Visual Translation Embedding Network for Visual Relation Detection

Hanwang Zhang[†], Zawlin Kyaw[‡], Shih-Fu Chang[†], Tat-Seng Chua[‡]
[†]Columbia University, [‡]National University of Singapore
{hanwangzhang, kzl.zawlin}@gmail.com; sfchang@ee.columbia.edu; dcscts@nus.edu.sg

Abstract

Visual relations, such as "person ride bike" and "bike next to car", offer a comprehensive scene understanding of an image, and have already shown their great utility in connecting computer vision and natural language. However, due to the challenging combinatorial complexity of modeling subject-predicate-object relation triplets, very little work has been done to localize and predict visual relations. Inspired by the recent advances in relational representation learning of knowledge bases and convolutional object detection networks, we propose a Visual Translation Embedding network (VTransE) for visual relation detection. VTransE places objects in a low-dimensional relation space where a relation can be modeled as a simple vector translation, i.e., subject + predicate ≈ object. We propose a novel feature extraction layer that enables object-relation knowledge transfer in a fully-convolutional fashion that supports training and inference in a single forward/backward pass. To the best of our knowledge, VTransE is the first end-to-end relation detection network. We demonstrate the effectiveness of VTransE over other state-of-the-art methods on two large-scale datasets: Visual Relationship and Visual Genome. Note that even though VTransE is a purely visual model, it is still competitive to the Lu's multi-modal model with language priors [27].

1. Introduction

We are witnessing the impressive development in connecting computer vision and natural language, from the arguably mature visual detection [16, 35] to the burgeoning visual captioning and question answering [2, 4]. However, most existing efforts to the latter vision-language tasks attempt to directly bridge the visual model (*e.g.*, CNN) and the language model (*e.g.*, RNN), but fall short in modeling and understanding the relationships between objects. As a result, poor generalization ability was observed as those models are often optimized on specialized datasets for specific tasks such as image captioning or image QA. [17, 40].

As illustrated in Figure 1, we take a step forward from the lower-level object detection and a step backward from the higher-level language modeling, focusing on the *visual relations* between objects in an image. We refer to a visual relation as a `subject-predicate-object` triplet[1], where the predicate can be verb (`person1-talk-person2`), spatial (`clock-above-person2`), preposition (`car-with-wheel`), and comparative (`person1-taller-person2`) [23, 27]. Visual relations naturally bridge the vision and language by placing objects in a semantic context of what, where, and how objects are connected with each other. For example, if we can detect `clock-above-person2` and `person2-wear-jacket` successfully, the reasoning behind the answer "gray" to the question asked in Figure 1 will be explicitly interpretable using dataset-independent inference, *e.g.*, QA over knowl-

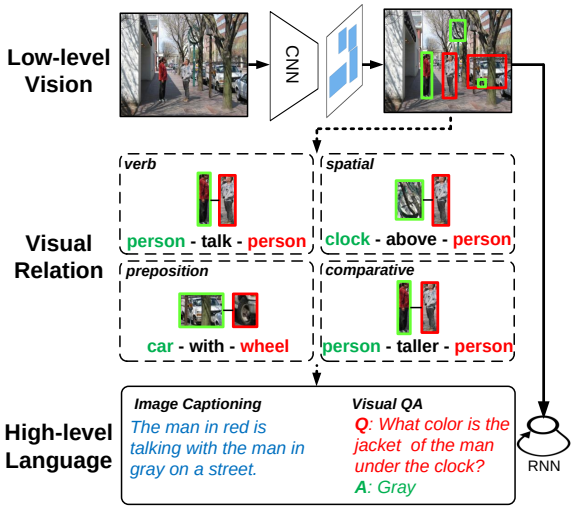

Figure 1. We focus on detecting *visual relations* (dashed boxes in the middle layer) in this paper. Different from the direct connection between low-level vision and high-level language, visual relations offer the direct understanding of object interactions, which provide further semantic information for applications such as image captioning and QA.

[1]When the context is clear, we always refer to object in normal font as a general object and `object` in teletype to the tail object in a relation.

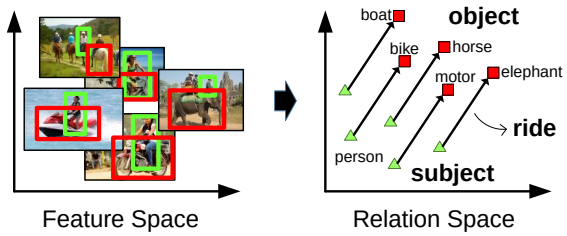

Figure 2. An illustration of translation embedding for learning predicate `ride`. Instead of modeling from a variety of `ride` images, VTransE learns consistent translation vector in the relation space regardless of the diverse appearances of subjects (*e.g.*, `person`) and objects (*e.g.*, `horse`, `bike`, *etc.*) involved in the predicate relation (*e.g.*, `ride`).

edge bases [8], and thus permits better generalization or even zero-shot learning [23, 41].

In this paper, we present a convolutional localization network for visual relation detection dubbed **V**isual **Trans**lation **E**mbedding network (**VTransE**). It detects objects and predicts their relations simultaneously from an image in an end-to-end fashion. We highlight two key novelties that make VTransE effective and distinguishable from other visual relation models [27, 36, 37]:

Translation Embedding. Since relations are compositions of objects and predicates, their distribution is much more long-tailed than objects. For N objects and R predicates, one has to address the fundamental challenge of learning $\mathcal{O}(N^2R)$ relations with few examples [33, 37]. A common solution is to learn separate models for objects and predicates, reducing the complexity to $\mathcal{O}(N + R)$. However, the drastic appearance change of predicates makes the learning even more challenging. For example, `ride` appearance largely varies from `person-ride-bike` to `person-ride-elephant`. To this end, inspired by Translation Embedding (TransE) in representing large-scale knowledge bases [5, 25], we propose to model visual relations by mapping the features of objects and predicates in a low-dimensional space, where the relation triplet can be interpreted as a vector translation, *e.g.*, `person+ride` $\approx$ `bike`. As shown in Figure 2, by avoiding learning the diverse appearances of `subject-ride-object` with large variance, we only need to learn the `ride` translation vector in the relation space, even though the subjects and/or objects can be quite diverse.

Knowledge Transfer in Relation. Cognitive evidences show that the recognition of objects and their interactions is reciprocal [6, 15]. For example, `person` and `bike` detections serve as the context for `ride` prediction, which in turn constrains the articulation of the two objects, and thus benefiting object detection. Inspired by this, we explicitly incorporate knowledge transfer between objects and predicates in VTransE. Specifically, we propose a novel feature extraction layer that extracts three types of object features used in translation embedding: classeme (*i.e.*, class probabilities), locations (*i.e.*, bounding boxes coordinates and scales), and RoI visual features. In particular, we use the bilinear feature interpolation [13, 18] instead of RoI pooling [11, 35] for differentiable coordinates. Thus, the knowledge between object and relation—confidence, location, and scale—can be transfered by a single forward/backward pass in an end-to-end fashion.

We evaluate the proposed VTransE on two recently released relation datasets: Visual Relationship [27] with 5,000 images and 6,672 unique relations, and Visual Genome [23] with 99,658 images and 19,237 unique relations. We show significant performance improvement over several state-of-the-art visual relation models. In particular, our purely visual VTransE can even outperform the multi-modal method with vision and language priors [27] in detection and retrieval, and a bit shy of it in zero-shot learning.

In summary, our contributions are as follows: 1) We propose a visual relation detection model dubbed Visual Translation Embedding network (VTransE), which is a convolutional network that detects objects and relations simultaneously. To the best of our knowledge, this is the first end-to-end relation detection network; 2) We propose a novel visual relation learning model for VTransE that incorporates translation embedding and knowledge transfer; 3) VTransE outperforms several strong baselines on visual relation detection by a large performance gain.

2. Related Work

Our work falls in the recent progress on grounding compositional semantics in an image [23, 32]. It has been shown that high-quality groundings provide more comprehensive scene understanding, which underpins many vision-language tasks such as VQA [1], captioning [21] and complex query retrieval [20]. Visual relation detection not only ground regions with objects, but also describes their interactions. In particular, our VTransE network draws on recent works in relation learning and object detection.

Visual Relation Detection. Different from considering relations as hidden variables [42], we relate to explicit relation models which can be divided into two categories: joint model and separate model. For joint models, a relation triplet is considered as a unique class [3, 9, 33, 37]. However, the long-tailed distribution is an inherent defect for scalability. Therefore, we follow the separate model that learns subject, object, and predicate individually [7, 14, 36, 27]. But, modeling the large visual variance of predicates is challenging. Inspired by TransE that has been successfully used in relation learning in large-scale knowledge base [5, 25], our VTransE extends TransE for modeling visual relations by mapping subjects and objects into a low-dimensional relation space with less variance, and modeling the predicate as a translation vector between the subject and

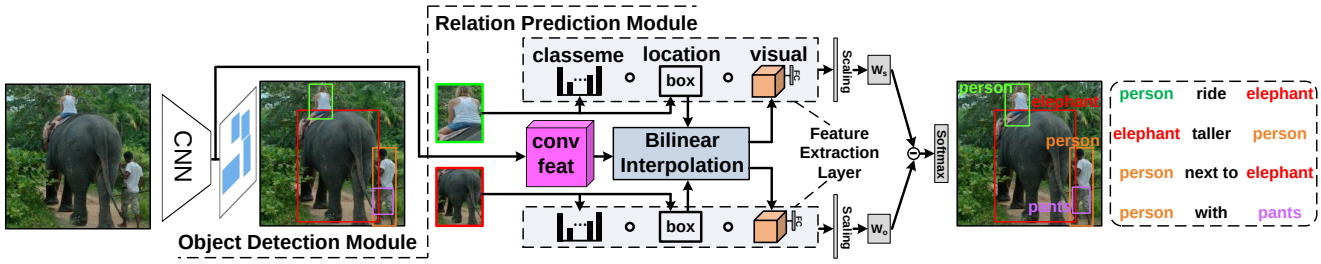

Figure 3. The VTransE network overview. An input image is first through the Object Detection Module, which is a convolutional localization network that outputs a set of detected objects. Then, every pair of objects are fed into the Relation Prediction Module for feature extraction and visual translation embedding. In particular, the visual feature of an object is smoothly extracted from the last convolutional feature map using Bilinear Interpolation. $\circ$ denotes vector concatenation and $\ominus$ denotes element-wise subtraction.

object. Note that there are works [3, 27, 33, 24] that exploit language priors to boost relation detection, but we are only interested in visual models.

Object Detection. VTransE is based on an object detection module composed of a region proposal network (RPN) and a classification layer. In particular, we use Faster-RCNN [35], which is evolved from its predecessors [11, 12] that requires additional input of region proposals. Note that VTransE cannot be simply considered as appending a relation prediction layer to Faster-RCNN. In fact, we propose a novel feature extraction layer that allows knowledge transfer between objects and relations. The layer exploits the bilinear interpolation [13, 18, 19] instead of the non-smooth RoI pooling in Faster-RCNN and thus the reciprocal learning of objects and predicates can be achieved in a single forward/backward pass. Note that VTransE can be married to any object detection network such as the very recent SSD [26] and YOLO [34].

3. Our Approach: VTransE Network

VTransE is an end-to-end architecture that completes object detection and relation prediction simultaneously. As illustrated in Figure 3, it builds upon an object detection module (*e.g.*, Faster-RCNN), and then incorporates the proposed feature extraction layer and the translation embedding for relation prediction.

3.1. Visual Translation Embedding

Given any valid relation, Translation Embedding (TransE) [5] represents `subject-predicate-object` in low-dimensional vectors $\mathbf{s}$, $\mathbf{p}$, and $\mathbf{o}$, respectively, and the relation is represented as a translation in the embedding space: $\mathbf{s} + \mathbf{p} \approx \mathbf{o}$ when the relation holds, and $\mathbf{s} + \mathbf{p} \not\approx \mathbf{o}$ otherwise. TransE offers a simple yet effective linear model for representing the long-tail relations in large knowledge databases [31]. Suppose $\mathbf{x}_s, \mathbf{x}_o \in \mathbb{R}^M$ are the M-dimensional features of `subject` and `object`, respectively. Besides learning a relation translation vector $\mathbf{t}_p \in \mathbb{R}^r$ ($r \ll M$) as in TransE[2], VTransE learns two projection matrices $\mathbf{W}_s, \mathbf{W}_o \in \mathbb{R}^{r \times M}$ from the feature space to the relation space, *i.e.*, $\mathbf{s} = \mathbf{W}_s \mathbf{x}_s$ and $\mathbf{o} = \mathbf{W}_o \mathbf{x}_o$:

$$\mathbf{W}_s \mathbf{x}_s + \mathbf{t}_p \approx \mathbf{W}_o \mathbf{x}_o. \quad (1)$$

Unlike the relations in a knowledge base that are generally facts, *e.g.*, `AlanTuring-bornIn-London`, visual relations are volatile to specific visual examples, *e.g.*, the validity of `car-taller-person` depends on the heights of the specific car and person in an image, resulting in problematic sampling negative triplets if the relation annotation is incomplete. Instead, we propose to use a simple yet efficient softmax for prediction loss that only rewards the deterministically accurate predicates[3], but not the agnostic object compositions of specific examples:

$$\mathcal{L}_{rel} = \sum_{(s,p,o) \in \mathcal{R}} -\log \text{softmax}\left(\mathbf{t}_p^T (\mathbf{W}_o \mathbf{x}_o - \mathbf{W}_s \mathbf{x}_s)\right), \quad (2)$$

where the softmax is computed over p. Although Eq. (2) learns a rotational approximation for the translation model in Eq. (1), we can retain the translational property by proper regularizations such as weight decay [30, 43, 44].

The final score for relation detection is the sum of object detection score and predicate prediction score in Eq. (2): $S_{s,p,o} = S_s + S_p + S_o$, where S_s or S_o is the object detection score and S_p is the relation predicate prediction score.

3.2. Feature Extraction

We propose a Feature Extraction Layer in VTransE to extract $\mathbf{x}_s$ and $\mathbf{x}_o$. There are three types of features that characterize the multiple facets of objects in relations:

Classeme. It is an $(N+1)$-d vector of object classification probabilities (*i.e.*, N classes and 1 background) from the object detection network. Classeme is widely used as semantic attributes in various vision tasks [39]. For example, in relation detection, classeme is a useful prior for rejecting

[2] In experiments, we tested $r \in \{100, 200, ..., 1000\}$ and found that $r = 500$ is a good default.

[3] In fact, predicate is multi-labeled, *e.g.*, both `person-on-bike` and `person-ride-bike` are correct. However, most relations are single-labeled in the datasets, *e.g.*, 58% in VRD [27] and 67% in VG [23].

unlikely relations such as `cat-ride-person`.

Location. It is a 4-d vector (t_x, t_y, t_w, t_h), which is the bounding box parameterization in [12], where (t_x, t_y) specifies a scale-invariant translation and (t_w, t_h) specifies the log-space height/width shift relative to its counterpart `object` or `subject`. Take `subject` as an example:

$$t_x = \frac{x - x'}{w'}, t_y = \frac{y - y'}{h'}, t_w = \log \frac{w}{w'}, t_h = \log \frac{h}{h'} \quad (3)$$

where (x, y, w, h) and (x', y', w', h') are the box coordinates of `subject` and `object`, respectively. Location feature is not only useful for detecting spatial or preposition relation, but also useful for verbs, *e.g.*, `subject` is usually above `object` when the predicate is `ride`.

Visual Feature. It is a D-d vector transformed from a convolutional feature of the shape $X \times Y \times C$. Although it is as the same size as the RoI pooling features used in Faster-RCNN, our features are bilinearly interpolated from the last conv-feature map, so as to achieve end-to-end training that allows knowledge transfer (cf. Section 3.3).

The overall feature $\mathbf{x}_s$ or $\mathbf{x}_o$ is a weighted concatenation of the above three features ($M = N + D + 5$), where the weights are learnable scaling layers since the feature contribution dynamically varies from relation to relation. As shown in Figure 3, the proposed feature extraction layer couples the Object Detection Module and the Relation Prediction Module.

3.3. Architecture Details

A training image for VTransE is labeled with a list of `subject-predicate-object` triplets, where every unique `subject` or `object` is annotated with a bounding box. At testing time, VTransE inputs an image and outputs a set of detected objects and the relation prediction scores for every pair of objects.

Object Detection Network. VTransE network starts from the Faster-RCNN [35] object detection network with the VGG-16 architecture [38]. At training time, we sample a mini-batch cotaining 256 region proposal boxes generated by the RPN of Faster-RCNN, each of which is positive if it has an intersection over union (IoU) of at least 0.7 with some ground truth regions and it is negative if the IoU $<$ 0.3. The positive proposals are fed into the classification layer, where each proposal outputs an $(N + 1)$ class probabilities and N bounding box estimations. Then, we perform non-maximum suppression (NMS) for every class with the IoU > 0.4, resulting in 15.6 detected objects on average, each of which has only one bounding box. The reasons of performing NMS for object detection are two folds: 1) we need a specific object class for each region to match with the relation ground truth, and 2) we need to down-sample the objects for a reasonable number of candidate relations. At test time, we sample 300 proposal regions generated by RPN with IoU > 0.7. After the classification layer, we perform NMS with IoU > 0.6 on the 300 proposals, resulting in 15–20 detections per image on average.

Bilinear Interpolation. By removing the final pooling layer of VGG-16, we use the last convolutional feature map $\mathbf{F}$ of the shape $W' \times H' \times C$ (the pink cube in Figure 3), where $C = 512$ is the number of channels, $W' = \lfloor \frac{W}{16} \rfloor$, and $H' = \lfloor \frac{H}{16} \rfloor$, where W and H are the width and height of the input image. $\mathbf{F}$ encodes the visual appearance of the whole image and is used for extracting visual features for the object detection and relation prediction.

In order to achieve object-relation knowledge transfer, the relation error should be back-propagated to the object detection network and thus refines the objects. However, the widely-used RoI pooling visual feature in Fast/Faster R-CNN is not a smooth function of coordinates since it requires discrete grid split for the proposal region, resulting in zero coordinate gradients back-propagated from the feature extraction layer.

To this end, we replace the RoI pooling layer with bilinear interpolation [18]. It is a smooth function of two inputs: the feature map $\mathbf{F}$ and an object bounding box projected onto $\mathbf{F}$, and the output is a feature $\mathbf{V}$ of the size $X \times Y \times C$ (the orange cube in Figure 3). Each entry value in $\mathbf{V}$ can be efficiently interpolated from $\mathbf{F}$ in a convolutional way:

$$V_{i,j,c} = \sum_{i'=1}^{W'} \sum_{j'=1}^{H'} F_{i',j',c} k(i' - G_{i,j,1}) k(j' - G_{i,j,2}), \quad (4)$$

where $\mathbf{G} \in \mathbb{R}^{X \times Y \times 2}$ records the positions of the $X \times Y$ grid split in the input bounding box and $k(x) = \max(0, 1 - |x|)$ is the bilinear interpolation kernel. Note that the grid position $\mathbf{G}$ matrix is a linear function of the input box. Therefore, the gradients from $\mathbf{V}$ can be back-propagated to the bounding box coordinates.

Optimization. We train the VTransE network end-to-end by stochastic gradient descent with momentum [22]. We follow the "image-centric" training strategy [35], *i.e.*, the mini-batch arises from a single image that contains many object regions and relations. The loss function is a multi-task loss combining the object detection loss $\mathcal{L}_{obj}$ and the relation detection loss $\mathcal{L}_{rel}$ in Eq. (2), allowing reciprocal learning for objects and relations. In particular, we find that a reasonable loss trade-off is $\mathcal{L}_{obj} + 0.4\mathcal{L}_{rel}$. Since object detection and relation prediction have different sample sizes, we normalize $\mathcal{L}_{obj}$ and $\mathcal{L}_{rel}$ by the mini-batch size.

For model initializations, we pre-train Faster-RCNN on the objects in the relation datasets to initialize the object detection network and randomly initialize the VTransE component with Gaussian weights. For end-to-end training, we also replace the RoI pooling layer in the object detection network with bilinear interpolation. For efficiency, we do not fine-tune the VGG-16 CNN. Generally, we need 2 – 3 epochs for the model to converge. For a single image that has been resized to the longer side of 720 pixels, the train-

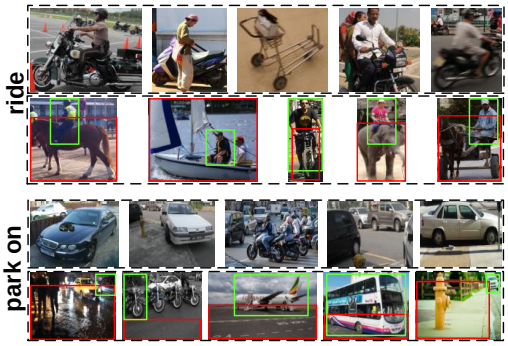

Figure 4. Top 5 confident regions of `subject` and `object` retrieved by `ride` and `park on` models of JointBox (1st row) and VTransE (2nd row with ground-truth bounding boxes) from VRD.

ing runs in 2.0 fps and the testing runs in 6.7 fps on a Titan X GPU using Caffe and Python. Note that we can always plug-in faster object detection networks such as SSD [26] and YOLO [34] for more efficient training and testing.

4. Experiments

We are going to validate the effectiveness of the proposed VTransE network by answering the following questions. **Q1**: Is the idea of embedding relations effective in the visual domain? **Q2**: What are the effects of the features in relation detection and knowledge transfer? **Q3**: How does the overall VTransE network perform compared to the other state-of-the-art visual relation models?

4.1. Datasets and Metrics

To the best of our knowledge, there are only two datasets for visual relation detection at a large scale. We used both:
VRD. It is the Visual Relationships dataset [27]. It contains 5,000 images with 100 object categories and 70 predicates. In total, VRD contains 37,993 relation annotations with 6,672 unique relations and 24.25 predicates per object category. We followed the same train/test split as in [27], *i.e.*, 4,000 training images and 1,000 test images, where 1,877 relationships are only in the test set for zero-shot evaluations.
VG. It is the latest Visual Genome Version 1.2 relation dataset [23]. Unlike VRD that is constructed by computer vision experts, VG is annotated by crowd workers and thus the objects and relations are noisy. Therefore, we contact the authors for an official pruning of them. For example, "young woman" and "lady" are merged to the WordNet hypernym "woman". We filtered out relations with less than 5 samples. In summary, VG contains 99,658 images with 200 object categories and 100 predicates, resulting in 1,174,692 relation annotations with 19,237 unique relations and 57 predicates per object category. We split the data into 73,801 for training and 25,857 for testing.

Following [27], we used Recall@50 (**R@50**) and Recall@100 (**R@100**) as evaluation metrics for detection. R@K computes the fraction of times a true relation is predicted in the top K confident relation predictions in an image. Note that precision and average precision (AP) are not proper metrics as visual relations are labeled incompletely and they will penalize the detection if we do not have that particular ground truth. For the relation retrieval task (cf. Section 4.4), we adopted the Recall rate@5 (**Rr@5**), which computes the fraction of times the correct result was found among the top 5, and Median rank (**Med r**), which is the median rank of the first correctly retrieved image [20]. In fact, for datasets with more complete annotations (*e.g.*, VG), even if the recall is low, the actual precision could be high since the number of ground truth in an image is usually larger than 50/100. Therefore, the retrieval task measured by Rr@5 and Med r provides a complementary evaluation.

Table 1. Predicate prediction performances of the two methods.

Method	JointBox		VTransE	
Dataset	VRD	VG	VRD	VG
R@50	25.78	46.59	**44.76**	**62.63**
R@100	25.78	46.77	**44.76**	**62.87**

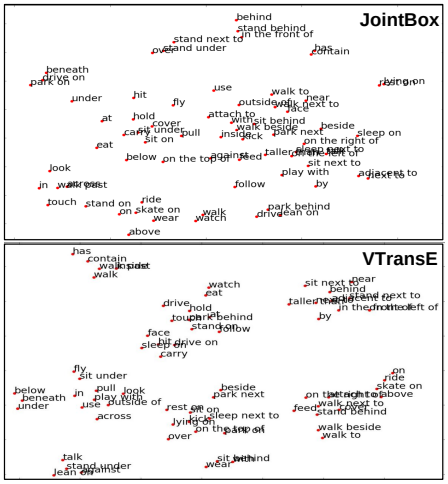

Figure 5. t-SNE visualizations [28] of the 70 predicate model parameters of JointBox and VTransE from VRD. Please zoom in.

4.2. Evaluations of Translation Embedding (Q1)

Setup. Visual relation detection requires both object detection and predicate prediction. To investigate whether VTransE is a good model for relations, we need to isolate it from object detection and perform the task of **Predicate Prediction**: predicting predicates given the ground-truth objects with bounding boxes.

Comparing Methods. We compared 1) **JointBox**, a softmax classifier that classifies the images of the `subject` and `object` joint bounding boxes into predicates, and 2) **VTransE** that classifies the predicate of a pair of `subject` and `object` boxes. For fair comparison, we only use the RoI pooling visual features of boxes for the

two methods. Note that JointBox represents many visual relation models in predicate prediction [9, 27, 33, 37]

Results. From Table 1, we can see that VTransE formulated in Eq. (2) outperforms conventional visual models like JointBox. This is because the predicate model parameters of VTransE—the translation vectors—are able to capture the essential meanings of relations between two objects mapped into a low-dimensional relation space. Figure 4 illustrates that VTransE can predict correct predicates with diversity while JointBox is more likely to bias on certain visual patterns. For example, JointBox limits `park on` in cars, but VTransE can generalize to other subjects like plane and bus. Moreover, by inspecting the semantic affinities between the predicate parameter vectors in Figure 5, we can speculate that JointBox does not actually model relations but the joint object co-occurrence. For example, in JointBox, the reason why `beneath` is close to `drive on` and `park on` is largely due to the co-occurrence of `road-beneath-car` and `car-drive on-road`; however, VTransE is more likely to understand the meaning of `beneath` as its neighbors are `below` and `under`, and it is far from `on` and `above`.

4.3. Evaluations of Features (Q2)

Setup. We evaluated how the features proposed in Section 3.1 affect visual relation detection. We performed **Relation Detection** [27, 37]: the input is an image and the output is a set of relation triplets and localizations of both `subject` and `object` in the image having at least 0.5 overlap with their ground-truth boxes simultaneously.

Comparing Methods. We ablated VTransE into four methods in terms of using different features: 1) **Classeme**, 2) **Location**, 3) **Visual**, and 4) **All** that uses classeme, locations, visual features, and the fusion of the above with a scaling layer (cf. Figure 3), respectively. Note that all the above models are trained end-to-end including the object detection module. To further investigate the feature influence on relations, we categorized the predicates into four categories: verb, spatial, preposition and comparative (cf. Supplementary Material for the detailed category list).

Results. From Figure 6, we can see the details of what features are good at detecting what relations: 1) fusing all the features with a learned scaling layer can achieve the best performance on all types of relations; 2) classeme can generally outperform visual features in various kinds of relations as it characterizes both the high-level visual appearances (*e.g.*, what an object looks like) and composition priors (*e.g.*, `person` is more likely to `ride-bike` than `cat`); 3) for spatial relations, location features are better; however, for preposition relations, all features perform relatively poor. This is because the spatial and visual cues of prepositions are volatile such as `person-with-watch` and `car-with-wheel`.

Table 2. Object detection mAP% before (Faster-RCNN) and after training VTransE from VRD (100 objects) and VG (200 objects). Low mAP is mainly due to the incomplete object annotation.

VRD		VG	
Before	After	Before	After
13.32	**13.98**	6.21	**6.58**

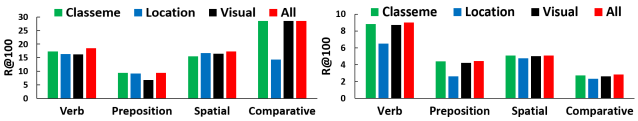

Figure 6. Performances (R@100%) of relation detection of the four relation types using the four ablated VTransE methods from VRD (left) and VG (right).

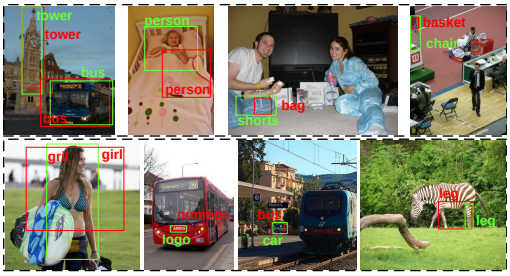

Figure 7. Qualitative object detection examples before (red box and font) and after (green box and font) training VTransE from VRD (top row) and VG (bottom row).

Table 2 shows that the end-to-end training of VTransE can improve the object detection. This is mainly due to that the proposed feature extraction layer allows knowledge transfer so that the errors made by relation prediction can be back-propagated to the front object detection module. In fact, the improvement can be expected since we incorporate additional relation labels besides object labels. As shown in Figure 7, compared to the pre-trained Faster-RCNN module, the object detection module trained by VTransE can generally improve bounding boxes, such as minor refinement or even recovery from drastic dislocation and corrections for wrong detections. This demonstrates that relations place objects in a contextual scene. For example, relation can recover `shorts` from the wrong detection `bag`, even though the correct detection should be `pants`, which is semantically similar to `shorts`. This correction is likely inferred by the relation `person-wear-shorts/pants`.

4.4. Comparison with State-of-The-Arts (Q3)

Setup. As we will introduce later, some joint relation models can only detect a joint bounding box for an entire relation; thus, besides relation detection, we performed **Phrase Detection** [27]: the input is an image and the output is a set of relation triplets and localizations of the entire bounding box for each relation that having at least 0.5 overlap with the ground-truth joint `subject` and `object` box.

For more extensive evaluations, we also performed two additional tasks. 1) **Relation Retrieval**: image search with the query of a relation triplet. We first detect the relation

Table 3. Performances of phrase detection, relation detection, relation retrieval using various methods on both datasets. "–" denotes that the result is not applicable. (cf. Supplementary Material for the incomplete annotation in VRD that causes low retrieval performances.)

Dataset	VRD [27]						VG [23]					
Task	Phrase Det.		Relation Det.		Retrieval		Phrase Det.		Relation Det.		Retrieval	
Metric	R@50	R@100	R@50	R@100	Rr@5	Med r	R@50	R@100	R@50	R@100	Rr@5	Med r
VisualPhrase [37]	0.54	0.63	–	–	3.51	204	3.41	4.27	–	–	11.42	18
DenseCap [19]	0.62	0.77	–	–	4.16	199	3.85	5.01	–	–	12.95	13
Lu's-V [27]	2.24	2.61	1.58	1.85	2.82	211	–	–	–	–	–	–
Lu's-VLK [27]	16.17	17.03	13.86	14.70	**8.75**	137	–	–	–	–	–	–
VTransE	**19.42**	**22.42**	**14.07**	**15.20**	7.89	**41**	**9.46**	**10.45**	**5.52**	**6.04**	**14.65**	**7**
VTransE-2stage	18.45	21.29	13.30	14.64	7.14	41	8.73	10.11	4.97	5.48	12.82	12
Random	0.06	0.11	7.14×10^{-3}	1.43×10^{-2}	2.95	497	0.04	0.07	1.25×10^{-3}	2.50×10^{-3}	3.45	1.28×10^{4}

Figure 8. Qualitative examples of relation detection (4 top-1 detections from 4 predicate types) and retrieval (top-5 images). We compare our VTransE with its best competitors: Lu's-VLK on VRD and DenseCap on VG. Green and red borders denote correct and incorrect results.

query in gallery (*i.e.*, test) images and then score them according to the average detection scores of the query relation. An image with at least one successful query relation detection is considered as a hit. This task is a representation of the compositional semantic retrieval [20]; We selected the top 1,000 frequent relations as queries. 2) **Zero-shot Learning** [27]: individual `subject`, `object`, and `predicate` are seen in both training and test, but some specific triplet compositions are only in the test set. Due to the long-tailed relation distribution, it is a practical setting since it is impossible to collect data for all triplets.

Comparing Methods. We compared the VTransE network to four state-of-the-art visual relation detection models. 1) **VisualPhrase** [37]: a joint relation model that considers every unique relation triplet as an relation class. For fair comparison, we replace the original DPM object detection model [10] with Faster-RCNN [35]; 2) **DenseCap** [19]: it detects sub-image regions and generate their descriptions simultaneously. It is an end-to-end model using bilinear interpolated visual features for region localizations. We replace its LSTM classification layer with softmax for relation prediction. Thus, it can be considered as an joint relation model; 3) **Lu's-V** (V-only in [27]): it is a two-stage separate model that first uses R-CNN [12] for object detection and then adopts a large-margin JointBox model for predicate classification; 4) **Lu's-VLK** (V+L+K in [27]): a two-stage separate model that combines Lu's-V and word2vec language priors [30]. In addition, we compared VTransE to its two-stage training model **VTransE-2stage** that apply Faster-RCNN for object detection and then perform predicate predication using translation embedding as in Q1.

As we have no training source codes of Lu's methods, we cannot apply them in VG and we quoted the results of VRD reported in their paper [27]. Moreover, as the joint relation models such as VisualPhrase and DenseCap can only detect relation triplet as a whole, they are not applicable in zero-shot learning. Therefore, we only report zero-shot results (detection and retrieval) on VRD for the official 1,877 zero-shot relations [27].

Results. From the quantitative results in Table 3 and the qualitative results in Figure 8, we have:
1) Separate relation models like VTransE and Lu's-V out-

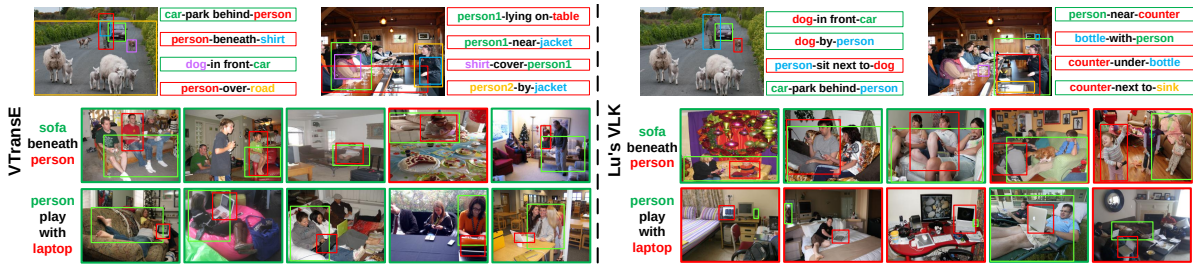

Figure 9. Qualitative examples of zero-shot relation detection (top-4) and retrieval (top-5) using VTransE and Lu's-VLK on VRD. Green and red borders denote correct and incorrect results.

perform joint models like VisualPhrase and DenseCap significantly, especially on VRD. This is because the classification space of joint models for all possible relationships is large (*e.g.*, 6,672 and 19,237 training relations in VRD and VG, respectively), leading to insufficient samples for training infrequent relations.

2) For separate models, better object detection networks, such as Faster-RCNN v.s. R-CNN used in VTrasnE and Lu's, are beneficial for relation detections. As shown in Figure 8, on VRD dataset, Lu's-VLK mistakes `soundbox` as `person` and `plate` as `bowl`. We believe that this is a significant reason why their visual model Lu's-V is considerably worse than VTransE.

3) Even though VTransE is a purely visual model, we can still outperform Lu's-VLK which incorporates language priors, *e.g.*, on VRD measured by R@50 and Med r, we are 20%, 2%, and 230% relatively better in phrase detection, relation detection, and relation retrieval, respectively. First, the classeme feature can serve as a similar role as language priors. Second, location feature is indispensable to relations. Take the `person-wear-tie` relation query as an example in Figure 8, when there are multiple `person` detections in an image, Lu's-VLK usually relates `tie` to the wrong `person`, regardless the fact that the spatial distance is far. Similar examples can be also found in the false detection `shirt-on-cup` of Lu's-VLK.

4) The end-to-end VTransE is better than VTransE-2stage across all the tasks on both datasets. Together with the results in Q2, they demonstrate the effectiveness of reciprocal learning between objects and relations.

From the zero-shot quantitative results in Table 4 and the qualitative results in Figure 9, we have:

1) The performances of ours and the compared methods degrade drastically, *e.g.*, for relation detection, VTransE and Lu's-VLK suffer 88% and 79% performance (R@100) drop, respectively. This is the key **limitation** of VTransE. Perhaps this is because our transformation from feature space to relation space in Eq. (1) is too generic, especially for verbs, and thus fails to capture the relation-specific visual deformations. For example, VTransE cannot discriminate between `person-lying on-table` and `person-sit next to-table`. One remedy is to incorporate

Table 4. Performances of zero-shot phrase detection, relation detection, relation retrieval using various methods on VRD. Note that joint models like VisualPhrase and DenseCap do not apply in zero-shot setting.

Task	Phrase Det.		Relation Det.		Retrieval	
Metric	R@50	R@100	R@50	R@100	Rr@5	Med r
Lu's-V [27]	0.95	1.12	0.67	0.78	0.54	454
Lu's-VLK [27]	**3.36**	**3.75**	**3.13**	**3.52**	1.24	434
VTransE	2.65	3.51	1.71	2.14	**1.42**	**422**
Random	0.02	0.04	7.14×10^{-3}	1.43×10^{-2}	0.45	499

predicate and object models [29], although it will increase the model complexity from $\mathcal{O}(N+R)$ to $\mathcal{O}(NR)$, where N is the number of objects and R is the number of predicates.

2) Both as visual models, our VTransE is significantly better than Lu's-V in zero-shot relation predictions; nevertheless, as a multi-modal model, Lu's-VLK surpasses VTransE by exploiting language priors. But, since visual relations are volatile to specific examples, language priors are not always correct—Lu's-VLK can be misled by frequent language collocations which are invalid in visual examples, *e.g.*, the mismatch of `subject` and `object` in `sofa-beneath-person` and `person-play with-laptop`.

5. Conclusions

We focused on the *visual relation detection* task that is believed to offer a comprehensive scene understanding for connecting computer vision and natural language. Towards this task we introduced the VTransE network for simultaneous object detection and relation prediction. VTransE is an end-to-end and fully-convolutional architecture that consists of an object detection module, a novel differentiable feature extraction layer, and a novel visual translation embedding layer for predicate classification. Moving forward, we are going to 1) model higher-order relations such as `person-throw-ball-to-dog`, 2) tackle the challenge of zero-shot relation learning, and 3) apply VTransE in a VQA system based on relation reasoning.

Acknowledgements

NExT research is supported by the National Research Foundation, Prime Minister's Office, Singapore under its IRC@SG Funding Initiative.

References

[1] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Deep compositional question answering with neural module networks. In *CVPR*, 2016. 2

[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh. Vqa: Visual question answering. In *ICCV*, 2015. 1

[3] Y. Atzmon, J. Berant, V. Kezami, A. Globerson, and G. Chechik. Learning to generalize to new compositions in image understanding. In *EMNLP*, 2016. 2, 3

[4] R. Bernardi, R. Cakici, D. Elliott, A. Erdem, E. Erdem, N. Ikizler-Cinbis, F. Keller, A. Muscat, and B. Plank. Automatic description generation from images: A survey of models, datasets, and evaluation measures. *JAIR*, 2016. 1

[5] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko. Translating embeddings for modeling multi-relational data. In *NIPS*, 2013. 2, 3

[6] L. L. Chao and A. Martin. Representation of manipulable man-made objects in the dorsal stream. *Neuroimage*, 2000. 2

[7] C. Desai, D. Ramanan, and C. C. Fowlkes. Discriminative models for multi-class object layout. *IJCV*, 2011. 2

[8] L. Dong, F. Wei, M. Zhou, and K. Xu. Question answering over freebase with multi-column convolutional neural networks. In *ACL*, 2015. 2

[9] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*, 2010. 2, 6

[10] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 2010. 7

[11] R. Girshick. Fast r-cnn. In *ICCV*, 2015. 2, 3

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 3, 4, 7

[13] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015. 2, 3

[14] A. Gupta and L. S. Davis. Beyond nouns: Exploiting prepositions and comparative adjectives for learning visual classifiers. In *ECCV*, 2008. 2

[15] A. Gupta, A. Kembhavi, and L. S. Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *TPAMI*, 2009. 2

[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. 2016. 1

[17] A. Jabri, A. Joulin, and L. van der Maaten. Revisiting visual question answering baselines. In *ECCV*, 2016. 1

[18] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015. 2, 3, 4

[19] J. Johnson, A. Karpathy, and L. Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *CVPR*, 2016. 3, 7

[20] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei. Image retrieval using scene graphs. In *CVPR*, 2015. 2, 5, 7

[21] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 2

[22] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[23] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 2016. 1, 2, 3, 5, 7

[24] X. Liang, L. Lee, and E. P. Xing. Deep variation-structured reinforcement learning for visual relationship and attribute detection. In *CVPR*, 2017. 3

[25] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, 2015. 2

[26] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed. Ssd: Single shot multibox detector. In *ECCV*, 2016. 3, 5

[27] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *ECCV*, 2016. 1, 2, 3, 5, 6, 7, 8

[28] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *JMLR*, 2008. 5

[29] A. Mallya and S. Lazebnik. Learning models for actions and person-object interactions with transfer to question answering. In *ECCV*, 2016. 8

[30] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013. 3, 7

[31] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 2016. 3

[32] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015. 2

[33] V. Ramanathan, C. Li, J. Deng, W. Han, Z. Li, K. Gu, Y. Song, S. Bengio, C. Rossenberg, and L. Fei-Fei. Learning semantic relationships for better action retrieval in images. In *CVPR*, 2015. 2, 3, 6

[34] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016. 3, 5

[35] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 1, 2, 3, 4, 7

[36] F. Sadeghi, S. K. Divvala, and A. Farhadi. Viske: Visual knowledge extraction and question answering by visual verification of relation phrases. In *CVPR*, 2015. 2

[37] M. A. Sadeghi and A. Farhadi. Recognition using visual phrases. In *CVPR*, 2011. 2, 6, 7

[38] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[39] L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classemes. In *ECCV*, 2010. 3

[40] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *TPAMI*, 2016. 1

[41] P. Wang, Q. Wu, C. Shen, A. v. d. Hengel, and A. Dick. Fvqa: Fact-based visual question answering. *arXiv preprint arXiv:1606.05433*, 2016. 2

[42] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, 2010. 2

[43] H. Zhang, X. Shang, H. Luan, M. Wang, and T.-S. Chua. Learning from collective intelligence: Feature learning using social images and tags. *TOMM*, 2016. 3

[44] H. Zhang, X. Shang, W. Yang, H. Xu, H. Luan, and T.-S. Chua. Online collaborative learning for open-vocabulary visual classifiers. In *CVPR*, 2016. 3