Video Object Grounding using Semantic Roles in Language Description

Arka Sadhu\textsuperscript{1}  Kan Chen\textsuperscript{2}  Ram Nevatia\textsuperscript{1}

\textsuperscript{1}University of Southern California  \textsuperscript{2}Facebook Inc.
\{asadhu|nevatia\}@usc.edu  kanchen18@fb.com

Abstract

We explore the task of Video Object Grounding (VOG), which grounds objects in videos referred to in natural language descriptions. Previous methods apply image grounding based algorithms to address VOG, fail to explore the object relation information and suffer from limited generalization. Here, we investigate the role of object relations in VOG and propose a novel framework VOGNet to encode multi-modal object relations via self-attention with relative position encoding. To evaluate VOGNet, we propose novel contrasting sampling methods to generate more challenging grounding input samples, and construct a new dataset called ActivityNet-SRL (ASRL) based on existing caption and grounding datasets. Experiments on ASRL validate the need of encoding object relations in VOG, and our VOGNet outperforms competitive baselines by a significant margin.

1. Introduction

Grounding objects in images \cite{7, 67, 68} and videos \cite{8, 27, 76} from natural language queries is a fundamental task at the intersection of Vision and Language. It is a building block for downstream grounded vision+language tasks such as Grounded-VQA \cite{14, 31, 32, 74, 77}, Grounded-Captioning \cite{35–37, 75} and Grounded Navigation \cite{22}.

In this work, we address the task of Video Object Grounding (VOG): given a video and its natural language description we aim to localize each referred object. Different from prior VOG methods on finding objects from query mentions \cite{76} or distinguishing spatio-temporal tubes from a referring expression \cite{8}, we formulate VOG as localizing only the specific referred objects in the query. Prior work has focused on attending to each object in isolation; our formulation additionally requires incorporating object-object relations in both time and space. Figure 1 illustrates the key differences.

Despite the importance of associating natural language descriptions with objects in videos, VOG has remained relatively unexplored due to two practical requirements: (i) a large-scale video dataset with object-level annotations, (ii) the videos should contain multiple instances of the same object category so making a distinction among them becomes necessary. Recently, \cite{75} released ActivityNet-Entities dataset which contains bounding box annotations relating the noun-phrases of the video descriptions \cite{29} to the corresponding objects instances in ActivityNet \cite{4} videos. Despite its scale, a majority of the videos in ActivityNet contain single instances of various objects. For instance, in Figure 1 “ball” can be localized simply using an object detection system such as FasterRCNN \cite{47} without...
relating “ball” to the “man” or the “kids”.

We mitigate this absence of multiple object instances in two steps. First, we sample contrasting examples from the dataset; these are examples that are similar to but not exactly the same as described by the language query. To sample contrasting examples, we obtain semantic-roles (SRLs) using a state-of-the-art Semantic Role Labeling (SRL) system [55] on the language descriptions. SRLs answer the high-level question of “who (Arg0) did what (Verb) to whom (Arg1)” [58]. We sample videos with descriptions of the same semantic roles structure as the queried description, but the role is realized by a different noun or a verb.

In the next step, we need to present the contrasting videos to a model. If the contrasting samples are processed independently, a model could easily “cheat” and find the associated video by simply adding the object detection and action recognition scores as per the query. To prevent this, we propose novel spatial and temporal concatenation methods to merge contrasting samples into one video. With contrasting objects and their relations in the same video, the model is forced to encode object relations in order to ground the referred objects (details in Section 3.1).

Clearly, encoding object relations is of primary importance for VOG. Recently, [16] and [75] show promising results using self-attention [61] to encode object relations. However, there are two issues in directly adapting self-attention on objects for VOG. First, such object relations are computed independent of the language creating ambiguities when two objects have multiple relations. For instance, in Figure 1 “The man is playing with a group of kids” is an accurate description for the same video but the queried relation between “the man” and “kids” is different. Second, the transformer module for self-attention [61] expects positional encoding for its input but absolute positions are not meaningful in a video.

We address these issues in our proposed VOGNet framework which applies self-attention to both the object features and fused multi-modal features to encode language dependent and independent object relations. To encode positions, we propose a relative position encoding (RPE) scheme based on [54]. Essentially, RPE biases the model to weigh related objects based on their proximity (details on model architecture in Section 3.2).

To evaluate our models, we contribute ActivityNet-SRL which adds semantic roles to the descriptions [29] and aligns with the noun-phrase annotations in [75]. We further show by pre-computing lemmatized noun-phrases, contrastive sampling process can be used in training (details on dataset construction in Section 4.1,4.2).

Our contributions are three-fold: (i) we explore VOG and propose contrastive sampling with temporal and spatial concatenation to allow learning object relations (ii) we design VOGNet which extends self-attention to encode language-dependent object relations and relative position encodings (iii) we contribute ActivityNet-SRL as a benchmark for VOG. Our code and dataset are publicly available1.

2. Related Work

Grounding objects in images is a heavily studied topic under referring expression [26, 39, 67, 68] and phrase localization [7, 44, 45, 48, 50]. In contrast, grounding objects in videos has garnered less interest. Apart from [8, 76], [27] enforces temporal consistency for video object segmentation and requires the target to be in each frame and [23] use structured representations in videos and language for co-reference resolution. Different from them, our proposed formulation of VOG elevates the role of object relations and supports supervised training due to use of a larger dataset.

Object relations is also fairly well-studied in images under scene-graph generation [30, 33, 40, 64] and human-object interaction [5, 6, 12, 17, 49, 78] and referring relations [28]. However, a majority of the relations are spatial (“to-the-left-of”, “holding”) with considerable biases caused due to co-occurrence [72]. On the video side, it has been explored for spatio-temporal detection [3, 16, 59]. In particular, [16] showed self-attention using transformers [61] to be more effective than relation-networks [51] based detectors [59]. For VOG, relation networks would not be effective due to high memory requirements and thus we only explore self-attention mechanism. Different from [16], we use bottom-up features [2] which don’t maintain any order. As an alternative, we employ relative position encoding.

Video relation detection [53, 53, 60] is closely related to VOG where relations between two objects need to be detected across video frames. However, the metrics used (recall@50/100) are difficult to interpret. Moreover, densely annotating the relations is expensive and results in less diverse relations. In contrast, ours uses sparsely annotated frames and leverages off-the-shelf SRL systems.

Visual Semantic Role Labeling in images has focused on situation recognition [57, 65, 66]. To annotate the images, [66] employed FrameNet [11] annotations and [57] shows using semantic parsers on image captions significantly reduces annotation cost. We instead PropBank annotations [42] which is verb-oriented and thus more suited to video descriptions. Finally, our use of semantic roles is guided by contrastive sampling and not assigning semantic roles to visual entities.

Contrastive Training via max-margin loss has been commonly used in vision-language tasks [24, 67, 73, 76]. Here, we don’t use contrastive losses, instead, the concatenation of the videos directly informs us which objects are related. As such, we train using binary cross-entropy.

1https://github.com/TheShadow29/vognet-pytorch
Figure 2. (a) illustrates contrastive sampling based on semantic roles. Q1 contains a single agent (“man”) and a single patient (“dog”). We use the SRL structure \( \text{Arg}_0 - \text{Verb} - \text{Arg}_1 \) but replace one queried object (Q2, Q4) or action (Q3). (b) shows temporal concatenation where we resize each video to the same width, height. (c) shows spatial concatenation where we resize the height and sample a fixed number of frames across the videos (d) shows an unreasonable spatial concatenation as videos have a top-down order (“ocean” is always below “sky”).

### 3. Method

We describe our sampling and concatenation process which enables learning object relations for VOG (Section 3.1), followed by details of VOGNet (Section 3.2) and relative position encoding scheme (Section 3.3).

#### 3.1. Contrastive Sampling

Most large scale video datasets [1,4,25] are curated from Internet sources like YouTube which rarely contain multiple instances of the same object in the same video. VOG on such datasets can be trivially solved using object detection.

To mitigate this issue, we propose a two-step contrastive sampling method. First, we assign semantic roles labels (SRLs) to every language descriptions of the videos (see Table 1) and sample other descriptions by replacing each role in a one-hot style (Figure 2(a)).

In the second step, we aggregate our samples. One simple method is to present each video separately, similar to standard multiple-choice in Question-Answering tasks [31,70,71]; we call this “separate” (SEP) strategy (i.e. the videos are viewed separately). However, SEP doesn’t force learning object relations, as one could independently add the scores for each referred object. For instance, in Figure 2-(a) we can score “man”, “petting”, “dog” individually and choose the objects in the video with the highest aggregate score essentially discarding object relations.

Alternatively, we generate new samples by concatenation along the time axis (TEMP) or the width axis (SPAT). For TEMP, we resize the sampled videos to have the same width and height (Figure 2(b)). For SPAT, we resize the height dimension and uniformly sample \( F \) frames for each video (Figure 2(c)). Generally, it is not reasonable to concatenate along the height dimension as most real-world images obey up-down order (“sky” is on the top while “ocean” is below) but not left-to-right order (Figure 2(d)). Such concatenated videos, by construction, have multiple instances of the same object category. To associate an instance described in the language query to its bounding box in the video, a model would need to disambiguate among similar object instances by exploiting their relations to the other objects. For e.g., in Figure 2(c) “man” or “dog” cannot be uniquely identified without considering other objects.

**Caveats:** (i) in TEMP, one could use an activity proposal network like [13,34] and bypass the problem altogether, (ii) in SPAT uniformly sampling \( F \) frames from two different videos, would result in different parts of the image moving faster or slower and could partially affect our results.

#### 3.2. Framework

**Notation:** We are given a video \( V \) sampled with \( F \) frames and a language description \( L \) with \( k \) roles. In gen-

<table>
<thead>
<tr>
<th>Agent</th>
<th>Verb</th>
<th>Patient</th>
<th>Modifier</th>
<th>Instrument</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person</td>
<td>washes</td>
<td>cups</td>
<td>in a sink</td>
<td>with water.</td>
</tr>
</tbody>
</table>

Table 1. An illustration of semantic-role assignment to a description. Here, the actor/agent (person) performs an action/verb (wash) using some instrument (water) at some location (sink).
no known apriori. Given $P$ proposals for each frame using an object detector, we denote $O = \{p_{i,j}\}$ ($i^{th}$ proposal in $j^{th}$ frame) as the set of proposals in the video. In VOG we learn the mapping $H : (V, O, L) \rightarrow \{\{p_{i,j}\}^P_{j=1}\}$ where $p^* \in O$. That is, for each of the $k$ roles, we output a proposal $p^*$ in each frame. We allow $p^* = \phi$ if the object is not visible in a particular frame, or the object cannot be localized.

We build a VOGNet framework that contains a Language Module to encode the query descriptions at the phrase level, a Visual Module to encode the object and frame level features in the video, and a Multi-Modal Relation Module to encode both a language independent and dependent object relations. Figure 3 gives an overview of VOGNet.

**Language Module** first encodes the query $q = \{w_i\}^n_{i=1}$ as $n$ hidden vectors $[h_1, \ldots, h_n]$ with a Bi-LSTM [20, 52]. The $j^{th}$ Semantic Role Label (SRL) in query $q$, $\lambda_{q,j}$, spanning a set of words $S_j$ (e.g., in Figure 3, $\lambda_{q,0}$ includes the words $S_0 = \{“The”, “man”\}$) is encoded as $\tilde{q}_j$ is

$$\tilde{q}_j = M_q(\mathcal{G}(\delta(w_i \in S_j) \cdot h_i)_{i=1}^n)$$

where $\delta(\cdot)$ is an indicator function, and $\mathcal{G}(\cdot)$ is an aggregation function. In VOGNet, we set $\mathcal{G}$ as the concatenation of the first word and the last word for each SRL, followed by $M_q$ which denotes a Multiple Layer Perceptron (MLP).

**Visual Feature Extraction**: An off-the-shelf object detector [47] returns $P$ proposals for each frame. Let $p_{i,j}$ be the $i^{th}$ proposal in $j^{th}$ frame and $v_{i,j} \in \mathbb{R}^{d_v}$ be its ROI-pooled feature. Similarly, an action classifier returns temporal features containing image-level and flow-level features of the video. In general, the number of frames considered by the action classifier could be greater than $F$. We consider the local segment feature corresponding to the $F$ frames to get $s_j \in \mathbb{R}^{d_s}$, and append it to each proposal feature in $j^{th}$ frame. The final visual feature is $\hat{v}_{i,j} = M_v(v_{i,j}||s_j)$, where $M_v$ is a MLP.

**Object Transformer** is a transformer [61] and applies self-attention over the proposal features $\hat{v}_{i,j}$, i.e. self-attention is applied to $P \times F$ proposals. We denote the self-attended visual features as $\hat{v}_{i,j}^{sa}$. Similar module is used in [75] but there are two differences: first, $\hat{v}_{i,j}$ contains additional segment features; second absolute positions are replaced with relative position encoding (Section 3.3).

**Multi-Modal Transformer**: We concatenate the self-attended visual features $\hat{v}^{sa}$ and the language features $\tilde{q}$ to get multi-modal features $m$ where $m_{i,j} = [\hat{v}_{i,j}^{sa}||\tilde{q}]$. We apply self-attention with relative position encoding to get self-attended multi-modal features $m^{sa}$. However, due to hardware limitations, it is extremely time consuming to perform self-attention over all the proposals especially when $P \times F \times k$ is large. Thus, we perform this self-attention per frame i.e. self-attention is applied to $P \times k$ features $F$ times. Subsequently, $m^{sa}$ is passed through 2-layered MLP to get prediction for each proposal-role pair to get $\hat{m}^{sa}$.

**Loss Function**: Let $L_g$ be the set of groundable roles i.e. have a corresponding bounding box. Thus, a proposal-role pair is considered correct if it has $IoU \geq 0.5$ and negative otherwise. We train using Binary Cross Entropy (BCE) loss.

Figure 3. An overview of VOGNet. It takes a video-query pair as an input. A visual encoder extracts object features for each frame and concatenates them with segment features (rgb+flow). A language encoder encodes the whole query with a BiLSTM [20, 52] and then maintains a separate encoding for each phrase in the query (Eq. 1). A Transformer [61] is first applied to the visual features to model object relations. These self-attended visual features are fused with the language features. Finally, a separate transformer models the interaction among the fused multi-modal features followed by a 2-layer MLP. VOGNet is trained with Binary Cross Entropy (BCE) loss.
and average over the phrases with a bounding box:

\[ L_{\text{pred}} = \frac{1}{|L_g|} \sum_{l_g \in L_g} \text{BCE}(\hat{m}_{l_g}^{\text{alg}}[l_g, i, j], g[l_i, j]) \] (2)

**Minor changes for SEP:** When training and evaluating models using SEP strategy we have access to the individual videos. Here, we use the temporal features to learn a Verb score which can be used to disambiguate between videos with the same objects but different verbs. In general, this didn’t translate to other strategies and hence it is not included in our framework.

### 3.3. Relative Position Encoding

Relative Position Encoding (RPE) uses the relative distances between two proposals as an additional cue for attention. We denote the normalized positions of the proposal \( p_{a,b} \), whose 5d coordinate is \( [x_t, y_t, x_b, y_b, j] \) with \( \text{pos}_{a,b} = [x_t/W, y_t/H, x_b/W, y_b/H, j/F] \). We encode the relative distance between two proposals \( A \) and \( B \) as \( \Delta_{A,B} = M_p(\text{pos}_A - \text{pos}_B) \), where \( M_p \) is a MLP.

Let the Transformer contain \( n_l \) layers and \( n_h \) heads. Here, \( \Delta_{A,B} \in \mathbb{R}^{n_h} \). When self-attention is applied to a batch

\[ A(Q, K, V) = \text{SoftMax}(QK^T / \sqrt{d_k})V \] (3)

We change this to

\[ A(Q, K, V) = \text{SoftMax}((QK^T + \Delta[h]) / \sqrt{d_k})V \] (4)

Note that \( \Delta[h] \) has the same dimensions as \( QK^T \) and leading to a simple matrix addition. That is, our relative position encoding (RPE) encodes the distance between each proposal pair and this encoding is different for each head. Intuitively, RPE biases the self-attention to weigh the contribution of other objects relative to their proximity.

Our solution is based on prior work \[54] but differs in two key aspects: (i) the relative positions are not embedding layers rather modeled by an MLP to encode the difference (ii) our relative encoding is different for different heads. Another way to extend \[54] to visual setting would be to categorize the distances into multiple bins and learn encoding for each bin. We leave this study for future work.

**Caveat:** While we resolve the issue of adding RPE to the transformer network efficiently, computation of \( \Delta_{i,j} \) remains expensive as it requires \( O(n^2) \) difference computation and is a bottleneck of our proposed solution.

### 4. Experiments

We briefly describe the dataset construction (see Appendix B for more details) followed by experimental setup, results and visualizations.

<table>
<thead>
<tr>
<th>Arg0</th>
<th>Arg1</th>
<th>Arg2</th>
<th>ArgN-Loc</th>
</tr>
</thead>
<tbody>
<tr>
<td>42472</td>
<td>32455</td>
<td>9520</td>
<td>5082</td>
</tr>
</tbody>
</table>

Table 2. Number of annotated boxes in ASRL training set.

### 4.1. Constructing ActivityNet-SRL

Our proposed dataset ActivityNet-SRL (ASRL) is derived from ActivityNet \[4\], ActivityNet-Captions (AC) \[29\] and ActivityNet-Entities (AE) \[75\]. There are two key steps in creating ASRL: (i) add semantic role labels (SRLs) to the descriptions in AC and filter it using heuristics (ii) add lemmatized words for each groundable phrase labeled as a semantic role for efficient contrastive sampling.

For (i) we apply \[55\], a BERT-based \[10\] semantic-role labeling system to the video descriptions in AC. We use the implementation provided in \[15\] trained on OntoNotes5 \[46\] which uses the PropBank annotation format \[42\]. The obtained semantic-roles are cleaned using heuristics like removing verbs without any roles usually for “is”, “are” etc. In general, each description contains multiple “verbs” and we treat them separately.

For (ii) we utilize bounding box annotations in AE. First, we align the tokens obtained from the SRL system with the tokens of AE using \[21\]. Then, for each phrase labeled with a semantic role, we check if the corresponding phrase in AE has a bounding box and mark the phrase as being groundable or not. Since AE provides object names derived from the noun-phrases parsed using \[38\] we use them as the lemmatized word for the phrase. Table 2 shows the top-4 semantic roles with bounding box annotations in the training set of ActivityNet-Entities. We confine to this set of SRLs for contrastive sampling.

For training, we use the training set of ActivityNet which is the same as AC and AE. However, to create test set for AE, we need the ground-truth annotations which are kept private for evaluative purposes. As an alternative, we split the validation set of AE equally to create our validation and test set. When contrastive sampling is used in training, we only sample from the train set. However, since the size of validation and test sets is reduced, it is difficult to find contrastive examples. As a remedy, we allow sampling of contrastive examples from the test set during validation and vice versa for testing but never used in training.

### 4.2. Dynamic Contrastive Sampling

While Contrastive Sampling is mainly used to create the validation and test sets to evaluate VOG, it can also be used for training where speed is the bottleneck. Given a particular description belonging to VOG, it can also be used for training where speed is the bottleneck. Given a particular description belonging to VOG, it can also be used for training where speed is the bottleneck. Given a particular description belonging to VOG, it can also be used for training...
sample other descriptions with the same semantic-roles but containing one different lemmatized word. That is, we need to sample indices \( T_i \) whose lemmatized words are \( S_i = \{s_1, \ldots, s'_i, \ldots, s_k\} \) for every \( 1 \leq i \leq k \).

To address this, we first create a separate dictionary \( D_i \) for each semantic role \( r_i \) containing a map from the lemmatized words to all the annotation indices where it appears as \( r_i \). Given \( S \), we can efficiently obtain \( T_i \) by randomly sampling from the set \( E_i = \cap_{j \in \{1 \ldots k\}, j \neq i} D_j(s_j) \).

Due to hardware limitations, we restrict \( k \leq 4 \). For \( k > 4 \), we randomly drop \( k - 4 \) indices. If \( k < 4 \), then we randomly sample a training index \( T_j \) with the only restriction that the \( T \) and \( T_j \) describe different videos.

### 4.3. Experimental setup

**Dataset Statistics:** In total, ASRL contains 39.5k videos with 80k queries split into training, validation, and testing with 31.7k, 3.9k, 3.9k videos and 63.8k, 7.9k, 7.8k queries. Each video contains around 2 queries containing 3.45 semantic roles and each query has around 8 words.

**Evaluation Metrics:** We compute the following four metrics: (i) **accuracy:** correct prediction for a given object in a query (recall that a query has references to multiple objects) (ii) **strict accuracy:** correct prediction for all objects in the query (iii) **consistency:** the predictions for each object lie in the same video (iv) **video accuracy:** predictions are consistent and lie in the correct video. While strict accuracy is the most important metric to note for VOG, other metrics reveal useful trends helpful for model diagnosis and building robust VOG models and datasets.

**Metric Computation:** In AE, the noun phrases are only localized in the frame where it is most easily visible. This complicates the evaluation process when the same objects appear across multiple frames (a common occurrence). Thus, we select the highest-scoring proposal box for each role in the query in every frame and set a score threshold. Given a phrase referring to a groundable object, we consider the prediction correct when the predicted box in an annotated frame has an \( IoU \geq 0.5 \) with a ground-truth box. This allows us to compute accuracy in a single video single query (SVSQ) setting.

For **SEP, TEMP, SPAT** we have additional information about which video frames and proposal boxes are not ground-truths. To evaluate **SEP**: we check if the predicted video is correct (which gives us video accuracy), and if so compute the accuracy similar to SVSQ.

In **TEMP** and **SPAT**, for a given role if the predicted boxes not belonging to the ground-truth video have a score higher than a threshold, then the prediction for the role is marked incorrect. If the boxes are in the ground-truth video, we evaluate it similar to SVSQ (see Appendix C for examples of each strategy).

**Baselines:** Prior work on VOG cannot be evaluated on ASRL due to their restrictive formulations. For instance, [76] grounds all objects when using TEMP and SPAT resulting in 0 accuracy and [8] needs spatio-temporal tubes.

Recently, [75] proposed GVD, a model for grounded video description. GVD calculates its grounding accuracy by feeding the ground-truth description into a captioning system and finding the highest scored objects. However, this is not applicable to our task because it considers the language in a sequential manner. For an input query “man throwing ball”, GVD would ground “man” without looking at the remaining description and thus fail at grounding in our proposed contrastive setting.

As an alternative, we propose two competitive baselines: (i) **ImgGrnd:** an image grounding system which treats each frame independently and does not explicitly encode object relations. (ii) **VidGrnd:** a video grounding system based on GVD using an object transformer to encode object relations. For fair comparisons, we use the same language features, visual features (the proposal and segment features) for both ImgGrnd and VidGrnd.

**Implementation details:** We re-use the extracted visual features provided by [75] for AE. The object proposals and features are obtained from a FasterRCNN [47] trained on visual genome [30]. Segment features (both RGB and Flow features) are obtained using TSN [62] trained on ActivityNet [4]. For each video, \( F=10 \) frames are uniformly sampled and for each frame, we consider \( P=100 \) proposals which gives a recall of 88.14%. However, training with 100 proposals is time-consuming and computationally expensive. Instead, we introduce GT5 setting where we use exactly 5 proposals per frame. In unannotated frames, it includes the highest-scoring proposals, and for annotated frames, for each ground-truth box, it prioritizes the proposal having the highest \( IoU \). GT5 maintains a similar recall score (86.73%), and allows experimenting with more variations and sets upper performance bound.

For self-attention, both Object Transformer (OTx) and Multi-Modal Transformer (MTx) use multi-head attention [61] with \( n_l=1 \) layer and \( n_h=3 \) heads unless mentioned otherwise. In general, Object Transformer (OTx) applies self-attention across all proposals and frames whereas the Multi-Modal Transformer (MTx) applies self-attention to each frame separately due to higher computation load. We train all models until the validation accuracy saturates. For **SEP, TEMP, SPAT** we found 10 epochs with batch size 4 for GT5 and 2 for P100, using Adam with learning rate \( 1e^{-4} \) to be sufficient for most models. For **SVSQ**, we set batch size 4 for all models. We use the model with the highest validation accuracy for testing. We set the threshold used in evaluating **TEMP** and **SPAT** as 0.2 for GT5 and 0.1 for P100 across all models. More implementation details are provided in Appendix D.
### 4.4. Results and Discussions

In Table 3, we compare VOGNet against two baselines, ImgGrnd and VidGrnd, across GT5 (5 proposal boxes per frame) and P100 (100 proposal boxes per frame).

**Comparison of Strategies:** We note that in the **SVSQ** column, all the models perform comparably. However, these results fail to generalize to other cases which indicates that evaluating on **SVSQ** is insufficient. Next, the **SEP** column shows that models can distinguish contrastive samples by considering the contribution of each object independently with very high accuracy and can easily distinguish similar examples achieving ≈50% on video accuracy even in the P100 setting. Such cues are not present in **SPAT** and **TEMP** where the model is given a single video and single query but now the video contains more than one actor performing some action. The performance in both **SPAT** and **TEMP** is still very low (strict accuracy for P100 is <5%), which suggests that VOG remains an extremely challenging problem for current state-of-art models.

**Comparison with Baselines:** For both **TEMP** and **SPAT**, we find ImgGrnd performs relatively well (~17% in GT5) despite not using any object relations. This is likely because the model can exploit attribute information (such as “red shirt”) in the phrases. VidGrnd which uses language independent object relations obtains gains of 2–3%. Finally, VOGNet, which additionally uses language-dependent object relations, outperforms VidGrnd by another 3–4%.

**GT5 vs P100:** We observe that both GT5 and P100 follow similar patterns across metrics suggesting GT5 is a good proxy to explore more settings. For the remaining experiments, we consider only the GT5 setting.

---

Table 3. Comparison of VOGNet against ImgGrnd and VidGrnd. GT5 and P100 use 5 and 100 proposals per frame. Here, Acc: Grounding Accuracy, V Acc: Video accuracy, Cons: Consistency, SAcc: Strict Accuracy (see Section 4.3 for details). On the challenging evaluation metrics of **TEMP** and **SPAT**, VOGNet (ours) shows significant improvement over competitive image and video grounding baselines.

<table>
<thead>
<tr>
<th>Model</th>
<th>SVSQ</th>
<th>TEMP</th>
<th>SPAT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc</td>
<td>VAcc</td>
<td>Acc</td>
</tr>
<tr>
<td>ImgGrnd</td>
<td>75.31</td>
<td>56.53</td>
<td>39.78</td>
</tr>
<tr>
<td>VidGrnd</td>
<td>75.42</td>
<td>57.16</td>
<td>41.59</td>
</tr>
<tr>
<td>VOGNet</td>
<td>76.34</td>
<td>58.85</td>
<td>42.82</td>
</tr>
</tbody>
</table>

Table 4. Evaluation of VOGNet in GT5 setting by training (first column) and testing (top row) on **SVSQ**, **TEMP**, **SPAT** respectively.

<table>
<thead>
<tr>
<th></th>
<th>SVSQ</th>
<th>TEMP</th>
<th>SPAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc</td>
<td>SAcc</td>
<td>Acc</td>
<td>SAcc</td>
</tr>
<tr>
<td>ImgGrnd</td>
<td>76.38</td>
<td>59.58</td>
<td>1.7</td>
</tr>
<tr>
<td>VidGrnd</td>
<td>75.4</td>
<td>57.38</td>
<td>23.07</td>
</tr>
<tr>
<td>VOGNet</td>
<td>75.15</td>
<td>57.02</td>
<td>22.6</td>
</tr>
</tbody>
</table>

Table 5. Comparison of Contrastive Sampling (CS) vs Random Sampling (Rnd) for training (row-1,2) and evaluation (row-2,3).

<table>
<thead>
<tr>
<th></th>
<th>SEp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc</td>
<td>VAcc</td>
</tr>
<tr>
<td>Rnd</td>
<td>CS</td>
</tr>
<tr>
<td>CS+Rnd</td>
<td>CS</td>
</tr>
<tr>
<td>CS+Rnd</td>
<td>Rnd</td>
</tr>
</tbody>
</table>

Table 6. Training VOGNet in **SPAT** setting with different number of concatenated videos and tested on **SPAT** with 4 videos.

<table>
<thead>
<tr>
<th>#vids</th>
<th>#epochs</th>
<th>Acc</th>
<th>VAcc</th>
<th>Cons</th>
<th>SAcc</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>20</td>
<td>20.18</td>
<td>10.18</td>
<td>52.45</td>
<td>8.84</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>21.7</td>
<td>13.33</td>
<td>55.55</td>
<td>10.68</td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>23.34</td>
<td>14.53</td>
<td>56.51</td>
<td>11.71</td>
</tr>
</tbody>
</table>

Table 7. Ablative study comparing gains from Multi-Modal Transformer (MTx) and Object Transformer (OTx) and Relative Position Encoding (RPE). L: Number of Layers, H: Number of Heads in the Transformer. Note that VOGNet = ImgGrnd + MTx(1L,3H) + OTx(1L,3H) + RPE.
Performance across Strategies: Table 4 shows that VOGNet trained in SPAT and TEMP settings performs competitively on SVSQ (maintaining $\approx 75\%$ accuracy). However, the reverse is not true i.e. models trained on SVSQ fail miserably in SPAT and TEMP (accuracy is $<3\%$). This suggests that both TEMP and SPAT moderately counter the bias caused by having a single object instance in a video. Interestingly, while VOGNet trained on TEMP doesn’t perform well on SPAT (performance is worse than VidGrnd trained on SPAT), when VOGNet is trained on SPAT and tested on TEMP it significantly outperforms VidGrnd trained in TEMP. This asymmetry is possibly because the multi-modal transformer is applied to individual frames.

Contrastive Sampling: Table 5 compares Contrastive Sampling (CS) to a Random Sampling (RS) baseline for evaluation and training. Using RS for validation, SEP video accuracy is very high $75\%$ implying that CS is a harder case; similarly, we find higher performance in both TEMP and SPAT cases. Interestingly, using RS for training is only slightly worse for SPAT, TEMP while outperforming in SEP. Thus, CS in SPAT and TEMP helps learn better object relations, but random sampling remains a very competitive baseline for training. Table 6 shows that using more videos in training helps; we use 4 videos due to GPU memory considerations and training time.

Ablation Study: In Table 7 we record the individual contributions of each module in SPAT. We observe: (i) self-attention via object is an effective way to encode object relations across frames (ii) multi-modal transformer applied on individual frames gives modest gains but falls short of object transformer due to lack of temporal information (iii) relative position encoding (RPE) boosts strict accuracy for both transformers (iv) object transformer with 3 layers and 6 heads performs worse than using a single multi-modal transformer i.e. adding more layers and attention heads to object transformer is not enough (v) using both object and multi-modal transformers with more layers and more heads gives the best performing model.

4.5. Visualizations

For qualitative analysis, we show the visualizations of SPAT and TEMP strategies in Figure 4. In the interest of space, we use $k=2$ contrastive sampling (visualizations with $k=4$ are provided in the Appendix F). In the first image, the videos are concatenated along the width axis and both contain a “man” and “ball”. Our model correctly identifies which “ball” is being thrown into the air and by whom. Note that only viewing the last frame doesn’t uniquely identify if the “man” visible in the current frame has thrown the ball. In general, our SPAT model performed with high consistency i.e. it chose objects nearer to each other which we attribute to RPE. In the second image, the videos are concatenated along the time-axis and in both videos, the person “spins” something. Using “board” as an additional cue, our model correctly finds both the “person” and the “board that he spins”. Our TEMP model performs slightly worse than SPAT model possibly because encoding temporal information is more challenging. Finally, in the third image, our model grounds “he” incorrectly likely due to not being able to ground “dummy”.

5. Conclusion

In this work, we analyze the problem of VOG which aims to localize the referred objects in a video given a language query. We show that semantic-role labeling systems can be used to sample contrastive examples. We then enforce that the model views the contrastive samples as a whole video so that the model explicitly learns object relations. We further propose an additional self-attention layer to capture language dependent object relations along with a relative position encoding. Finally, we validate our proposed model VOGNet on our dataset ActivityNet-SRL which emphasizes the role of object interactions.

Acknowledgement: We thank the anonymous reviewers for their suggestions. This research was supported, in part, by the Office of Naval Research under grant #N00014-18-1-2050.
References


[37] Chih-Yao Ma, Asim Kadav, Iain Melvin, Zsolt Kira, Ghasan Al-Regib, and Hans Peter Graf. Grounded objects and interactions for video captioning. CVPR Workshop, 2017. 1
[38] Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In ACL Workshop, 2014. 5, 10
[40] Alejandro Newell and Jia Deng. Pixels to graphs by associative embedding. In NIPS, 2017. 2
[57] Carina Silberer and Manfred Pinkal. Grounding semantic roles in images. In EMNLP, 2018. 2
[64] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph R-CNN for scene graph generation. In ECCV, 2018. 2


[76] Luowei Zhou, Nathan Louis, and Jason J. Corso. Weakly-supervised video object grounding from text by loss weighting and object interaction. In *BMVC*, 2018. 1, 2, 4, 5, 6
