Detecting Human-Object Relationships in Videos

Jingwei Ji  Rishi Desai  Juan Carlos Niebles
Stanford University
{jingweij, rdesai2, jniebles}@cs.stanford.edu

Abstract

We study a crucial problem in video analysis: human-object relationship detection. The majority of previous approaches are developed only for the static image scenario, without incorporating the temporal dynamics so vital to contextualizing human-object relationships. We propose a model with Intra- and Inter-Transformers, enabling joint spatial and temporal reasoning on multiple visual concepts of objects, relationships, and human poses. We find that applying attention mechanisms among features distributed spatio-temporally greatly improves our understanding of human-object relationships. Our method is validated on two datasets, Action Genome and CAD-120-EVAR, and achieves state-of-the-art performance on both of them.

1. Introduction

As we develop intelligent agents to understand images more comprehensively, the computer vision research problems we are solving have become more and more complex. The computer vision community has moved from classifying images and detecting objects, to detecting object relationships and understanding object interactions. In real-world applications, we often need to infer human behaviors from videos. In human-centered applications such as human-robot interaction, senior care [36], and healthcare [19], understanding the interactions people have with their environment is pivotal. One important problem at the heart of action recognition is detecting human-object relationships in videos: given the frames of a video, we would like to detect which objects a person is interacting with and classify the relationships between the person and objects.

Rather than generating scene graphs on static images [29, 53, 58, 45], human-object relationship (HOR) detection in videos focuses on the human and the active objects, that is the objects the person is actively interacting with. Unlike human-object interaction (HOI) detection [18, 7, 39, 33], HOR detection classifies not only the verbs that describe human actions, but also the prepositions that describe the relationships between human and objects, such as “behind”, “beneath” and “in”. While many

verbs are only associated with certain objects, prepositions are often applicable to numerous object categories.

Compared to scene graph generation and HOI detection on images, HOR detection in videos faces several challenges. First, the model needs to find out which objects are the protagonists of the scene. For instance, the clip in Figure 1 contains plenty of background objects that are not of our interest. How can we accurately focus on only the active objects? Second, the object detector, a key component in the detection model, will be confused during model training as video datasets [24, 60] typically only provide annotations on active objects. Without the knowledge of human-object interactions, a simple object detector will mistakenly fire on both the sat-on chair and those stacked chairs not relevant to the action. Third, videos can often be blurred at some
frames, and static image models have difficulty with performing inference on blurred frames. Considering these key issues, how can we leverage information from neighboring frames to produce more accurate predictions?

To tackle these issues, we propose a Human-Object Relationship Transformer (HORT) model for HOR detection in videos. Our model has two stages: i) static image prediction and ii) using visual concepts from the first stage to recognize active objects and their relationships. HORT first performs static image prediction, where one can plug in an existing model from various choices of scene graph generation or HOI detection. In the second stage, HORT gathers the encoded visual concepts from the first stage (specifically the object, relationship, and human pose embeddings) and feeds them into transformers with intra- and inter-attention mechanisms. The attention mechanisms allow the model to integrate information from spatially and temporally scattered visual cues to find out which interactions are happening. In the transformer modules, we also pass messages from human pose and relationship features to object encoding, enabling an object scorer to focus on active objects.

To validate our HORT model, we have benchmarked its detection performance on two video datasets: Action Genome [24] and CAD-120-EVAR [60]. Our model outperforms state-of-the-art methods in scene graph generation and HOI detection. We have also conducted ablation studies to examine the contribution of each part of our model.

2. Related Work

Scene graph generation. Scene graphs are a symbolic representation of images, where objects are encoded as nodes and their relationships are encoded as connecting edges [26, 29]. This structured representation has bolstered many down-stream image tasks such as image retrieval [26, 41], visual question answering [25], visual reasoning [42], and image captioning [1]. A large body of work has focused on improving scene graph generation from single images. Lu et al. [35] propose to use both visual and language modules to generate scene graphs. Xu et al. [52] utilize RNNs to iteratively leverage node and edge information. Zellers et al. [57] highlight the regularly occurring graph structures existing in commonly used databases [29]. Li et al. [30] showcase the importance of context in hierarchical regions. Yang et al. [53] propose a relationship proposal network to prune edges in scene graphs and use attentional graph convolutional networks (GCNs) to integrate node information. Zhang et al. [58] introduce graphical contrastive losses. Guo et al. [16] apply a transformer on object features to explore contextual information among objects. Inspired by causal inference, Tang et al. [45] tackle the issue of biased representations. Zareian et al. [56] bridge commonsense knowledge graphs with scene graphs.

However, all these methods are limited to static images without modeling the spatio-temporal dynamics of relationships in videos. Moreover, most of the existing scene graph generation models implicitly assume a single-class relationship between each pair of objects [29], while this does not always hold, especially for human-object relationships [24] (e.g. <person – looking at, holding, eating – food> has three concurrent relationships).

Human-object interaction detection. Human-object interaction (HOI) detection [18, 7] aims to understand how a person is interacting with objects in an image. Our task, HOR detection, is similar to HOI detection but considers a broader class of edges. Relationships in HOI detection are verbs, such as “riding”, “typing on”, and “hugging”, which often exclusively relate to certain object classes. Relationships in HOR detection can be verbs or prepositions [29, 24], such as “in”, “behind”, and “on the side of”, which are more general and object-class-agnostic.

The task of HOI detection has resulted in a series of research [17, 39, 38, 48, 27, 33, 34, 23, 11]. Our model design shares the spirit of the multi-stream approach [7, 12, 50, 32], benefits from the knowledge of human poses [51, 31, 59], and leverages an attention mechanism among visual concepts. We further model the temporal dependencies between instances in our intra- and inter-transformer model, leading to better understanding of human-object interactions.

Transformer models in video analysis. Transformers [49] have emerged as one of the most powerful building blocks in natural language processing [9, 3]. Recent studies also showcase the capability of transformers on 2-D image tasks [37, 5, 2] and graph-structured data [55, 4]. Transformers have also been utilized in video analysis. Sun et al. [44] propose VideoBERT for action classification and video captioning on instructional videos. Girdhar et al. [15] introduce an action transformer network for action localization. Gavrylyuk et al. [14] tackle group activity recognition with actor transformers. Garcia et al. [13] include transformers in their model for video question answering. In our Human-Object Relationship Transformers, we leverage knowledge with both intra- and inter-attention from three visual concepts – human pose, object, and relationship – scattered in the 3-D spatio-temporal space.

3. Human-Object Relationship Transformers

The video HOR detection problem is defined as follows: we wish to build a model that takes a video clip as input, and on each frame outputs the location of the human, the locations of the active objects, and the multiple relationships between each human-object pair. Most recent scene graph generation [53, 58, 45] and HOI detection models [39, 32] for static images consist of three modules: a backbone image feature extractor, an object detection head, and another
Video Clip comes with an encoded object feature vector \( \mathbf{O} \) bounding boxes \( s \). It takes in image feature maps \( \mathbf{F} \) and relationship detection heads. Our Human-Object Relationship Transformer (HORT) model. Our model extracts object, relationship and human pose features from the static image modules. Then these features are fed into an object branch and a relationship branch of intra- and inter-transformers. Finally the model generates human-object relationship detection by combining the object detection and relationship classification outputs.

head to predict the interactions or relationships between the detected objects. Our HORT model (Figure 2) adds an object branch and a relationship branch of intra- and inter-transformer modules on top of this framework. The transformer modules take in the temporal sequences of object, relationship, and pose features extracted from static images, integrate these features with an attention mechanism along both spatial and temporal dimensions, and finally generate more accurate human-object relationship detection.

### 3.1. Feature extraction on static images

Given a video clip \( V = \{I_1, I_2, ..., I_T\} \), where \( I_t \) is the RGB frame at time step \( t \), we first extract the feature map of each image: \( x_t = \mathbf{F}(I_t) \). \( \mathbf{F} \) refers to a backbone image feature extractor, typically implemented as a fully convolutional neural network. \( x_t \in \mathbb{R}^{H \times W \times C} \) is the extracted image feature map that will be shared by both the object and relationship detection heads.

Our object detection head \( \mathcal{H}_O \) follows Faster R-CNN [40] and consists of a region proposal network and a box head. It takes in image feature maps \( x_t \) and generates bounding boxes \( \{b^o_i\}_t \) of object proposals, each of which comes with an encoded object feature vector \( O_t \in \mathbb{R}^{d_O} \) and a proposal confidence score \( s^o_{t, static} \in (0, 1) \):

\[
\{O_t\}_t, \{b^o_i\}_t, \{s^o_{i, static}\}_t = \mathcal{H}_O(x_t), i \in \{1, ..., N_t\} \tag{1}
\]

where \( N_t \) is the number of detected objects on frame \( t \). \( N_t \) is typically larger than the number of ground truth objects in the scene, as the object detector outputs many false positive object proposals that are not being interacted with. The denotation \( s^o_{i, static} \) indicates that these confidence scores are generated only with the knowledge of static images. \( s^o_{i, static} \) highly depends on the appearance of objects rather than the context of human-object interactions.

The relationship head \( \mathcal{H}_R \) infers the relationships between each human-object pair detected by \( \mathcal{H}_O \). Following the designs of many relationship heads [53, 46, 58], we take the union box of each pair of human and object boxes \( b^o \) as a region of interaction \( b^r \), pool an ROIAlign feature [20] from \( x_t \), and then apply a neural network (e.g. ResNet-50 [21]) to extract a pairwise relationship feature \( R_j \in \mathbb{R}^{d_R} \). \( \mathcal{H}_O \) also takes a set of object features \( \{O_t\}_t \) as input and then outputs the logits \( z_j \in \mathbb{R}^{C_R} \) (\( C_R \) referring to the number of relationship classes) for classifying the relationships between the \( j \)-th pair of human and object (in static image baseline methods). The function of \( \mathcal{H}_R \) is summarized as

\[
\{R_j\}_t, \{b^o_j\}_t, \{z_j\}_t = \mathcal{H}_R(x_t, \{b^o_i\}_t, \{O_t\}_t) \tag{2}
\]

for all \( j \in \{1, 2, ..., M_t\} \), where \( M_t \) denotes the number of human-object proposal pairs on frame \( t \).

Research [54, 10, 31] has shown that understanding human-object interactions in static images can benefit from the knowledge of human poses. Hence, we believe that the temporal dynamics of human poses are helpful for inferring human-object relationships across time. Using a pose estimator \( \mathcal{P} \), we generate keypoints and determine bounding boxes of body parts following [10, 31]: \( \{b^p_k\}_t = \mathcal{P}(I_t), k \in \{1, 2, ..., K_t\} \). \( K_t \) denotes the number of body parts in a person (head, shoulders, wrists, pelvic, knees, and ankles). Then we extract the feature \( \{P_k\}_t \in \mathbb{R}^{d_p} \) for each body part from \( x_t \): \( \{P_k\}_t = \mathcal{H}_P(x_t, \{b^p_k\}_t) \).

So far we have extracted three sets of static image features: objects \( \{O_t\}_t \), relationships \( \{R_j\}_t \), and human pose \( \{P_k\}_t \). We will now show how spatio-temporally integrate these features with our transformer models.
3.2. 3-D positional encoding

While Recurrent Neural Networks [22, 8] maintain the orders of tokens and Temporal Convolutional Neural Networks [47, 6] operate on temporal neighborhoods of features, the Transformer [49] is a sequential model that utilizes a fully-connected attention mechanism, thereby removing restrictions caused by positions and allowing dependencies between any pair of tokens to be modeled. However, in video analysis the knowledge of positions is still crucial. Before feeding the features extracted from static images into the transformer models, we need to reconstruct the positional information for each feature vector, i.e. where an object/relationship/pose feature is extracted from in the 3-D space of a video clip.

The original transformer model [49] uses a sinusoidal encoding for word positions. This positional encoding has been generalized to the $x$-$y$ image plane in image applications [37, 5]. We further adapt this positional encoding for three dimensions, $x$, $y$, and time step $t$, so that the encoding indicates the position of each feature vector in 3-D space. After normalizing the $x$-$y$ coordinates by $\bar{x} = 2\pi x/W$, $\bar{y} = 2\pi y/H$, the sinusoidal positional encoding along any dimension can be written as

$$\mathcal{P}E(*)(\bar{x}) = \sin(\pi \bar{x} / {10000}^{2i/d_x})$$

$$\mathcal{P}E(*)(\bar{y}) = \cos(\pi \bar{y} / {10000}^{2i/d_y})$$

where $*$ can either be $\bar{x}$, $\bar{y}$ or $t$. With the dimensionality of transformer input $d_{Tx}$ being 512, we set $d_x = 128$, $d_y = 128$, $d_t = 256$, such that $d_{Tx} = d_x + d_y + d_t$. By concatenating the positional encoding from spatial and temporal dimensions, we get the 3-D positional encoding:

$$\mathcal{P}E(\bar{x}, \bar{y}, t) = \text{concat}(\mathcal{P}E(\bar{x}, t), \mathcal{P}E(\bar{y}, t), \mathcal{P}E(t))$$

Note that the positional encodings are similar for features that are spatio-temporally close and different for those that are spatio-temporally far.

We use the $(x, y)$ coordinates of the center point of each box $b^o$, $b^r$ and $b^p$ for its positional encoding. Because the dimensions of the features output by $\mathcal{H}_O$, $\mathcal{H}_R$ and $\mathcal{H}_P$ may be different, we apply linear projections to align the dimensions into $d_{Tx}$ for inputs to the transformer models. After adding the linearly projected features and positional encodings, we have the inputs for the transformers as follows:

$$O'_{i,t} = W^O_{Ot}O_{i,t} + \mathcal{P}E(\bar{b}^o_{i,t}), W_o \in \mathbb{R}^{d_o \times d_{Tx}}$$

$$R'_{j,t} = W^R_{Rt}R_{j,t} + \mathcal{P}E(\bar{b}^r_{j,t}), W_r \in \mathbb{R}^{d_r \times d_{Tx}}$$

$$P'_{k,t} = W^P_{Pt}P_{k,t} + \mathcal{P}E(\bar{b}^p_{k,t}), W_p \in \mathbb{R}^{d_p \times d_{Tx}}$$

3.3. Intra- and Inter-Transformers

One of the key components of the original transformer model is computing the multi-head self-attention of representations, i.e. constructing an attention map between all pairs of features in a sequence, and then using this attention map to integrate features. Mathematically, the attention function computes the scaled inner product of a query feature sequence $Q$ and a key feature sequence $K$ to generate an attention map $A$. Then $A$ is used to look up in a value feature sequence $V$. In the self-attention or intra-attention setup, $Q$, $K$, and $V$ are linear projections of the same feature sequence. We are omitting the multi-head details here and we refer the readers to [49] or our supplementary materials for a more detailed description of the original intra-attention transformer model.

When detecting human-object relationships in videos, we need not only attention within each modality (object features $\{O'_t\}$ or relationship features $\{R'_t\}$), but also attention that is inter-modality. Pose and object features are critical cues for classifying relationships; pose and relationship features are helpful in determining which objects are being interacted with.

Our transformers are divided into two symmetric branches: an object branch and a relationship branch (Figure 3). For simplicity, we will only describe the details in the relationship branch (Figure 3(b)). The relationship branch consists of one intra-transformer and two inter-transformers, and each transformer contains an encoder and a decoder. The intra-transformer simply follows the original transformer design, where all $Q$, $K$, and $V$ are linear projections of the relationship features $R$.

The first of the two inter-transformers in the relationship branch considers the attention between human poses and relationships. Intuitively, by simply looking at the human poses (especially a temporal sequence of them), one can make a reasonable impression of which relationships are occurring in the scene. Taking Figure 2 as an example: the pose suggests that the person is sitting somewhere and may be holding something close to his face.

The second inter-transformer utilizes attention among all three visual concepts: human poses, objects and relationships. The encoder computes inter-attention between the pose and object features, allowing the model to determine which objects are salient by looking at the trajectory of human poses and all objects. The encoder outputs a pose-object memory which is then passed into the decoder as a reference for classifying relationships.

We will now describe the architecture of inter-transformers. We prepare the input by flattening each set of features $\{O'_t\}$, $\{R'_t\}$ and $\{P'_t\}$ along spatial and temporal dimensions. Note that the positional information has been reserved in each feature vector, so flattening these features will not lose knowledge of proximity. We denote each resulting feature matrix as $O \in \mathbb{R}^{N \times d_{Tx}}$, $R \in \mathbb{R}^{M \times d_{Tx}}$, and $P \in \mathbb{R}^{K \times d_{Tx}}$, where $N = \sum_i N_i$, $M = \sum_i M_i$ and $K = \sum_i K_i$. 
In the relationship branch, each of the three transformers outputs a new relationship feature matrix. After computing the linear projections for each respective feature matrix, we add them together to create the logits for relationship classification. Prior work [30, 53, 58, 45] implicitly assumes only a single relationship can exist between each pair of subject and object; therefore, a softmax function is applied upon the logits to get the relationship scores. Because we are inferring multiple classes of relationships between human and objects (e.g. attentional, spatial, and contacting relationships [24]), we use the sigmoid function to generate per-class relationship scores $s^r \in (0, 1)^{M \times C_R}$.

Having an architecture symmetric to the relationship branch, the object branch outputs object scores $s^o_{\mathcal{P}} \in (0, 1)^{N}$ for object proposals. These scores indicate if objects are salient in the context of human-object interactions.

### 3.4. Training and post-processing

Although the entire model can be trained end-to-end, we opt to pre-train the backbone and object detection head, fix their weights, and then only train the feature extractor in the relation head and the transformer models. This way, we can fairly compare our model with other baseline methods using the same backbone and object detector.

We utilize two loss functions while training our model: a binary cross entropy loss $L_o$ for object saliency classification and a binary cross entropy loss $L_r$ for relationship classification. We add the two losses for the total loss $L = L_o + \lambda L_r$.

Our method generates two scores for each object: $s^o_{\text{static}}$ from the static image object detector and $s^o_{\mathcal{P} \mathcal{T}_x}$ from the object branch. We observed that $s^o_{\text{static}}$ are highly biased towards the saliency of object appearance rather than whether an object is involved in any human-object interactions. False positive objects are often assigned with very high $s^o_{\text{static}}$ scores. Conversely, $s^o_{\mathcal{P} \mathcal{T}_x}$ pay more attention to the interaction context as the object branch has integrated features from temporal sequences of poses, objects, and relationships; thus, $s^o_{\mathcal{P} \mathcal{T}_x}$ are usually low on false positive object proposals. We combine the two scores by choosing the minimum: $s^o = \min(s^o_{\text{static}}, s^o_{\mathcal{P} \mathcal{T}_x})$. We have found that this fusion effectively suppresses false positive object proposals.

Finally, we compute a total score for each triplet (subject - relationship - object):

$$s = s^o * s^r * s^p,$$

where $s^p$ is the confidence score for the human box generated in pose estimator. We rank all possible human-object
relationship triplets in each frame by their total scores.

4. Experimental Results

4.1. Datasets

We have evaluated our HORT model over two third-person view video databases: Action Genome [24] and the re-annotated CAD-120-EVAR dataset [28, 60].

Action Genome. The Action Genome dataset [24] is built upon the crowdsourced videos from the Charades dataset [43], which captures indoor human activities and behaviors in daily life. Action Genome provides annotations of 476,229 bounding boxes of interacted objects and 1,715,568 relationship classes between the person and objects on 234,253 frames. Action Genome contains labels for 35 classes of objects and 25 classes of relationships. The relationships in Action Genome can be categorized into three types: attentional relationships indicating whether a person is looking at something, spatial relationships such as \(<\text{chair - beneath - person}>\), and contacting relationships indicating if a person is contacting an object and what type of contact is happening.

CAD-120-EVAR. The CAD-120 video dataset [28] consists of 4 subjects performing 10 different high-level household activities (e.g., arranging objects, taking food). Each subject performs each household activity 3 or 4 times, totaling 124 video sequences. In our experiments, we utilize the newly re-annotated version, which we call CAD-120-EVAR [60]. CAD-120-EVAR consists of 551 video clips with 32,327 frames. These frames are re-annotated to contain 6 classes of relationships between objects (e.g., holding, containing), the attributes of objects (e.g., open, closed), and the regions of interest of all of the objects in the frames.

4.2. Implementation details

The model is implemented in PyTorch. We use ResNet-101 [21] as our backbone image feature extractor and take the C4 features as \(x_t\). For our experiments on both datasets, we pre-train the backbone and object detection head with the object detection task on Visual Genome [29]. For the Action Genome experiments, we further finetune the backbone and object head on Action Genome’s training set. The same backbone and object detector is shared in all baseline experiments except for [33]. We do not finetune on CAD-120-EVAR as the “ground truth” object bounding boxes are generated by an object detector. We use an off-the-shelf Keypoint R-CNN [20] to estimate all human keypoints.

In our Action Genome experiments, we choose the clip length \(T = 5\). Because Action Genome’s annotation sampling rate is \(\sim 1\) FPS, our clip length effectively covers approximately 5 seconds on average. In our CAD-120-EVAR experiments, we set \(T = 10\). We use a clip batch size of 4 in training on both datasets. Our models are trained on 4 Nvidia TITAN XP GPUs for 80,000 iterations with a learning rate starting at \(5e^{-4}\) and shrinking to \(5e^{-5}\) and \(5e^{-6}\) at iteration 30,000 and 50,000, respectively. For all the transformers, \(d_{T_x} = 512\), 8 parallel heads are used, the feed forward dimension is 2048, and both the encoder and decoder contain 2 layers. Code will be released after acceptance.

4.3. HOR detection on Action Genome

Evaluation metrics. We follow the three standard evaluation modes for image-based scene graph prediction [35], which are also the evaluation metrics provided by Action Genome [24]: (1) predicate classification (PredCls) which assumes ground truth object classes and bounding boxes are given and only evaluates the predicate/relationship labels between each subject-object pair, (2) scene graph classification (SGCls) which assumes ground truth object bound-
ing boxes are given and evaluates the triplet labels of \( \langle \text{subject} - \text{relationship} - \text{object} \rangle \), and (3) scene graph detection (SGDet) which evaluates all predictions including bounding box locations and triplet labels. Action Genome further proposes the video version of these three metrics, where the per-frame measurements are first averaged in each video, then averaged across all videos in the test set. We report these metrics with recall@20 and recall@50, where recall@x computes the fraction of correct relationships in the top x ranked triplet predictions.

**Baselines.** We report the performance of various HOI detection and scene graph generation methods (Table 1). GPNN [39] is the only recent HOI detection method that also generalizes to video analysis. PPDM [33] is one of the state-of-the-art models in HOI detection. PPDM makes use of a different object detector than all other baselines. We pre-trained this hourglass-based object detector with the same curriculum as our Faster R-CNN detector: first on Visual Genome, then on Action Genome. PPDM's object detector achieves better performance (measured by AP\(_{50}\)), but PPDM does not perform as well as several other scene graph generation baselines in HOR detection.

Among the scene graph generation models, we compare with VRD [35], IMP [52], MSDN [30], Graph R-CNN [53], ReIDN [58] and VCTree [46, 45]. When the backbone and object detector are fixed and shared by the baseline models, many models show similar performance, as they are designed for static images only. We have also extended the ReIDN model with a simple approach of integrating temporal contextual information (Temporal ReIDN in Table 1): when predicting relationships in both training and testing, the final logits of a frame is obtained by averaging the logits from a 5-frame temporal window around this frame.

Note that our reported measurements of baseline methods are significantly higher than those in [24]. This is because [24] has the restriction that only one relationship can be predicted between each pair of human and object during training and testing. Here, we remove this restriction, resulting in much higher measurements in all baseline methods. As a reference, we have also reported the evaluation comparisons with the single-relationship constraint in the supplementary material. Due to the intra- and inter-transformer models, HORT outperforms all baseline methods. Figure 4 illustrates examples of the predictions output by the HORT model. Please see the supplementary materials for more qualitative results.

### 4.4. Relationship classification on CAD-120-EVAR

CAD-120-EVAR does not provide manually labeled ground truth of object bounding boxes, so we do not train and test the object detector on CAD-120-EVAR. Therefore, the task of HOR detection is simplified to classifying the relationship between each pair of objects. The metric used in [60] is the accuracy of relationship classification, which is essentially the same as PredCls.

We report the classification accuracy of all 6 relationship categories in Table 2. Note that only holding, not holding, contacting and apart are human-object relationships, whereas containing and separate are actually relationships between microwave and other objects. Still, our model can handle the cases of non-human relationships as well. HORT outperforms both the baseline method reported in [60] and a ReIDN [58] baseline we constructed.

### 4.5. Ablation study

We have conducted ablation experiments on Action Genome to inspect the effectiveness of each transformer, the object branch, and different positional encodings.

**Transformer modules.** In total, we have 6 transformers in our model: two branches for object scoring and relationship classification, each of which consists of an intra-transformer and two inter-transformers. We found that all of...
Table 2: Accuracy of relationship classification on CAD-120-EVAR [28, 60]. Abbreviation for classes of relationships: HO - Holding; NH - Not holding; CTC - Contacting; AP - Apart; CTN - Containing; SP - Separate. Our method achieves similar or better performance on all relationship classes.

<table>
<thead>
<tr>
<th>Method</th>
<th>HO</th>
<th>NH</th>
<th>CTC</th>
<th>AP</th>
<th>CTN</th>
<th>SP</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVAR w/o obj</td>
<td>0.86</td>
<td>0.91</td>
<td>0.59</td>
<td>0.60</td>
<td>0.93</td>
<td>0.67</td>
</tr>
<tr>
<td>EVAR w/ obj</td>
<td>0.82</td>
<td>0.96</td>
<td>0.80</td>
<td>0.96</td>
<td>0.95</td>
<td>0.96</td>
</tr>
<tr>
<td>RelIDN [58]</td>
<td>0.88</td>
<td>0.96</td>
<td>0.88</td>
<td>0.95</td>
<td>0.95</td>
<td>0.94</td>
</tr>
<tr>
<td>HORT (Ours)</td>
<td>0.89</td>
<td>0.96</td>
<td>0.98</td>
<td>0.97</td>
<td>0.96</td>
<td>0.97</td>
</tr>
</tbody>
</table>

Table 3: Ablation study: effectiveness of different transformers. InterTx-2 refers to inter-transformers with two sets of features as inputs, either pose-object in the object branch or pose-relationship in the relationship branch. InterTx-3 refers to inter-transformers with three modalities as inputs.

<table>
<thead>
<tr>
<th>Object branch</th>
<th>InterTx</th>
<th>InterTx-2</th>
<th>InterTx-3</th>
<th>SGD-Eva@20</th>
<th>SGD-Eva@50</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>36.03</td>
<td>46.10</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>36.92</td>
<td>47.51</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>37.19</td>
<td>47.76</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Relation branch</th>
<th>InterTx</th>
<th>InterTx-2</th>
<th>InterTx-3</th>
<th>SGD-Eva@20</th>
<th>SGD-Eva@50</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td>35.93</td>
<td>45.96</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>36.36</td>
<td>46.55</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>37.19</td>
<td>47.76</td>
</tr>
</tbody>
</table>

Figure 6: Ablation study: we measure the SGD-Eva of our model with different types of positional encoding. t refers to temporal encoding only, xy to spatial encoding only, and xyt to the 3-D positional encoding we use in our full model.

5. Conclusion

In this paper, we propose a Human-Object Relationship Transformer (HORT) model for the problem of detecting human-object relationships in videos. Our model is composed of two stages, static image prediction and feature extraction, followed by intra- and inter-transformers performing spatio-temporal reasoning. By integrating features from different instances scattered across image planes and time steps, our model filters out false positive object proposals, identifies the active objects, and refines the relationship classification. We conducted experiments on two video datasets, Action Genome and CAD-120-EVAR, and showed how our model enables a better understanding of human-object relationships. We also inspected the contribution of each submodule in our ablation study, verifying the efficacy of our spatio-temporal reasoning.

Acknowledgment. We gratefully acknowledge the support of Panasonic and ONR (N00014-16-1-2127 and N00014-19-1-2477).
References


[38] Julia Peyre, Ivan Laptev, Cordelia Schmid, and Josef Sivic. Detecting rare visual relations using analogies. 2019. 2


