Social Fabric: Tubelet Compositions for Video Relation Detection

Shuo Chen, Zenglin Shi, Pascal Mettes, and Cees G. M. Snoek
University of Amsterdam

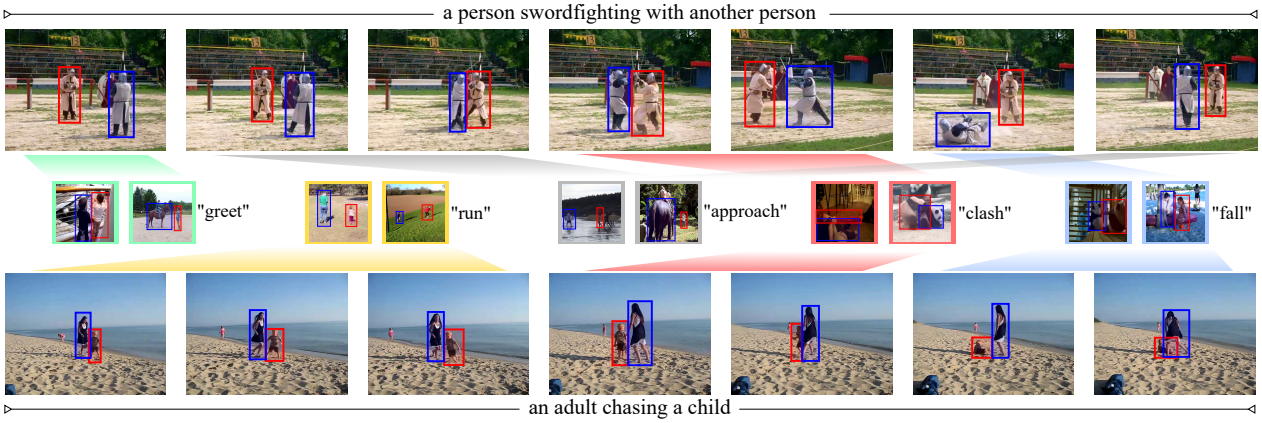

Figure 1: **Social Fabric** encodes compositions of interaction primitives defined over tubelet pairs. The primitives are data driven and may correspond to interactions like "greet", "clash" and "fall". Using the primitives, our two-stage network can classify, detect, and search for complex relations across entire videos.

Abstract

This paper strives to classify and detect the relationship between object tubelets appearing within a video as a ⟨subject-predicate-object⟩ *triplet. Where existing works treat object proposals or tubelets as single entities and model their relations* a posteriori*, we propose to classify and detect predicates for pairs of object tubelets* a priori*. We also propose Social Fabric: an encoding that represents a pair of object tubelets as a composition of interaction primitives. These primitives are learned over all relations, resulting in a compact representation able to localize and classify relations from the pool of co-occurring object tubelets across all timespans in a video. The encoding enables our two-stage network. In the first stage, we train Social Fabric to suggest proposals that are likely interacting. We use the Social Fabric in the second stage to simultaneously fine-tune and predict predicate labels for the tubelets. Experiments demonstrate the benefit of early video relation modeling, our encoding and the two-stage architecture, leading to a new state-of-the-art on two benchmarks. We also show how the encoding enables query-by-primitive-example to search for spatio-temporal video relations. Code:* https://github.com/shanshuo/Social-Fabric.

1. Introduction

To understand what is happening where in videos, it is necessary to detect and recognize relationships between individual instances. Effectively capturing these relationships could improve captioning [55], video retrieval [41], visual question answering [1] and many other visual-language tasks. In this paper, we strive to classify and detect the relationship between object tubelets appearing throughout a video as a *⟨subject-predicate-object⟩* triplet, like *⟨dog-chase-child⟩* or *⟨horse-stand_behind-person⟩*.

Shang *et al*. [38, 39] pioneered this challenging problem by their definition of video datasets with dense bounding box annotations, temporal bounds, and relationship-triplet labels. Following their guidance, a leading approach to date is to generate proposals for individual objects on short video snippets, encode the proposals, predict a relation and associate the relations over the entire video, *e.g*. [34, 42, 53]. To better detect long-term interactions, Liu *et al*. [30] forego the need for snippets by first localizing individual object tubelets throughout the entire video, filter out unlikely pairs and predict predicates for the remaining ones. Different from all these existing works on video relation prediction, which treat object proposals or tubelets as single entities and model their relations *a posteriori*, we propose to clas-

sify and detect predicates for pairs of object tublets *a priori*.

Considering objects as tubelet pairs from the start requires an encoding that enables us to localize and classify interactions from the pool of all co-occurring object tubelets across all timespans in a video. This is reminiscent of many classical problems in computer vision that need to aggregate spatial, *e.g*. [2, 22, 40, 47], temporal, *e.g*. [28, 50, 57] or spatio-temporal, *e.g*. [15, 16, 32] primitives into a common representation. We take inspiration from ActionVLAD by Girdhar *et al*. [16], which encodes actions as a composition of local action primitives to capture the entire spatio-temporal extent of actions. In this paper, we also learn to encode local spatio-temporal video features in a compositional manner. Different from ActionVLAD, which operates on an entire video, our Social Fabric encoding operates on tubelet pairs, *i.e*. on inputs from multiple object tubelets and multiple modalities, with a set of interaction primitives that is dynamically learned during video relation training. Social Fabric captures information across the entire scope of tubelet pairs, which is especially beneficial when interactions last long. See Figure 1 for an illustrative example.

We make three contributions. First, we propose to classify and detect video relations for pairs of object tubelets from the start. Second, we introduce Social Fabric, a compositional encoding suited for multi-tubelet and multi-modal inputs. The interaction primitives that form the encoding are learned and updated dynamically, akin to the NetVLAD layer from Arandjelović *et al*. [2] for visual place recognition. Third, to leverage the Social Fabric, we propose a two-stage network for video relation classification and detection. In the first stage, we localize interactions by training Social Fabric to propose tubelet pairs that are likely interacting. In the second stage we use the Social Fabric to simultaneously fine-tune and learn to predict predicate labels for the tubelets. Experiments on the benchmarks for video relation detection of Shang *et al*. [38, 39] show the benefits of our approach, especially when interactions are long and complex. Social Fabric outperforms alternative video encodings and our two-stage architecture sets a new state-of-the-art for both video relation classification and detection. Besides classification and detection, we show that our encoding enables searching for relations in videos by providing primitive-examples as queries.

2. Related Work

Image relation detection. Visual relation recognition has a long-standing tradition for static images [8, 17, 18, 20, 21, 26, 27, 31, 49, 56]. Besides recognizing visual relationships between objects, Chao *et al*. [7] introduce the problem of detecting human-object-interactions in static images and contribute a corresponding dataset. It inspired many to contribute to human-object-interaction detection, *e.g*. [10, 26, 49, 51, 54]. Li *et al*. [26], for example, learn the knowledge between human and object categories from the provided datasets and use this knowledge as a prior while performing detection. Wan *et al*. [49] introduce a pose-aware network that employs a multi-level feature strategy. Where image-based relation detection requires two boxes (subject and object) and a predicate, we aim to perform video-based relation detection, which requires us to also localize and track subjects and objects over time.

Snippet relation detection. Many before us have investigated relation detection in videos [5, 11, 25, 30, 34, 38, 39, 42, 43, 44, 46, 53, 59]. Relation in videos provide additional temporal information, important for interactions such as pushing or pulling a closed door. Shang *et al*. [39] pioneered this problem and introduced the ImageNet-VidVRD dataset, the first video relation detection benchmark in which all video relation triplets, along with their object and subject trajectories, are labelled. Building on the foundational work of Shang *et al*. [39], Tsai *et al*. [46] propose a gated spatio-temporal energy graph using conditional random fields to model video relations. In a similar spirit, Qian *et al*. [34] built a spatio-temporal graph between adjacent video snippets and used multiple layers of graph convolutional networks to pass messages between nodes. Shang *et al*. [38] later introduced VidOR, the largest video relation detection benchmark to date. On this dataset, Sun *et al*. [43] utilize language context features along with spatio-temporal features for predicate prediction.

All the aforementioned methods adopt a three-stage framework. A video is first split into short snippets and subject/object tubelets are generated per snippet. Then, short-term relations are predicted for each tubelet. The subject/object proposals are obtained in the short snippets using an image object detector and tracker [34, 39, 46]. In the second stage, spatio-temporal features of each pair of object tubelets are extracted and used to predict short-term relation candidates. Xie *et al*. [53] combine a wide variety of multi-modal features for each pair to predict the relations with impressive relation classification accuracy. In the third stage, the short-term relation proposals are merged by a greedy relational association algorithm. Su *et al*. [42] maintain multiple relation hypotheses during the association process to accommodate for inaccurate or missing proposals in the earlier steps. Instead of treating the relations independently at the various analysis stages, we consider the objects tubelets as interacting pairs from the start.

Proposal relation detection. Liu *et al*. [30] are the first to avoid the need to split videos into snippets. In a first stage they generate object tubelets for the whole videos. The second stage refines the tubelet-features and finds relevant object pairs using a graph convolutional network. The third stage focuses on predicting the predicates between related pairs. In this manner, interactions can be detected without a need for snippet splitting. Like Liu *et al*., we also avoid the

need for snippets. Different from them, we view subjects and objects as interactions from the start. As a result, we only need two stages, one for interaction proposal generation from the tubelet pairs and one for predicting the appropriate predicate. At the core of both our stages is the Social Fabric, which allows us to encode a set of interaction primitives, like the ones in Figure 1, from which we classify and detect different video relations.

3. Social Fabric Encoding

The goal in video relation detection is to localize interactions between two entities in space and time. Formally, a spatio-temporal interaction $\mathcal{I}$ is defined as a triplet $\mathcal{I} = \{O_1, P, O_2\}$, with subject tubelet $O_1 \in \mathbb{R}^{4\times(T_2-T_1)}$, object tubelet $O_2 \in \mathbb{R}^{4\times(T_2-T_1)}$ and their relation predicate category P. Here, T_1 and T_2 denote the start and end frame of the interaction and each frame contains box coordinates. To address both video relation classification and detection, we propose a two-stage approach that encodes subjects and objects as pairs from the start. Central to both stages is our Social Fabric encoding for representing compositions of tubelet pairs. Below, we outline how to learn the encoding, how to use it to represent tubelet pairs and how the encoding relates to existing video encodings.

Learning the encoding. The idea behind the encoding is that a pair of tubelets, which form a video relation triplet, are composed of multiple interaction primitives. These primitives can represent different relations by varying their combinations. For example, let {"approach","run","watch","touch"} denote a set of primitives, then a hugging relation can be represented by {"watch","approach","touch"}, while a chasing relation can be represented by {"run","approach"}. In the object detection and action recognition literature, compositional learning and encoding is well established, with advantages such as sharing components amongst categories *e.g.* [13], efficient and compact encoding *e.g.* [58], and high discriminative ability *e.g.* [23, 24]. By introducing a compositional encoding for video relation detection we share the same benefits and show some examples of the primitives we learned in Figure 2.

For each task, we are given a training set of tubelet pairs, denoted as $\mathcal{R}$, where the input representation of each tubelet pair is denoted as $S_i \subset \mathcal{R} \in \mathbb{R}^{N\times F}$, with N the number of frames of the tubelets and F the feature dimensionality for each frame, denoting the combined subject and object representations. On top of the features, we apply layer normalization [3], followed by a linear layer to obtain embedded representation $R_i \subset \mathcal{R} \in \mathbb{R}^{N\times D}$. In this D-dimensional embedding space, we learn a set $C \in \mathbb{R}^{K\times D}$ consisting of K primitives. The idea behind our encoding is to describe a tubelet pair entirely as a weighted combination of these primitives. So tubelet pair i is encoded with our approach

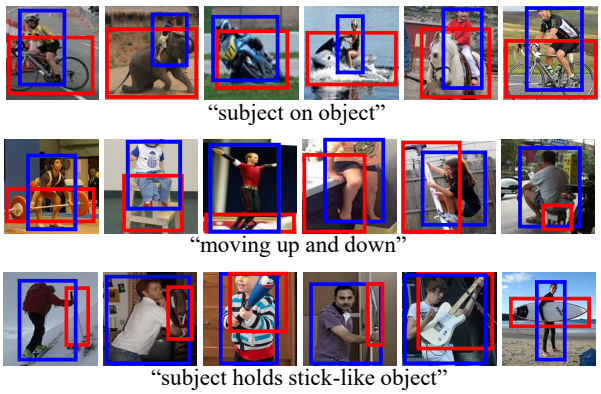

Figure 2: **Interaction primitives** that our Social Fabric encoding learns when trained for multi-modal features. Each row shows several frames from videos that get assigned to one specific primitive. Blue boxes indicate the subject while red boxes denote the object. Here we show some easy to interpret primitives.

as a concatenation of weighted primitive locations:

$$E_i{=}[E_{i,1}, \cdots, E_{i,K}],\ E_{i,k}{=}\sum_{j=1}^{N} z_{ijk} C_k, \tag{1}$$

where the weight is inversely proportional to the distance between a local relational feature vector and the primitive:

$$z_{ijk} = \frac{\exp\left[-\beta \left\|R_{ij} - C_k\right\|^2\right]}{\sum_{l=1}^{K} \exp\left[-\beta \left\|R_{ij} - C_l\right\|^2\right]}, \tag{2}$$

where $\beta > 0$ denotes a temperature parameter to tune how soft or hard the assignments should be, fixed to $1/\sqrt{D}$ throughout this work. Intuitively, our encoding describes how much a relation is in line with each primitive in C. Each portion $E_{i,k}$ of the encoding forms a line between the primitive C_k and the origin; the stronger the agreement, the closer $E_{i,k}$ is to the primitive and the more its values contribute to the next layer. The diagram of the Social Fabric Encoding is shown in Figure 3.

On top of the representation E_i, we learn a fully-connected layer classification head, which can be used to determine whether a tubelet pair makes for a good proposal or to predict its predicate using a shallow network head. The

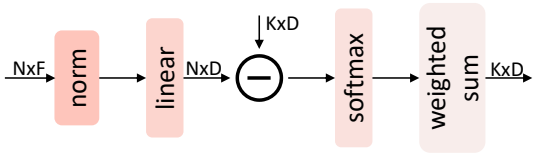

Figure 3: **Social Fabric Encoding**.

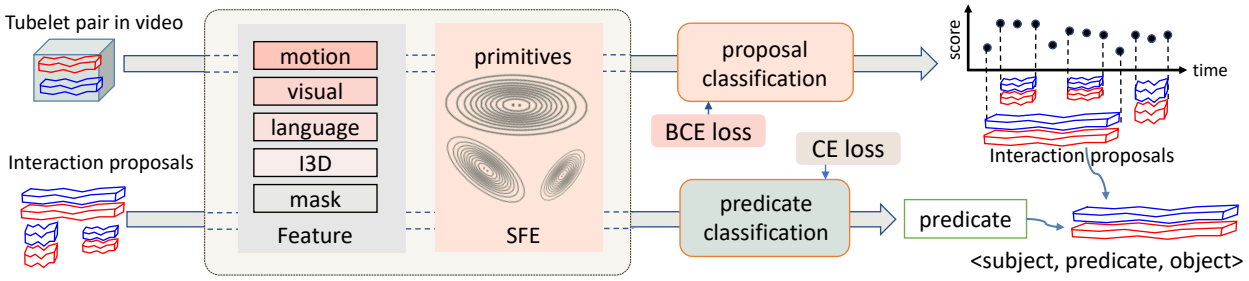

Figure 4: **Two-stage video relation network**. We first obtain interaction proposals and then predicate predictions. Social Fabric Encoding (SFE) is essential to both stages as to represent an object tubelet with a composition of interaction primitives. BCE loss and CE loss represents binary cross-entropy loss and cross-entropy loss separately.

layers of the network and the set C are jointly learned during the optimization.

Relation to alternative encodings. A common encoding in video-based representations is average pooling [57]. In our encoding, average pooling is a special case where the codebook contains a single primitive. Average pooling implicitly assumes that the features of the input representation follow a single mode. Video relations, however, consist of multiple interaction primitives that evolve over time. Moreover, these primitives are shared between different relations, which we capture. Encodings such as transformers follow the self-attention architecture, where each feature is a weighted sum of other features [48]. Compared to transformers, our approach provides a fixed-sized representation, important because tubelet pairs are of varying length. Other encodings like NetVLAD [2] and ActionVLAD [16] operate on whole images and videos, while residuals between local features and clusters are used to obtain a representation. In contrast, our encoding operates on pairs of spatio-temporal tubelets, accepts multi-modal features, and we directly use the primitives to encode inputs. Lastly, we are the first to rely on a compositional encoding for the task of video relation detection.

4. Two-stage video relation network

We utilize the Social Fabric Encoding to both classify and detect video relations using two stages, rather than three stages common in the literature. In the first stage, we sift through all combinations of co-occurring tubelets across all timesteps to obtain a set of interaction proposals that likely cover all ground truth video relations. In the second stage, we classify each proposal with a predicate label. An overview of our approach is visualized in Figure 4. Next, we detail both stages and show how to obtain the final classification and spatio-temporal detection results.

Stage 1: Interaction proposals. We initialize the video relation optimization by performing object detection in each frame, followed by linking over time based on [52]. For a video V, this results in M object tubelets. We consider all unique combinations of tubelets for proposal generation and train a binary classifier to determine interactivityness at the frame-level using a local window around the box pairs in a frame [9]. For the two objects (O_1, O_2) in a tubelet pair and frame f, we consider a neighbourhood of $m/2-1$ frames in both temporal directions of the tubelets. We compute and stack the multi-modal features for the windowed tubelet pair, resulting in $R^1_f(O_1, O_2) \in \mathbb{R}^{m \times D}$ for frame f. We feed this as input to Social Fabric, resulting in $E^1(O_1, O_2) \in \mathbb{R}^{K \times D}$. During training, the encoding is used to train a binary classifier to separate potential interactions from non-interactions with a binary cross-entropy loss $\mathcal{L}=\big(y \log(s) + (1-y)\log(1-s)\big)$, where s denotes the interactivityness. Simultaneously, the primitives in the Social Fabric are learned. For each frame in a tubelet pair, this results in a score indicating its interactivityness. Over the array of scores over all timesteps of the tubelet pair, we employ a 1D watershed algorithm [9,36] to generate spatio-temporal interaction proposals. We repeat this procedure for all co-occurring tubelets and combine the outputs per pair into a final set of interaction proposals for a video.

Stage 2: Predicate prediction. Once a video is decomposed into a set of interaction proposals, each consisting of two tubelets with a similar start and end time, we seek to score all proposals for their predicate. For interaction proposal (O_1, O_2), we sample n frames uniformly. For each sampled frame, we extract a single uni-modal or several multi-modal features. Then we stack the features over all frames and obtain $R^2(O_1, O_2) \in \mathbb{R}^{N \times D}$ for this tubelet. This is fed into Social Fabric and the output representation is in $E^2(O_1, O_2) \in \mathbb{R}^{K \times D}$. In stage 2 we fine-tune the Social Fabric trained in stage 1 to accelerate the convergence. After encoding each proposal, we feed the representation into a final linear layer to obtain predicate scores. The predicate prediction is optimized with softmax cross-entropy. After obtaining predicate predictions, we multiply the predicate score and corresponding subject and object scores as

the relation triplet prediction score. The subject and object scores are obtained from the tubelet pairs in stage 1. Relation triplets are the predicted results for relation classification. The relation triplet associated with subject and object tubelets act as the predicted results for relation detection.

Search-by-primitive-example. The Social Fabric encoding is optimized for video relation classification and detection, but is not limited to these tasks. Here, we show how we can also search for spatio-temporal video relations in a collection of videos by querying primitive examples. As input, a user can provide one or more frames with a subject and object performing a basic interaction. We compute the non-temporal features for each input and use it to find the nearest learned primitive. To find the interaction proposal across all videos that best describes the primitive examples, we use the weights from Equation 2 to score the relevance of each primitive for an entire proposal. In turn, we simply sum the scores for the few primitives determined by the user and output the interaction proposal with the highest score. As a result, we can search on-the-fly for video relations that are composed of example primitives provided by a user, without the need for search optimization.

5. Experimental setup

5.1. Datasets

To evaluate the proposed methods, we perform experiments on ImageNet-VidVRD [39] and Video Object Relation (VidOR) [38].

ImageNet-VidVRD. [39] consists of 1,000 videos, created from the ILSVRC2016-VID dataset [37]. There are 35 object categories and 132 predicate categories. The videos are densely annotated with relation triplets in the form of ⟨*subject-predicate-object*⟩ as well as the corresponding subjects and objects trajectories. Following [39,46], we use 800 videos for training and the remaining 200 for testing.

VidOR. [38] contains 10,000 user-generated videos selected from YFCC-100M [45], for a total of about 84 hours. There are 80 object categories and 50 predicate categories. Besides providing annotated relation triplets, the dataset also provides the bounding boxes of objects. The dataset is split into a training set with 7,000 videos, validation set with 835 videos, and a testing set with 2,165 videos. Since the ground truth of the test set is not available, we use the training set for training and the validation set for testing, following [30, 34, 42, 53].

5.2. Implementation and evaluation details

Tubelet pairing. We first detect all the objects per video frame by Faster R-CNN [35] with a ResNet-101 [19] backbone. The detector is trained on MS-COCO [29]. The detected bounding boxes are linked with the Deep SORT tracker [52] to obtain individual object tubelets. Finally, each tubelet is paired with any other tubelet to generate the tubelet pairs. We use the object trajectories of ImageNet-VidVRD and VidOR adopted in [34,39,42,43] for fair comparison.

Feature extraction. In the video relation literature, features from multiple modalities are commonly used, *e.g*. Sun *et al*. [43] use motion features and language features. Liu *et al*. [30] use motion features, visual features and I3D features. Xie *et al*. [53] use motion features, visual features, language features and location mask features. We consider all features and arrive at motion features, visual features, language features, I3D features, and location mask features. We follow [43] to calculate the spatial location feature as motion features. The visual features are extracted using the detection backbone in Faster R-CNN and followed by an RoI pooling layer. For the language features we use a word2vec module, pre-trained on GoogleNews [33], to encode the subject and object classes into language features with dimension of 600. We use the I3D module from [6] to extract I3D features with fixed dimension of 832. We follow the method of [53] to generate a mask based on the bounding boxes of the subject and object in the tubelet pair.

Two-stage network optimization. The size of the linear layer for embedding representation is D=512. In the first stage, we consider m=30 neighbourhood frames on both temporal directions. The interaction proposal generation network is trained for 20 epochs using an SGD optimizer with a mini-batch of 128. We use a fixed learning rate and set its value to 0.01. In the second stage, we sample n=25 frames for each interaction proposal. The predicate prediction network is trained for 10 epochs using an SGD optimizer with a mini-batch of 128. We use a fixed learning rate and set its value to 0.01.

Evaluation metrics. Following [39], we adopt Precision@1, Precision@5 and Precision@10 to measure the ability of classifying visual relations. We will refer to the classification task as relation tagging in the experiments for consistency with current literature. For video relation detection we report mAP (mean Average Precision), Recall@50 and Recall@100.

6. Results

Benefit of multi-modal features. We first evaluate the benefit of the use of multi-modal features on VidOR in Table 1. With only motion features, our method achieves a P@1 of 50.97 for relation tagging and an mAP of 6.14 for relation detection. With all features included, the performance is clearly improved with a P@1 of 68.86 for relation tagging and an mAP of 11.21 for relation detection. The results show that our encoding benefits from incorporating information from many modalities. In the following ablations, we use all features.

Influence of encoding size. Next, we evaluate the influ-

Feature type					Relation tagging			Relation detection		
motion	visual	language	I3D	mask	P@1	P@5	P@10	mAP	R@50	R@100
✓					50.97	39.57	31.58	6.14	6.74	8.70
✓	✓				56.89	44.76	34.07	8.93	7.38	9.22
✓	✓	✓			59.24	47.24	35.99	9.54	8.49	10.17
✓	✓	✓	✓		61.52	50.05	38.48	10.04	8.94	10.69
✓	✓	✓	✓	✓	**68.86**	**55.16**	**43.40**	**11.21**	**9.99**	**11.94**

Table 1: **Benefit of multi-modal features** on VidOR. More is better. The increasing gaps indicate Social Fabric effectively captures multi-modal features for relation classification and detection.

Clusters	1	8	32	64	128
mAP	10.05	10.69	10.91	**11.21**	11.01

Table 2: **Influence of encoding size** on VidOR for relation detection. Using multiple primitives results in a more accurate predicate prediction, where we achieve best performance for 64 primitives.

		Relation tagging			Relation detection		
Stage 1	Stage 2	P@1	P@5	P@10	mAP	R@50	R@100
		60.72	46.40	36.62	9.61	8.73	10.81
✓		64.19	49.60	39.22	10.16	9.62	11.63
✓	✓	**68.86**	**55.16**	**43.40**	**11.21**	**9.99**	**11.94**

Table 3: **Importance of two stages** on VidOR. Incorporating Social Fabric into the two stages of our pipeline (third row) is preferred over baselines based on average pooling of features with video snippet proposals (first row) and using Social Fabric only for the proposals (second row).

	Relation tagging	Relation detection
Encoding	P@1	mAP
average pooling	62.73	10.05
transformer	63.86	10.07
NetVLAD	65.34	10.15
NetRVLAD	66.80	10.55
Social Fabric	**68.86**	**11.21**

Table 4: **Comparison with alternative encodings** on VidOR. Social Fabric performs well.

ence of the number of interaction primitives in the Social Fabric Encoding. Intuitively, the more primitives, the finer commonalities between interactions are modelled. In Table 2, we find that multiple primitive components indeed improves over a single component (which resembles conventional average pooling). When increasing the number of primitives, we further improve the performance. The Social Fabric Encoding performs best at K=64, where it provides a balance between coverage of the space and sharing amongst relations. We use this encoding size for further experiments.

Importance of two stages. Next, we show the importance of the interaction proposal stage and the predicate predication stage on VidOR in Table 3. The baseline (first row) splits the video into short snippets. Relationships are separately detected in each snippet and merged afterwards, akin to [34,42,53]. It average pools the features before predicate prediction. With the interaction proposal stage added (second row), we have spatio-temporal proposals covering long-range interactions. It provides the necessary context to recognize long duration interactions. Accordingly, both recall and precision improve. The Recall@50 is improved by 1.09 and P@1 is improved by 3.47 compared to the baseline. Upon adding the second stage (Third row), the P@1 increases by 4.67 compared to when we only use interaction encoding in proposal generation. We conclude that both stages matter in combination with our encoding.

Comparison with alternative encodings. We compare to the following encodings on VidOR: average pooling, transformer encoding, NetVLAD [16], NetRVLAD [32]. Average pooling corresponds to our encoding with a single mixture component. Transformers were proposed in [48] for textual sequence-to-sequence tasks and recently adopted in video tasks [4, 14, 15]. Here, we investigate their potential for interaction detection. We feed the frame-level representations to the transformer encoder. The output representation is average pooled and then fed into the predicate classifier. NetVLAD was first introduced for place recognition and later adopted for video action classification in [16]. We train a classifier over the NetVLAD layer initialized by k-means on all features to initialize the cluster centroids (and keep it fixed). As our method, we use 64 cluster centroids. NetRVLAD [32] is a simplification of the original NetVLAD architecture that averages the actual descriptors instead of the residuals.

We report the P@1 and mAP on VidOR dataset in Table 4. All encodings take the same multi-modal representa-

	ImageNet-VidVRD						VidOR				
	Relation tagging			Relation detection			Relation tagging		Relation detection		
	P@1	P@5	P@10	mAP	R@50	R@100	P@1	P@5	mAP	R@50	R@100
Shang *et al*. [39]	43.00	28.90	20.80	8.58	5.54	6.37	-	-	-	-	-
Tsai *et al*. [46]	51.50	39.50	28.23	9.52	7.05	8.67	-	-	-	-	-
Qian *et al*. [34]	57.50	41.00	28.50	16.26	8.07	9.33	-	-	-	-	-
Sun *et al*. [43]	-	-	-	-	-	-	51.20	40.73	6.56	6.89	8.83
Su *et al*. [42]	57.50	41.40	29.45	19.03	9.53	10.38	50.72	41.56	6.59	6.35	8.05
Liu *et al*. [30]	60.00	43.10	32.24	18.38	11.21	13.69	48.92	36.78	6.85	8.21	9.90
Xie *et al*. [53]	-	-	-	-	-	-	67.43	-	9.93	9.12	-
This paper, features as Su *et al*. [42]	57.50	43.40	31.90	19.23	12.74	16.19	54.57	43.58	8.93	9.15	11.13
This paper, features as Liu *et al*. [30]	61.00	47.50	36.60	19.77	12.91	16.32	55.40	45.74	9.13	9.36	11.30
This paper, features as Xie *et al*. [53]	-	-	-	-	-	-	68.62	53.34	11.05	9.91	11.89
This paper, our features	**62.50**	**49.20**	**38.45**	**20.08**	**13.73**	**16.88**	**68.86**	**55.16**	**11.21**	**9.99**	**11.94**

Table 5: **Comparison with state-of-the-art** for relation tagging and detection on ImageNet-VidVRD and VidOR. We outperform the recent snippet relation detection methods of both Su *et al*. and Xie *et al*. for almost all metrics when using their features. We also outperform the proposal relation detection method of Liu *et al*. when using their features. When we rely on our full set of features results improve further and set a new state-of-the-art on both tasks for both benchmarks.

tions as input. The transformer and average pooling baselines obtain similar performance. NetVLAD improves over average pooling and transformers, highlighting the effectiveness of codebook-based encodings. NetRVLAD further improves over NetVLAD, potentially because aggregating the actual feature instead of residuals may benefit the performance [12]. Our encoding uses a similar strategy with a dynamic learning scheme and outperforms all baselines, with an mAP of 11.21% compared to 10.55% for NetRVLAD as the best performing alternative.

Comparison with state-of-the-art. We compare with the state-of-the-art in video relation classification and detection in Table 5 for both ImageNet-VidVRD and VidOR. Liu *et al*. [30] report good results for relation classification and detection on both sets. When we compare with them using the same input features, *i.e*. visual, I3D and motion feature, we improve over their work on all metrics. Most notably, the mAP for relation detection improves from 18.38 to 19.77 on ImageNet-VidVRD and from 6.85 to 9.13 on VidOR. We also compare favorably against the recent snippet-based video relation detection of Su *et al*. [42] using their features. We are on par for the relation classification P@1 on ImageNet-VidVRD, but outperform them on all other metrics and datasets, demonstrating the benefit of detecting predicates for social tubelets from the start. Xie *et al*. [53] improved the state-of-the-art considerably by combining a motion feature, visual feature, language feature and location mask feature for each trajectory pair before predicting their relation. Our method profits from such a rich set of multi-modal features also. When we use the same features as Xie *et al*. our results get better as well, obtaining 68.62 P@1 and 11.05 mAP for relation classification and detection respectively. Our features adds I3D feature to the feature set used by Xie *et al*. [53]. Using our features we obtain state-of-the-art performance with 11.21 mAP and 68.86 P@1. We also consider the computational aspects of our method. We test using a GTX 1080 Ti GPU. With the same features as Liu *et al*. [30], the average time to process one ImageNet-VidVRD validation video is 58.2s for Liu *et al*. [30], and 48.3s for our method.

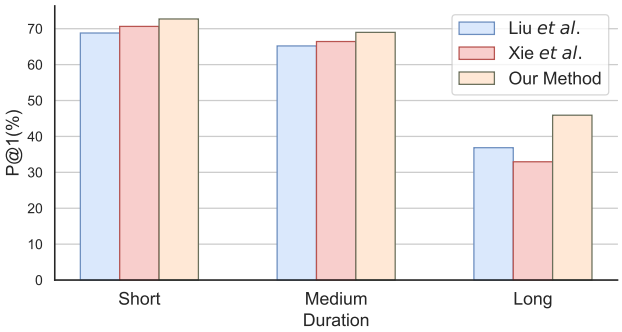

Figure 5: **Comparison along relation duration** on VidOR. We observe our method's performance improves over alternatives as the duration of the video relation increases.

Comparison along relation duration. To verify the effectiveness of our approach on long-range relations. we break down the performance into three bins according to the duration of the relation instances: "short", "medium" and "long". We compare our method with Liu *et al*. [30] and Xie *et al*. [53] on the VidOR validation set. Results are shown in Figure 5. The three methods use the same features as Xie *et al*. [53] for fair comparison. The results of Xie *et al*. [53] are provided by the authors. The results of Liu *et al*. [30] are obtained by running the pro-

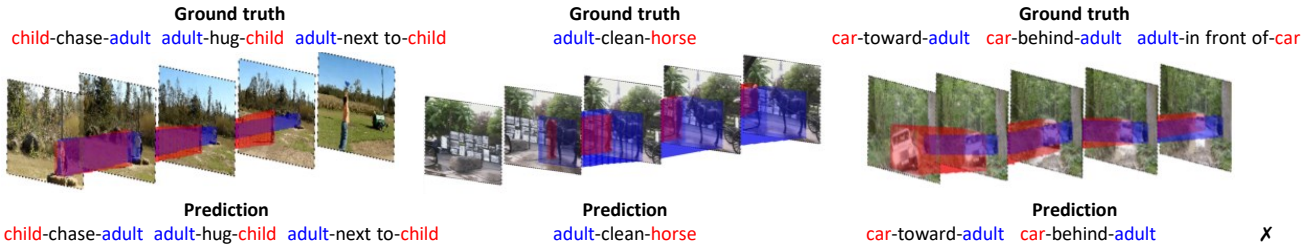

Figure 6: **Success and failure cases** on VidOR. For the left example, we detect all the ground truth relation instances and successfully predict the long-range relation `chase`. The middle case needs temporal context information to detect an adult cleaning a horse. Our method's detection proves its ability to detect long-range relations. In the right example, our approach detects `behind` and `toward` relations. But since the object detector wrongly recognizes `car` as `truck`, the final triplet predictions are wrong even though the relation predicates are correct. Incorrect object categories also lead to imprecise semantic features, which may contribute to the missing of a relation prediction. We provide more qualitative results and example videos with success and failure in the supplemental material.

vided code. As expected, Liu *et al*. [30] surpasses Xie *et al*. [53] for long duration relations as they are designed to be effective beyond short-snippets. Our method is beyond both Liu *et al*. and Xie *et al*. for all durations. Compared to Xie *et al*. [53] who do not consider long-range relations, our method's performance gain increases as the relation length increases. We conclude our approach is beneficial for encoding multi-modal features for relation detection especially at long-range. Besides, we have split the predicates in VidOR into two super categories: action-based and spatial-based relations, following [37]. We obtain a mAP of 7.33% for action-based relations and a mAP of 12.89% for spatial-based relations, while the state-of-the-art by Xie *et al*. [51] obtains a mAP of 6.25% for action-based relations and a mAP of 11.23% for spatial-based relations. We show some success and failure cases in Figure 6.

Video relation query-by-primitive-examples. In Figure 7 we show three search cases, where for each case three primitive examples are given as input. We use the VidOR validation set for the search. The results show that we can find relevant video relations in space and time across many videos, simply by providing a few primitive examples, further highlighting the importance of compositions for video relations.

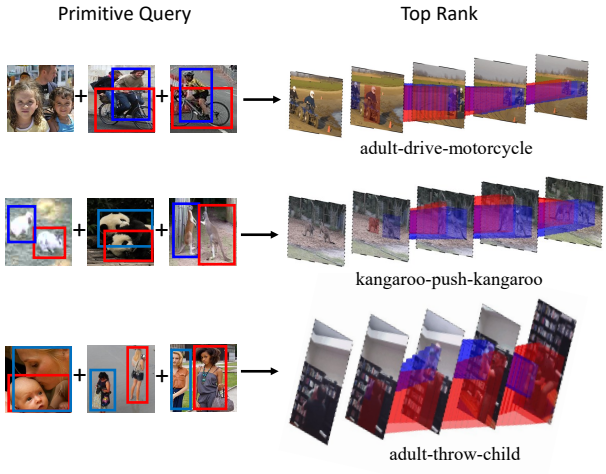

Figure 7: **Query-by-primitive-examples.** We use three examples of primitives as query. Among the VidOR validation set, the relation whose primitive weights are closest to the three examples is selected. *e.g*., in third row, three examples represent primitives of "subject touches object", "subject and object moving away" and "subject and object are person". And the top ranked relation we return is *adult, throw, child*.

7. Conclusion

We propose an approach to video relation classification and detection that operates on pairs of object tubelets from the start. By doing so we no longer have to scatter the video into snippets or individual object tubelets and gather them at the end. To represent all pairs of object tubelets appearing in a video, we propose Social Fabric: an encoding built on a composition of data-driven interaction primitives, akin to the classical codebook approach. We use the encoding in a two-stage network, that first suggest proposals that are likely interacting and then fine-tunes and predicts it most likely predicate label. Experiments demonstrate the benefit of early video relation modeling, our encoding, as well as the two-stage architecture, leading to a new state-of-the-art on two video relation benchmarks. We also show how the encoding enables spatio-temporal video search by query-by-primitive-examples.

Acknowledgements. The authors thank Pengwan Yang for his help on figure design and comments.

References

[1] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *CVPR*, 2015. 1

[2] Relja Arandjelović, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *CVPR*, 2016. 2, 4

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv*, 2016. 3

[4] Petr Byvshev, Pascal Mettes, and Yu Xiao. Heterogeneous non-local fusion for multimodal activity recognition. In *ICMR*, 2020. 6

[5] Qianwen Cao, Heyan Huang, Xindi Shang, Boran Wang, and Tat-Seng Chua. 3-D Relation Network for visual relation recognition in videos. *Neurocomputing*, 2021. 2

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 2017. 5

[7] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *WACV*, 2018. 2

[8] Yu-Wei Chao, Zhan Wang, Yugeng He, Jiaxuan Wang, and Jia Deng. Hico: A benchmark for recognizing human-object interactions in images. In *ICCV*, 2015. 2

[9] Shuo Chen, Pascal Mettes, Tao Hu, and Cees GM Snoek. Interactivity proposals for surveillance videos. In *ICMR*, 2020. 4

[10] Qiongjie Cui, Huaijiang Sun, and Fei Yang. Learning dynamic relationships for 3d human motion prediction. In *CVPR*, 2020. 2

[11] Donglin Di, Xindi Shang, Weinan Zhang, Xun Yang, and Tat-Seng Chua. Multiple hypothesis video relation detection. In *BigMM*, 2019. 2

[12] Matthijs Douze, Jérôme Revaud, Cordelia Schmid, and Hervé Jégou. Stable hyper-pooling and query expansion for event detection. In *ICCV*, 2013. 7

[13] Adrien Gaidon, Zaid Harchaoui, and Cordelia Schmid. Temporal localization of actions with actoms. *PAMI*, 2013. 3

[14] Kirill Gavrilyuk, Ryan Sanford, Mehrsan Javan, and Cees GM Snoek. Actor-transformers for group activity recognition. In *CVPR*, 2020. 6

[15] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *CVPR*, 2019. 2, 6

[16] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. In *CVPR*, 2017. 2, 4, 6

[17] Colin Graber and Alexander G. Schwing. Dynamic neural relational inference. In *CVPR*, 2020. 2

[18] Abhinav Gupta, Aniruddha Kembhavi, and Larry S Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *PAMI*, 2009. 2

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[20] Zhiwei Hu, Guang Feng, Jiayu Sun, Lihe Zhang, and Huchuan Lu. Bi-directional relationship inferring network for referring image segmentation. In *CVPR*, 2020. 2

[21] Sho Inayoshi, Keita Otani, Antonio Tejero-de Pablos, and Tatsuya Harada. Bounding-box channels for visual relationship detection. In *ECCV*, 2020. 2

[22] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, 2010. 2

[23] Mayank Juneja, Andrea Vedaldi, C.V. Jawahar, and Andrew Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013. 3

[24] Adam Kortylewski, Qing Liu, Angtian Wang, Yihong Sun, and Alan Yuille. Compositional convolutional neural networks: A robust and interpretable model for object recognition under occlusion. *IJCV*, 2020. 3

[25] Anna Kukleva, Makarand Tapaswi, and Ivan Laptev. Learning interactions and relationships between movie characters. In *CVPR*, 2020. 2

[26] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. Transferable interactiveness knowledge for human-object interaction detection. In *CVPR*, 2019. 2

[27] Yue Liao, Si Liu, Fei Wang, Yanjie Chen, Chen Qian, and Jiashi Feng. Ppdm: Parallel point detection and matching for real-time human-object interaction detection. In *CVPR*, 2020. 2

[28] Rongcheng Lin, Jing Xiao, and Jianping Fan. Nextvlad: An efficient neural network to aggregate frame-level features for large-scale video classification. In *ECCV*, 2018. 2

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 5

[30] Chenchen Liu, Yang Jin, Kehan Xu, Guoqiang Gong, and Yadong Mu. Beyond short-term snippet: Video relation detection with spatio-temporal global context. In *CVPR*, 2020. 1, 2, 5, 7, 8

[31] Li Mi and Zhenzhong Chen. Hierarchical graph attention network for visual relationship detection. In *CVPR*, 2020. 2

[32] Antoine Miech, Ivan Laptev, and Josef Sivic. Learnable pooling with context gating for video classification. In *CVPRW*, 2017. 2, 6

[33] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *ICLRW*, 2013. 5

[34] Xufeng Qian, Yueting Zhuang, Yimeng Li, Shaoning Xiao, Shiliang Pu, and Jun Xiao. Video relation detection with spatio-temporal graph. In *MM*, 2019. 1, 2, 5, 6, 7

[35] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 5

[36] Jos BTM Roerdink and Arnold Meijster. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta Informaticae*, 2000. 4

[37] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy,

Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet large scale visual recognition challenge. *IJCV*, 2015. 5

[38] Xindi Shang, Donglin Di, Junbin Xiao, Yu Cao, Xun Yang, and Tat-Seng Chua. Annotating objects and relations in user-generated videos. In *ICMR*, 2019. 1, 2, 5

[39] Xindi Shang, Tongwei Ren, Jingfan Guo, Hanwang Zhang, and Tat-Seng Chua. Video visual relation detection. In *MM*, 2017. 1, 2, 5, 7

[40] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003. 2

[41] Cees GM Snoek and Marcel Worring. *Concept-based video retrieval*. Now Publishers Inc, 2009. 1

[42] Zixuan Su, Xindi Shang, Jingjing Chen, Yu-Gang Jiang, Zhiyong Qiu, and Tat-Seng Chua. Video relation detection via multiple hypothesis association. In *MM*, 2020. 1, 2, 5, 6, 7

[43] Xu Sun, Tongwei Ren, Yuan Zi, and Gangshan Wu. Video visual relation detection via multi-modal feature fusion. In *MM*, 2019. 2, 5, 7

[44] Sai Praneeth Reddy Sunkesula, Rishabh Dabral, and Ganesh Ramakrishnan. Lighten: Learning interactions with graph and hierarchical temporal networks for hoi in videos. In *MM*, 2020. 2

[45] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. YFCC100M: The new data in multimedia research. *Comm. of the ACM*, 2016. 5

[46] Yao-Hung Hubert Tsai, Santosh Divvala, Louis-Philippe Morency, Ruslan Salakhutdinov, and Ali Farhadi. Video relationship reasoning using gated spatio-temporal energy graph. In *CVPR*, 2019. 2, 5, 7

[47] Jan C. van Gemert, Cor J. Veenman, Arnold W. M. Smeulders, and Jan-Mark Geusebroek. Visual word ambiguity. *PAMI*, 2010. 2

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 4, 6

[49] Bo Wan, Desen Zhou, Yongfei Liu, Rongjie Li, and Xuming He. Pose-aware multi-level feature network for human object interaction detection. In *ICCV*, 2019. 2

[50] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 2

[51] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using interaction points. In *CVPR*, 2020. 2

[52] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, 2017. 4, 5

[53] Wentao Xie, Guanghui Ren, and Si Liu. Video relation detection with trajectory-aware multi-modal features. In *MM*, 2020. 1, 2, 5, 6, 7, 8

[54] Bingjie Xu, Yongkang Wong, Junnan Li, Qi Zhao, and Mohan S Kankanhalli. Learning to detect human-object interactions with knowledge. In *CVPR*, 2019. 2

[55] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *ICML*, 2015. 1

[56] Bangpeng Yao and Li Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, 2010. 2

[57] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *CVPR*, 2015. 2, 4

[58] Alan L Yuille. Towards a theory of compositional learning and encoding of objects. In *ICCVW*, 2011. 3

[59] Sipeng Zheng, Xiangyu Chen, Shizhe Chen, and Qin Jin. Relation understanding in videos. In *MM*, 2019. 2