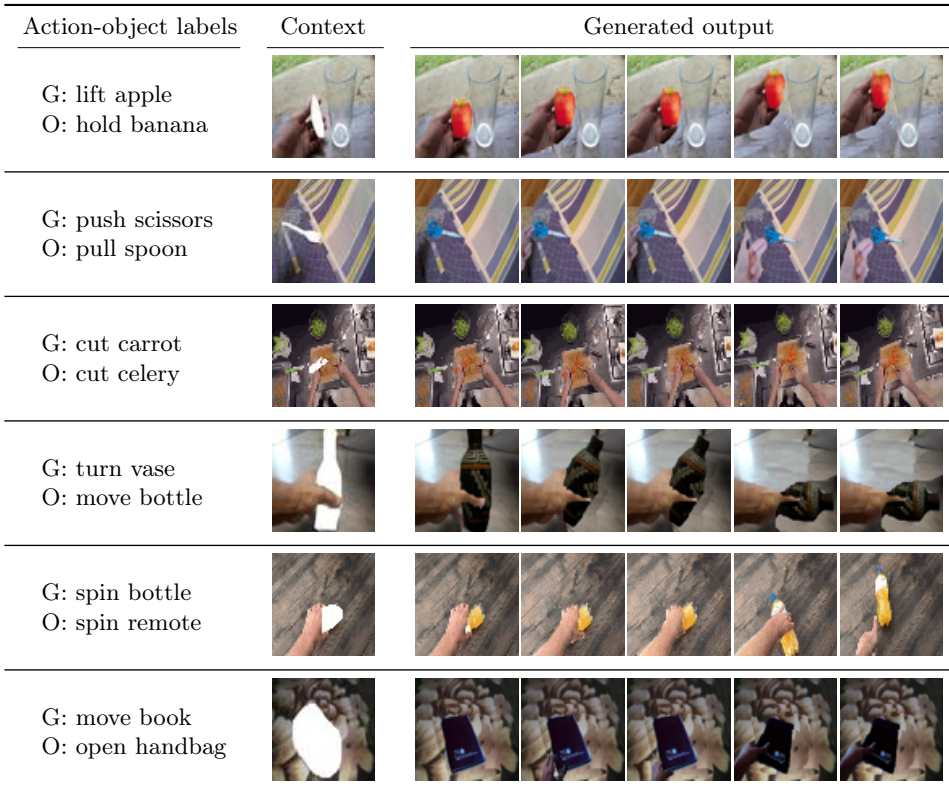


Fig. 7. Qualitative Evaluation (GS2). Samples generated using our model in Generation Scenario 2, *i.e.*, target action-object composition are unseen during training but target context image is seen with an object different from target object and a same/different action from target action. Thus, the overall target compositions comprising object, action and context are unseen during training. 'G' indicates the target action-object composition and 'O' indicates the action-object composition of the video (in the training set) from which the context image is chosen. We provide 5 frames for each generated video sample in the figure – please refer to the video `1586_video.mp4` to see the corresponding video samples.

Furthermore, we also generate videos corresponding to various sets of compositions with the same target context, same target object and different target actions. Samples in Figure 9 indicate that our model is able to synthesize videos with different actions being performed on same object. In particular, the model is able to generate the same object with different and diverse set of visual appearances (*e.g.* the *bowl*s in Figure 9 look different) and perform the different actions upon them.

How does HOI-GAN generalize over compositions? Recall, the generation in this paper is performed in a zero-shot compositional setting, i.e., actions

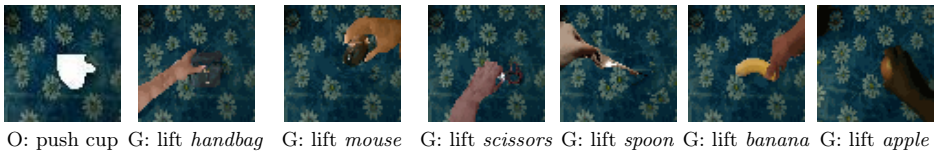


Fig. 8. Qualitative Evaluation (GS2 - same action, same context, different objects). Samples generated using HOI-GAN in Generation Scenario 2 corresponding to a set of compositions with same context frame, same action and different objects. 'G' indicates the target action-object composition and 'O' indicates the action-object composition of the video (in the training set) from which the context image is chosen. We show the context frame with mask on the left in each row. We provide 1 frame for each generated video sample in the figure – please refer to the video `1586_video.mp4` to see the corresponding video samples.

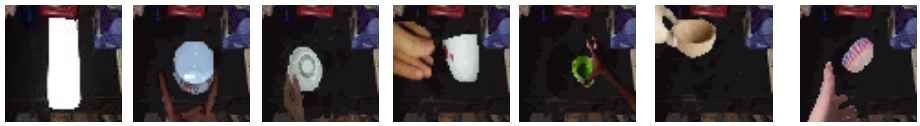


Fig. 9. Qualitative Evaluation (GS2 - same object, same context, different actions). Samples generated using HOI-GAN in Generation Scenario 2 corresponding to a set of compositions with same context frame, same object and different actions. 'G' indicates the target action-object composition and 'O' indicates the action-object composition of the video (in the training set) from which the context image is chosen. We show the context frame with mask on the left in each row. We provide 1 frame for each generated video sample in the figure – please refer to the video `1586_video.mp4` to see the corresponding video samples.

and objects are seen independently in certain compositions during training but the target action-object compositions are unseen during training. Intuitively, during this process, our model is able to effectively disentangle actions and objects. Therefore, when given a previously unseen target action-object composition for generation, our model is able to bring together or combine the information (distributed over the training set) in a meaningful way to synthesize realistic videos corresponding to the unseen composition. Consider the video corresponding to *lift handbag* in Figure 10, the model has seen different handbags in different contexts with different actions (other than *lift*), and has also seen different instances of the action *lift* being performed on objects other than *handbag*

Table 5. Quantitative Evaluation (Effect of Word Embeddings). Comparison of HOI-GAN with C-VGAN, C-TGAN, and MoCoGAN baselines using one-hot encoded labels instead of embeddings as conditional inputs(default version). (see section 4.3). Arrows indicate whether lower (↓) or higher (↑) is better. [I: inception score; S: saliency score; D: diversity score]

Model	EPIC						SS					
	GS1			GS2			GS1			GS2		
	I↑	S↓	D↑	I↑	S↓	D↑	I↑	S↓	D↑	I↑	S↓	D↑
C-VGAN [67]	1.1	52.1	0.4	1.1	52.1	0.4	2.1	45.6	0.8	1.9	45.1	0.5
C-TGAN [58]	1.6	65.4	0.4	2.2	28.1	0.5	2.4	36.2	1.1	1.7	42.8	0.6
MoCoGAN [65]	2.6	25.4	1.0	2.0	34.9	1.0	2.9	22.8	1.3	2.4	27.4	1.5
(ours) HOI-GAN (bboxes)	3.8	18.5	2.1	3.2	24.1	2.4	4.9	26.2	2.7	4.0	25.2	2.4
(ours) HOI-GAN (masks)	**4.3**	**16.5**	**2.5**	**3.9**	**20.2**	**1.6**	**5.8**	**15.8**	**3.0**	**4.5**	**23.7**	**2.8**

in different contexts. Given all this information, our model is able to combine the relevant information and synthesizes a video corresponding to a handbag being lifted in the given context. Similarly, we show two other compositions and the corresponding training samples of the action and object that might have helped the model during the particular generations.

Failure Cases (Additional Discussion). We showed two failure cases in Section 4.3. Particularly, for *open microwave*, while the model is able to generate a microwave object having seen it in training, it is not able to blend it into the background context. This is because the mask covers most of the background and the model gets very little information about the context. In the case of *cut peach*, the model is unable to generate the pieces well because the interior of a peach differs from its exterior. This is in contrast to *cut carrot* (see Figure 7) wherein the interior of the carrot is similar to its exterior, and hence the model is able to generate the pieces properly.

6.2 Qualitative Evaluation of Baselines

In this section, we provide the middle frame of samples generated using the baselines: C-VGAN, C-TGAN, and MoCoGAN for a given composition of context frame, action and object as conditional inputs. Figure 11 shows the samples generated from these baselines. Figure 11 also shows the samples generated using HOI-GAN corresponding to the given composition for comparison. The results clearly show that our HOI-GAN is able to synthesize more realistic videos. Moreover, this also supports the quantitative evaluation conducted in the main paper. Please open the video file `1586_video.mp4` in a suitable video player to see the samples together.

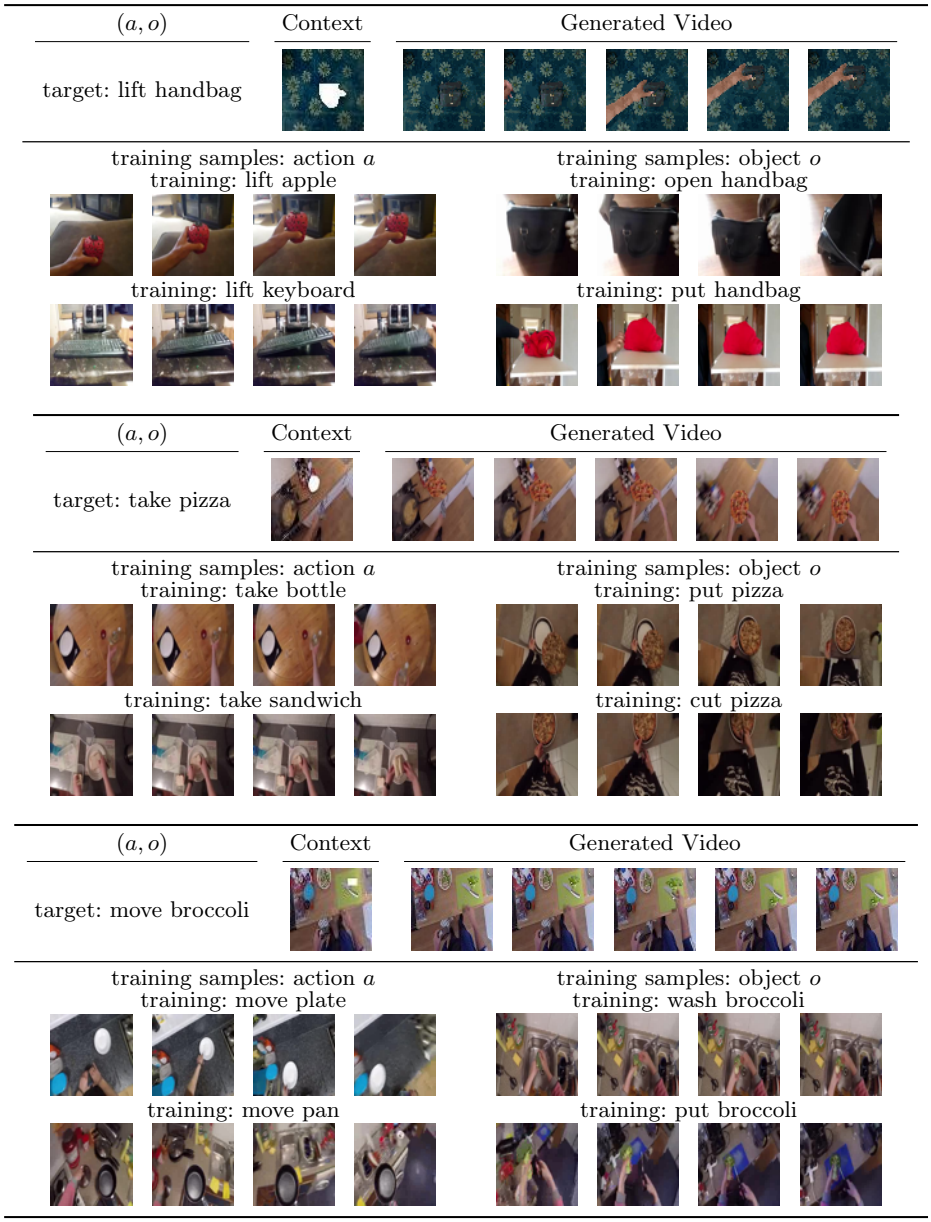


Fig. 10. How does HOI-GAN generalize over compositions?. Training samples in the data to illustrate that HOI-GAN leverages the information available during training and learns to combine them in a meaningful way. This ability allows HOI-GAN to generalize over unseen compositions of action, object and context. We provide a few frames for each sample in the figure – please refer to the video `1586_video.mp4` to see the corresponding video samples.

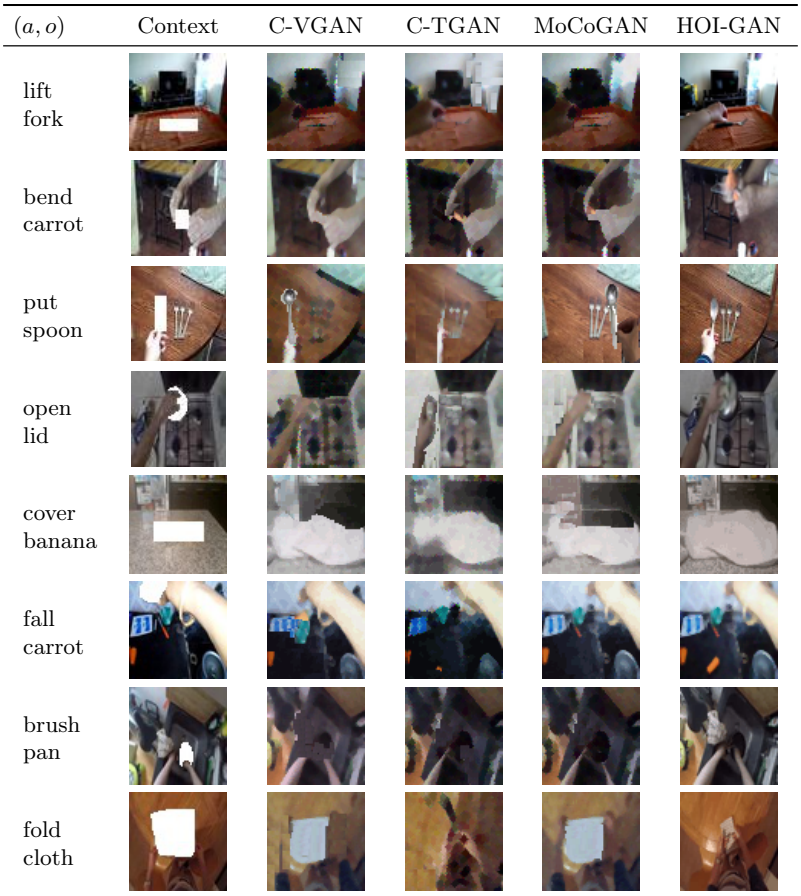


Fig. 11. Qualitative Evaluation (Baselines). Samples generated using the baseline models (C-VGAN, C-TGAN, and MoCoGAN) in different generation scenarios. We also present the sample generated using HOI-GAN in the given composition of action-object (a, o) pair and context image for comparison. We provide 1 frame for each generated video sample in the figure – please refer to the video `1586_video.mp4` to see the corresponding video samples.

Table 6. Quantitative Evaluation (FID). Fréchet Inception Distance (FID) comparison of HOI-GAN with C-VGAN, C-TGAN, and MoCoGAN baselines. Lower FID implies higher quality.

Model	EPIC		SS	
	GS1	GS2	GS1	GS2
C-VGAN [67]	18.8	23.7	15.1	20.5
C-TGAN [58]	17.2	21.3	13.6	18.2
MoCoGAN [65]	14.6	19.9	11.4	17.5
HOI-GAN (ours)	**8.1**	**10.2**	**7.2**	**8.3**

6.3 Additional Quantitative Evaluation

In this section, we provide results of the additional quantitative evaluation of our HOI-GAN to illustrate the effect of using semantic embeddings.

Effect of Word Embeddings. In our approach, we use word embeddings for the action and object labels to share information among semantically similar categories during training. To demonstrate the impact of using embeddings, we also trained HOI-GAN using one-hot encoded labels corresponding to both actions and objects. We observe that these models perform worse than the models trained using semantic embeddings (refer last two rows of Table 2 in the main paper and Table 5). Nevertheless, our models still outperform the baselines (refer to Table 5).

Evaluation using FID We primarily used video classifier based Inception score as a metric for quantitative evaluation. As an additional measure to evaluate the quality of generated samples, we also report another Fréchet Inception Distance (lower is better) in Table 6. We compute the scores following [71]. Specifically, we use a Kinetics-pretrained ResNext-101 video classification model as the feature extractor. The results show that videos generated using HOI-GAN are more realistic than those created using baselines.

Classification Experiments To further demonstrate the effectiveness of our model, we conduct classification experiments using generated videos in different settings. The experiments are described as follows.

Finetuning on real and evaluation on generated videos. We finetuned a Kinetics-pretrained ResNext-101 classifier model (same as the one used to compute evaluation metrics). We used this finetuned video classifier to classify generated videos. We report the classification performance of the classifier in Table 7. The evaluation is done for the generated videos corresponding to the unseen compositions only. For our HOI-GAN and baseline MoCoGAN, we calculated the accuracy on the videos generated by the models (with unseen compositions as conditional input). For comparison, we also report the classification performance

Table 7. Classification Experiments. Accuracy of a video classifier when finetuned on real videos from the dataset and evaluated on generated videos corresponding to unseen action-object compositions.

Classifier Setting	EPIC	SS
Chance	<0.1	<0.1
Finetuned on real / Evaluated on generated (MoCoGAN)	11.0	20.6
Finetuned on real / Evaluated on generated (HOI-GAN)	35.4	53.6
Finetuned on real / Evaluated on real	51.7	68.8

Table 8. Classification Experiments. Accuracy of a video classifier when finetuned on generated videos and evaluated on real videos for unseen action-object compositions.

Classifier Setting	EPIC	SS
Finetuned on generated / Evaluated on real	33.1	46.3
Finetuned on real / Evaluated on real	51.7	68.8

on a test set containing real videos of the same compositions – this serves as the upper bound. We observe that the performance on videos generated using HOI-GAN is considerably better than that on videos generated using MocoGAN (best performing baseline) and much closer to the performance on real videos. This indicates that our proposed framework is consistently generating realistic videos conditioned on given action-object compositions.

Finetuning on generated and evaluation on real videos. We used a Kinetics-pretrained ResNext-101 video classifier (same as the one used to compute evaluation metrics) and fine-tuned it on a dataset containing only generated samples corresponding to the unseen action-object compositions. We report the classification performance in terms of accuracy of this classifier when evaluated on a test set containing real videos corresponding to unseen compositions (from the original dataset) in Table 8. For reference, we also report the classification performance on the same test set for the classifier fine-tuned on real videos. As expected, performance is lower than that using real videos, but the generated ones serve as a reasonable proxy for learning to recognize unseen compositions.

6.4 Preprocessing and Data Splits

As described in Section 4.1, we perform new splits of the dataset for the task of zero-shot HOI video generation. In this section, we provide the details of preprocessing and zero-shot compositional splits for datasets EPIC-Kitchens (EPIC) and 20BN-Something-Something V2 (SS).

EPIC: Processing and Splits. The EPIC-Kitchens dataset originally consists of 39,594 video samples of the form (V, a, o), *i.e.*, video V with action label a

and object label o, spanning 125 unique actions and 352 unique objects. We further filtered the dataset to ensure that the video samples contain both ground truth bounding box annotation and MaskRCNN output (NMS threshold = 0.7) in the frames uniformly sampled from a video. We interpolated the sequence if the number of such frames is less than 16. We then split the filtered dataset by action-object compositions to obtain train and test splits suitable for the zero-shot compositional setting, *i.e.*, all the unique object and action labels in combined dataset appear independently in the train split, however, a certain pair of action and object present in the test split is absent in train split and vice versa. Subsequently we obtained two splits: (1) train split containing 19,895 videos that overall depict 1,128 unique action-object compositions, and (2) test split containing 7,805 videos (568 unique action-object compositions). The final splits consist of compositions spanning 204 unique actions and 63 unique objects.

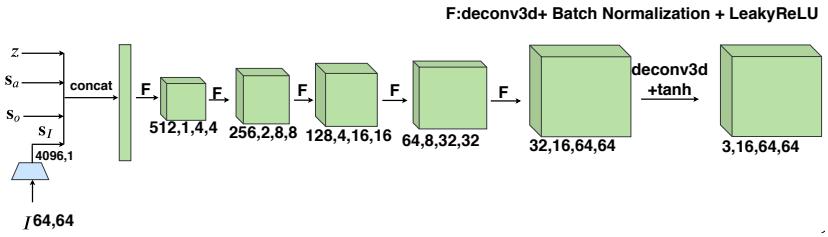
SS: Processing and Splits. The 20BN-Something-Something V2 dataset originally consists of 220,847 video samples of the form (V, l), *i.e.*, video V having a label l. To transform the dataset instances to the form (V, a, o), we applied NLTK POS-tagger on l and obtained verb a and noun o. In particular, we considered the verb tag (after stemming) in l as action label a. We observe that all instances of l begin with the present continuous form of a which is acting upon the subsequent noun. Therefore, we used the noun that appears immediately after the verb as object o. We merged the train and validation split of the transformed dataset. We further filtered the dataset to ensure that the video samples contain objects that can be detected using MaskRCNN (NMS threshold = 0.7) in the frames uniformly sampled from a video. We then split the transformed dataset by compositions of action a and object o to obtain the train and test splits suitable for the zero-shot compositional setting (same as EPIC). Subsequently, we obtained two splits: (1) train split containing 23,511 videos overall that overall depict 671 unique action-object compositions, and (2) test split containing 3,515 videos overall (135 unique action-object compositions). The final splits consist of compositions spanning 48 unique actions and 62 unique objects.

6.5 Implementation Details

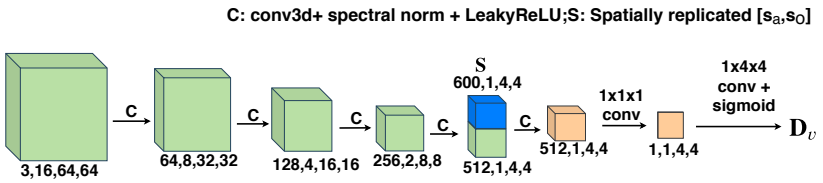
In our experiments, the convolutional layers in all networks, namely, $\mathbf{G}$, $\mathbf{D}_f$, $\mathbf{D}_g$, $\mathbf{D}_v$, $\mathbf{D}_r$ have kernel size 4 and stride 2. We generate a video clip consisting of $T = 16$ frames having $H = W = 64$. The noise vector $\mathbf{z}$ is of length 100. The parameters $w_0 = h_0 = 4$, $d_0 = 1$ and $N = 512$ for $\mathbf{D}_v$ and $w_0^t = h_0^t = 4$ and $N^{(t)} = 512$ for $\mathbf{D}_f$, $\mathbf{D}_g$, and $\mathbf{D}_r$. To obtain the semantic embeddings $\mathbf{s}_a$ and $\mathbf{s}_o$ corresponding to action and object labels respectively, we use Wikipedia-pretrained GLoVe [55] embedding vectors of length 300. We provide further implementation details of our model architecture in the supplementary section. For training, we use the Adam [37] optimizer with learning rate 0.0002 and $\beta_1 = 0.5$, $\beta_2 = 0.999$. We train all our models with a batch size of 32. We use dropout (probability = 0.3) [59] in the last layer of all discriminators and all layers (except first) of the generator.

Relational discriminator. We used the final output layer of MaskRCNN, that comprises a list of bounding boxes, a list of segmentation masks and a list of labels corresponding to each detection. We used `https://github.com/facebookresearch/maskrcnn-benchmark` repository to obtain the detection output. The same list of bounding boxes have been used for real and generated. Then, using each bounding box in the output, we crop the visual region from the corresponding frame. These crops will correspond to the nodes of spatio-temporal graph. These cropped visual regions are resized to $3 \times 16 \times 16$ ($C \times H \times W$) and their position (bounding box top-left coordinates normalized with respect to the image size) and their original aspect ratio (bounding box height and width normalized with respect to image size) are collectively used for node feature representation (Refer to Figure 3 for illustration). We used a `conv` module (shared weights for all crops), i.e., convolutional layers (stride=2, kernel size=4) and obtain a convolutional embedding for resized visual regions of size 4096 appended with 4 additional numbers corresponding to position and aspect ratio. We design Graph Convolution Layer using the implementation of Graph Convolution Network (GCN) available at `https://github.com/tkipf/pygcn`. We used 7 such Graph Convolution layers: initial layer converts the feature size to 4096 and output feature size of the node is doubled every two layer in next 6 layers. Until this stage, the node is represented using single dimensional vector. After pooling along the temporal axis, the channel dimension is reshaped to $256 \times 8 \times 8$ and the resulting tensor is of shape $K \times 256 \times 8 \times 8$ where K is the number of crops.

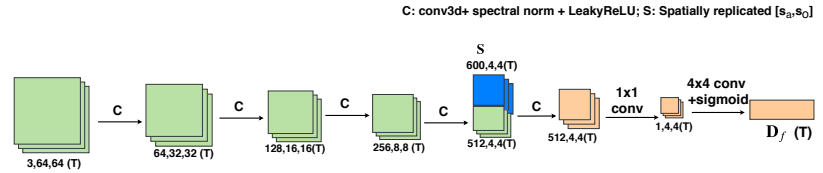
Architecture Details. As described in Section 3, our model comprises 5 networks involving a generator network and four discriminator networks. We provide the details of the architectures used in our implementation for the generator network, video discriminator, frame discriminator and relational discriminator in Figure 12. The architecture for gradient discriminator is same as that of the frame discriminator.



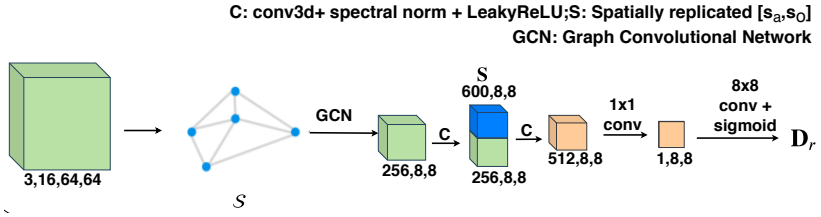
(i) Generator Network in HOI-GAN



(ii) Video Discriminator Network in HOI-GAN



(iii) Frame Discriminator Network in HOI-GAN



(iv) Relational Discriminator Network in HOI-GAN

Fig. 12. Architecture Details. Model architectures used in our experiments for: (i) Generator, (ii)Video Discriminator, (iii) Frame discriminator (gradient discriminator has similar architecture), (iv) Relational Discriminator. Best viewed in color on desktop.