Show, Attend and Relate: Scenegraph Generation with Visual Attention and Depth Perception

Saravana Kumar
The Ohio State University
shanmugamsakthivadivel.1@osu.edu

Arnab Banerjee
The Ohio State University
banerjee.146@osu.edu

Paul Linville
The Ohio State University
linville.50@osu.edu

Robert Gross
The Ohio State University
gross.567@osu.edu

Abstract

1 Introduction

Recent advances in deep neural networks and computer vision has placed a disproportionate focus on object detection and identification. However, the goal of computer vision is to identify what is in an image, where it is located in the image, and what is it doing. Although detection and identification are important tasks, understanding an image fully also requires that there exist comprehension of the activity of an object such as the interaction with other objects in the image. For example, in Figure 1 the knowledge of the existence of man, dog and sheep in the image is important information, however, true visual understanding comes from knowing that the man is herding sheep and the dog is playing in the grass nearby. Understanding an image in this way can help in various other tasks such as image generation [4], Visual Question Answering (VQA) [8, 2] and video captioning [7].

Scenegraphs are a natural structure to represent the relationship between the objects in an image. A scenegraph is a graph in which the nodes are representations of objects and the edges represent the relationship between them. An example of a scenegraph for 1 is show in 2

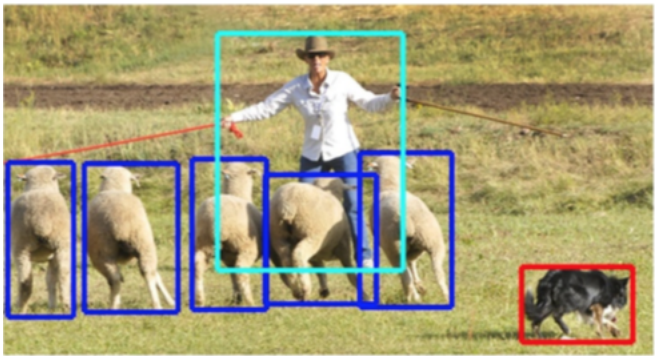


Figure 1: A sample figure from MS COCO of a man herding sheep while the dog is eating grass

The task of scenegraph generation is inherently difficult. In the past few years alone [10, 12, 5, 6] have all tried to tackle this problem with no promising results. We investigate the problem of scenegraph generation and make the following contributions in our work:

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

1. We use visual attention to predict the edges between objects
2. We investigate the use of depth maps in predicting the relationships between objects in images

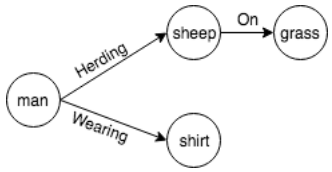


Figure 2: Scenegraph for the image show in Figure 1

2 Dataset

The visual genome dataset was utilized for this undertaking. This dataset includes:
1.) 108,077 Images
2.) 2.3 Million Relationships
3.) 3.8 Million Object Instance
The dataset can be found with the following link:
`http://visualgenome.org/api/v0/api_home.html`

3 Visual Attention

3.1 Object Detection

The first step in our visual attention approach is to perform object detection on each image in the visual genome dataset. For this, we used a TensorFlow implementation of Faster RCNN. This model classifies object proposals using deep convolutional networks and is comparitively fast at training and testing compared to other object detection algorithms.

Faster RCNN takes an entire image and a set of object proposals as input. It then processes the whole image with several convolutional (conv) and max pooling layers to produce a conv feature map, and for each object proposal a region of interest (RoI) pooling layer extracts a fixed-length feature vector from the feature map. Each feature vector is fed into a sequence of fully connected (fc) layers that finally branch into two sibling output layers. These output layers consist of a softmax for K object classes and "background class" and bounding box positions for each.

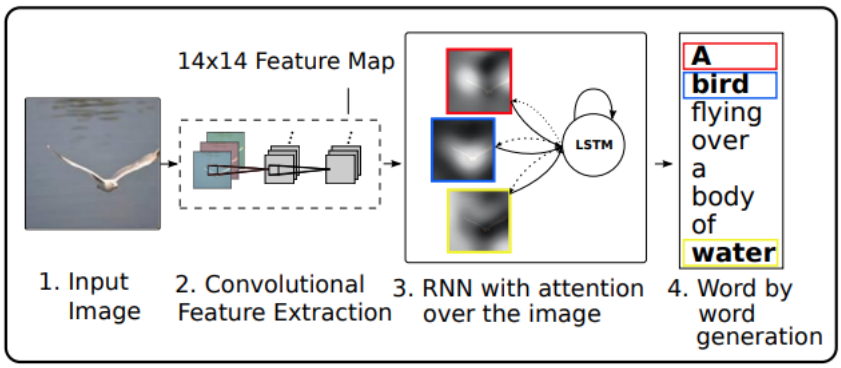


Figure 3: T
he Show Attend and Tell Image Captioning model that uses visual attention for generating captions.

3.2 Caption Generation

Visual Attention is the process of focusing only on a part of the input image and not on the entire image. This has for long been used in caption generation [11]. The Show Attend and Tell caption

generation model is shown in Figure 3. It uses visual attention and Long Short Term Memory [3] models to generate captions for an image. In a similar fashion we convert the scenegraphs in the Visual Genome dataset into captions of the form <SUBJECT> <RELATIONSHIP> <OBJECT> as shown in Figure 4. We then use these captions to train the Show Attend and Tell caption generation model and generate the relationships as captions for the testing set as shown in figure **??**. The results of this approach was about 14% recall. The limitation with this approach and perhaps the reason why we get miserable results is because of the fact that we do not leverage information about the objects in the image and just let the caption generation model figure it out.

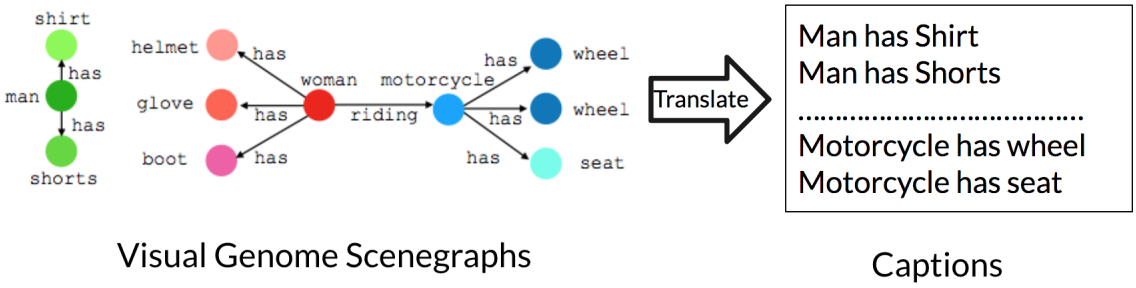
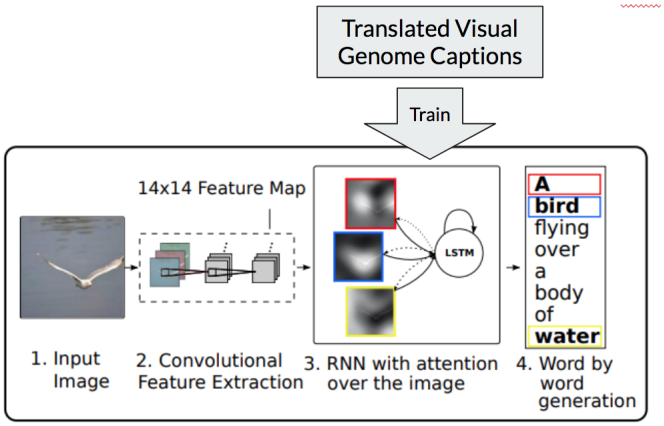


Figure 4: Transforming scenegraphs into captions



3.3 Show Attend and Relate Model

The Show Attend and Relate model is an extension of the Show Attend and Tell caption generation model in which we leverage the object information of the image and also use visual attention to predict the relationship between objects in the image. The Show Attend and Relate model is shown in figure 5. In the Show Attend and Relate model, we break down the scenegraph generation problem into 2 separate sub problems. The first problem is the problem of object detection for which established models exists such as the Faster R-CNN method described above. We use the Faster R-CNN model to first propose a set of objects in the image. We then develop a separate Recurrent Neural Network model to do the relationship prediction that learns to focus its attention on different parts of the image based on the objects in the image. The RNN model that we develop is based on the seq2seq model developed in [9]. In order to draw parallels with the Neural Machine Translation problem described in [9] we transform the problem of scenegraph generation into a Neural Machine Translation problem where the source sentence is the two objects and the target sentence is the relationship to be predicted. This process is shown in Figure 6. We also apply the attention layer to the image and pass it as the initial hidden state to the seq2seq model. This model performs considerably better with a recall of 30

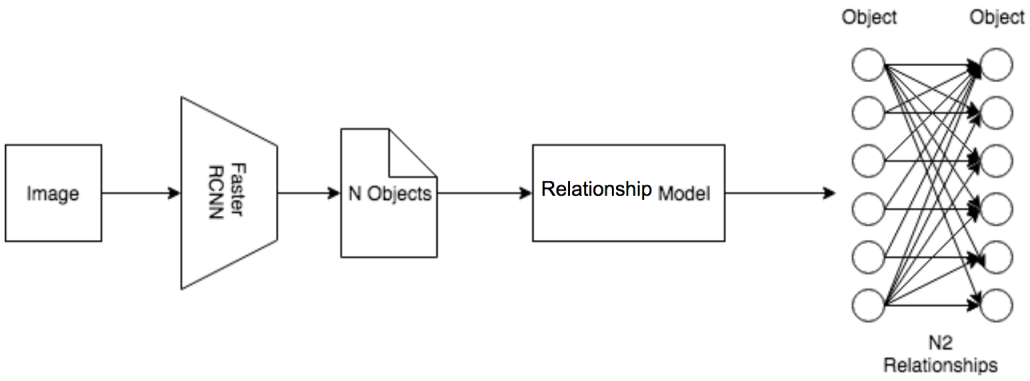


Figure 5: The show attend and relate model

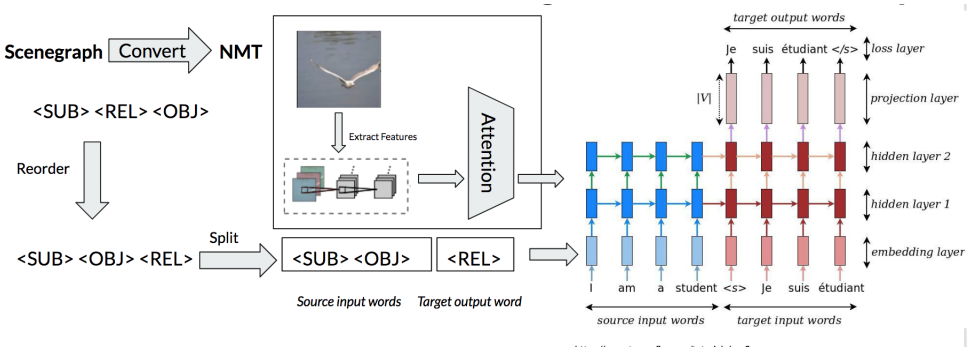


Figure 6: Applying a Seq2Seq model for relationship prediction

4 Depth Perception

In 3D computer graphics a depth map is an image or image channel that contains information relating to the distance of the surfaces of scene objects from a viewpoint. A depth map can provide us distance information which can be used for several purposes like autostereograms.etc. For example if we consider the below image the corresponding depth map is as shown

(a) Input (b) Depth Map

Figure 7: Depth Map Generation

Implementation

The implementation is based on the paper "Depth Map Prediction from a Single Image using a Multi-Scale Deep Network"[1]. The implementation was done using Tensorflow in order to develop the Convolutional Neutral Network Model.

The overall model used can be summarized using the following picture.

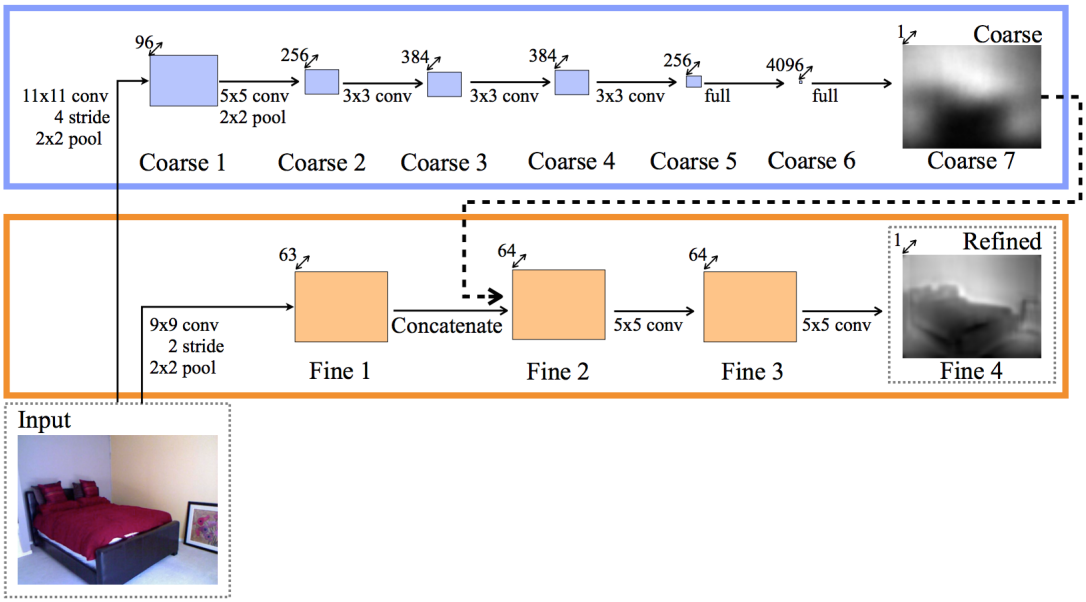


Figure 8: Model used for training

As illustrated in the above picture there are two stacks of deep networks: one is called coarse global predictor that makes a global prediction based on the entire image and the other is called fine local predictor that refines this prediction locally. Both stacks are applied to the original input, but the coarse network's output is passed to the fine network as additional first-layer image features. In this way, the local network can edit the global prediction to incorporate finer-scale details.

The global coarse network contains five feature extraction layers of convolution and max-pooling, followed by two fully connected layers. All hidden layers use rectified linear units for activations, with the exception of the coarse output layer 7, which is linear.

The task of the local fine network component is to edit the coarse prediction it receives to align with local details such as object and wall edges. The fine-scale network stack consists of convolutional layers only, along with one pooling stage for the first layer edge features. The coarse output is fed in as an additional low-level feature map. By design, the coarse prediction is the same spatial size as the output of the first fine-scale layer (after pooling),and we concatenate the two together. Subsequent layers maintain this size using zero-padded convolutions. All hidden units of this layer use rectified linear activations. The last convolutional layer is linear, as it predicts the target depth.

The loss function which is used to compute the training loss is given by the below formula:

$$L(y, y^*) = \frac{1}{n}\sum_i d_i^2 - \frac{\lambda}{n^2}\left(\sum_i d_i\right)^2$$

Figure 9: Training Loss

where

$d_i = \log y_i - \log y_i^*$ and $\lambda \in [0, 1]$

Results :

(a) Input(NYU Dataset) (b) Output

Figure 10: Depth Map Generated from trained model

(a) Input(Visual Genome Dataset) (b) Output

Figure 11: Depth Map Generated from trained model

5 Visual Translation Embedding Network

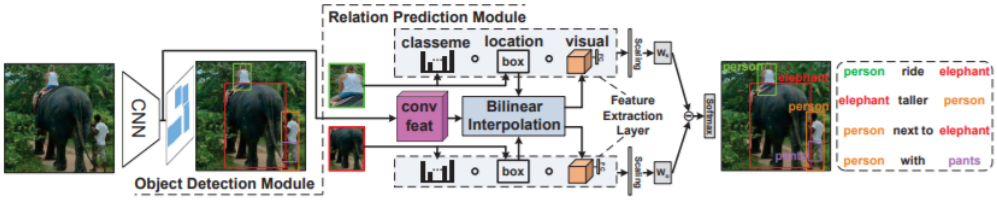
The depth map becomes useful as it is utilized as a portion of the Visual Translation Embedding Network (VTEN) [13]. Based on the referenced paper, This network has a couple main parts which can each be subdivided into smaller portions themselves. In figure 12 below, the basic design can be visualized. The first and simplest part of the complete network is the object detection phase. Similar

to the other solution to the scene graph problem evaluated in this paper, this is done with Faster RCNN. Assuming an understanding of this is already made the most important part comes next.

The network seeks to create two relation translation vectors to evaluate the predicate in the subject, predicate, object trio. These matrices are created by concatenation of three parts including the probabilities from classeme, location, and visual relation. Classeme seeks to evaluate and remove unlikely relations based on the trio segmentation. For example, it is highly unlikely to have a situation where "cat rides person". On the other hand, location assesses whether the two objects location in the image make sense in relation to the predicate joining them. An example of this would be checking whether object A is directly above object B in the image to say "A on B". Lastly, the visual relations seeks to get visual features through a bilinear interpolation, or smoothing operation, in the last layer of the detection phase. These visual features suggest that if it is known that "person above skateboard" it is more likely that "person on skateboard" etc.

With this understanding, it is here where the depth network comes into play. The network of the paper only looks at distances in terms of x and y coordinates of the image when evaluating location understanding. However, looking at depth should naturally be useful as well since two objects which are close to the camera can be further apart than two objects further in the image, but in reality be comparatively closer and more likely to have a relation. In other words, this is basically an enhancement of the paper's location feature filter to include a z-direction.

Finally, the vectors which resulted from the middle are used to produce the results. This is done by optimization through learning the projection matrices to be multiplied against the vectors. After this multiplication is done, the difference is computed between the two. This becomes a part of the sum over all possible combinations of subject, predicate, objects, such that with the negative log softmax of the previous result, a probability is produced and the top results are kept.



(a) Faster RCNN used for detection, concatenation of the 3 features in the middle section, and optimization of the sum of the softmax to get results.

Figure 12: VTEN Basic Design

Observations

1. The NYU Dataset was used for training so a relatively good image depth map was developed in the validation instance of the NYU Dataset after training.
2. The output corresponding to the input taken from Visual Genome Dataset shows two human forms in black. Though the human form is not completely understandable in the output.
3. There are two ways we can increase the accuracy of the model:

- Increase the number of training epochs.
- The dataset used for training(NYU Dataset) only contains images of rooms and other interior parts of a house. Thus the training dataset is not diverse enough for the model to learn parameters for any general image.

6 Evaluation

In the scenegraph generation area, evaluation has been done for the following tasks:

PREDCLS: Feed the model no ground truth information **SGCLS**: Feed the model with only the bounding box information and predict the class and the relationships. **SGDET**: Feed the model with class labels and bounding boxes and predict the relationships.

		Scene Graph Detection			Scene Graph Classification			Predicate Classification			Mean
	Model	R@20	R@50	R@100	R@20	R@50	R@100	R@20	R@50	R@100	
models	VRD [29]		0.3	0.5		11.8	14.1		27.9	35.0	14.9
	MESSAGE PASSING [47]		3.4	4.2		21.7	24.4		44.8	53.0	25.3
	MESSAGE PASSING+	14.6	20.7	24.5	31.7	34.6	35.4	52.7	59.3	61.3	39.3
	ASSOC EMBED [31]⋆	6.5	8.1	8.2	18.2	21.8	22.6	47.9	54.1	55.4	28.3
	FREQ	17.7	23.5	27.6	27.7	32.4	34.0	49.4	59.9	64.1	40.2
	FREQ+OVERLAP	20.1	26.2	30.1	29.3	32.3	32.9	53.6	60.6	62.2	40.7
	MOTIFNET-LEFTRIGHT	21.4	27.2	30.3	**32.9**	**35.8**	**36.5**	**58.5**	**65.2**	**67.1**	**43.6**
ablations	MOTIFNET-NOCONTEXT	21.0	26.2	29.0	31.9	34.8	35.5	57.0	63.7	65.6	42.4
	MOTIFNET-CONFIDENCE	**21.7**	**27.3**	**30.5**	32.6	35.4	36.1	58.2	65.1	67.0	43.5
	MOTIFNET-SIZE	21.6	**27.3**	30.4	32.2	35.0	35.7	58.0	64.9	66.8	43.3
	MOTIFNET-RANDOM	21.6	**27.3**	30.4	32.5	35.5	36.2	58.1	65.1	66.9	43.5

Figure 13: Results of scenegraph detection from previous work

The metric for measurement has been R@K, which means what fraction of the ground truth information occurs in the top K predictions made by the model.

7 Result

The results for the scene graph generation tasks can be seen in Table 13

As can be seen from Table 13 the scenegraph generation task is pretty hard with very low R@K metrics.

We were unable to put together out Object Detection and Relationship prediction modules together. We used ground truth information for class labels and predicted the relationships only for which the absolute recall is 30%. Although these numbers are not compatible to be compared, it gives and indication that these results are promising, since even without knowing the location of the objects, it is able to predict to a reasonable degree.

8 Conclusion

The depth network results were not too great when tested on the visual genome dataset which causes a great breakdown in its value since if its inaccurate, it does not really contribute towards understanding z-location features in the image. Another complication occurred with the second approach in that the depth network was built in tensorflow with python, whereas the main network was created in caffe through matlab. It would be great to get more depth annotated data to train with, especially the visual genome dataset to offset the first problem.
The visual attention method provides promising results for scene graph generation. Great improvements can be made on the model by using the location of the objects and not just the labels of the objects. **Special thanks to Professor Andrew Plummer for helping to acquire access to OSC resources to assist in running the networks.**

9 Contribution

1. Saravana Kumar: All Visual Attention Models
2. Arnab Banerjee: Depth Map
3. Paul Linville: Object Detection
4. Robert Gross: Visual Translation Embedding Network

References

[1] R. F. David Eigen, Christian Puhrsch. Depth map prediction from a single image using a multi-scale deep network. *arXiv preprint arXiv:1406.2283*, 2014.

[2] F. F. de Faria, R. Usbeck, A. Sarullo, T. Mu, and A. Freitas. Question answering mediated by visual clues and knowledge graphs. In *Companion of the The Web Conference 2018 on The Web Conference 2018*, pages 1937–1939. International World Wide Web Conferences Steering Committee, 2018.

[3] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[4] J. Johnson, A. Gupta, and L. Fei-Fei. Image generation from scene graphs. *arXiv preprint arXiv:1804.01622*, 2018.

[5] X. Liang, L. Lee, and E. P. Xing. Deep variation-structured reinforcement learning for visual relationship and attribute detection. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on*, pages 4408–4417. IEEE, 2017.

[6] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision*, 2016.

[7] C.-Y. Ma, A. Kadav, I. Melvin, Z. Kira, G. AlRegib, and H. P. Graf. Grounded objects and interactions for video captioning. *arXiv preprint arXiv:1711.06354*, 2017.

[8] A. Mahendru. *Role of Premises in Visual Question Answering*. PhD thesis, Virginia Tech, 2017.

[9] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[10] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing.

[11] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057, 2015.

[12] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi. Neural motifs: Scene graph parsing with global context. *arXiv preprint arXiv:1711.06640*, 2017.

[13] H. Zhang, Z. Kyaw, S. Chang, and T. Chua. Visual translation embedding network for visual relation detection. 2017.