IMPROVING VISUAL RELATION DETECTION USING DEPTH MAPS

Anonymous authors
Paper under double-blind review

ABSTRACT

State of the art visual relation detection methods mostly rely on object information extracted from RGB images such as predicted class probabilities, 2D bounding boxes and feature maps. In this paper, we argue that the 3D positions of objects in space can provide additional valuable information about object relations. This information helps not only to detect spatial relations, such as *standing behind*, but also non-spatial relations, such as *holding*. Since 3D information of a scene is not easily accessible, we propose incorporating a pre-trained RGB-to-Depth model within visual relation detection frameworks. We discuss different feature extraction strategies from depth maps and show their critical role in relation detection. Our experiments confirm that the performance of state-of-the-art visual relation detection approaches can significantly be improved by utilizing depth map information.

Relational learning is an established research area in machine learning. State-of-the-art approaches (Nickel et al., 2016) describe relations as triples of the form *(subject, predicate, object)*, such as *(Man, rides, Bike)*. The triples, which all together form a knowledge graph, are typically extracted from structured data such as the infoboxes in Wikipedia and other sources.

In the last years, the detection of relations from images, i.e., visual relation detection, has also gained a lot of interest. One reason is that understanding relations between entities can play an important role in decision making. For example, detecting whether a man is *on* a bike or *next to* a bike is a crucial challenge in autonomous driving. State-of-the-art works in this area utilize information of objects in the scene such as class labels, bounding boxes and RGB features, to capture pairwise relations using different models. In this paper, we argue that relation detection can additionally benefit from objects' 3D information. This information can help to distinguish between many relations such as *standing behind*, *standing in front* and even improve detection in situations where the objects are nearby such as *standing next to*.

Unfortunately, most available datasets, specifically the ones with relational annotations such as Visual Relation Detection (VRD) (Lu et al., 2016) and Visual Genome (VG) (Krishna et al., 2017), lack this 3D information since acquiring them is a cumbersome task requiring specialized hardware. We propose to solve this issue by synthetically generating the corresponding depth maps of images in these datasets. Depth maps provide the objects' distance from the camera and the availability of large corpora of RGB-D pairs, e.g. NYU-Depth-v2 (Nathan Silberman & Fergus, 2012) dataset, enables us to learn the mapping between any RGB image to its corresponding depth map using a fully convolutional neural network. We can then apply the trained network on images from VRD and VG, converting them into depth maps. We call these dataset extensions *VRD-Depth* and *VG-Depth*[1].

The extracted features from depth maps, together with other object information extracted from the RGB images, are the basis for relation detection in our framework. Depth maps have already been widely employed in other tasks, e.g. image classification and segmentation. In these applications, it is common to simply encode a depth image as a rendered RGB image and extract features using a pre-trained convolutional neural network, such as VGG (Simonyan & Zisserman, 2014). However, this might be sub-optimal, as these are two modalities with different information. Indeed, in our experimental results in Section 3.5 we show that only if the feature extraction network had particularly been trained on depth maps, visual relation detection can be improved.

In summary, our contributions are as follows:

[1]These datasets will be made publicly available upon publication.

Figure 1: An image from the VRD Dataset showing several men sitting next to each other and the corresponding generated depth map. Bright colors indicate a larger distance to the camera. Knowing the depth of, e.g., the individuals, the house and the car, plays an important role in detecting relations such as *next to* and *behind*.

1. We are the first to utilize 3D information in visual relation detection. To compensate for the lack of 3D data, we use an RGB-to-Depth model trained on separate corpora of available pairs in both modalities and apply it to images from VRD and VG.
2. We discuss and empirically investigate different strategies to extract features from depth maps for relation detection.
3. We study the quantitative and qualitative benefits of incorporating depth maps. We show in our empirical evaluations using the VRD and VG datasets, that models using depth maps can outperform competing methods by a margin of up to 3% points, even those using information extracted from external language sources.

In general, in this work we aim to answer the following questions:

1. If we are given only depth maps of unknown objects in a scene, how accurately can we infer the distribution of possible pairwise relations?
2. Is it better if we extract features from the *depth maps*, relevant to *relation detection*, by utilizing convolutional filters that have been pre-trained on *RGB images* for *object detection*?
3. Current visual relation detection frameworks commonly rely on extensive object information such as class labels, bounding boxes, RGB features, contextual information, etc. Does it bring any additional information, if we also include depth representations in these frameworks or would that only bring redundant knowledge about the scene?

1 Related Works

Knowledge Graph Learning In Knowledge Graph learning, the aim is typically to find embeddings or latent representations for entities and predicates, which then can serve to predict the probability of unseen triples. These methods mostly differ in how they model relations. In RESCAL (Nickel et al., 2011) each relation is defined as a transformation in the embedding space of entities, producing a triple probability. TransE (Bordes et al., 2013) employs a similar idea but limits each relation to a translation. In comparison to RESCAL, it has fewer parameters; as a disadvantage, it cannot model symmetric relations. DistMult (Yang et al., 2014) considers each relation as a vector, similar to TransE, but minimizes the trilinear dot product of subject, predicate and object vector. DistMult can also be understood as a form of RESCAL, where the transformation matrix is diagonal. ComplEx (Trouillon et al., 2016) extends DistMult to complex-valued vectors of embeddings. A multilayer perceptron (MLP) architecture (Dong et al., 2014) extends these methods to non-linear transformations and has shown to be competitive to the other discussed approaches on most benchmarks (Nickel et al., 2016; Socher et al., 2013).

Visual Relation Detection Visual relation detection received a huge boost by the availability of large corpora of annotated images such as the Visual Relation Detection (VRD) (Lu et al., 2016) and the VG (Krishna et al., 2017), containing the visual form of entities, and relations. In VRD,

Word2Vec representations of the subject, object, and the predicate were used to train a model jointly with the corresponding image section describing the predicate. In particular, they consider the joint bounding box of subject and object as the image representation for the predicate. Follow-up work achieved improved performance by incorporating a knowledge graph, constructed from the annotated triples in the training set (Baier et al., 2017). In general, the distribution of the predicate bounding box in these works is much more long-tailed than of entities alone. Therefore, separating the models for objects and predicates, as employed in VTransE (Zhang et al., 2017) reduces the complexity of training such a model. VTransE is a generalization of TransE to visual relation detection in which the last convolutional layer of the image detector, together with the location of entities and their class labels, is taken as the input vector to the TransE algorithm. More recently, Yu et al. (Yu et al., 2017) proposed a teacher-student model to distill external language knowledge to improve visual relation detection. Iterative Message Passing (Xu et al., 2017), Neural Motifs (Zellers et al., 2018) and Graph R-CNN (Yang et al., 2018) incorporate context within each prediction using RNNs and graph convolutions respectively.

Depth Maps While several works have leveraged depth maps to improve object detection (Bo et al., 2013; Eitel et al., 2015; Gupta et al., 2014), to the best of our knowledge this is the first time that depth maps are used in the relation detection task.

2 FRAMEWORK

In this section we introduce the general framework employed for this study. Let $\mathcal{E} = \{e_1, e_2, ..., e_n\}$ be the set of all entities, including subjects(s) and objects(o), and $\mathcal{P} = \{p_1, p_2, ..., p_m\}$ the set of all predicates. Each entity e_i can appear in images within a bounding box $bb_i = (x_i, y_i, w_i, h_i)$, where (x_i, y_i) are the coordinates of the bounding box and (w_i, h_i) are its width and height. In this work we apply the pre-trained and fine-tuned Faster R-CNN (Ren et al., 2015) on each image I to extract a feature map $\mathbf{fmap_I}$, together with object proposals as a set of bounding boxes bb and class probability distributions $\mathbf{c}$. For each RGB image, we generate a depth map $\mathbf{D}$ where the same bounding box areas encompass the entities' distance from the camera. In the next section, we first describe the employed network for synthetic generation of $\mathbf{D}$s and then discuss the extraction of depth features. In the end, we describe the relation detection head where the late fusion of pairwise features and their relational modelling takes place.

2.1 DEPTH MAPS FOR RELATION DETECTION

2.1.1 GENERATION

We incorporate an RGB-to-Depth model within our visual relation detection framework. As shown in Figure 2, this is a fully convolutional neural network (CNN) that takes an RGB image as input and generates its predicted depth map. This model can be pre-trained on any datasets containing pairs of RGB and depth maps regardless of having object or predicate annotations. This enables us to work with currently available visual relation detection datasets without requiring to collect additional data, and also mitigates the need for specialized hardware in real-world applications. The architectural details are explained in Section 3.

2.1.2 FEATURE EXTRACTION

Depth maps have been employed in tasks such as *object detection* and *segmentation* (Eitel et al., 2015; Hazirbas et al., 2016). In those works, it is common to simply render a depth map as an RGB image and extract depth features using a CNN pre-trained for RGB images. There, it has been argued that the edges in depth maps might yield better object contours than the edges in cluttered RGB images and that one may combine edges from both RGB and depth to obtain more information (Hazirbas et al., 2016). Therefore, they aimed to get similar, complementary features from both modalities.

However, the practice of employing a model pre-trained on a particular source modality, e.g. RGB, and applying it on a different target modality, e.g. depth map, is sub-optimal in many applications[2].

[2]One should also keep in mind that even fine-tuning some layers of a network does not change the very early convolutional filters.

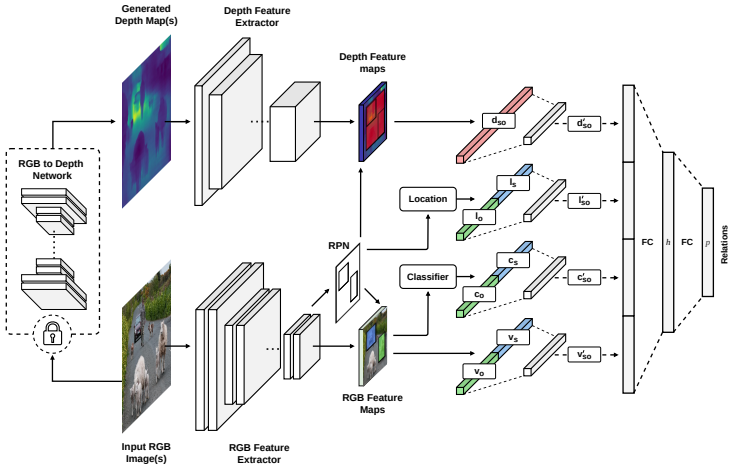

Figure 2: We propose utilizing 3D information in visual relation detection by synthetically generating depth maps using an RGB-to-Depth model incorporated within relation detection frameworks. On the left side, we see the RGB image and its depth map. We use CNNs to extract feature maps from both modalities and create pairwise feature vectors $\mathbf{d'_{so}}$ (from depth feature maps), $\mathbf{l'_{so}}$ (from bounding boxes), $\mathbf{c'_{so}}$ (from class labels) and $\mathbf{v'_{so}}$ (from the pooled visual features). These vectors are then concatenated and fed into a relation detection layer to infer the predicate.

Therefore, similar to other works on depth-based image segmentation or classification we employ a CNN, to generate a feature map $\mathbf{fmap_D}$ from an input depth map, but unlike those works, we train this network from scratch, using depth maps and specifically for relation detection. In Section 3 we show that empirical investigations of this effect support our proposal. The object-level features are then pooled from the feature map which will be described in the next section.

2.2 Relation Model

In the previous section, we described methods for the extraction of individual object features. Here, we outline the model that infers relations using pairwise combination of these features. For each pair of detected objects within an image, we create a scale-invariant location feature $\mathbf{l_s} = (t_x, t_y, t_w, t_h)$ with: $t_x = (x_s - x_o)/w_o, t_y = (y_s - y_o)/h_o, t_w = \log(w_s/w_o), t_h = \log(h_s/h_o)$ and similarly $\mathbf{l_o}$. We then pool the corresponding features $\mathbf{v_s}$ and $\mathbf{v_o}$ from $\mathbf{fmap_I}$ and create a pairwise visual feature vector $\mathbf{v_{so}} = [\mathbf{v_s}; \mathbf{v_o}]$, a pairwise class vector $\mathbf{c_{so}} = [\mathbf{c_s}; \mathbf{c_o}]$, and a pairwise depth feature vector $\mathbf{d_{so}}$ extracted from the union of bb_s and bb_o within $\mathbf{fmap_D}$. Each of these vectors are then fed into separate layers, producing $\mathbf{v'_{so}}$, $\mathbf{l'_{so}}$, $\mathbf{c'_{so}}$ and $\mathbf{d'_{so}}$ before being concatenated altogether and fed to the relation head which projects them to the relation space such that:

$$\mathbf{e_p} = f(\boldsymbol{W}[\mathbf{d'_{so}}; \mathbf{l'_{so}}; \mathbf{c'_{so}}; \mathbf{v'_{so}}]). \tag{1}$$

Here, $\boldsymbol{W}$ describes a linear transformation and f is a non-linear function. We realize this as fully connected layers in a neural network with ReLU activations and dropout. $\mathbf{e_p}$ is the embedding vector for predicate which will be learned jointly with other parameters. This simple relation prediction model is a generalization of the model used by (Dong et al., 2014) to construct knowledge graphs. In that paper, latent features for the predicates are also part of the network input, whereas here we use a separate output for each predicate which is easier to train regarding negative sampling, and also has fewer parameters. As shown in earlier works, using more sophisticated models for context propagation between objects with RNNs or graph convolutions, can further improve the prediction accuracy. However, the aim here is to study the effect of including depth maps as additional object representations in visual relation detection and as will be shown later, even with this simple model, they can be a more effective supplement than e.g. propagated context[3]. Clearly, those other

[3]We carried on Visual Genome experiments in an isolated framework, where all pre-trained weights and hyper-parameters, except for the depth map channel, were kept identical as the ones employed by Neural Motifs (Zellers et al., 2018).

models can also further enrich their understanding of object relations, by employing these additional representations.

To learn the parameters, we consider each relation (s, p, o) with an associated Bernoulli variable y_{spo} that takes 1 if the triple is observed and 0 otherwise, following a locally closed world assumption (Nickel et al., 2016). Given the set of observed triples T, the loss function is the categorical cross entropy between the one-hot targets and the distribution obtained by softmax over the network's output defined as:

$$\mathcal{L} = \sum_{(s,p,o)\in T} -\log \frac{\exp\left(\mathbf{e}_p^{\mathrm{T}} f(\boldsymbol{W}[\mathbf{v}'_{\mathbf{so}}; \mathbf{l}'_{\mathbf{so}}; \mathbf{c}'_{\mathbf{so}}; \mathbf{d}'_{\mathbf{so}}])\right)}{\sum_{p'\in P} \exp\left(\mathbf{e}_{p'}^{\mathrm{T}} f(\boldsymbol{W}[\mathbf{v}'_{\mathbf{so}}; \mathbf{l}'_{\mathbf{so}}; \mathbf{c}'_{\mathbf{so}}; \mathbf{d}'_{\mathbf{so}}])\right)} \tag{2}$$

3 EVALUATION

3.1 DATASETS

We test our approach on the *Visual Relation Detection (VRD)* (Lu et al., 2016) and *Visual Genome* (Krishna et al., 2017) datasets. The VRD dataset contains 100 objects categories and 70 predicates. It has 37,993 triples from which 6,672 are unique. Similar to other works (Baier et al., 2017; Lu et al., 2016; Zhang et al., 2017), we split the data into 4,000 training and 1,000 test images, where 1,877 relationships are in the test set for zero-shot evaluations. The more commonly used subset of VG dataset proposed by (Xu et al., 2017) contains 150 object classes and 50 relations with 75,651 images used for training and (5000 for validation) and 32,422 images for testing.

3.2 ARCHITECTURES

RGB-to-Depth Network: We use the RGB-to-Depth model architecture introduced in (Laina et al., 2016) which is a fully convolutional neural network built on ResNet-50 (He et al., 2015), and trained in an end-to-end fashion on data from NYU Depth Dataset v2 (Nathan Silberman & Fergus, 2012). Training on the outdoor images from Make3D dataset (Saxena et al., 2007) instead, did not show promising results in our framework. This observation is not surprising since Make3D images contain mostly outdoor scenes with too few objects.

RGB Feature Extraction: To extract embeddings and class probabilities of RGB images, we use the VGG-16 architecture (Simonyan & Zisserman, 2014) pre-trained on ImageNet (Russakovsky et al., 2015) and fine-tuned to our data.

Depth Map Feature Extraction: For depth map extraction we use an enhanced version of AlexNet proposed in (Krizhevsky, 2014). We also equipped the network with batch normalization layers referring to as *AlexNet-BN*. We trained this model from scratch following the earlier discussions in Subsection 2.1.2. In the experimental section (Subsection 3.5), we show that this leads to much better solutions than using pre-trained convolutional layers. We trained this network on a pure depth-based, relation detection task using Adam (Kingma & Ba, 2014), with a learning rate of 10^{-4} and batch size of 16 for eight epochs.

Relation Detection Network: Finally, given the features extracted from previous models with the location features described in Subsection 2.1.2, we trained our relation detection model. We connected each feature pair described in the previous section, with a fully connected hidden layer of 64, 200, 4096 and 20 neurons and a dropout rate of 0.1, 0.8, 0.8 and 0.1, with a scaling layer initialized as 1.0, 0.3, 0.5 and 1.0. The penultimate layer contains 4096 neurons with 0.2 dropout. We trained this network by Adam (Kingma & Ba, 2014), with a learning rate of 10^{-4}. We used a batch size of 16 and six epochs of training. All of the layers were initialized with Xavier weights (Glorot & Bengio, 2010).

3.3 METRICS

Some relations might not be annotated in the test set while because of the model's generalization, they might get higher prediction values than the annotated ones. Therefore, we report R@K where it tells us whether the specific predicate in test set ended up as one of the top K listed probable predicates (Baier et al., 2017; Krishna et al., 2017; Lu et al., 2016). It is important to note that relation

Table 1: Predicate prediction and zero-shot accuracies on VRD and VG test-set. When the depth maps are utilized together with RGB features (*Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}, \mathbf{d}'_{\mathbf{so}}$), we gain a large improvement. This improvement is almost as large as *Ours*-$\mathbf{c}'_{\mathbf{so}}$ to *Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}$, demonstrating the importance of depth features in relation detection. One can also see that even using depth maps alone (*Ours*-$\mathbf{d}'_{\mathbf{so}}$) gives a surprisingly significant detection accuracy. Additionally, comparing *Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}$ to *VTransE* and *Neural Motifs* reveals the advantage of our simple model.

	Dataset	VRD				VG		
	Task	Zero-shot Pred.		Predicate Pred.		Predicate Pred.		
	Metric	R@100	R@50	R@100	R@50	R@100	R@50	R@20
models	Lu's-V (Lu et al., 2016)	32.34	23.95	37.20	28.36	-	-	-
	Lu's-VLK (Lu et al., 2016)	50.04	29.77	84.34	70.97	-	-	-
	Yu's-S (Yu et al., 2017)	74.65	54.20	86.97	74.98	49.88	-	-
	Yu's-S+T (Yu et al., 2017)	-	-	94.65	85.64	55.89	-	-
	IMP (Xu et al., 2017)	-	-	-	-	53.00	44.80	-
	Graph R-CNN (Yang et al., 2018)	-	-	-	-	59.10	54.20	-
	Neural Motifs (Zellers et al., 2018)	-	-	-	-	67.10	65.20	58.50
	ComplEx (Baier et al., 2017)	76.05	51.92	93.81	84.17	-	-	-
	VTransE (Zhang et al., 2017)	83.66	67.15	96.22	90.00	62.87	62.63	-
ablations	Ours - d'_{so}	72.80	52.27	86.61	74.17	51.81	49.07	40.87
	Ours - c'_{so}	81.69	62.70	95.39	88.22	66.02	65.03	58.11
	Ours - $c'_{so}, v'_{so}, l'_{so}$	84.94	70.06	96.12	90.34	67.86	66.06	59.34
	Ours - $c'_{so}, v'_{so}, l'_{so}, d'_{so}$	**87.43**	**72.28**	**96.54**	**90.47**	**68.22**	**66.46**	**59.68**

detection is a multilabel setting and that multiple predicates can describe the relation between an entity pair. For example, the correct predicate, given entity pair (Man, Horse) could be at the same time *above* and *riding*. Therefore, given each subject and object pair, some works (Yu et al., 2017) not only consider the predicate with the highest prediction score but also the ranked prediction scores of all possible 70 predicates. In the literature this setting is mainly employed in VRD experiments whereas VG experiments are mostly reported by taking the highest scored prediction given each pair. For a better comparison, we report them similarly.

3.4 Comparing Methods

We compare our results with earlier approaches of *Lu's-V* (Lu et al., 2016), which takes the joint bounding boxes of subject and object as the predicate's image and applies an image classifier on it, *Lu's-VLK* (Lu et al., 2016) that combines the previous approach with Word2Vec (Mikolov et al., 2013) embeddings, and Baier et al. (2017) that constructs a knowledge graph with *ComplEx* (Trouillon et al., 2016) model trained on label distributions. We also compare with *VTransE* (Zhang et al., 2017) that takes visual embeddings and projects them to relation space using TransE. For a fair comparison, we implemented a basic version of the last two methods. The other reported results are from Yu et al. (2017). We consider their student network (*Yu's-S*) which is trained given images from each dataset, and their full model (*Yu's-S+T*) that also employs external language data from Wikipedia. In the context propagating methods we report Neural Motifs (Zellers et al., 2018), Graph R-CNN (Yang et al., 2018) and IMP Xu et al. (2017).

In an ablation study, we report our relation prediction results in several settings: (1) When only class labels from images are available (*Ours*-$\mathbf{c}'_{\mathbf{so}}$). (2) When RGB features are also utilized (*Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}$). (3) When depth maps are also available (*Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}, \mathbf{d}'_{\mathbf{so}}$). To see the influence of including depth maps as additional object information, one can compare the last two settings. (4) *Ours*-$\mathbf{d}'_{\mathbf{so}}$ are the results when using *only* the depth values with no image or label information.

3.5 Experiments

In this section, we describe the experimental settings. Each experiment is accompanied by a discussion investigating the outcomes.

Predicate Prediction Our main goal is to investigate the role of depth maps in relation detection. Therefore, we do not focus on improving the object detection accuracy and report predicate prediction results. In this setting, the relation detection performance is isolated from the object detector's error

Table 2: Comparing predicate prediction performance on VRD given different feature extractors. When a model pre-trained on RGB images is employed for depth map feature extraction, the accuracy drops to even less than Ours - $c'_{so}, v'_{so}, l'_{so}$ with no depth maps. Highest accuracy is achieved when the feature extractor is trained from scratch and specifically for the relation detection task.

Task	**Zero-shot Predicate Pred.**		**Predicate Pred.**	
Metric	R@100	R@50	R@100	R@50
AlexNet-BN - Raw	**87.43**	**72.28**	**96.54**	**90.47**
AlexNet-BN - Pre-trained	83.59	67.20	60.29	45.77
VGG16 - Pre-trained	82.74	66.44	71.17	48.93

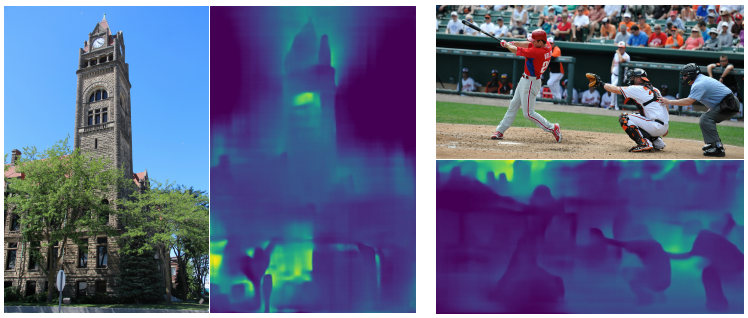

Figure 3: Using depth maps improves the understanding of *perspective* and increases the detection rate of relations such as *(Tower, taller, Trees)* even under noisy conditions such as in the left image. On the other hand, detecting the relation *(Person, stand behind, Person)* in the right image is not improved. One should note that here the athletes have similar depth values, which is more commonly observed in other relations such as *next to*. It might be difficult for the model to learn this case, unless supportive samples are presented during training.

by using ground truth bounding boxes of the entities and their class labels to predict the most likely predicates.

Some triples e.g. *(Man, rides, Horse)* might appear in both test and training sets (with a differently appearing man and horse) while some triples in test set might have never been observed during training at all. To evaluate our model's potential in generalizing to such unseen triples, we also report *zero-shot predicate prediction* where we evaluate the prediction accuracy of relations that are never seen during the training. This is only reported in VRD dataset as there are no comparable results available from other works on VG.

Discussion: The results are shown in Table 1. The upper part of the table demonstrates the results directly reported from those works while the lower two parts present the results from own implementations. We can see that our full model achieves the highest accuracy in comparison to the others in all settings in both datasets. It is also interesting to note that when using *only* depth maps we can already achieve a significant accuracy in predicate prediction, emphasizing on the value of relational information stored within depth maps. As will be shown later, gaining this improvement would not have been possible without cautious employment of the feature extractor.

Considering the imbalanced number of relations within the datasets (e.g. *taller than* appears much less often than *has*) and to get a better intuition of the improvements that using different types of information bring, we plotted the top 10% absolute changes in prediction accuracy of VRD images, for each predicate class (Figure 4). The left plot shows the performance changes from *Ours*-$\mathbf{c}'_{\mathbf{so}}$ to *Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}$. The right plot shows the changes from *Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}$ to *Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}, \mathbf{d}'_{\mathbf{so}}$. When depth maps are included, predicting some of the predicates such as *across*, *sleep on* and *taller than* gains a large improvement while some of the predicates such as *lying on*, *look* and *stand behind* either get worse or stayed the same. To explore the potential reason behind these results, we further examined the test images. Figure 3 as an example, shows the relation *(tower, taller than, trees)* on the left, which was correctly detected after including the depth maps. In general, as shown in Figure 4, the accuracy of relations including the predicate of *taller than*, has been improved. As sometimes the

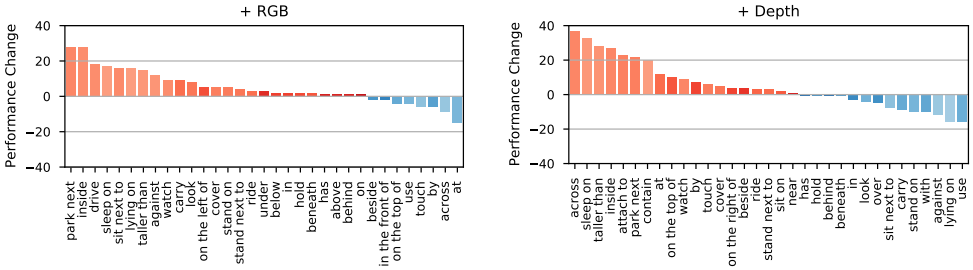

Figure 4: This plot shows the top 10 percent absolute changes in prediction performance per predicate, comparing *Ours*-$\mathbf{c}'_{\mathbf{so}}$ to *Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}$ and *Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}$ to *Ours*-$\mathbf{c}'_{\mathbf{so}}, \mathbf{v}'_{\mathbf{so}}, \mathbf{l}'_{\mathbf{so}}, \mathbf{d}'_{\mathbf{so}}$. The aim is to understand the effect of using RGB and depth maps on detection rate of each predicate. The bars are sorted according to the larger improvements. The darker shades within each blue and red area indicate larger frequency of those predicates appearing in the test set. An improvement in predicates with more frequency has a larger effect on the total accuracy presented in Table 1.

large and distant objects might appear smaller than the nearby, smaller objects, understanding such relations requires a good understanding of perspective, which is provided here by depth maps. On the other hand, we expected to have improved accuracy in detecting relations such as stand behind when using depth maps, while we see no improvements. One of the test images containing the predicate *stand behind* is shown in Figure 3, on the right. Here, the annotations indicate that *(Person, stand behind, Person)* while based on the corresponding depth map, they are in the same distance from the camera. It is important to note that humans describe a predicate from their point of view which is not necessarily associated with the camera's point of view. This effect might appear in many other instances within the dataset, leading to worse performances when dealing with such predicates. A simple way to overcome such problems is by having a richer set of data. Note that the imprecision within the generated depth map, e.g. sky and reflective objects such as glasses, are inevitable as there are no ground truth depth maps available.

Depth Feature Extraction In this experiment, we elaborate on the discussion presented in Section 2.1.2 and evaluate the effect of feature extraction on predicate prediction accuracy. Table 2 presents the results for this experiment. We compare the VRD results when (1) The AlexNet-BN (Krizhevsky, 2014) architecture is pre-trained for object detection using the RGB images of the 100 object categories and then fine-tuned to the depth maps for relation detection (*AlexNet-BN - Pre-trained*). (2) The AlexNet-BN (Krizhevsky, 2014) architecture is trained from scratch using the depth maps for relation detection (*AlexNet-BN - Raw*). (3) A deeper network (VGG16 (Simonyan & Zisserman, 2014)) is pre-trained on a larger dataset (ImageNet (Russakovsky et al., 2015)) and fine-tuned on the depth maps for relation detection (*VGG16 - Pre-trained*).

Discussion: As the results show, a small network trained specifically on depth maps for the relation detection task (*AlexNet-BN - Raw*) is superior to a network using weights pre-trained on RGB images for object detection (*AlexNet-BN - Pre-trained*) even if the network is deeper and pre-trained on a much larger corpus such as ImageNet (*VGG16 - Pre-trained*). In fact, when we used pre-trained networks there was no improvement in performance at all.

4 Conclusion

We identified 3D information as an important attribute for visual relation detection. Since this information is typically not provided in visual data sets, we employed an RGB-to-Depth network, which was pre-trained on a large corpus of data, to generate depth information. We showed that for relation detection, one gets significant improvements by training a CNN feature extractor for the depth images from scratch, rather than using a CNN optimized for RGB data, as it is common practice. In empirical evaluations, we demonstrate that by using depth information, one achieves significantly better performance compared to other state-of-the-art methods.

REFERENCES

Stephan Baier, Yunpu Ma, and Volker Tresp. Improving visual relationship detection using semantic modeling of scene descriptions. In *International Semantic Web Conference*, pp. 53–68. Springer, 2017.

Liefeng Bo, Xiaofeng Ren, and Dieter Fox. Unsupervised feature learning for rgb-d based object recognition. In *Experimental Robotics*, pp. 387–402. Springer, 2013.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 26*, pp. 2787–2795. Curran Associates, Inc., 2013. URL http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf.

Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 601–610. ACM, 2014.

Andreas Eitel, Jost Tobias Springenberg, Luciano Spinello, Martin Riedmiller, and Wolfram Burgard. Multimodal deep learning for robust rgb-d object recognition. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 681–687. IEEE, 2015.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010.

Saurabh Gupta, Ross Girshick, Pablo Arbeláez, and Jitendra Malik. Learning rich features from rgb-d images for object detection and segmentation. In *European Conference on Computer Vision*, pp. 345–360. Springer, 2014.

Caner Hazirbas, Lingni Ma, Csaba Domokos, and Daniel Cremers. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In *Asian Conference on Computer Vision*, pp. 213–228. Springer, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

Alex Krizhevsky. One weird trick for parallelizing convolutional neural networks. *CoRR*, abs/1404.5997, 2014. URL http://arxiv.org/abs/1404.5997.

Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab. Deeper depth prediction with fully convolutional residual networks. In *3D Vision (3DV), 2016 Fourth International Conference on*, pp. 239–248. IEEE, 2016.

Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision*, pp. 852–869. Springer, 2016.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119, 2013.

Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In *ICML*, volume 11, pp. 809–816, 2011.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2016.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pp. 91–99, 2015.

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. *International Journal of Computer Vision (IJCV)*, 115 (3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Ashutosh Saxena, Min Sun, and Andrew Y Ng. Learning 3-d scene structure from a single still image. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pp. 1–8. IEEE, 2007.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. Reasoning with neural tensor networks for knowledge base completion. In *Advances in neural information processing systems*, pp. 926–934, 2013.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *International Conference on Machine Learning*, pp. 2071–2080, 2016.

Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5410–5419, 2017.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.

Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 670–685, 2018.

Ruichi Yu, Ang Li, Vlad I Morariu, and Larry S Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5831–5840, 2018.

Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 3107–3115. IEEE Computer Society, 2017. ISBN 978-1-5386-0457-1. doi: 10.1109/CVPR.2017.331. URL `https://doi.org/10.1109/CVPR.2017.331`.