Home Action Genome: Cooperative Compositional Action Understanding Supplementary Material

A. Dataset

1. Sensors and Modalities

We build multi-modal sensor kits for data collection as shown in Figure 5. This kit assists the creation of the multi-modal dataset by dramatically simplifying the data collection process through simple recording and timing synchronization. The data from all viewpoints are collected by these sensor-kits. Figure 6 shows the photo of the multi-modal sensor mounted on the head of a subject participant.

The audio and video data from the sensor is saved to a video file, and the sensor data is saved in the same file as additional tracks. By using lossless codecs like the Free Lossless Audio Codec (FLAC) or WavPack, we can save the sensor data with high fidelity. Both codecs support multi-channel audio in 8-32 bit integer format at frequencies as low as 1Hz. Sensor data is acquired over I2C with constant timing adjustments to maintain synchronization with audio and video.

HOMAGE contains 12 modalities with multiple viewpoints. Specifically, the infrared data is obtained by the Grid-EYE 8x8 pixel infrared array sensor. The RGB light data is obtained by a photodiode array sensor that provides an RGB spectral response with IR blocking filter. The sensor kit also includes an ambient light sensor that combines a broadband photodiode and an infrared-responding photodiode on a single CMOS-integrated circuit to provide ambient light data. The human presence sensor is a 4-channel nondispersive infrared (NDIR) sensor. The magnetic field data is acquired from a magnetometer in the sensor kit.

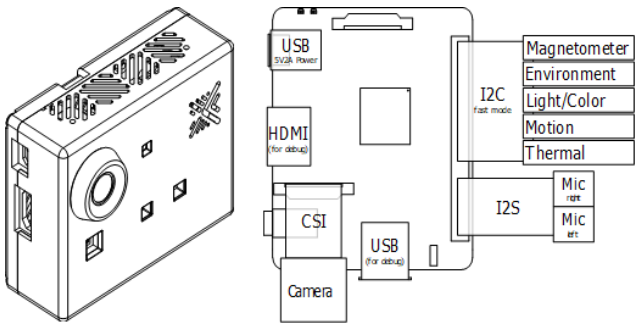

Figure 5: The multi-modal sensor kit used in data collection.

2. Data Synchronization

With the multimodal sensor kits, we collect human action data from different viewpoints. Specifically, we synchronize the data from different modalities by using the scheme below.

Figure 6: The multi-modal sensor, mounted on the head of the participant.

(1) The participants were instructed to start the activity displayed on the screen after they heard the start tone.

(2) The content of the participants was specified by activity unit (e.g. make bed). We do not specify a detailed sequence of atomic actions.

(3) We sounded the end tone when the participant's activity is finished. We synchronized the data of multiple sensor-kits using the signal of start/end tone.

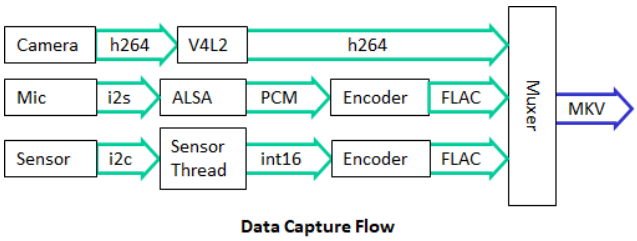

Figure 7: The flow chart of data collection.

3. Data Statistics

In this section, we include further details about the HOMAGE dataset. For the spatio-temporal scene graph, Figure 8 shows the most frequent object classes and Figure 9 shows the most frequent object relationships. Figure 10 shows the joint distribution of object classes and relationships. Figure 11 shows the distribution of the durations of

atomic actions.

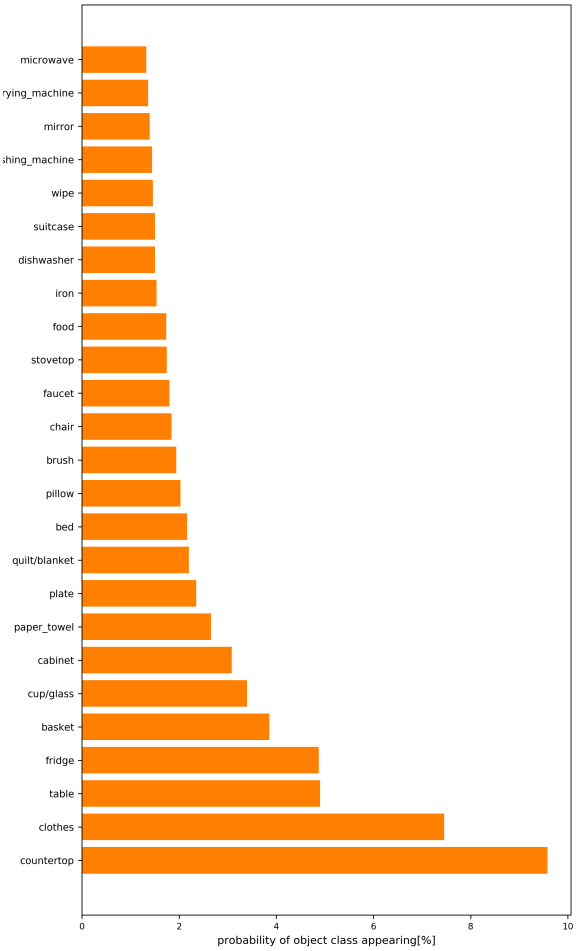

Figure 8: Distribution of object classes (top 25 objects)

B. Additional Experiments

1. Self-Supervised Pre-Training

Approach Our base backbone remains similar to the one we discuss in the main paper and the overall approach is inspired by [54]. To summarize, an aggregation function, $g(\cdot)$ takes a sequence $\{z_1, z_2, \ldots, z_j\}$ as input and generates a context representation $c_j = g(z_1, z_2, \ldots, z_j)$. In our setup, $z_j \in \mathbb{R}^{H' \times W' \times D}$ and $c_j \in \mathbb{R}^D$. D represents the embedding size and H', W' represent down-sampled resolutions

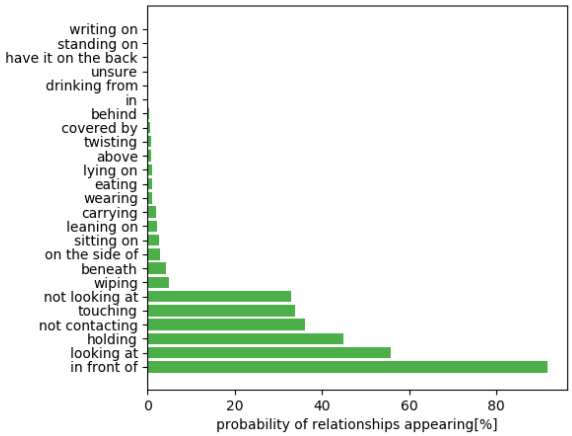

Figure 9: Distribution of relationship classes

as different regions in z_j represent features for different spatial locations. We define $z'_j = Pool(z_j)$ where $z'_j \in \mathbb{R}^D$ and $c = F(V)$ where $F(\cdot) = g(f(\cdot))$. In our experiments, $H' = 4, W' = 4, D = 256$.

To learn effective representations, we create a prediction task involving predicting z of future blocks similar to [54]. In the ideal scenario, the task should force our model to capture all the necessary contextual semantics in c_t and all frame-level semantics in z_t. We define $\phi(\cdot)$ which takes as input c_t and predicts the latent state of the future frames. The formulation is given in Eq. (3).

$$\begin{aligned} \widetilde{z}_{t+1} &= \phi(c_t), \\ \widetilde{z}_{t+1} &= \phi(g(z_1, z_2, \ldots, z_t)), \\ \widetilde{z}_{t+2} &= \phi(g(z_1, z_2, \ldots, z_t, \widetilde{z}_{t+1})), \end{aligned} \tag{3}$$

where $\phi(\cdot)$ takes c_t as input and predicts the latent state of the future frames. We then utilize the predicted $\widetilde{z}_{t+1}$ to compute $\widetilde{c}_{t+1}$. We can repeat this for as many steps as we want, in our experiments we restrict ourselves to predict till 3 steps in to the future.

Note that we use the predicted $\widetilde{z}_{t+1}$ while predicting $\widetilde{z}_{t+2}$ to force the model to capture long-range semantics. We can repeat this for a varying number of steps, although the difficulty increases tremendously as the number of steps increases as seen in [54]. In our experiments, we predict the next three blocks using the first five blocks.

Results To study the value of multiple viewpoints of the video data, we perform pre-training with the above learning framework weights to get a self-supervised initialization for our experiment. We first train our model in the self-supervised setting for 500 epochs. We use the pre-trained weights to initialize the ego-view and third-person view encoders and train with supervision loss to the same number

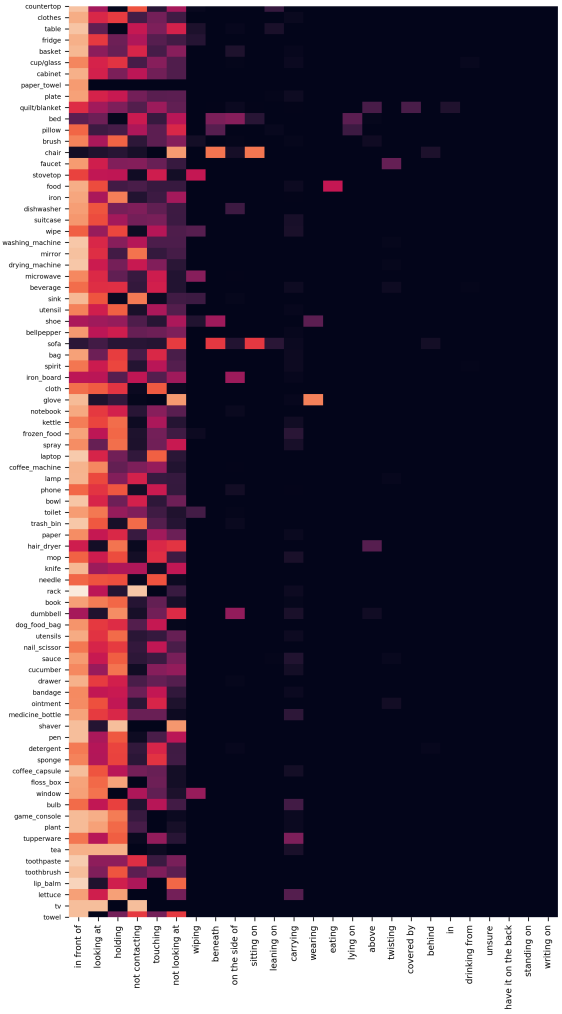

Figure 10: The co-occurrence statistics for objects and relationships in Home Action Genome.

of epochs as the randomly initialized baseline. Note that in the supervision phase, each modality is trained separately and no cross-modality loss is used. Table 8 shows that cooperative learning with different modalities results in distinctively improved performance compared to random initialization as we are able to utilize structural information naturally present in the examples. We also observe that the model with self-supervised pre-training converges faster than the baseline. This demonstrates the additional possibility of utilizing Home Action Genome to evaluate multi-modal self-supervision approaches.

2. Baseline with Oracle Scene Graphs

We provide a baseline for human action classification using oracle scene graphs. This experiment gives a rough reference of the upper bound of action inference using spatio-temporal information.

We represent the ground-truth scene graph input as a matrix M of size $n_{obj} \times n_{rel}$, with n_{obj} and n_{rel} be the number of object and relationship categories, respectively, initialized to be filled with 0. We encode a relationship with object category s, and relationship category r by setting $M[s, r]$ to be 1. The input representation is then flattened and fed into

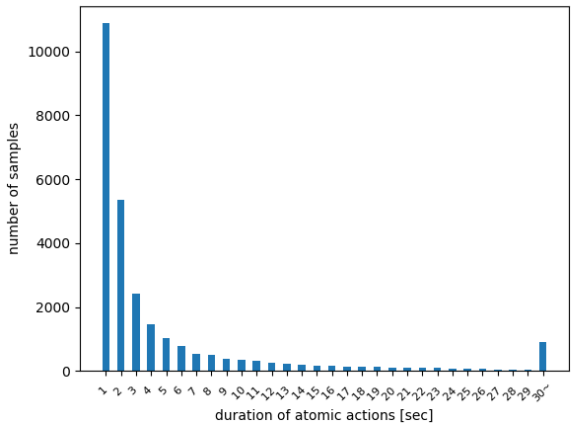

Figure 11: Duration of atomic action

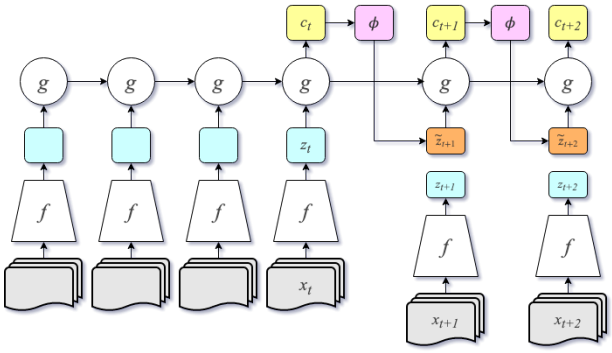

Figure 12: A diagram of the learning framework utilized. We look at features in a sequential manner while simultaneously trying to predict representations for future states.

Method	Ego-View	3^{rd} Person
SV	31.8	21.8
SS + SV	**33.1**	**24.8**

Table 8: Effect of self-supervised pre-training on atomic action classification. We see considerable performance improvements when initializing our model with pre-training using multi-modal self supervision.

an MLP-based encoder.

Table 9 shows the performance of activity classification using ground-truth scene graphs, with the encoding scheme described above. We observe that the modality of the ground-truth scene graph is very informative compared with the other modalities, highlighting the potential for scene graph prediction on human action understanding.

Acc1	Acc3
76.0	91.7

Table 9: Classification of activities using ground-truth scene graphs. Results are averaged over the two test splits.

3. Multi-Task Loss

As discussed in Section 4.3, we utilize two variants for our multi-task losses. The first is an equally weighed variant where both $\mathcal{L}_a$ and $\mathcal{L}_v$ have the same weights, while the other is similar to the one proposed in [53] utilizing task-dependent uncertainty to automatically weigh losses. The loss is defined as:

$$\mathcal{L}_c = \mathcal{L}_v \,/\, \sigma_v^2 + \mathcal{L}_a \,/\, \sigma_a^2 + \log(\sigma_v.\sigma_a) \tag{4}$$

Where σ_i refers to the task dependent uncertainty (aleatoric homoscedastic). Although the latter has shown improved results in numerous settings, we noticed that it led to slower convergence and the performance improvements were not consistent across modalities. For this reason, all results reported utilize the simple equally weighted multi-task loss.

4. Learning Attention

As mentioned in the main text, we also explore the usage of an attention module that allows auto-learning of associations between different modalities similar to [52] which do it for audio and visual modalities. We setup attention in a slightly different manner by predicting weights over the grid. Recall that our features are arranged in a grid of shape $H' \times W'$. We predict $H' \times W'$ values $\alpha_{i,j}$ representing the weight of each feature corresponding to spatial location (i, j). Given an original context c of shape $D \times H' \times W'$, we extract c_{agg} from it as given in Eq. (2). Note that we generate attention weights for each pair of modalities to capture the associations between them.

In our experiments, we did not notice any differences between choosing various values of temperature as it seems the network modulated the learned α's accordingly. p's are utilized to infer regions of interest, as cells with higher p correspond to relevant portions of the modalities. Another thing worth noting is that this attention module is only used in conjunction with image modalities, as we found attention over an audio spectrogram was not directly interpretable in the traditional sense.

Figure 13: Visual results for multi-modal attention between egocentric and third person view. We show four instances where the left image refers to the third person view, while the right shows the predicted attention weights (White represents higher importance for attention). As we can see, CCAU is loosely able to predict areas of interest using our proposed self-supervised losses.

5. Knowledge Distillation

We discuss Knowledge Distillation briefly in the main text as one of the important baselines in Section 5.3.1. The framework we used is similar to the famously used one proposed in [48]. Without going into details, the overall loss is given in Eq. (5).

$$\mathcal{L}_{kd} = \alpha \cdot \mathcal{H}(y, \sigma(zs)) + \beta \cdot \mathcal{H}(\sigma(zt, \tau), \sigma(zs, \tau)) \tag{5}$$

Eq. (5) is an instance of matching logit distributions leading to the distillation of knowledge from the teacher to the student. Where H represents the cross-entropy loss, τ represents the temperature. zs and zt are outputs for the student and teacher, respectively.

For multiple modalities, the loss is just repeated multiple times for each modality. For our experiments we use $\alpha = 1$ and $\beta = 0.1$. We choose $\tau = 2.5$ as the models are similar in capacity. We also experiment with two variants i.e. Static and Cooperative Knowledge Distillation. The difference being Static KD involves static teachers while the cooperative variants allow all modalities to serve as both students and teachers.