Exploiting Attention for Visual Relationship Detection

Tongxin Hu[1(✉)], Wentong Liao[1], Michael Ying Yang[2], and Bodo Rosenhahn[1]

[1] Leibniz University Hannover, Hanover, Germany
hutong@tnt.uni-hannover.de
[2] University of Twente, Enschede, Netherlands

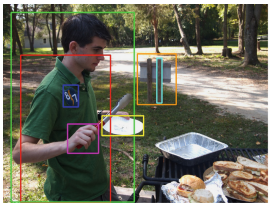
Abstract. Visual relationship detection targets on predicting categories of predicates and object pairs, and also locating the object pairs. Recognizing the relationships between individual objects is important for describing visual scenes in static images. In this paper, we propose a novel end-to-end framework on the visual relationship detection task. First, we design a spatial attention model for specializing *predicate* features. Compared to a normal ROI-pooling layer, this structure significantly improves Predicate Classification performance. Second, for extracting relative spatial configuration, we propose to map simple geometric representations to a high dimension, which boosts relationship detection accuracy. Third, we implement a feature embedding model with a bi-directional RNN which considers *subject*, *predicate* and *object* as a time sequence. We evaluate our method on three tasks. The experiments demonstrate that our method achieves competitive results compared to state-of-the-art methods.

1 Introduction

In recent years, deep learning technology has achieved great success in computer vision tasks, such as object detection techniques [13,26,27], pose estimation [32], tracking [10], AI games [1,29]. However, visual scene understanding remains open challenging tasks. Particularly, recognizing the relationships between objects is important for describing visual scenes in static images. It provides rich information for other visual tasks, such as Visual Turing Test [9]. Reasoning about the pair-wise interactions between objects is a visual-language task which builds the connection between visual images and human natural languages. A visual relationship is generally defined as two interacting objects combined together via a predicate, as illustrated in Fig. 1. The interacting objects are divided into *subject* and *object*. Following the definition in [20], we represent the visual relationship as a triplet <*subject, predicate, object*>.

Traditional methods [6,28] consider this problem as a pure classification task which treats the combination of <*subject, predicate, object*> as a single category. Due to a large number of different possible combinations, such method requires a huge amount of training samples. It is difficult to collect enough samples of phrase combinations for training a reliable model, especially the unusual

G. A. Fink et al. (Eds.): DAGM GCPR 2019, LNCS 11824, pp. 331–344, 2019.
https://doi.org/10.1007/978-3-030-33676-9_23



(a) object detection

< logo, on, shirt >
< man, holding, plate>
< man, has, hand>
< sign, with, pole>
...

(b) visual relationship detection

Fig. 1. The ground truth of visual relationship detection for a given image. With the localized objects, relationships of pair-wise objects are represented.

phrases which might appear rarely in images (*i.e.* long-tail problem). For example, the phrase <*man, ride, horse*> is usual but combination <*kid, ride, dog*> is rare in the dataset. Another approach is to separately detect the object and predicate classes, which treats predicates as individual categories [4,20,37]. Our approach adopts such a strategy to reduce the dimensional complexity and avoid the long-tail problem caused by unusual phrases. Objects and predicates provide supplementary information to each other [17,18,34]. To leverage the dependencies, many recent works [17,18,34] propose to recognize objects and predicate jointly.

Based on the observation that objects and predicates affect each other, we propose a feature embedding model in our network. In the relationship triplet <*subject, predicate, object*>, as shown in Fig. 1, each element has semantic correlations with the next one. For each relationship triplet, we consider that the three elements are in a time sequence. The feature embedding process encodes the semantic correlations among three branches. Since the recurrent neural network is mostly adopted to deal with time sequence input, we apply a bi-directional RNN to compute the feature embedding for three branches. By observing the previous approaches, we find two problems: (1) Current strategies group the detected objects or object proposals to object pairs and use the union bounding boxes to extract union features, which are leveraged as the fundamental feature expressions of predicates [4,17,18,20,34]. Since union features are extracted using the union boxes of object pairs, background information is also included. The background information may distract the model's attention from the interaction between two objects, since the background is normally very complex and contains noise. However, the background also provides local contextual information which is useful for understanding interactions. Motivated by this observation, we propose a spatial attention model for specializing predicate features. (2) Spatial information, including relative locations and relative scales, is important for understanding the interactions between objects. However, using simple geometric representations [22,37,40] is not so effective. To solve this problem, we propose a new spatial feature extraction structure which maps the simple geometric representation to a high dimension.

Contributions. In this work, we propose a novel end-to-end model to incorporate attention and semantic correlations for visual relationship detection. The novelties of our work are: (1) We design a spatial attention model which constrains the network focusing more on the most important regions. (2) We introduce a new spatial feature extraction model which significantly improves the detection performance. (3) We implement a feature embedding model which encodes the semantic correlations among *subject*, *predicate* and *object* branches. Experimental results display that our method achieves competitive performance, compared to state-of-the-art methods.

2 Related Work

Over the decade, a number of researches [4,17,18,35,37–39] investigate the visual relationship detection task. In earlier days, efforts focused on learning specific relationships, *e.g.* spatial relations [3,15], physical support relations [12,36] or actions [8,25,33]. Many previous works proposed to use the visual relationship as a complementary tool for other visual tasks, such as image retrieval [21,24], image captioning [2,7] and scene understanding [11,41]. Fundamentally different from these previous works, our approach targets on generic visual relationship detection. Our method extends the variety of relationships. It can not only recognize positional relations (*"above"*) and verbs (*"walk on"*) but also prepositions (*"with"*) and functional relations (*"of"*).

Contemporary works pay more attention to recognize more general relationships. In [5,6,28], the visual task is considered as a pure classification task through recognizing visual phrases, which are the alliance of object and predicate categories. Such methods face difficulties because of the large combination space and long-tail problem. Another strategy is to implement object and predicate classifiers separately [19,23,39,40]. Lu *et al.* leverage semantic word embeddings (*i.e.* language priors) for recognizing predicates in [20]. Dai *et al.* [4] introduce Deep Relational Network for exploring statistical relations between object and predicate categories. Yu *et al.* [37] leverage both internal and external linguistic knowledge to regularize training process. In [17], Li *et al.* converge several subgraphs whose feature information was exchanged with object features. In [35], Yang *et al.* exploit the contextual information between objects and predicates through Graph Convolutional Network. Zellers *et al.* [38] analyze repeated substructures in the dataset and design to let the model learn from scene graph priors.

The most relevant works are [18,34], which propose to jointly detect objects and pair-wise relationships. In [34], messages are passed iteratively between object and predicate branches through the construction of the scene graph. In [18], Li *et al.* additionally introduce new convolutional features (*i.e.* region captions), features of three semantic branches are jointly refined. Different from their methods, we propose a feature embedding model which encodes the time dependencies and semantic correlations between objects and relationships. We also design a spatial attention model which constrains a higher attention for predicates. Experiments show that our proposed method performs better.

3 Framework

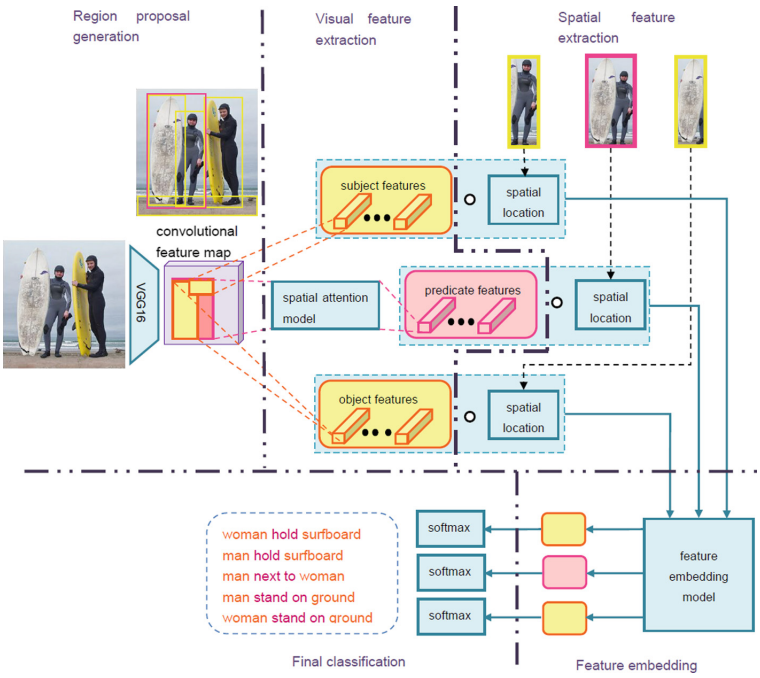


Fig. 2. An overview of our proposed framework.

An overview of our proposed framework is illustrated in Fig. 2. It contains object and predicate branches, where the object branch is divided into *subject* and *object* branch. Our overall network consists of five parts: (1) Object proposal generation. It aims to generate Region of Interests for objects and predicates. (2) Visual feature extraction. It targets on obtaining feature vectors based on obtained convolutional feature maps. (3) Spatial configurations. In addition to visual feature information, we also extract relative spatial information. (4) Feature embedding. Feature vectors of parallel branches are fused to generate an embedded feature vector, which indicates the semantic correlations of *subject*, *predicate* and *object*. (5) Final classification. Categories of *subject*, *predicate* and *object* are predicted, which are the final outputs of our network. In the following, we describe the involved parts in more detail.

3.1 Object Proposal Generation

The input of our entire network is an original image without any preprocessing. The foundation of visual relationship detection is object recognition. Therefore, we remove the last three fully connected layers and the last max pooling layer of

VGG-16 [30], and use 13 convolutional layers to generate convolutional feature map $X \in \mathbb{R}^{W \times H \times K}$, where K is the channel size. Then we apply the Region Proposal Network [27] to generate object ROIs (Region of Interests) [17,18]. We randomly group the obtained object proposals into object pairs. If we retain N object proposals from RPN, in this step we group them into $N \times (N-1)$ object pairs, which are considered as *predicate* proposals in the following detection procedure. We split the object proposals into two subsets, i.e. *subject* and *object* proposals. These two subsets share the same layers and the same parameters.

3.2 Visual Feature Extraction

For different branches in our network, we require their corresponding features. For *subject* and *object* branches, we feed the shared convolutional feature map and the generated object proposals to a ROI-pooling layer. We acquire the *subject* and *object* visual feature vectors of size $512 \times 7 \times 7$ and flatten them. Through two 512-dimensional FC layers, we obtain 512-dimensional *subject* and *object* visual feature vectors.

A different feature specialization structure is implemented for the *predicate* branch. The grouped object pairs, i.e. *predicate* ROIs, contain background information. The most important parts of each *predicate* ROI are the object parts which form the specific *predicate* ROI. On the one hand, the background information may distract the network's attention on *subject* and *object*; on the other hand, it may be the supplemental information for *predicate* branch. So we wish to strengthen the information of object parts and also weaken the background information. In [16], Laskar *et al.* propose a model which combines background information and object proposals' features in a single feature representation for image retrieval task. Motivated by this observation, we propose a spatial attention model for *predicate* branch in our visual relationship detection task.

Spatial Attention Mechanism. We develop the spatial attention model for specializing *predicate* features. The grouped object pair, i.e. the *predicate* ROI $\mathcal{R}_{pr}$ is mapped to the convolutional feature map $X \in \mathbb{R}^{W \times H \times K}$. The mapped *predicate* ROI is represented as $\mathcal{R}'_{pr} \in \mathbb{R}^{W_{pr} \times H_{pr}}$. We define an attention map $A \in \mathbb{R}^{W_{pr} \times H_{pr}}$. This attention map is computed for all the K channels. The two mapped objects, which are employed to form *predicate* ROI, are denoted as $\mathcal{R}'_s$ and $\mathcal{R}'_o$. For each spatial position p on attention map A:

$$A_p = \begin{cases} 1, & \text{if } p \in \mathcal{R}'_s \cup \mathcal{R}'_o \\ M_p, & \text{if } p \in \mathcal{R}'_{pr} \text{ and } p \notin \mathcal{R}'_s \cup \mathcal{R}'_o \end{cases} \tag{1}$$

$p \in \mathcal{R}'_s \cup \mathcal{R}'_o$ denotes that the location point p lies inside the mapped *subject* ROI $\mathcal{R}'_s$ or *object* ROI $\mathcal{R}'_o$. $p \in \mathcal{R}'_{pr}$ and $p \notin \mathcal{R}'_s \cup \mathcal{R}'_o$ means that p lies in the background region of mapped *predicate* ROI $\mathcal{R}'_{pr}$. The saliency map M is defined as:

$$M_p = \sum_{k=1}^{K} X_{k,p}, \text{for } p \in \mathcal{R}'_{pr} \tag{2}$$

where $X_k \in X, k = 1...K$. We compute the max-normalization to ensure $M_p \in [0, 1]$. For each $X_k \in X$, activation occurs:

$$\tilde{X}_{k,p} = \begin{cases} A_p X_{k,p}, & \text{if } p \in \mathcal{R}'_s \cup \mathcal{R}'_o \\ g(A_p) X_{k,p}, & \text{if } p \in \mathcal{R}'_{pr} \text{ and } p \notin \mathcal{R}'_s \cup \mathcal{R}'_o \end{cases} \tag{3}$$

The applied $g(\cdot)$ function is:

$$g(a) = \lambda_1 + \lambda_2 a^{\phi} \tag{4}$$

Constants $\lambda_1, \lambda_2 \in (0, 1)$ are selected to satisfy the constraint $g(\cdot) < 1$.

Through this method, for each *predicate* ROI, the feature values of object pairs are activated by 1. The background information on *predicate* ROI is activated by values smaller than 1. So the effectiveness of background information for *predicate* branch is weakened.

After the computation with the attention map, max-pooling is implemented for *predicate* ROI as in normal ROI-pooling operation. The following process is the same as for *subject* and *object* branch, we use another two 512-dimensional FC layers for obtaining *predicate* visual feature vector.

3.3 Spatial Configurations

Previous attempts have proven that relative spatial configuration is important to the visual relationship detection task. We implement and compare two approaches for extracting spatial features.

Dual Mask. For each object pair, we crop the *subject* bounding box on the original image and define a binary mask for *subject*, where the pixels inside this bounding box are set to 1 and the others are 0. We perform the same process for the *object* bounding box, too. The two binary masks for *subject* and *object* are down-sampled and stacked to form the dual mask $M_D \in \mathbb{R}^{32 \times 32 \times 2}$. Through three convolutional layers and an FC layer, we obtain the *predicate* spatial feature vector. We concatenate this spatial feature vector with the *predicate* visual feature vector obtained from spatial attention model. The concatenated feature vector passes through a 512-dimensional FC layer and we obtain the new *predicate* feature.

Mapping Geometric Representation to a High Dimension. In another method, we first use 6-dimensional, 8-dimensional, and 6-dimensional vectors to express spatial information for three branches and then map the location information to a high dimension. From the object detector, we obtain *subject* $o_s = [x_s, y_s, w_s, h_s]$, *object* $o_o = [x_o, y_o, w_o, h_o]$ and also *predicate* bounding boxes $o_{pr} = [x_{pr}, y_{pr}, w_{pr}, h_{pr}]$. For *subject*, the geometric representation is:

$$\left[\frac{x_s - x_o}{w_o}, \frac{y_s - y_o}{h_o}, \log \frac{w_s}{w_o}, \log \frac{h_s}{h_o}, x_{s,central}, y_{s,central} \right] \tag{5}$$

where $[x_{s,central}, y_{s,central}]$ is the central point coordinate of the *subject* bounding box. $\frac{x_s - x_o}{w_o}$ and $\frac{y_s - y_o}{h_o}$ encode the normalized translation between *subject* and *object* bounding box. $\log \frac{w_s}{w_o}$ and $\log \frac{h_s}{h_o}$ represent the weight and height ratio of two boxes.

For *object*, the representation is:

$$\left[\frac{x_o - x_s}{w_s}, \frac{y_o - y_s}{h_s}, \log \frac{w_o}{w_s}, \log \frac{h_o}{h_s}, x_{o,central}, y_{o,central}\right] \tag{6}$$

And for *predicate* branch:

$$\left[\frac{x_s - x_{pr}}{w_{pr}}, \frac{y_s - y_{pr}}{h_{pr}}, \log \frac{w_s}{w_{pr}}, \log \frac{h_s}{h_{pr}}, \frac{x_o - x_{pr}}{w_{pr}}, \frac{y_o - y_{pr}}{h_{pr}}, \log \frac{w_o}{w_{pr}}, \log \frac{h_o}{h_{pr}}\right] \tag{7}$$

All three geometric representations are embedded in a high dimension. Sine and cosine calculations with different wavelengths [31] are calculated to compute the embedding:

$$E_{(g,2i)} = \sin\left(\frac{g}{10000^{\frac{2i}{D}}}\right), i = 0, ..., \left(\frac{D}{2} - 1\right), i \in \mathbb{N} \tag{8}$$

$$E_{(g,2i+1)} = \cos\left(\frac{g}{10000^{\frac{2i+1}{D}}}\right), i = 0, ..., \left(\frac{D}{2} - 1\right), i \in \mathbb{N} \tag{9}$$

where E denotes the embedded features in high dimension. g means the current spatial representation. D is the dimension of this mapping model and we select $D = 32$.

After the embedding process, we concatenate the spatial feature vectors respectively with the *subject*, *predicate*, and *object* visual feature vectors. We use one 512-dimensional FC layer for *subject* and *object* branches, and another 512-dimensional FC layer for *predicate* branch, to obtain new feature vectors for these three branches.

3.4 Feature Embedding

Information for objects and relationships is correlated. Through the iterative message processing among different branches, prediction performances for three branches will all be improved. Motivated by this thought, we add a feature embedding architecture after our feature extraction structure.

In our feature embedding model, we consider *subject*, *predicate* and *object* features as a time sequence. We apply the bi-directional RNN to compute the feature embedding for three branches. Our bi-RNN network accepts *subject* feature as the input of a sequence at the first time point t_1. *Predicate* feature is the input at the second time point t_2 of the time sequence and *object* feature the input at the third time point t_3. For our visual relationship detection task, the order of input features is really important for the final predictions. We take the hidden states in the forward and backward directions at the last time point, i.e. the third time point, as the embedded feature. It embeds the time dependency

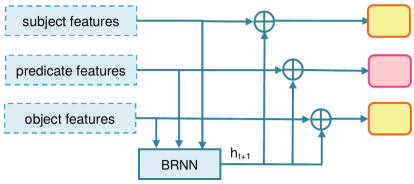


Fig. 3. Feature embedding model with a bi-directional recurrent neural network. We take the hidden states at the last time point as the embedded feature and then concatenate it respectively with *subject*, *predicate* and *object* features obtained from the third step.

of three branches. We concatenate this time-sequence-based embedded feature respectively with the previous *subject*, *predicate* and *object* features. The concatenated features are directly used for the final classification task. The feature embedding procedure using Bi-RNN is illustrated in Fig. 3.

4 Experiments

4.1 Implementation Details

Model Details. We initialize our model by a pre-trained VGG-16 [30] network on ImageNet. Instead of using original 4096, we employ 512 neurons in the fully connected layers, and the weights are initialized using the weights of the pre-trained model. The other parameters are initialized randomly.

Training Details. At first, we train the Region Proposal Network. Then we train the complete network with a mini-batch which contains only one image. After RPN, we use NMS with a threshold of 0.7 and keep at most 2000 object proposals (In testing, we set the threshold to 0.3 and keep at most 300 object proposals). Then we sample 256 object ROIs per image. We sample 512 predicate ROIs, of which 25% are positive. Our loss is the weighted sum of the cross-entropy for objects, the cross-entropy for predicates and the smooth L1 loss for box regression, the ratio is 1:1:0.2. We optimize using SGD with gradient clipping on GTX 1080, with a learning rate of $lr = 0.01$ which will be divided by 10 after every three epochs. The training process stops after 15 epochs and the running time is about three days.

4.2 Dataset and Evaluation

Dataset. We evaluate our proposed method on the cleansed Visual Genome [14] dataset. In previous works, there are different ways of data cleaning and dataset splitting. We use the filtered data from [34] where the most frequent 150 object and 50 predicate categories are chosen. We follow the train/test dataset splits in [34], where the training set contains 57723 and the testing set 26223 images.

Performance Metric. We adopt the same performance metric reported in [20], i.e. the *Top-K recall*, or represented as Rec@K. Rec@K denotes the number of correctly detected relationships in the top K relationship predictions. Following [20], we apply Rec@50 and Rec@100 for evaluation. We use this metric instead of using *mean average precision* mAP because the ground truth annotations applied for evaluation are incomplete.

Task Settings. We evaluate our methods on three tasks: (1) Predicate Classification(PredCls): The inputs are the object ground truth boxes and labels together with the image. We only evaluate the classification performance of *predicate* in this task. (2) Phrase Recognition(PhrRecog): Taken the image as input only, the model predicts *subject*, *predicate*, *object* together, and also the union bounding boxes of object pairs. If the overlap between the predicted union box and the ground truth box is larger than 0.5, the prediction will be considered as correct. (3) Relationship Recognition(RelRecog): The input is an image only. This task targets on localizing object pairs and predicting categories of *predicate* together with object pairs. Both two bounding boxes are required to have an overlap larger than 0.5 with ground truth boxes, for correct recognition.

4.3 Ablation Study

In our network, we propose a spatial attention model, a spatial feature extraction structure which embeds simple geometric representations in a higher dimension, and a feature embedding model. To evaluate how these parts influence the predictive performance of our final model, we perform ablation studies. The left columns of Table 1 display whether the spatial attention model (SA), the spatial feature extraction network (G or DM) and the feature embedding model (bi-RNN) is used or not.

In Table 1, we find that with the spatial attention model, the prediction performances on three tasks are all improved, especially on Predicate Classification. With SA, our network focuses more on the most important parts for *predicate* branch and it is not distracted by the other regions. This proves the effectiveness of 'attention'. The relative spatial configuration boosts prediction performances significantly. It provides supplementary information to simple visual feature representations. Comparing the two spatial feature extraction methods, we find that DM is slightly better than G. So we adopt DM to distill relative spatial information in our final network. Adding the feature embedding model to our network further improves the prediction performance. The feature embedding model is implemented to encode the dependencies of different branches. The improvement indicates that the features of different branches affect each other.

4.4 Comparison with Existing Methods

Previous works use different dataset splitting methods. Since we follow the data cleaning and train/test splitting in [34], we compare our results with those computed using the same dataset splitting. The comparison is listed in Table 2. In

Table 1. Ablation studies on our proposed network. We evaluate our method on the cleansed Visual Genome dataset and report the results for three evaluation tasks. All numbers in %. 'SA' represents the spatial attention model. 'G' indicates the spatial feature extraction structure which maps the geometric representation to a high dimension. 'DM' means the dual mask for extracting spatial features. 'bi-RNN' denotes the feature embedding model with the bi-directional recurrent neural network.

SA	G	DM	bi-RNN	PredCls		PhrRecog		RelRecog	
				Rec@50	Rec@100	Rec@50	Rec@100	Rec@50	Rec@100
–	–	–	–	29.0	40.0	7.3	10.1	3.0	4.8
✓	–	–	–	32.1	44.0	8.3	11.6	3.7	6.0
✓	✓	–	–	44.7	57.1	11.8	14.9	7.3	10.9
✓	–	✓	–	45.7	58.3	13.0	16.4	8.2	12.2
–	–	–	✓	47.5	60.5	17.4	20.9	**10.4**	**14.1**
✓	–	✓	✓	**48.9**	**61.5**	**17.5**	**21.1**	10.2	13.8

Table 2. Comparison of our proposed framework with the existing methods. The results of LP [20] are taken from [18]. The results of ISGG [34], MSDN [18], Factorizable Net [17] are taken from [17]. ours* reports the results where the network is first trained with ground truth boxes and then fine-tuned using pre-trained RPN.

Comparison	PredCls		PhrRecog		RelRecog	
	Rec@50	Rec@100	Rec@50	Rec@100	Rec@50	Rec@100
LP [20]	26.6	33.3	10.1	12.6	0.08	0.14
ISGG [34]	–	–	15.9	19.5	8.2	10.9
MSDN [18]	–	–	20.0	24.9	10.7	14.2
Factorizable Net [17]	–	–	22.8	28.6	13.1	16.5
Graph R-CNN [35]	54.2	59.1	–	–	11.4	13.7
Ours	48.9	61.5	17.5	21.1	10.2	13.8
Ours*	**58.2**	**60.7**	**29.4**	**34.5**	**19.4**	**22.7**

our original experiments (*i.e.* ours), we train RPN first and then apply the pre-trained model to train the entire network. From Table 2 we find that our model improves the Predicate Classification performance on Rec@100, and achieves comparative results on the Relationship Recognition, but performs not so well on the Phrase Recognition task. In an additional experiment (*i.e.* ours*), we first train our network using ground truth bounding boxes, and then fine-tune it using the pre-trained RPN. The object detector (*i.e.* RPN) is also fine-tuned. In this way, the performance improves significantly, especially on Phrase Recognition and Relationship Recognition tasks.

4.5 Qualitative Results

We show the qualitative results of our final network in Fig. 4, which displays the correct predictions, and Fig. 5, which illustrates the incorrect ones. The input of our network is an original image. Our network predicts *subject* and *object* bounding boxes and categories of *subject*, *predicate* and *object*. In the actual recognition process, the model attempts to find out all the possible relationship triplets for each image. For illustration in a simpler way, we only display one primal relationship triplet for one image.

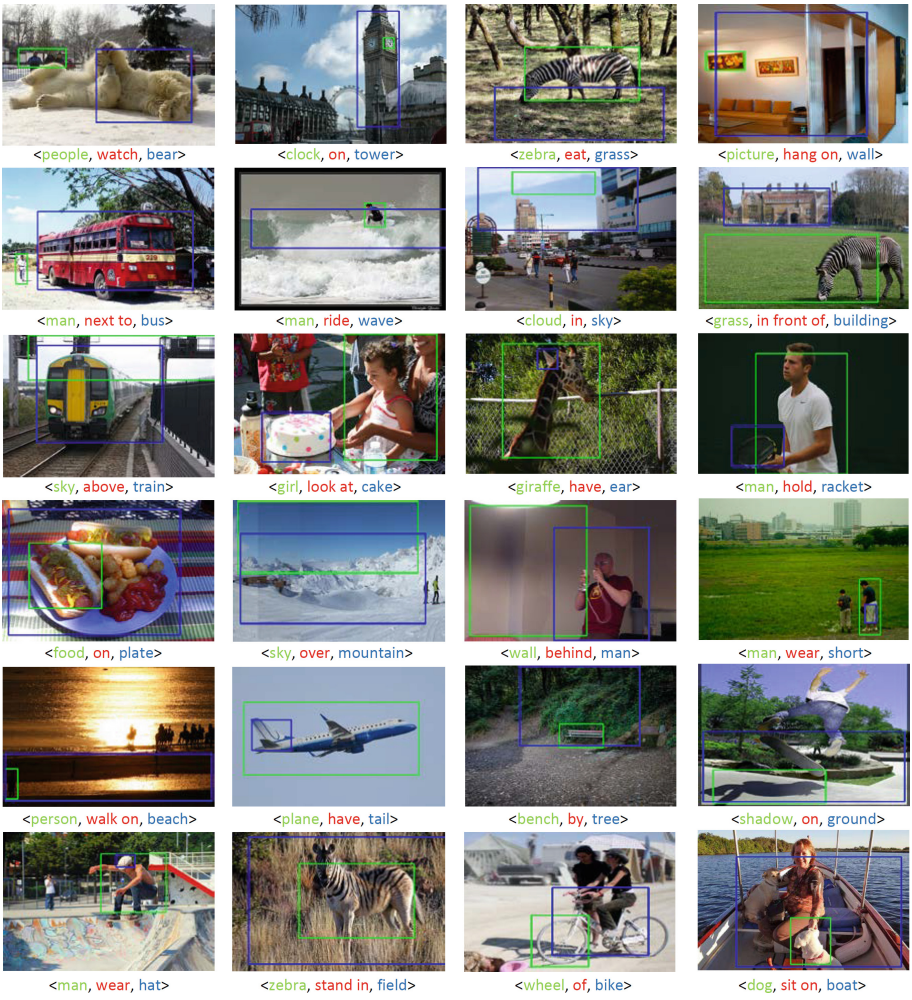


Fig. 4. Qualitative results of our final network. The green and blue bounding boxes correspond to *subjects* and *objects* respectively. (Color figure online)

In Fig. 4, the green and blue boxes represent predicted *subject* and *object* respectively. It displays the effectiveness of our network on the visual relationship detection task. However, there are also some false predictions. In the left sub-image of Fig. 5, *shadow* is falsely detected as *light*, this might be caused by the predicted object bounding box which contains both shadow and light region. In the middle sub-image of Fig. 5, the correct category for *predicate* should be *beside* since there is no physical contact between *man* and *bike*. However, the network might consider that there exists physical contact between them. In the right sub-image of Fig. 5, the two glasses are close to each other and their colors are quite similar, which may lead to the mistake.

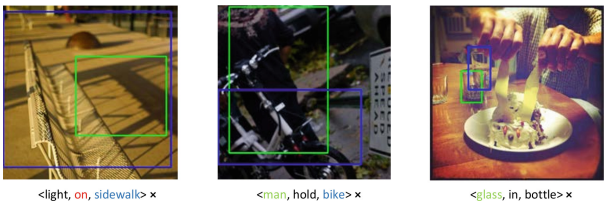


Fig. 5. Incorrect results. In the left image, *shadow* is falsely detected as *light*. In the middle one, the correct label for *predicate* should be *beside*. And for the right one, the ground truth label is *<glass, next to, glass>*.

5 Conclusion

In this work, we propose a new framework for precise visual relationship detection. The proposed framework learns the predicate features between two objects by using a spatial attention module. To capture the contextual information between two objects which are involved in a likely relationship, an RNN module is utilized, which also integrates the spatial information with the visual features together. The framework is trained in an end-to-end fashion. The proposed method outperforms the previous works in the experiments w.r.t. the task of visual relationship detection. There remain some directions for improvement. Instead of using spatial attention module, another way is to exclude background information and only feed the features of two objects to the network. Another interesting direction is to replace the RNN embedding structure with other effective modules.

Acknowledgements. The work is funded by DFG (German Research Foundation) YA 351/2-1 and RO 4804/2-1 within SPP 1894. The authors gratefully acknowledge the support. The authors also acknowledge NVIDIA Corporation for the donated GPUs.

References

1. Awiszus, M., Rosenhahn, B.: Markov chain neural networks. In: CVPR Workshops, pp. 2180–2187 (2018)

2. Berg, A.C., et al.: Understanding and predicting importance in images. In: CVPR, pp. 3562–3569. IEEE (2012)
3. Choi, W., Chao, Y.W., Pantofaru, C., Savarese, S.: Understanding indoor scenes using 3D geometric phrases. In: CVPR, pp. 33–40 (2013)
4. Dai, B., Zhang, Y., Lin, D.: Detecting visual relationships with deep relational networks. In: CVPR, pp. 3076–3086 (2017)
5. Das, P., Xu, C., Doell, R.F., Corso, J.J.: A thousand frames in just a few words: lingual description of videos through latent topics and sparse object stitching. In: CVPR, pp. 2634–2641 (2013)
6. Divvala, S.K., Farhadi, A., Guestrin, C.: Learning everything about anything: Webly-supervised visual concept learning. In: CVPR, pp. 3270–3277 (2014)
7. Fang, H., et al.: From captions to visual concepts and back. In: CVPR, pp. 1473–1482 (2015)
8. Farhadi, A., et al.: Every picture tells a story: generating sentences from images. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 15–29. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_2
9. Geman, D., Geman, S., Hallonquist, N., Younes, L.: Visual turing test for computer vision systems. Proc. Natl. Acad. Sci. **112**(12), 3618–3623 (2015)
10. Henschel, R., von Marcard, T., Rosenhahn, B.: Simultaneous identification and tracking of multiple people using video and IMUs. In: CVPR Workshops (2019)
11. Izadinia, H., Sadeghi, F., Farhadi, A.: Incorporating scene context and object layout into appearance modeling. In: CVPR, pp. 232–239 (2014)
12. Jia, Z., Gallagher, A., Saxena, A., Chen, T.: 3D-based reasoning with blocks, support, and stability. In: CVPR, pp. 1–8 (2013)
13. Kluger, F., et al.: Region-based cycle-consistent data augmentation for object detection. In: 2018 IEEE International Conference on Big Data (Big Data), pp. 5205–5211. IEEE (2018)
14. Krishna, R., et al.: Visual genome: connecting language and vision using crowdsourced dense image annotations. Int. J. Comput. Vis. **123**(1), 32–73 (2017)
15. Kulkarni, G., et al.: Baby talk: understanding and generating image descriptions. In: CVPR. Citeseer (2011)
16. Laskar, Z., Kannala, J.: Context aware query image representation for particular object retrieval. In: Sharma, P., Bianchi, F.M. (eds.) SCIA 2017, Part II. LNCS, vol. 10270, pp. 88–99. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59129-2_8
17. Li, Y., Ouyang, W., Zhou, B., Shi, J., Zhang, C., Wang, X.: Factorizable net: an efficient subgraph-based framework for scene graph generation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018, Part I. LNCS, vol. 11205, pp. 346–363. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01246-5_21
18. Li, Y., Ouyang, W., Zhou, B., Wang, K., Wang, X.: Scene graph generation from objects, phrases and region captions. In: ICCV, pp. 1261–1270 (2017)
19. Liao, W., Rosenhahn, B., Shuai, L., Ying Yang, M.: Natural language guided visual relationship detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (2019)
20. Lu, C., Krishna, R., Bernstein, M., Fei-Fei, L.: Visual relationship detection with language priors. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016, Part I. LNCS, vol. 9905, pp. 852–869. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_51

21. Mensink, T., Gavves, E., Snoek, C.G.: Costa: co-occurrence statistics for zero-shot classification. In: CVPR, pp. 2441–2448 (2014)
22. Nagaraja, V.K., Morariu, V.I., Davis, L.S.: Modeling context between objects for referring expression understanding. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016, Part IV. LNCS, vol. 9908, pp. 792–807. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_48
23. Peyre, J., Sivic, J., Laptev, I., Schmid, C.: Weakly-supervised learning of visual relations. In: ICCV, pp. 5179–5188 (2017)
24. Prabhu, N., Venkatesh Babu, R.: Attribute-graph: a graph based approach to image ranking. In: ICCV, pp. 1071–1079 (2015)
25. Ramanathan, V., et al.: Learning semantic relationships for better action retrieval in images. In: CVPR, pp. 1100–1109 (2015)
26. Reinders, C., Ackermann, H., Yang, M.Y., Rosenhahn, B.: Object recognition from very few training examples for enhancing bicycle maps. In: 2018 IEEE Intelligent Vehicles Symposium (IV), pp. 1–8. IEEE (2018)
27. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Advances in Neural Information Processing Systems, pp. 91–99 (2015)
28. Sadeghi, M.A., Farhadi, A.: Recognition using visual phrases. In: CVPR 2011, pp. 1745–1752. IEEE (2011)
29. Silver, D., et al.: Mastering the game of go without human knowledge. Nature **550**(7676), 354 (2017)
30. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint: arXiv:1409.1556 (2014)
31. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
32. Wandt, B., Rosenhahn, B.: RepNet: weakly supervised training of an adversarial reprojection network for 3D human pose estimation. In: CVPR, pp. 7782–7791 (2019)
33. Xiong, Y., Zhu, K., Lin, D., Tang, X.: Recognize complex events from static images by fusing deep channels. In: CVPR, pp. 1600–1609 (2015)
34. Xu, D., Zhu, Y., Choy, C.B., Fei-Fei, L.: Scene graph generation by iterative message passing. In: CVPR, pp. 5410–5419 (2017)
35. Yang, J., Lu, J., Lee, S., Batra, D., Parikh, D.: Graph R-CNN for scene graph generation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018, Part I. LNCS, vol. 11205, pp. 690–706. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01246-5_41
36. Yang, M.Y., Liao, W., Ackermann, H., Rosenhahn, B.: On support relations and semantic scene graphs. ISPRS J. Photogramm. Remote Sens. **131**, 15–25 (2017)
37. Yu, R., Li, A., Morariu, V.I., Davis, L.S.: Visual relationship detection with internal and external linguistic knowledge distillation. In: ICCV, pp. 1974–1982 (2017)
38. Zellers, R., Yatskar, M., Thomson, S., Choi, Y.: Neural motifs: scene graph parsing with global context. In: CVPR, pp. 5831–5840 (2018)
39. Zhang, H., Kyaw, Z., Chang, S.F., Chua, T.S.: Visual translation embedding network for visual relation detection. In: CVPR, pp. 5532–5540 (2017)
40. Zhuang, B., Liu, L., Shen, C., Reid, I.: Towards context-aware interaction recognition for visual relationship detection. In: ICCV, pp. 589–598 (2017)
41. Zitnick, C.L., Parikh, D., Vanderwende, L.: Learning the visual interpretation of sentences. In: ICCV, pp. 1681–1688 (2013)