Visual Question Answering using Structured Image Representation

Khushboo Mehra

4th Year Project Report
Cognitive Science
School of Informatics
University of Edinburgh

2017

Abstract

In this project, we investigate the effect of using structured image representation on Visual Question Answering. First, we design an augmented Visual Dependency Representation that is based on how humans see and understand images. We propose a heuristic based approach to automatically generate VDRs for the Abstract Scenes VQA dataset. We show that our approach can identify all significant interactions in scenes that contain multiple actors and depict complex interactions. Then, we explore the use of VDRs for the VQA task. We use a classification based approach that predicts scene interactions. We experiment with different feature sets and evaluate the model on open ended questions from the VQA dataset. We find that the including spatial dependencies of a scene gives a large boost in accuracy when predicting the objects involved in an interaction. However, no significant improvement was observed in predicting verbs. Predicting complex interactions requires reasoning based on the scene structure and semantics- which is a challenging task. Overall, we show that a knowledge of image structure is an essential component to understanding scenes.

Acknowledgements

I would like to thank my project supervisor, Dr. Frank Keller for his patience and support throughout the project. I'm grateful for his guidance and feedback, which has been integral to shaping the work done in this project.

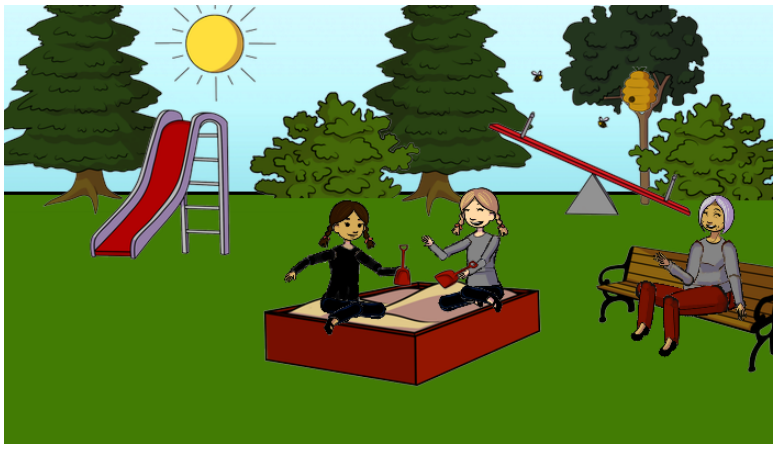
I would also like to thank my brother for his feedback on the writing and for helping me with the proof reading.

Table of Contents

Chapter 1

Introduction

Humans do not see an image as a simple collection of objects- they can easily interpret how the objects relate to each other and understand what is happening in a scene. Two scenes can have the same objects yet very different meanings based on the interaction between the objects. When images are represented as a bag-of-objects, this information is lost. Scene perception literature from cognitive psychology has long established that humans use structural and semantic relationships between objects to represent scenes (for example, Biederman, Mezzanotte, & Rabinowitz, 1982). To develop intelligent systems capable of high level visual tasks such as describing scenes or answering questions about a scene, we need models that can go beyond reasoning based on simple object co-occurrence. They should have the ability to analyse how the objects relate to each other, the actions and events they are involved in, and ultimately try to infer the intentions of the actors depicted in the scene. Thus, the focus should be on analysing the structure of the scene and the understanding the interactions of the scene objects.



```
Are the two girls playing with each other?
How many females are present?
Where is the beehive?
```

Figure 1.1: An example of the VQA task: given an image and associated questions, the goal of the VQA model is to generate answers based on the image.

Elliott and Keller (2013) proposed a novel framework for encoding the spatial structure of images known as Visual Dependency Representation (VDR). VDRs express the dependency between different objects in a scene. Based on their work, we will develop an augmented-VDR for a new dataset and explore its use in Visual Question Answering (VQA). The VQA task involves the following: for a given image and an associated natural language question, the goal is to generate an accurate answer based on the image (Fig 1.1). The generated answer typically consists of a few words, or a short phrase. VQA can be used to automatically evaluate a model's capacity for scene understanding. Alternatively, VQA can be a standalone application in itself. A rich understanding of scenes would, nonetheless, pave way for applications in areas such as improved assisted technology for the visually impaired or robotic interactions.

1.1 Contributions

1.1.1 Augmenting existing Visual Dependency Representation

The VDR developed by Elliott & Keller aims to capture the key object interactions in a given scene. The structural encoding obtained from this VDR has been used for tasks such as image description generation and image retrieval. As we will discuss in Chapter 3 we need to augment the existing VDRs for the VQA task. The main effort lies in designing the augmented Visual Dependency Grammar (VDG) and a graph pruning algorithm that uses a heuristic based approach to identify and remove relations that do not contribute to the meaning of a scene.

1.1.2 Creating augmented-VDR for Abstract Scenes dataset:

Prior work on VDR generation mainly uses datasets of real world images. Currently, there is no existing VDR set for the Abstract Scenes VQA dataset. Note that there is a VDR set for an earlier version of abstract scenes, however, that dataset does not contain the question/answer annotations, so it cannot be used for VQA.

The second aim of this project is to create silver standard augmented-VDRs for the Abstract Scenes VQA dataset. We do not have gold standard VDRs for this dataset, which rules out the use of a standard learning approach in our work. Therefore, will use a heuristic based approach to generate VDRs, which will be subsequently used for the VQA task.

1.1.3 Visual Question Answering

There is no prior work on using image structure for the VQA task. The third aim of this project is to examine whether adding the structure annotations to images in the form of VDRs leads to improved performance on the VQA task. Since VDRs capture the interactions among scene objects, we are particularly interested in investigating if

using structural information will lead to an improved accuracy on questions that ask about the spatial properties of the scene or about the interactions among scene objects.

1.2 Outline

The remaining chapters of this dissertation are organised as follows:

- Chapter 2 contains a brief overview of related work in the computer vision and natural language processing fields – the types of tasks they address and the different approaches and datasets used.
- Chapter 3 describes the process of generating augmented-VDRs for the Abstract Scenes VQA dataset. We discuss the design, methodology, implementation and evaluation of a system to automatically produce silver standard VDR for the dataset.
- Chapter 4 discusses the use of VDRs in the VQA task. We present a model based on template matching and a maximum entropy classifier. We evaluate this model on a subset of open ended questions in the Abstract Scenes VQA dataset. We show that the use of image structure encoding can be used to improve answer generation.
- Finally, in Chapter 5, we conclude with an analysis of the strengths and shortcomings of the system developed and suggest some improvements that can be made in future work on using VDRs for VQA.

Chapter 2

Related Work

Interest in cross-modal tasks such as image captioning, image description generation, image retrieval and visual question answering has been rising rapidly in the recent years. These are challenging tasks as they combine natural language processing (NLP) and computer vision (CV). They require a full understanding of scenes in terms of the constituent objects, their attributes, spatial relations and the interactions among those objects. Additionally, there is a natural language generation component involved which requires steps such as deciding which aspects of the scene to address, selecting the relevant information related to these aspects, and generating a natural language output.

This project focuses on the task of visual question answering (VQA). A large amount of literature exists on automatic image description generation while visual question answering is a relatively new research problem. In the last few years, significant improvements have been made in the state-of-the-art models for these tasks. This is mainly due to changes in three areas— how models capture and represent the relevant information, the availability of large, more realistic datasets and the use of deep learning techniques that allows the models to learn a better representation of the input and a mapping between the linguistic and visual modalities. This chapter reviews the existing work in these areas.

2.1 Image Description Generation

As stated, most of the work in the combined CV/ NLP domain has been on image description generation. We will begin with a brief review of the approaches used for this task. The description generation and question answering tasks are very similar- both require a visual input (image) and a linguistic output (description/ answer) based on the image, which in turn requires a comprehensive image understanding. We will look at the image description generation literature for key insights on how to approach the VQA task.

2.1.1 Approach

Existing approaches to image description generation address this task as either a generation problem or a retrieval problem. The former approach involves analysing the visual content using an array of CV techniques to detect objects and their attributes. The description are then directly generated based on the image content. The final generation step can use templates (Yang, Teo, Daumé III, & Aloimonos, 2011; Elliott & Keller, 2013) or language models based on n-grams (Kulkarni et al., 2013; Li, Kulkarni, Berg, Berg, & Choi, 2011), maximum entropy (Fang et al., 2015), Recurrent Neural Networks (Socher, Karpathy, Le, Manning, & Ng, 2014), etc. On the other hand, the retrieval based approach uses the training data to retrieve images that are similar to the query (e.g. Gupta, Verma, & Jawahar, 2012; Karpathy, Joulin, & Li, 2014). This step is followed by ranking the descriptions of the retrieved images using visual or linguistic features to produce an output.

2.1.2 Feature space

The feature space used to represent the input and perform retrieval can vary. Some approaches use simple visual features (Gupta et al., 2012) while some use a joint feature space constructed from the visual and linguistic inputs (Socher et al., 2014; Karpathy et al., 2014). Typically, Convolution Neural Networks (CNNs), trained on object recognition, are used to learn visual features while Recurrent Neural Networks (RNNs), trained on large text corpora, are used to get word embeddings and model language.

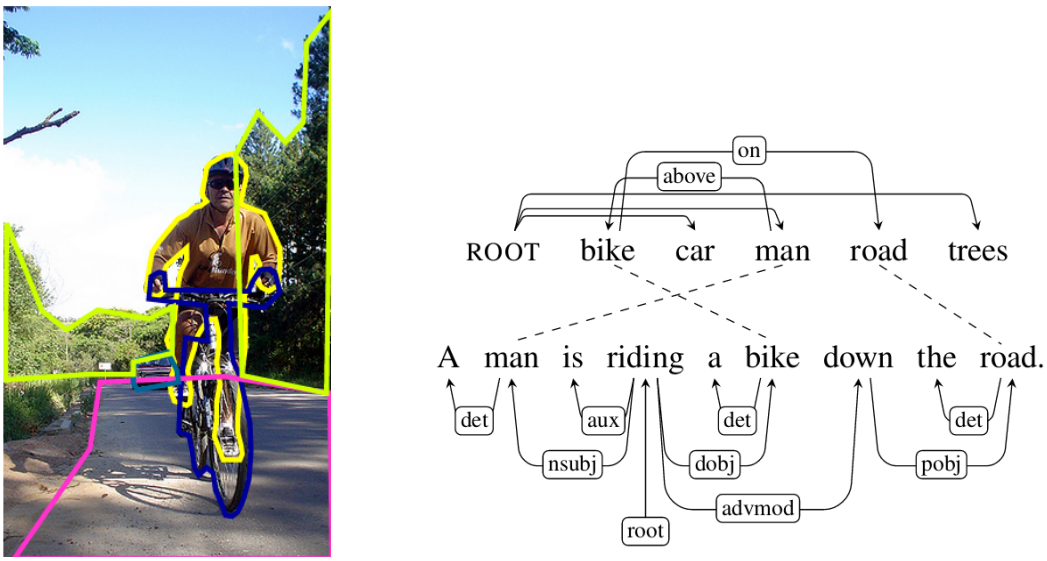
Approaches that use multimodal representation of the input significantly outperform those using unimodal image and text spaces. Features extracted from image-description pairs can be used to learn a common multimodal space. Socher et al. (2014) use Dependency Tree RNN to obtain word and image embeddings separately, which are then projected to a joint space. Then, the task can be treated as a cross-modal retrieval and ranking problem. Karpathy et al. (2014) extend this model by adding fine-grained embeddings (objects and fragments of dependency tree of the sentences). More recently, extensions of these models that also can generate sentences have been developed. Inspired by work in Machine Translation, these approaches use an encoder-decoder framework (e.g. Kiros, Salakhutdinov, & Zemel, 2014). First, the encoder constructs the joint visuo-linguistic space and ranks the images and descriptions. Next, the decoder is used for the generation process. Another approach by Karpathy and Fei-Fei (2015) shows an improved accuracy on image description when the alignments between the image regions and sentence segments are used to learn inter-modal correspondence. These alignments are used to generate novel descriptions for the test image.

2.1.3 Images Representation

The work discussed so far represents images as unstructured bag of objects. That is, images are encoded as vectors that represent the presence or absence of objects. Other

approaches have used representations that include additional annotations of object attributes, spatial relations or scene properties. For instance, Farhadi et al. (2010) use object-action-scene triplets to represent images and descriptions. Similarly, encodings that combine image visual features with their spatial configuration have been used to detect and classify human-object interactions more effectively (Yao & Fei-Fei, 2010; Prest, Schmid, & Ferrari, 2012).

Recently, a few approaches have been proposed in the literature that parse images into explicit structured representations. Elliott and Keller (2013) developed the Visual Dependency Representation (VDR) which encodes the spatial structure of an image as a dependency graph. These relations are derived using a Visual Dependency Grammar (VDG) which links object regions through geometric relations (4.5). This graph is aligned to the syntactic dependency graph of the associated image descriptions to yield a multimodal graph. VDRs have shown improved performance in tasks such as image retrieval (Elliott, Lavrenko, & Keller, 2014) and image description generation. The initial work on description generation by Elliott and Keller (2013) used manually generated gold standard VDRs. They used a template based approach to generate descriptions from the interactions represented in the scene VDR. Later, Elliott and de Vries (2015) showed that VDRs inferred automatically using a statistical parser based on the Maximum Spanning Tree Parser (McDonald, Pereira, Ribarov, & Hajič, 2005), also give results comparable to the state-of-the-art multimodal deep neural networks. Further, (Ortiz, Wolff, & Lapata, 2015) used VDRs as a part of their MT based model for image description on abstract scenes.



(a) (b)

Figure 2.1: An example showing (a) an image with (b) its visual dependency representation (Elliott et al., 2014). VDRs encode the spatial relationship between different object regions *man, bike* and *road*, (b) the VDR graph is aligned to the syntactic dependency parse of the image description to establish cross-modal correspondence.

Another framework based on detailed image semantics is known as scene graphs (D. Lin, Kong, Fidler, & Urtasun, 2015). Scene graphs are generated from images annotated with objects, their attributes and relationship to each other (Fig 2.2). These relationships can represent the spatial configuration or the action being performed by the subject. Scene graphs have been used for description generation and image retrieval (D. Lin et al., 2015; Johnson et al., 2015). Together, these studies show that knowledge of the spatial structure is essential to a comprehensive understanding of images.

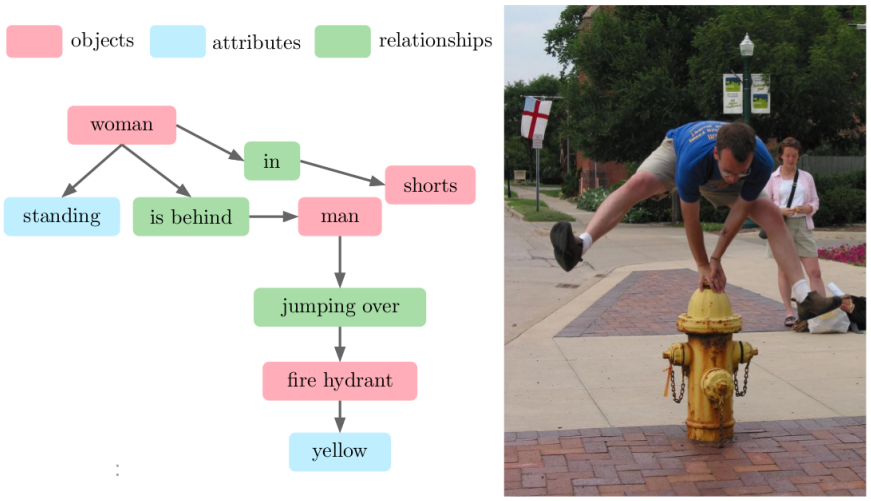


Figure 2.2: An example of a scene graph (Lin et al., 2015). Scene graphs encode the objects (*woman*), attributes (*woman is standing*) and relationships (*woman is behind man*) present in an image.

2.2 Visual Question Anwsering

In this section we will review the existing work on Visual Question Answering, with a particular focus on different approaches used and their specific advantages. A detailed description of the model architectures is beyond the scope of this chapter; it can be found in Wu, Teney, et al. (2016).

Geman, Geman, Hallonquist, and Younes (2015) developed the Visual Turing Test which measures the ability of a CV system to answer binary questions based on a test image. In contrast, Antol et al. (2015) introduced the concept of open-ended Visual Question Answering. These questions require a free-form answer; and generating these requires a combination of vision, language and knowledge-base reasoning together. The questions can be about any aspect of the scene, such as the object attributes (e.g. What is the colour of the dog?), scene attributes (e.g. *Is it sunny?*), scene structure (e.g. *Where is the man sitting?*) or object interactions (e.g. *What are the children doing?*) .

Tasks that involve textual question answering — like reading comprehension, information retrieval have long been the focus of research in NLP. The goal in textual QA is to

find an answer from textual narratives or knowledge bases. The typical approach to the textual QA task involves syntactic parsing and regular expression matching. We can think of VQA as an extension to textual QA. However, the added visual information from images is richer and less structured as compared to text, so standard textual QA approaches cannot be used for VQA.

Literature on VQA proposes a number of methodologies that have been motivated by the work in the related CV/NLP tasks discussed earlier. One of the first approaches to VQA was based on probabilistic inference (Malinowski & Fritz, 2014). More recently, models use a combination of CNNs and RNNs to learn the visual and linguistic representation of the input image and the question, respectively, in a common feature space. This method enables the models to perform inference over the question and the image content together. In terms of problem formulation, VQA has been approached as both a sequence generation problem (Malinowski, Rohrbach, & Fritz, 2015) and a classification problem (Ren, Kiros, & Zemel, 2015). The former approach can produce answers of variable lengths while in the latter, the model learns to predict classes which are essentially the different answers present in the training set. B. Zhou, Tian, Sukhbaatar, Szlam, and Fergus (2015) developed a simple baseline using a traditional bag-of-words approach that gives a comparable performance to other models that use RNNs. Other work includes models based on image-text grounding (Zhu, Groth, Bernstein, & Fei-Fei, 2016), memory-augmented and attention based VQA models (Lu, Yang, Batra, & Parikh, 2016). An advantage of using the additional attention mechanism is that it allows the model to use local image features, and therefore, generate answers based on specific image regions. An alternative approach that uses an external knowledge base (Wu, Wang, Shen, Dick, & van den Hengel, 2016) has shown improved accuracy on questions that rely on common sense knowledge.

We are not aware of any previous work on VQA that uses structured image representation.

2.3 Datasets

The work on image description generation and visual question answering discussed in this chapter uses a wide range of datasets that are annotated with natural language image descriptions. Since the first step in the image processing pipeline for any high level vision task is object detection, a good performance at this stage is crucial for the rest of the pipeline to work. If object detection is poor, then the generated image description or answer will be automatically incorrect. For a given model, there are two main ways of improving object detection: first, using more training data and second, using datasets that are close to the real world. In real-world scenes, objects are often present in a non-canonical perspective, or against a cluttered background, or they may be partially occluded.

The introduction of new datasets such as MS COCO (T.-Y. Lin et al., 2014) has addressed these areas. This dataset contains real world images with rich contextual information. It contains 300K images, which results in a large training instances to category

ratio. This, in turn, has not only helped make object detection more robust, but also enabled the use of models with complex architectures. The VQA dataset (Antol et al., 2015), built using images from MS COCO is the most widely used dataset for the VQA task. It is annotated with question-answer pairs of two types- multiple choice and open-ended. The DAQUAR is another dataset used for VQA (Malinowski & Fritz, 2014). In this dataset, the questions focus on the type, number and colour of objects.

Zitnick and Parikh (2013) proposed the use of clipart based abstract scenes to study semantic scene understanding. In contrast to using real images, the abstract scenes dataset gives information about object labels, location, size without the use of noisy object detectors. Additionally, using abstract images gives the ability to create a large number of semantically similar scenes in the dataset, which would not be possible using real world images. Later, Antol et al. (2015) developed the Abstract Scenes VQA dataset. It is alternative that allows researchers to isolate the problem of understanding high level semantics and the complex relationships in scene objects from the task of object detection. Research using abstract scenes has focused on areas such as capturing common sense knowledge (Fouhey & Zitnick, 2014), learning models of human interactions (Antol, Zitnick, & Parikh, 2014), exploring the semantic relevance of image features (Zitnick & Parikh, 2013) and generating scenes based on natural language descriptions (Zitnick, Parikh, & Vanderwende, 2013).

2.4 Conclusion

Out review of the existing literature on image description generation and visual question answering leads us to three key conclusions which have influenced the approach used in the project. These are:

- using *cross-modal alignments* between the visual and linguistic feature space is effective because it provides a more holistic understanding of the scene by exploiting the correspondences between the visual and linguistic data.
- using *image structure* is a powerful way of capturing the interactions among scene objects. This helps in understanding the semantics of a scene.
- the use of *abstract scenes* gives the ability to address the research question in isolation as it helps avoid the noise introduced during object detection.

For the VQA task, we propose a methodology with a structured image representation that aligns the visual and linguistic feature space. We will evaluate this approach on the Abstract Scenes VQA dataset. Additionally, we have seen some approaches that use features which are better at representing the input, but use simpler models, that can often outperform models employing deep learning architecture. Since there is no prior work that uses image structure for VQA, we will use a relatively simple approach. We formulate the VQA task as a classification problem that uses multinomial logistic regression to predict the interaction represented between object pairs.

Chapter 3

Generating Augmented Visual Dependency Representation

The first aim of the project is to develop an augmented Visual Dependency Representation for the Abstract Scenes VQA dataset, which will used in the Visual Question Answering task in Chapter 4. As mentioned in Chapter 1, we need to augment the existing VDR. In this chapter, we begin by introducing the VDR as proposed by Elliott and Keller (2013). Then we will discuss why the VQA task requires us to augment the existing VDRs. Next, we will present the details on the dataset, methodology and evaluation methods used for automatically generating augmented VDR for the VQA dataset. Finally, we will discuss and evaluate the results obtained.

3.1 Visual Dependency Representations

The VDR was developed by Elliott and Keller to bridge the semantic gap between how humans perceive and reason about images and how AI models represent them. In computational linguistics, dependency representations are widely used to model the syntactic relationship between the words of a sentence. Inspired by the natural language dependency syntax, the VDR of a scene is a directed acyclic graph that defines the spatial relationships over annotated regions of a scene. Thus, the arcs in a VDR graph represent the geometric relations between objects pairs that correspond to the head and the argument of the arcs. These relations are defined based on a Visual Dependency Grammar (VDG) which is an set of spatial relations that express how objects in an image can be related to each other. This knowledge of image structure allows us to distinguish between image objects that simply co-occur, and the objects that are interacting with each other (Fig 3.1).

The methodology used by the authors to generate the VDRs of an image can be described in four steps:

1. Identify the key objects in the scene from image regions mentioned in scene descriptions.

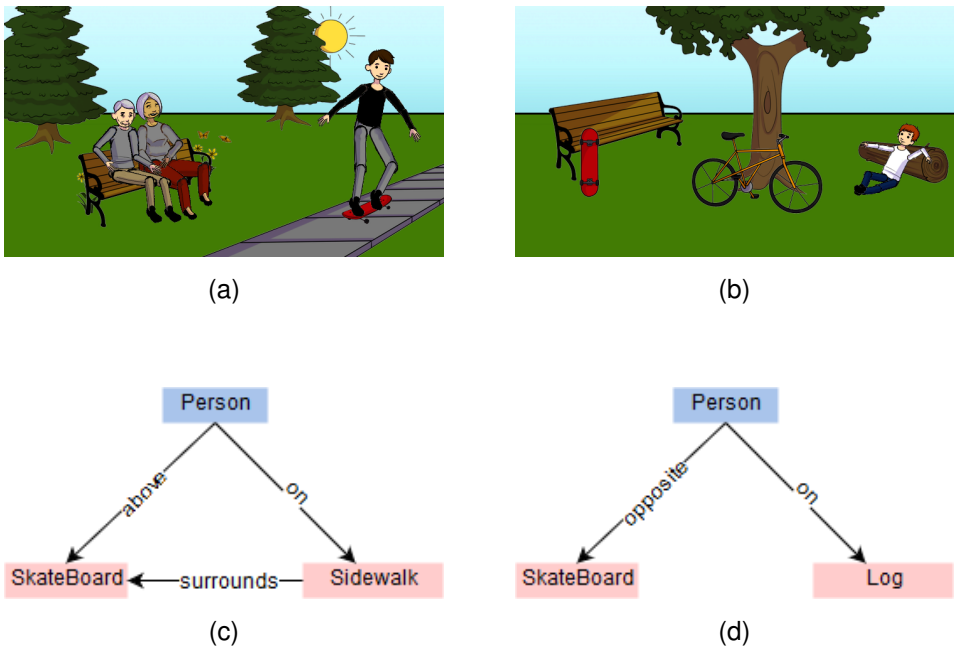


Figure 3.1: An example showing how VDR helps distinguish between scene objects that are interacting and the ones that simply co-occur.(a) shows a boy riding a skateboard and (b) shows a boy relaxing against a log, (c) and (d) show subgraphs from the VDRs of the respective scenes. Even though the scenes have same objects, i.e., a skateboard and a boy, the different structure dependencies help understand the actions being depicted. In (d), the boy is opposite the skateboard, so he cannot be riding it.

2. Start the VDR graph with the ROOT node which represents the image. To this, the main actor is attached.
3. Objects that are directly related to the actor in the description are then attached to the actor node.
4. All remaining objects are considered background objects that get attached to the ROOT node.

This results in a VDR structure that is a simple tree. The VDR tree is aligned to the Dependency Parse Tree (DPT) of the image description that was used to generate the VDR. These alignments enable the VDRs to exploit the isomorphism between image and linguistic structure, which is crucial to cross-modal tasks. To generate an image description, object pairs that occur in a parent-child relation in the VDR are used to generate verbs during a depth-first traversal of the VDR.

3.2 Abstract Scenes VQA Dataset

We introduced the Abstract Scenes VQA dataset (Antol et al., 2015) in Chapter 2. In this section, we will discuss the annotations available in the dataset. The VQA dataset contains 50K scenes with training/validation/test splits of 20K/10K/20K scenes respec-

tively. The scenes were generated using a fixed set of over 150 distinct clipart objects. The objects include 20 human clipart models from three age groups and 31 birds and animals in varying poses. Given this diversity of objects, the scenes in this dataset are more realistic and depict more complex relationships between objects, compared to the original Abstract Scenes dataset (Zitnick & Parikh, 2013).

Object Annotations

In the VQA dataset, all objects are annotated with a set of visual features (Table 3.1). Hence, we can avoid the use of object detectors. All objects have an associated pose/-configuration type, category and encoding of the direction they are facing. The human models are additionally annotated with facial expressions and details of the pose of each body part (present as the angle and position of the each joint, relative to the center point). The depth information of objects can be inferred from their clips' scaling factor (encoded 0 to 4). The objects do not have bounding boxes annotations.

Feature	Description
Idx	number particular clips (objects) present
name	clip ID
$type$	object category: human, animal, large, small
x	position along the x axis
y	position along the y axis
z	position along the z axis
$flip$	the direction the object is facing
poseID	object pose (animate) or configuration (inanimate)
numPose	total number of poses the object can assume
$deformable$	whether the object is deformable, note that this is only $True$ for human models

Table 3.1: A description of object annotations present in the abstract scenes VQA dataset.

Linguistic Annotations

Each scene is annotated with five natural language descriptions and three questions. The questions in the dataset can be split into three major groups of roughly equal size: yes/no, number and others. Each question has associated answer annotations for the two types of QA tasks (multiple choice and open-ended). For the multiple-choice type, there is a set of twenty possible options from which the answer is chosen while for

Image Descriptions:

```
The lady chases the cat who is chasing the mouse through the house.
The woman is chasing the cat walking on the carpet.
The cat is chasing the mouse across the rug.
The woman is chasing the cat who is chasing the mouse.
A woman runs after a cat that is chasing a mouse.
```

Question-Answers

```
Q1:  Is there a rat in the image?
A: yes
Q2:  How many books are above the cat?
A: 2
Q3:  What is the woman on the left doing with the dog?
A: chasing it
A: nothing
A: there is no dog it's cat
```

Figure 3.2: Sample image showing the linguistic annotations present in the Abstract Scenes VQA dataset. We can see questions from the three main categories: yes/no, numbers and other. Note that for brevity, we have only included the unique answers and not all ten answer annotations.

open-ended type, there are ten gold standard answers per question. In total, this results in 1500K questions and 15 million free-form answers. Fig 3.2 shows an example of a scene from the dataset and the associated linguistic annotations.

We chose to work with abstract scenes as opposed to real-world images, because identifying objects and their attributes in an image is still a challenging CV problem, and

despite all the recent breakthroughs, there is no perfect solution for it. Abstract Scenes allow us to skip this feature extraction step which eliminates a lot of noise. Moreover, scene perception research from cognitive psychology shows that the absence of photo realism does not effect how humans process and understand scene semantics (Heider & Simmel, 1944). Thus, using this dataset, we can focus on the VQA task in isolation and examine how incorporating image structure affects the model's understanding of subtle nuances in scene semantics.

3.3 Designing an augmented Visual Dependency Representation

To add structural information to the VQA model, we propose an augmented Visual Dependency Representation based on Elliott and Keller's work (Section 3.1). Our aim in developing this representation is to enable the model to identify *all* the important object interactions in a scene.

3.3.1 Why do we need to augment VDRs?

Our reasons for redefining what the VDRs capture are based on the differences in the nature of the VQA task and the dataset. The main difference in VQA and other CV/NLP tasks is that in problems such as description generation, the question that needs to be answered by the algorithm is predetermined (we can think of the generated description as an answer to the question *What is happening in the scene?*). During test time, only the input image changes. On the other hand, for the VQA task, the form that a test question can take is not determined until test time. Moreover, the question can refer to any object in the scene. As a result, VQA requires a more detailed understanding of the scene along with information about the scene objects and their relationships.

As mentioned in Chapter 2, VDRs have been used for image description generation and image retrieval. Intuitively, these tasks involve identifying and describing the main event being depicted in a scene. Recall that the existing VDRs only identify a main actor in an image and the objects it is interacting with (Section 3.1). Thus, although this approach provides sufficient data to describe or retrieve images, it is not enough to answer questions about images. For instance, in Fig 3.2, to answer a hypothetical question 'Where are the books?', we need the book node in the VDR to be connected to the bookshelf. The existing approach will not record this relationship since neither the book nor the bookshelf is an actor, or involved in an interaction with any actor in the scene.

The second reason for augmenting VDRs is that the VQA dataset often contains more than two human models in a single scene. The existing approach does not handle this scenerio because the VLT dataset (Elliott & Keller, 2011) used by the authors contained more than one actor in only 14% images. These images were simply ignored

because they wanted the VDR graph to be a tree. In the Abstract Scenes dataset used by Ortiz et al. (2015) the number of human objects in a scene was limited to a maximum of two. However, for the VQA dataset, using this approach would cause us to lose a lot of vital information. Therefore, we want a structured representation that is able to capture all the interactions that are depicted in any given scene. The resulting structure is no longer a tree, but a directed acyclic graph.

3.3.2 Defining *important* relations

Based on the discussion in the previews section, the VQA task requires a rich image structure information. The simplest strategy to achieve this would be to encode the spatial relations between all the scene objects. However, including the spatial relations between every possible object pair would add too much noise and the model will not be able to generate relevant answers. This is a common problem faced in the field of Natural Language Generation, where we need to decide— from all the possible information, which is relevant and should be included in the output. This is known as *content selection*. Similarly, for VQA, we will need to identify spatial relationships that are informative and contribute to meaning of a scene.

When humans look at an image, what are they interested in? Research in scene perception has tried to answer this question using various methods such as studying how the brain represents scene objects, the scan patterns people follow while viewing scenes, how they describe scenes, and so on. A number of visual and linguistic features have been shown to influence attention allocation and scan paths. For instance, factors such as position, saliency, animacy, and semantic proximity affect participants' responses in an object naming task (Clarke, Coco, & Keller, 2013). An intelligent question-answering model needs to be able to identify scene features that might interest a person. We will design a set of heuristics that can find which spatial relations should be encoded in a scene VDR. In this section, we discuss how we can use object and Scene features to establish which relations in a scene are important.

3.3.2.1 Animacy

Animate objects rapidly draw viewers' attention during scene perception (Rayner, 2012). Therefore VDR relations that involve an animate object will be considered important. Intuitively, this makes sense because understanding what is happening in a scene involves inferring the actions and intentions of the actors.

3.3.2.2 Position

Literature on perception and sensorimotor control categorises the space around the body into distinct regions (Previc, 1998). These regions are illustrated in Fig 3.3). The space immediately surrounding the body is known as the *peripersonal space*

(Rizzolatti, Fadiga, Fogassi, & Gallese, 1997). It is the seat of all body-object interactions, where objects can be grapsed and manipulated. Beyond this region is the *extrapersonal space*, and objects present here cannot normally be reached without moving toward them. The neural representation of objects situated in peripersonal space varies from that of objects in the extrapersonal space (e.g. Halligan & Marshall, 1991). Based on this, we can assume that relations between actors and the objects which are close to them, are important.

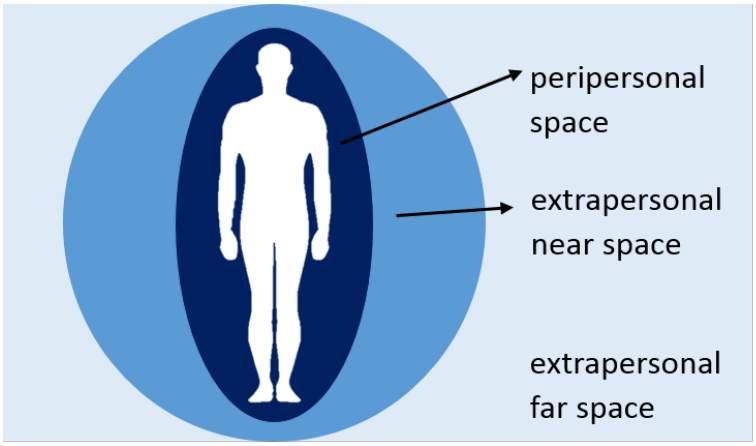


Figure 3.3: Neuropsychological literature classifies space into distinct regions based on how humans perceive spatial environment with respect to themselves. These regions determine the possibility of functional interactions with objects.

3.3.2.3 Semantic Similarity

The effect of scene semantics on the fixation location and duration is well established in the eye movements literature (e.g. Henderson & Hollingworth, 1999). In visual search tasks, attention is guided by the expectations about the target's likely position (Brockmole, Castelhano, & Henderson, 2006). For instance, in indoor scenes, a food item is more likely to be found on a plate than on a bookshelf. Therefore, we will consider relations between semantically similar objects to be important.

3.3.3 Augmenting Visual Dependency Grammar

As mentioned in Section 3.1, the VDG defines a set of spatial relationships that determine the dependency relation between two objects in a scene. These relationships are based on three geometric properties of the object regions- pixel overlap of bounding boxes, angle and distance between the centroids of the objects. Together, these properties determine the label that defines the spatial relation. To create an augmented VDR, we use findings from neuropsychological literature on 3-D spatial interactions (refer to 3.3.2.2) and redefine relations in Elliott and Keller's VDG. The augmented VDG will incorporate two additional image properties:

1. **Distance:** The existing VDG uses distance measures to distinguish relations along the x axis into two types: *beside* and *opposite*. We would like to extend this distance based thresholding to define discrete regions of space around all objects (Fig 3.4) based on the discussion in Section 3.3.2.2. The value of threshold changes according to the z-axis location of the objects so that the perceived spatial relations correspond to real world depth perception. This will result in finer grained spatial relationships, and thereby help distinguish the objects an actor is possibly interacting with from the ones that happen to be nearby.

2. **Depth:** The image descriptions in the VLT dataset on which VDRs were developed contained two sentences which describe the main objects and background objects separately. We will use the depth of the objects (determined using the scale of the clip) to distinguish background objects from the foreground objects. The z axis relations are concatenated to the XY relations which makes the VDR annotations 3-dimensional.

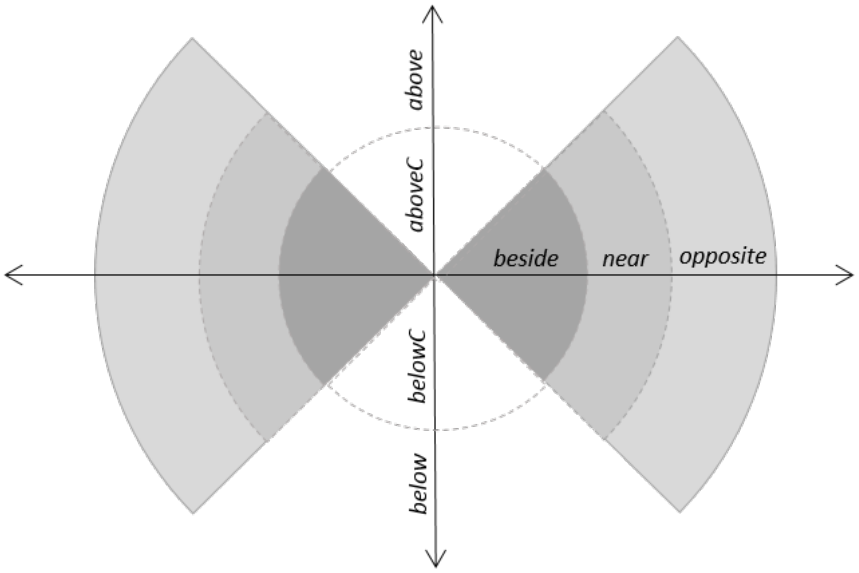


Figure 3.4: A visual explanation of how the angles and distances determine the spatial relation between two objects. The augmented VDG uses distance based thresholds to define fine-grained spatial relationships in the xy-plane.

The augmented VDG is presented in Table 3.2.

3.4 Methodology

In this section, we will describe the implementation of an automatic VDR generator that uses multimodal features to create structured representation of scenes.

Relation	Description
A *surrounds* B	The entire region B is overlapped by region A
A *on* B	70% of region A overlaps with region B
A *beside* B	Angle between centres of A and B is between 315 and 45 or 135 and 225 and A and B are close together ($d < \theta_{near}$)
A *near* B	same as *beside* but A and B are near ($\theta_{near} < d > \theta_{far}$)
A *opposite* B	same as *beside* but A and B are far apart ($d > \theta_{far}$)*
A *above* B	Angle between centres of A and B is between 225 and 315 and A and B are near ($d < \theta_{far}$)
A *aboveC* B	same as *above* but A and B are close ($d < \theta_{near}$)
A *below* B	Angle between centres of A and B is between 45 and 135 and A and B are near ($d < \theta_{far}$)
A *belowC* B	same as *below* but A and B are close ($d < \theta_{near}$)
A *infront* B	difference in z axis location of A and B is >1
A *behind* B	difference in z axis location of A and B is >-1

Table 3.2: Augmented Visual Dependency Grammar: distance thresholds θ_{near} and θ_{far} are used to subcategorise the relations in the xy-plane. z-axis relations are defined in addition to the xy-relations.

3.4.1 Data Pre-processing

The preprocessing steps outlined here are for both the VDR Parsing and VQA task.

3.4.1.1 Parsing Linguistic Annotations

The scene descriptions and question annotations were cleaned up by fixing spelling errors and removing unnecessary punctuations such as periods and question marks present in the middle of the sentences. These annotations were then parsed using the Stanford CoreNLP generator (Manning et al, 2014) which contains a set of natural language analysis tools. We used it to we obtain Part of Speech (POS) tags, word lemmas and universal dependencies for each sentence. As we will discuss later, these annotations were used in several steps of both the VDR generation and VQA pipeline.

Additionally, we also carried out constituency parsing on questions. The resultant phrase structure annotations were used to convert the question-answer pairs into declarative statements. Using the constituency annotations, as opposed to POS tags, allows to define a handful of rules that can be generalised to question with different lengths and POS tag sequences. The number of rules needed is further limited because the questions begin with a finite set of phrases. The conversion takes place using a two

step template matching algorithm.

1. **Prefix generation:** In this step, a question is converted to a statement with a blank. The constituent phrases of the question are matched against a set of pre-defined templates. We traverse the tree leaf to root, looking for the top-most level phrase that matches the next item in a given template. The constituent phrases are then rearranged based on the rule for that template 3.3. Next, we remove verbs such as 'doing','trying' that do not convey any information. The generated prefix contains an empty placeholder for the answer. This step is illustrated in Figure 3.5

2. **Concatenating answers**: The next step is to fill in the empty placeholder in the prefix. Most question were short and simple, so we found that the answers usually attached at the end of the prefix 85% cases. Note that this step does not take place for the yes/no questions. If an answer if no, we will simply ignore the statement during training for the VQA task.

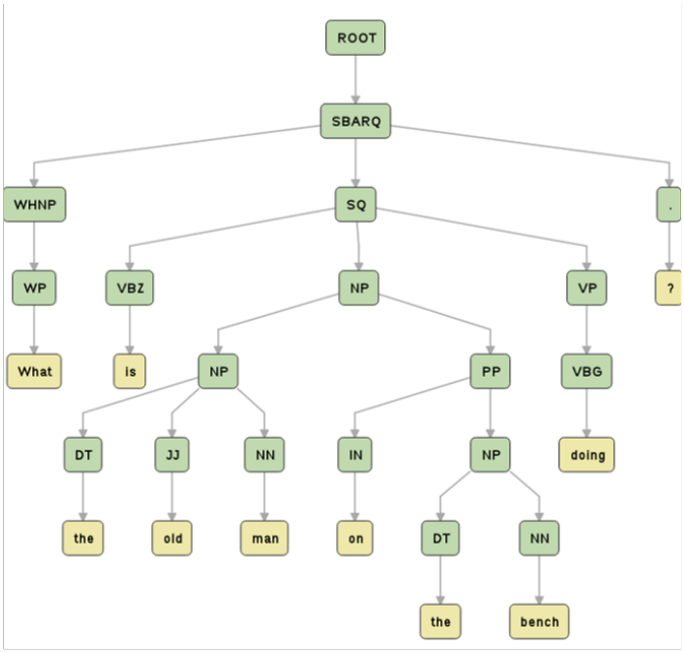
Question Template	Generated Prefix
WH VBZ NP VBG (PP) ?	NP VP VBZ VBG $ (PP) .
WH VBZ NP PP VBG ?	NP PP VBZ VBG $.
WH VBZ NP PP IN NP VP ?	NP PP IN NP VP VBZ $.
WH VBZ PP ?	Det $ VBZ PP .
VBZ/VBP NP (PP) ADJP/VP ?	NP (PP) VBZ/VBP ADJP/VP.
VBZ/VBP NP PP ?	NP VBZ/VBP PP.
VBZ/VBP NP ?	NP .

Table 3.3: Templates and conversion rules for questions based on the constituency parse annotations. We defined the 'WH' tag to include tags of all WH-words that the questions can begin with. The placeholder for the answer is denoted by '$'.

3.4.1.2 Extracting Visual Features

We did not use a CNN to learn visual features of a scene. Instead, we annotated the scenes with visual features that were either not present in the dataset (bounding boxes) or had to be inferred using object attributes (age, gender).

We obtained the bounding boxes of scene objects using the size of the corresponding clipart file. Since human clipart models are deformable, their image files are present as separate parts, so we defined a constant height and width for 8 major poses. This value was scaled differently for adult, child and baby models. Additionally, we scale down the size of the bounding boxes along the z axis to account for variations due to perspective in depth perception. As Figure 3.6 shows, this estimation technique, while not perfect, is fairly robust.



Input	What is the old man on the bench doing?
Template matched	WH VBZ NP PP VBG ?
Rule applied	WH VBZ NP PP VBG ? ⟶ NP PP VBZ VBG $.
Prefix generated	The old man on the bench is $.

Figure 3.5: An overview of the prefix generation step during the conversion of questions to declarative statements.

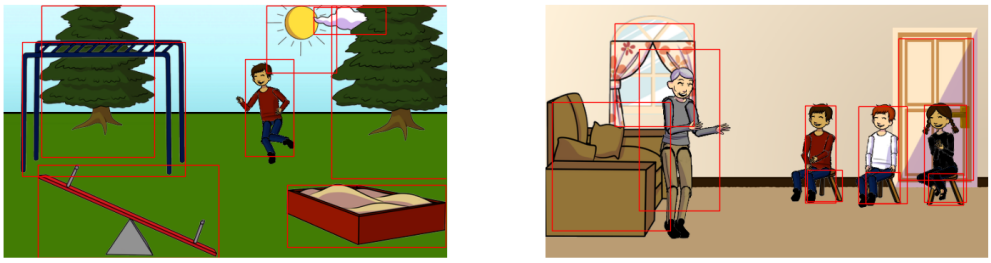
We manually added two additional annotations to the 20 human clipart models, gender and age category in order to resolve object ambiguity. The lack of these annotations would make it distinguish which scene objects the nouns in the descriptions or questions. For example, if a scene contained a *man* and a *woman*, it would not be possible to distinguish between the two without using additional annotations.

The step is to create a database of image object along with their relevant attributes (3.1) from the raw scene renderings. We did not include the details of the pose of the human models such as position and orientation of each limb/ joint. Instead, we will just use the pose ID of the model that refers to the one of the eight predefined poses (that is, before the clip was modified during the creation of the scene).

3.4.1.3 Object Grounding

Object grounding refers to the alignment between scene regions and the image descriptions. These groundings are key to determining the inter-modal correspondences between the visual and linguistic features as they will be used for aligning scene VDRs to the dependency trees of corresponding scene descriptions.

The nouns present in linguistic annotations were clustered based on their similarity (obtained using WordNet similarity measures) to the objects present in the dataset.



(a) (b)

Figure 3.6: Sample images showing the estimated bounding boxes of scene objects.

This generates a set of vocabulary that can be used to refer to any object in the scenes. Using this, generated a mapping between each object in each scene and the nouns from the scene's descriptions.

3.4.2 Generating VDR Graphs

This section describes how scene VDRs are generated. We start by applying the augmented VDG to each pair of objects present in a scene. The algorithm uses three rules to determine the order of objects while calculating the dependencies to avoid cycles in the VDR. These are:

- animate objects are always chosen as the head of a dependency. Where both objects are animate the preference is given to human objects.
- pairs involving a large and small object, the large object is always the head
- if VDR contains $A \rightarrow B$, then $B \rightarrow A$ is not added to the VDR.

Recall from Section 3.3.3 that the VDG uses the four geometric properties to calculate the dependency label: distance, angle, pixel overlap and depth. The algorithm used to calculate these properties closely follows the approach used in prior work on VDR generation.

- **Angle between regions:** This measure defines the xy-relation of the objects. The angle between the centres of two objects is used to apply the *above*, *below* and *besides* relationships.
- **Overlapping regions:** The amount of overlap between object calculated using the ratio of the intersection of their bounding boxes to the union of the bounding boxes. This measure is used to apply the *surrounds*, *on* relationships.
- **Distance between regions:** The relation obtained from the angle is converted to a more fine grained relation based on the Euclidean distance between the centres of the object. The relations *above* and *below* get converted to *aboveC* and *belowC* respectively if the distance is less than the thresholds set. Similarly, *besides* changes to *near* or *opposite* if the distance exceeds their respective thresholds. The distance thresholds decrease with an increase in object depth.

- **Relative depth of regions:** The difference in the z-axis location of the objects is used to apply additional z-relations *infront*, and *behind*.

This step generates a fully connected graph that contains all possible spatial relations for each scene.

Pruning VDR graphs

As discussed in Section 3.3.2, we do not need to know the spatial relation between every object in a scene to understand the meaning of a scene. The fully connected graphs VDR graphs are pruned to remove relations estimated to be of low importance to the scene.

Recall that we designed a set of heuristics in Section 3.3.2 to recognize object interactions that are not mentioned in the associated scene description, but still contribute the the meaning of a scene. The pruning algorithm applies these heuristics by defining a set of conditions that need to be met in order to keep an arc. These truth value of these conditions is used to assign a score of 0 or 1 to each dependency arc in the scene VDR. Arcs that are classified as not important, i.e, score 0 are removed. The implementation of the heuristics and their use in pruning the VDRs are summarised below.

Before we can determine which values of the heuristic functions generate a score of 1, we need to rank the values the functions can take. This was done as follows:

1. **Distance** This heuristic ranks the spatial relations such that, other things being constant, a relations corresponding to smaller distances are ranked higher, in the following order: `surrounds, on > beside, aboveC, belowC > near, above, below > opposite`.
2. **Depth** For the z-relations, object pair the same z-region (defined as $\pm$ 1) gets a higher rank: `same > infront, behind`.
3. **Similarity** Originally implemented using WordNet similarity measures, we changed our approach because an analysis of the WordNet similarity scores showed unexpectedly high similarity measures for some object pairs such as *man* and *bush*. Similarity is now defined by comparing the object category annotations present in the dataset. Object pairs that belong to the same category, or are the same, are more similar and get higher ranks. Based on the categories of the source and target objects, the order of the ranking is: human-human $>$ human-other $>$ animal-animal $>$ animal-other $>$ other-other.

Pruning Strength An additional parameter, pruning strength was added to the model after experimentation. Pruning strength controls how aggressive the pruning is. The rules used to score arcs can be varied to change the strength of the pruning algorithm. The factors that control the strength are:

1. **Presence in Description** Keeping relations between all object pairs mentioned in the scene descriptions resulted in noisy VDRs. Instead we defined a heuristic

that gets activated when the source and target objects are both present in any of the 5 scene descriptions. This feature controls how aggressive the pruning is, which effects the criteria that need to be met before an arc can be classified as relevant. For instance, the pruning strength van be reduced by relaxing the *distance* heuristic to increase the rank for `near` relations. Thus, if a pair of objects occurs in the scene descriptions, it will bias to that relation during pruning. It will not, however, guarantee that the relation does not get pruned.

2. **Scene Density** The scenes in the dataset have a large variation in term of object density. The number of objects in a scene n can vary from anywhere between 3 and 20. As a result, using the same pruning strength for these scenes will result in very sparse or very dense VDRs. To control for the scene density, dense scenes ($n > 12$) are pruned more aggressively.

The use of this dual pipeline ensures that the extra relations added to the VDR are actually important.

To prune a VDR graph, the algorithm combines the relation rankings from different heuristic functions and defines a set of rules that determine which arcs are removed. These rules were determined experimentally and depend on the category of the source and target objects (similarity heuristic) and the xy-relation (distance) and z-relation (depth).

To see how the algorithm works, consider the following hypothetical dependency arc

$$Dog \rightarrow Table, \text{ label} = belowC$$

lets assume that the objects occur in the scene description. In Table 3.4, the arc matches the rule in row 4 so it gets a score of 1. Consider a second scenario where the objects are present in different z-regions. If the dependency label was changed to $belowC\_infront$, the arc's closest match would now be row 1. However the corresponding rule does not contain $XYrel = belowC$, so the arc would score 0 and get pruned. Intuitively, this means that if the dog is in front of the table, there is no interaction between the two even if they appear to be close on the y-axis relation. In contrast, when they were located on the same z-region (as was the case in the former scenario), it could imply that the dog is sleeping or hiding under the table, so the relation would be important.

Sigma The generated VDRs were visualised using a javascript based library Sigma. The nodes in the graph represent the scene objects and the arcs correspond to the inferred dependency relations. The position of the nodes corresponds to the real location of the objects in the scenes and the size of the node reflects the encoded depth of the objects.

3.5 Results & Discussion

In this section we will present and review the results obtained in the automatic VDR generation task. We will also discuss the evaluation process and analyse the perfor-

SrcCat	*TgtCat*	*XYrel*	*Zrel*
all	all	*on*, *surrounds*, *beside*	all
human	*animate*	all	all
human	all	*aboveC*, *belowC*, *above*, *near*	same
animal	all	*aboveC*, *belowC*, *near*	same
all	all	*on*, *surrounds*, *aboveC*, *belowC*	same
human	all	*beside*, *above*	same
animal	all	*beside*	same
human	*human*	all	same

Table 3.4: Combinations of feature values that result in a score of 1 for pruning with moderate strength(top table) and high strength (bottom table). The feature values specified for different object categories are added to the values defined for all categories.

mance of our approach.

3.5.1 VDR Generation

We generated a silver standard set of augmented-VDRs for the Abstract Scenes VQA dataset. The training set VDRs (n= 200K), the generator produced a total of 1014K relations which were pruned to approximately 162K. The average pruning rate was 83.9% which resulted in an average of 8 dependencies per scene. Note that the scenes vary a lot in object density and hence the VDR density also has a large standard deviation.

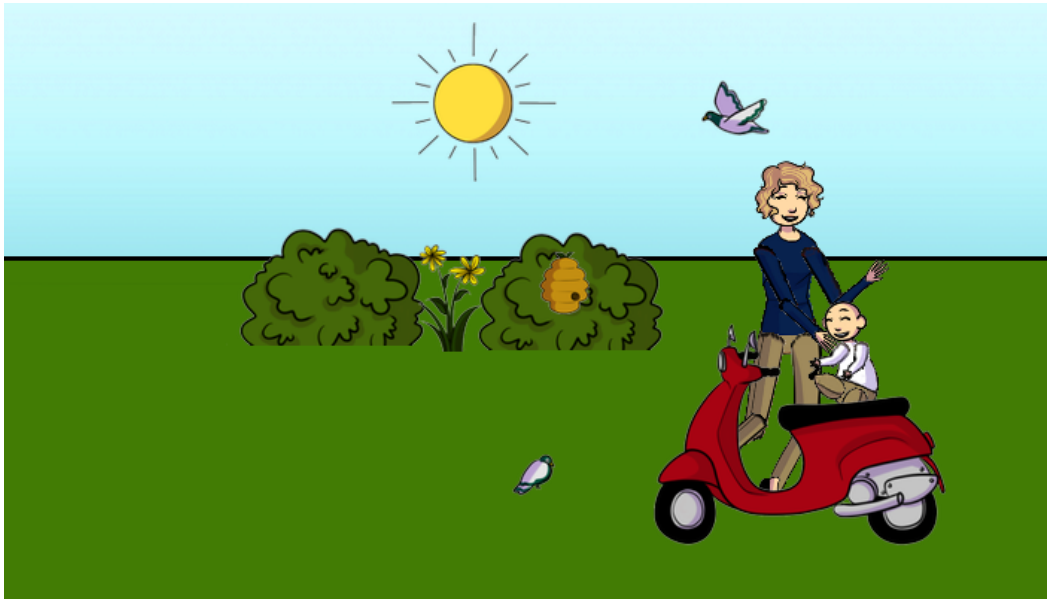
Figure 3.7 shows a sample VDR generated by the algorithm. A more comprehensive set of examples can be found in Appendix A where we show how the different types of scenes and handled by the algorithm.

In the next section, we do a quantitative as well as a qualitative analysis of the generated VDRs.

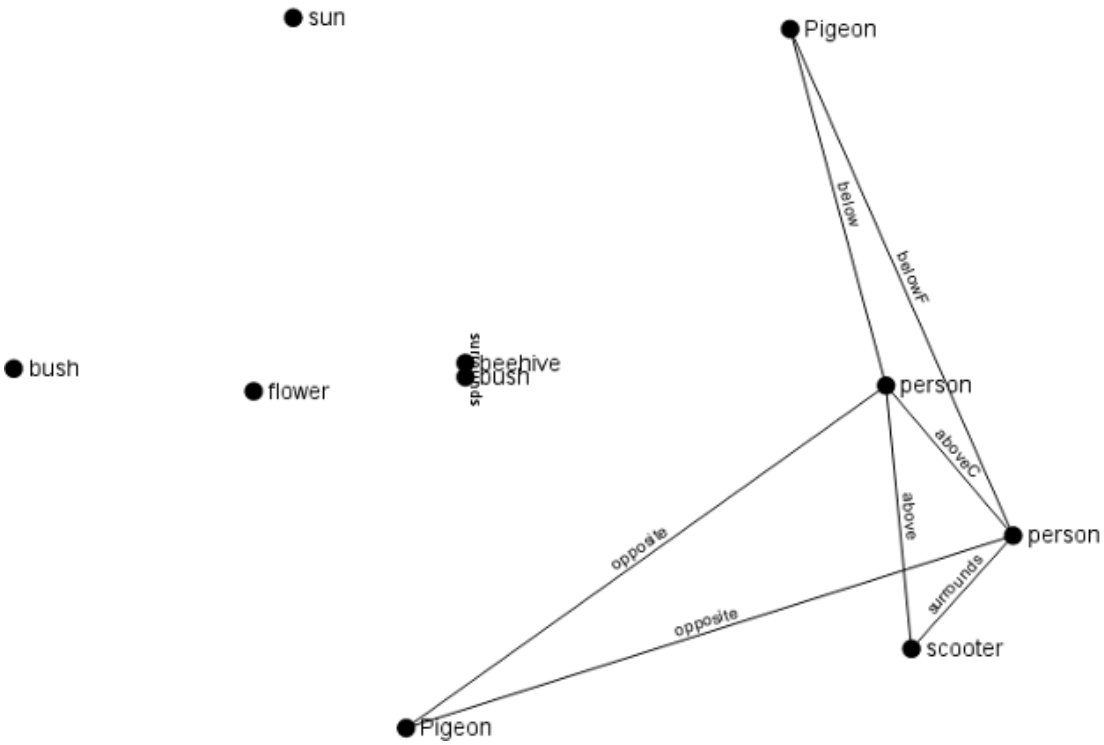
3.5.2 Evaluating Silver Standard VDRs

The absence of gold standard VDRs makes it impossible to do automatic evaluation of the generated VDR set. Instead, we evaluated the performance based on human judgment.

A subset of the pruned VDR graphs was manually evaluated by two people. To carry out the evaluation, we defined a set of guidelines using the criteria set out for creating gold standard VDRs by Elliott and Keller (2013). The aim of this evaluation is to



(a)



(b)

Figure 3.7: A sample scene from the Abstract Scenes VQA dataset and its corresponding output produced by the VDR generator. From the VDR we can see that the algorithm has identified all interesting relations in the scene. Thus the augmented VDRs can depict all that happening in a scene.

determine whether the generated graphs are correct (that is, the VDR relations are correct) and complete (identifying if important relations are missing). The guidelines used for VDR evaluation are presented below.

3.5.2.1 Method

For each scene, go through every relation in the VDR graph. Each arc is scored in a 2 step process. First, we determine whether the arc is correct, and second, whether the dependency label is correct. The criteria for correct arc, correct label and missing arc is defined below:

- Correct Arc: A correct arc gets a score of 1. An arc between two nodes is considered correct if it meets one of the following conditions-
  - there is a visible interaction between the objects based on the image
  - there a relationship between objects based on the scene descriptions
- Correct Label: If the relation label between the two objects is correct according to the VDG, then the label gets a score of 1. Note that in some cases, even labels that follow the grammar might actually be incorrect based on the scene because the depth information and bounding boxes are not perfect.
- Missing Arc: For every object node in the graph, decide if there should be an arc to other objects based on the scene descriptions or any visible interactions among the scene objects. All missing arcs are listed with a score of -1.

3.5.2.2 Evaluation Results

A subset of 50 scene VDRs, containing a total of 369 relations was manually evaluated. The results are presented in Table 3.5. 91.8% relations were correct. There were 1% irrelevant relations present in the VDRs. We identified 6 object pairs whose relations were missing in the VDRs even though the objects occurred in corresponding scene descriptions. Additionally, the VDRs for 2 scenes did not contain any relations. Overall, the algorithm generates scene VDRs with a precision of 90.4%.

	Arcs	Labels	Relations
Correct	97%	91.8%	91.8%
Incorrect	1%	7.2%	8.2%
Missing	1.6%	-	1.6%
Inter-annotator agreement	98%	92.5%	

Table 3.5: VDR Evaluation results, averaged over two independent human evaluations.

3.5.3 Error Analysis

Subsequent analysis of the incorrect arc labels showed that 6.9% of these errors were due to incorrect depth information, that is, instances where the object scaling did not reflect the depth, but the clip was just scaled to change the object size. The remaining errors were because of incorrect bounding box annotations of human objects and some large objects such as trees. Given that we had estimated bounding boxes of the human models based on a set of 8 poses, we have reason to expect some error in the estimation process because the limbs in the clipart models were deformable to allow for a continuous variation in pose. In contrast, for large objects like trees, sidewalk- that are not rectangular in shape, using rectangular bounding boxes provides incorrect information regarding the region they occupy. Recall that the pixel overlap between two objects was calculated based on their bounding box annotations. Where the bounding boxes contain large amounts of empty space, it often results in a wrong spatial relation. Fig A.4 shows a few examples of such cases where the VDR generator produces the wrong spatial label. For instance, the football is under the bars but the VDR relation generated is monkeybars *surrounds* football.

We also observed the depth confound in the case of missing relations. Ideally, we would expect interacting objects to be present in similar z-axis locations. However, as we pointed out earlier, some objects have been scaled to achieve a different size, and not to reflect a change in depth. In these cases, the generator misclassifies a foreground object as a background object. Fig A.6 shows such scenario. Based on the depth encoding, the jump rope is in the background and logically, the girl cannot be skipping a rope that happens to be in the foreground. Similarly, extra relations also occurred due to the incorrect depth information. In this case, background objects can get misclassified as foreground objects and the generator assigns a higher score to that relation because of the possibility of an interaction (Fig A.5).

3.6 Conclusion

In this chapter, we presented an approach to generate silver standard VDRs for the Abstract Scenes VQA dataset. This approach used heuristics inspired from scene perception literature to score the arcs in a VDR graph. As the results show, the generated VDRs are good at capturing the relevant image structure using a combination of features from the visual and linguistic modalities. Further, the high accuracy of pruned arcs suggests that the algorithm can predict which relations in a scene are important.

We saw that there is a small amount of noise present in the VDRs due to incorrect depth and bounding box annotations. A way around the first problem would be to use a combination of both y-axis and z-axis values to determine the depth of an object. However, this approach is not foolproof either. For the second issue, the solution would be use polygon bounding boxes using an object detector. Since we already know the identity and location of objects, we can assume that we will still be free from the noise that automatic object detectors introduce in photo-realistic datasets.

Overall, the VDR generator gave good results and we have completed our aim of generating richer VDRs for the Abstract Scenes VQA dataset . Next, we will use the generated VDR set for the VQA task.

Chapter 4

Visual Question Answering

In Chapter 3, we described the process of modeling image structure using augmented Visual Dependency Representation (VDR) and how to automatically create silver standard VDRs for the Abstract Scenes VQA dataset. We will now use these VDRs for the task of Visual Question Answering to test our hypothesis that encoding image structure would improve performance on the VQA task.

As seen in Chapter 2, prior work on VQA uses a combination of different strategies such as low-level computer vision, deep learning and common sense reasoning to generate answers (Section 2.4). In this chapter, we present an classification based approach to VQA task, that uses the spatial dependencies of a scene to infer relationships among the scene objects. We will evaluate this model on a subset of open-ended questions that rely on image structure to generate the answer, and not on object attributes or linguistic annotations alone.

4.1 Question Annotations

The main aim of Antol et al. (2015) in creating the Abstract Scenes VQA dataset was to have questions that would be difficult for an AI system to answer. Although the annotators were instructed to generate questions that require the image to answer the question, one of the main criticisms of the VQA dataset is that it has strong language priors (Goyal, Khot, Summers-Stay, Batra, & Parikh, 2016). This is evident from the 'blind' or language-only VQA model (Antol et al., 2015) which gives an accuracy of approximately 51% without including associated images during training. Therefore, we will evaluate performance on questions that require knowledge of image features to generate a correct answer. Further, for this project, we are particularly interested in questions that rely on the image structure (e.g. *What is the girl doing?*) and not on low level visual attributes such as the colours of an objects. This allows us to focus on evaluating the usefulness of scene structure for VQA.

Question Types

In Chapter 3, we discussed the visual and linguistic annotations present in the Abstract Scenes VQA dataset. To recap briefly, each scene is annotated with three questions and each question has ten answers, with a total of 150K questions and 1,950K answers. For the open ended questions, the task is to generate a natural language answer, which can range anywhere from a single word to short phrase.

We will begin by analysing the question annotations in the training set. Fig 4.1 shows the distribution of different types of questions present, where the questions have been clustered based on their starting words ((Antol et al., 2015)). About a third of the questions (question types: *Is...*, *Are...*, *Does...*) require a binary answer. While it might seem trivial, generating a yes or no response often involves using inference based on object attributes or common sense knowledge to verify the question content. Another 30 % of the questions require low level visual information such as the number or colour of objects (question type: *How many...* and *What colour is...* respectively) to answer them. Question content can be determined by analysing the objects and actions they refer to. The 60K questions in the training set contains approximately 565 distinct verb and 1232 distinct noun (after stemming the words and filtering out those with a count of 1). Fig 4.2 shows the frequency distribution of top 25 nouns and verbs in the training set questions. We can see that most often, questions talk about animate objects. The most frequent actor-object interaction about which questions are asked is *sit*.

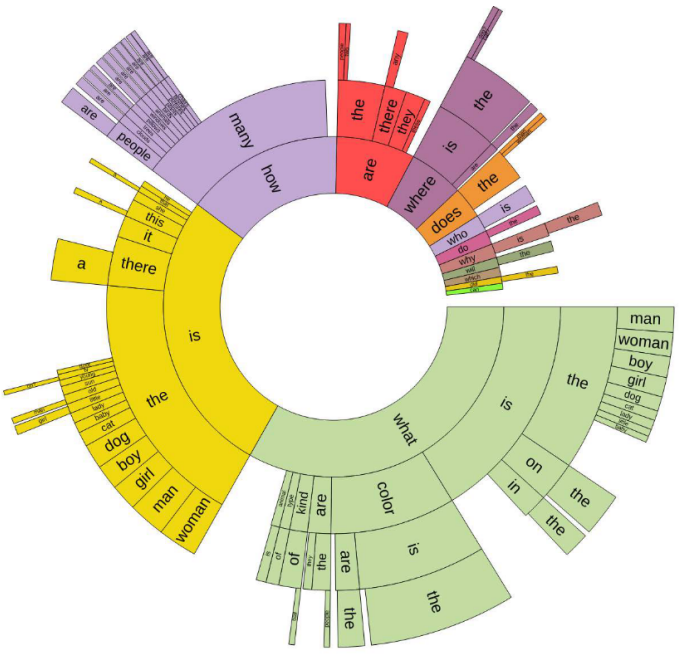
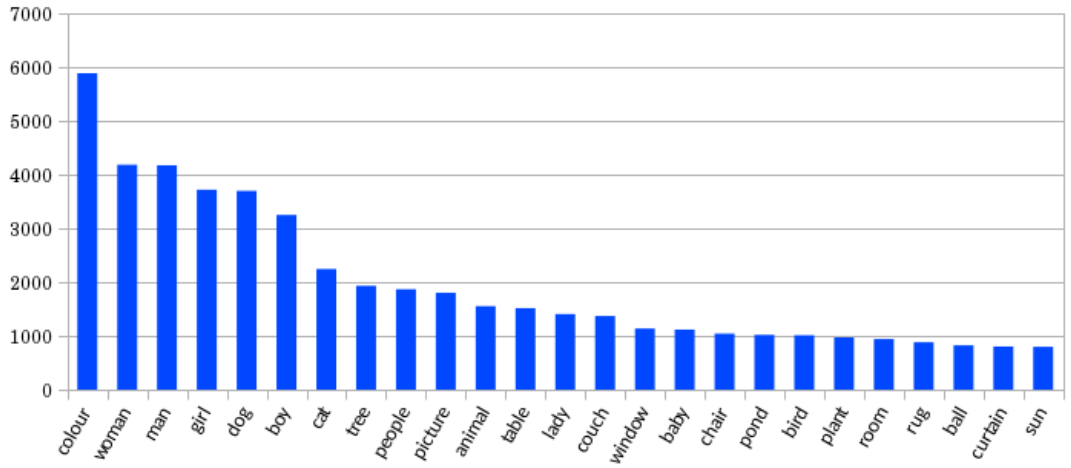


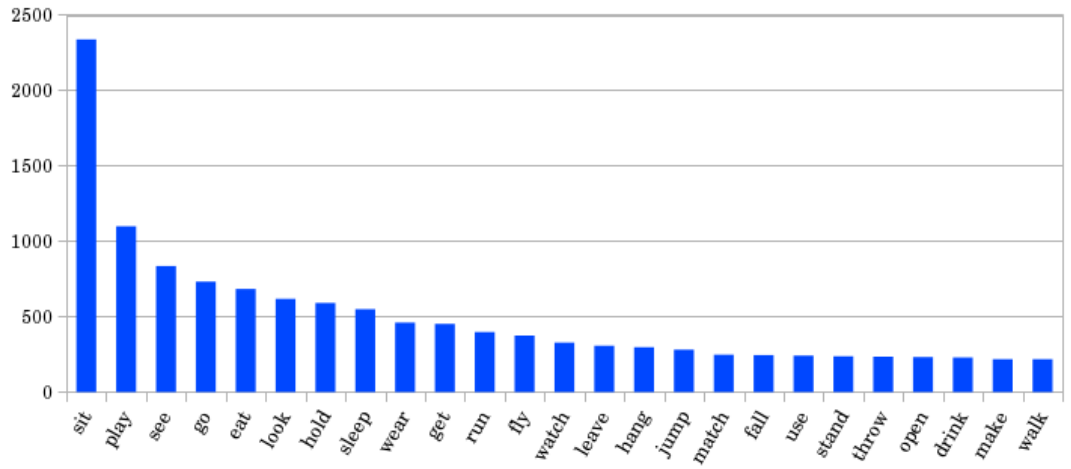
Figure 4.1: Distribution of questions based on their type (Antol et al., 2015). The question types determine the kind of answer that should be generated- binary, number or other (word/phrase).

Based on the general distribution of verbs, we can expect to see an improvement in the VQA accuracy for interactions such as *sit, play, eat* where we can expect specific

spatial relationships between objects. However, image structure encoding may not influence the performance for interactions such as *see, go, look, wear* that rely more on attributes such as pose or colour.



(a)



(b)

Figure 4.2: An analysis of question content based on frequency of (a) top 25 noun lemmas and (b) top 25 verb lemmas present in the training set. The auxiliary verbs such as *be, do, have,* etc have been filtered out.

4.2 Approach

In order to test the hypothesis that structured image representations are useful for Visual Question Answering, we present a model that uses a classification based approach to generate answers.

Elliott and Keller (2013) used VDRs to generate image descriptions in a two step process:

1. identify key image regions, i.e., where the objects are present
2. predict interactions between image regions present in a parent-child relationship in the scene VDR.

The interactions in the scene were predicted by estimating two probability distributions over verbs- first, conditioned on the subject, object and the label of spatial dependencies in the VDR set, and the second, conditioned on just the subject and object:

$$P(verb|o_{subj}, o_{obj}, rel_{subj,obj}) \tag{4.1}$$

$$P(verb|o_{subj}, o_{obj}) \tag{4.2}$$

where, the second distribution is used as a backoff if there is no arc connecting a pair of objects in the VDRs of the training set.

In Section 3.3.1, we briefly outlined the key differences between the VQA and image description tasks. We saw that generating an image description entails predicting the interaction between the scene actor scene objects. So, the task is to predict a verb given a pair of objects. In addition to this, question answering can also involve identifying the objects involved in the interaction. This means we need to estimate additional probability distributions over objects:

$$P(o_{obj}|o_{subj}, verb, rel_{subj,obj}) \tag{4.3}$$

$$P(o_{obj}|o_{subj}, rel_{subj,obj}) \tag{4.4}$$

The first distribution, which conditioned on verbs, will be used when the interaction refers to an action (e.g. *Where is the man sitting?*) while the second will be used when the interaction refers to a spatial relation (e.g. *What is under the table?*).

Problem Formulation

Since an answer can involve multiple components (object, verb, or both), we will not model the final answers directly, but the verbs and objects required to generate the answer. We formulate the problem of answering questions based on images as follows: given a set of features $\mathbf{x}$ that depend on the test scene, its VDR and the test question, predict the answer component y based on its parametric probability distribution, which can be defined as —

$$\hat{y} = \underset{y_j \in S}{\arg\max}\ P(y_j \mid \mathbf{x};\ \theta) \tag{4.5}$$

where, θ is the set of parameters the model learns and S is the set of fixed vocabulary from which the answer component is drawn. For the verbs, the vocabulary is the set of all the actions depicted in the training images while for objects, the vocabulary is restricted to the objects present in the test scene.

To predict objects, the probability distribution in equation 4.5 can be estimated using equations 4.3 and 4.4, which will be based on relative frequencies of the tuples involved.

To predict verbs, we will modify Elliott & Keller's approach because of two reasons. First, the scenes in the VQA dataset contain multiple actors and depict multiple interactions. As a result, our VDRs are richer and contain more complex relations than those used in prior work (Elliott & Keller, 2013; Ortiz et al., 2015). Second, as we saw in Section 4.1, predicting the interactions among objects requires the use of additional features such as pose, the direction the objects are facing in, and so on. We would like to condition the probability distribution on a larger set of set of visual and liguistic features as compared to equations 4.1 and 4.2.

Therefore, for verbs, the probability distribution in equation 4.5 will be estimated using a multinomial logistic regression (Maximum Entropy or MaxEnt) classifier. The MaxEnt classifier is a discriminative classifier; it assigns a class $y = c$ to an observation $\mathbf{x}$ by calculating the probability from an exponential function of a weighted feature set (Jurafsky & James, 2000). The probability of a verb belonging to class c can be defined as:

$$P(y = c \mid \mathbf{x}) \;=\; \frac{1}{Z} \exp \sum_i w_i \, f_i(c, \, \mathbf{x}) \tag{4.6}$$

where, Z is the normalisation constant, w_i are weights learnt by the model, features f_i are indicator features which are a property of both the observation $\vec{x}$ and the output class c. extracted from the input $\vec{x}$.

4.3 Methodology

Our VQA model uses a template based approach to understand the input question and a classification based approach to generate answers. Answer generation uses two components: a MaxEnt classifier to predict verbs and frequency estimates to predict objects. In this section, we describe the implementation of the different components of the VQA pipeline.

4.3.1 Extracting Subject-Object-Verb Tuples

The interactions among the scene objects are linguistically represented as verbs. These verbs are present in the associated scene descriptions. We use the image descriptions that were converted into dependency parse trees (DPTs) in Section 3.2, to extract subject-object-verb (SOV) tuples. Intuitively, these tuples represent the interactions being depicted in the scenes. The tuples are extracted using template matching on the DPT graphs. As the description annotations are long and complex sentences, they can contain several verbs, each with a different dependency type, such as `ROOT`, `acl`, `advcl`. We ignore the auxillary verbs such as *is*, *have* because they do not refer to any interaction.

The subject and object nouns are dependent on the verb; their location in the DTP graph changes based on the verb's dependency (Fig 4.3 a). We search for the verb's noun subject among the nouns present in the relevant configuration (which can be parent-child or c-command). The final assignment is done by matching the dependency of the

noun (`nsubj, nsubjpass`). We impose an additional criterion that the noun subject of a verb must refer to an animate object. The verb's object (if present) will be either a child node or a descendant of the verb. For intransitive verbs, where no object is present, we create a dummy object *ROOT* which refers to the image, and use it as the object in the tuple representation. Fig 4.3 (b) shows the verbs extracted from scene description shown in Fig 4.3 (a).

Next, the extracted verbs from each scene are grounded to arcs in the scene VDR, shown in Fig 4.3 (c), by mapping the subject and object of the verb to the corresponding source and target objects in the VDR arcs. This alignment gives us information on how different visuospatial configurations of objects are related to the type of interaction taking place between the objects. This information can be specific to the objects in the relation and it can also generalise across the object categories as the model learns these alignments over the whole training set. Thus, from the verb-VDR alignment in Fig 4.3 (b) and (c), the model can learn that the for the action *stand*, an (animate) object has to be present *above* another object.

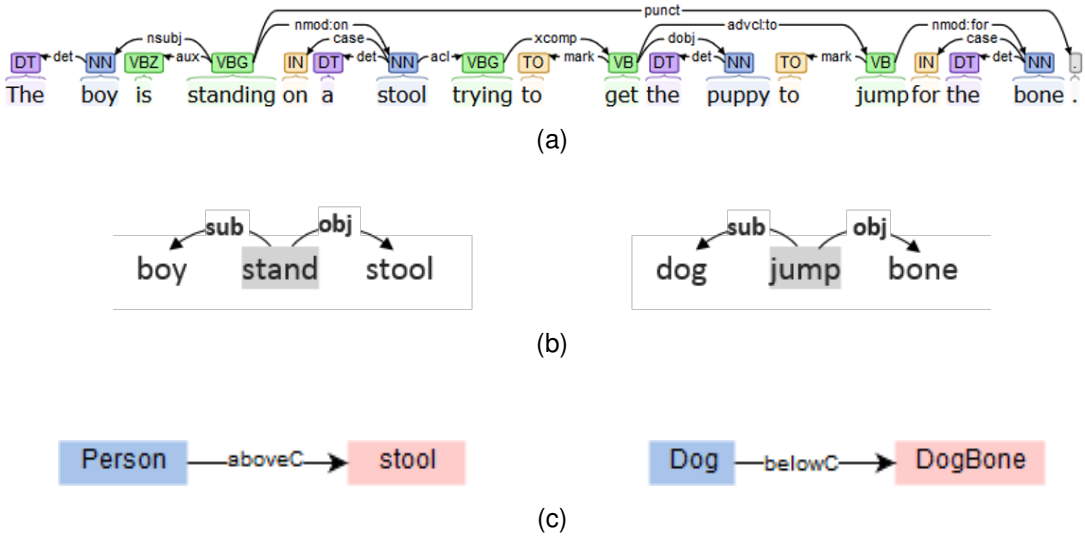


Figure 4.3: (a) Extracting verbs from the DPT of an image description: for verbs of type *ROOT* (here, *standing*), the subject (*boy* : *nsubj*) and the object (*stool* : *nmod*) of the verb are either in a sibling or c-command configuration; for verbs that are relative head clauses (*try* : *acl*), the subject and the object (*boy* : *nsubj*) are descendants; for verbs that are the adverb clauses (*jump* : *advcl*), the subject (*puppy* : *dobj*) and the object (*bone* : *nmod*) are always in a c-command configuration. (b) Verbs extracted from the scene description (c) the arcs from the scene VDR that correspond extracted verbs.

The SOV tuples and their alignments to the scene VDRs are used to directly estimate the probability distribution of objects (equation 4.3 and 4.4).

4.3.2 Verb Classifier

We train a multinomial logistic regression model to predict the verbs conditioned on the scene features. This verb classifier uses a combination of features extracted from

the visual and linguistic inputs to generate a probability distribution over the verbs. Some of these features would be correlated but logistic regression is robust to that. The feature extraction algorithm takes a SOV tuple and generates a set of features which are explained below.

Linguistic Features: features based on associated scene descriptions and questions from which the SOV tuple was taken. These include-

- subj(o1): this feature represents if an object is present as the subject of a verb.
- obj(o2): represents if an object is present as an object of a verb.

VDR Features: these features use the alignments between the VDR dependencies and the SOV tuples. They provide statistical information for each arc (object pair relation) over the verbs-

- connected(subj,obj): represents whether the object pair is connected in the VDR, regardless of spatial relation that connects them
- rel(subj): represents if an object is likely to appear as the head of a dependency with a given spatial relation.
- rel(obj): represents if an object to appear as an argument of a dependency with a given spatial relation.
- rel(subj,obj): a trigram feature that captures the likelihood of the object pair to be present in a certain spatial configuration.
- rel(subjCat,objCat): a trigram feature helps generalise the verbs across for the object categories; whether they occur in a given spatial configuration.

Object Features: This set of features is directly taken from the scene annotations (Section 3.2), or measures that are calculated using those annotations. They include-

- category(o): the category of an object.
- size(o): the type of object (large/small)
- pose(o): used for all animate objects.
- expression(o): refers to the facial expression, this feature is used for human objects only. present(o1,o2): captures how likely it is for the objects in the SOV pair to co-occur in a scene.
- centrality(o): distance of an object from the centre of a scene.
- distance(o1,o2): encodes the distance between the objects in the SOV tuple.
- category(o1,o2): bigram feature referring to the category of both objects in the SOV tuple. It captures information on how likely it is for the objects of two categories to co-occur in a scene

- facing(o1,o2): whether the two object are facing each other

Scene Features: these capture global properties of a scene-

- type: the type of the scene, indoor or outdoor

The features extracted using the above-mentioned templates are binarized (except distance, which is used as a numeric feature). Table 4.1 shows some sample features that would be generated using these templates for a training instance taken from Fig 4.3. The MaxEnt model learns weights such that they maximise the probability of the verb class over the set of observed features.

Template	Feature
subj(o1)	subj='child' & V='stand'
obj(o2)	obj='stool' & V='stand'
cat(subj)	subjCat='person' & V='stand'
cat(obj)	objCat='furniture' & V='stand'
connected(subj,obj)	subj='child' & obj='stool' & connected= True & V='stand'
rel(subj)	subj='child' & rel= 'aboveC' & V='stand'
rel(obj)	obj='stool' & rel= 'aboveC' & V='stand'
rel(subj,obj)	subj='child' & obj='stool' & rel= 'aboveC' & V='stand'
rel(subjCat,objCat)	subjCat='person' & objCat='furniture' & rel= 'aboveC' & V='stand'
pose(subj)	subj='child' & pose='02' & V='stand'
sceneType	sceneType='indoor' & V='stand'

Table 4.1: Example of some of the features generated using feature templates for the MaxEnt verb classifier, for an SOV tuple $boy, stand, stool$ where the target verb y = $stand$. Note that the table shows a selection of features whose value is 1.

4.3.3 Generating Answers

Once we have models that can predict verbs given an object pair or predict an object given an object and the verb, we can use this for the next step: generating natural language answers. The process of generating an answer for a given question and the scene VDR includes three main steps which are discussed in this section. Fig 4.4 shows an example input during test time and 4.5 illustrates the processing steps in generating an answer.

4.3.3.1 Identify object of interest-

The first step is to identify the object that the question is referring to. The object is identified by getting the grounding of the nouns in the question, using the scene alignments generated in Section 3.1. There can be some ambiguity at this stage, especially when the scene contains multiple objects of the same type. We have to consider other information such as object attributes (eg. age, gender), or the adjectives preceding the nouns (eg. *young man* vs *old man*) or spatial cues (eg. *man on the left*) to prevent false matches.

4.3.3.2 Identifying the information needed to generate the answer-

A question can be asking about the attributes of an object, the action being carried out by an object, or the object(s) taking part in an interaction. The kind of information needed to answer a question depends on the type of the question. For instance, questions of type *Where is...* require an object as an answer, questions of type *What is...* can require a verb or an object, or both. To find this, the questions are converted to statements using template matching on the POS sequence (Fig 4.4b). This is similar to Section 3.2, where training set questions were converted to statements, except during test time, the derived statements contain blanks. We then analyse the question type and POS sequence preceding the blank in the generated statements. Table 4.2 summarises the information needed to answer different question types. Not all question types will use VDRs for answer generation and we will limit our evaluation to the types highlighted in the table.

4.3.3.3 Identifying answer candidates-

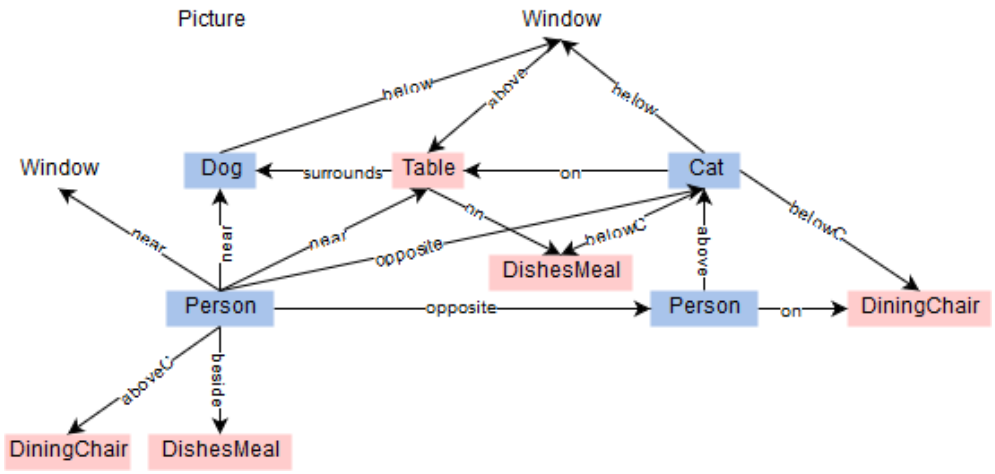
SCENE OBJECTS

After identifying the subject of the question, we can look for potential candidates for the answer from the set of objects that are in a parent-child relation in the scene VDR (Fig 4.3c). Questions that require an object as an answer can be of two types: questions about what objects are involved in a particular interaction (e.g. *What is the man holding?*), or questions about what objects are in a given spatial relationship (e.g. *What is on the table?*) For the first scenario, we have a subject and a verb and the task is to find the object. This can be done using equation 4.3 by finding an object o from that maximises the probability of the given verb and subject. For the second scenario, we have an object and a spatial relation and we simply need to find the node in the VDR connected to the object using that relation.

VERBS

The MaxEnt classifier can predict verbs given both the subject and object. However, questions contain both an subject and object in rare cases. Consider the question: *What is the dog doing?*. It contains a subject *dog* and no object. In this case, we first identify potential target objects from the scene VDR that the subject is likely to be interacting

(a)



(b)

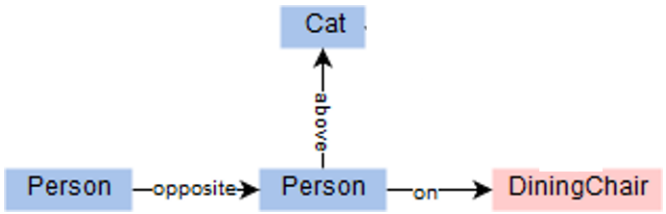
Figure 4.4: The input for the VQA task during test time (a) the scene and the question (b) silver standard Visual Dependency Representation of the scene: the actors are shown in blue and the remaining foreground object are in pink.

with. Often, the subject is interacting with more than object. We choose the most likely candidates for the answer based on the labels of the VDR arcs. For this step, other information such as the pose of the actor and the direction it is facing is also essential along with the structure encoding. The identified candidates are paired with the subject to predict a verb.

FACT CHECKING

The questions that require a yes/no response involve verifying whether the statement generated from the question is True or False. This may require a simple analysis of object attributes, spatial configurations or verbs or reasoning based on these features.

(a)



(b)

Figure 4.5: Processing the input to generate an answer involves (a) identifying the object of the question using POS tags and converting the question to a statement. The POS sequence preceding the blank indicates the kind of information (object/action) required in the answer. (b) The identified object is grounded to the scene and we consider the objects it is interacting to choose top candidates based on the VDR arcs, which are fed into the verb prediction model to generate an answer.

Question Type	Object Attributes	Scene Objects	Verbs	Reasoning
Is/ Are	✓	✓	✓	
Does/ Do	✓	✓	✓	
How many	✓			
Where is/are		✓		
What is [IN]		✓		
What is the [NN*]		✓	✓	
What colour	✓			
What kind	✓			✓
Who	✓	✓		✓
Why	✓		✓	✓
Which	✓	✓		
Will/Would/Can				✓

Table 4.2: The different question types in the VQA Abstract Scenes dataset and the kind of information required to answer each type. The first two rows denote types that require yes/no answers. The highlighted rows indicate the subsets of questions where answers depend on image structure.

4.4 Results & Discussion

In this section, we present the results obtained on the VQA task. First, we carried out experiments to optimise the verb prediction model before using it for VQA.

4.4.1 Predicting Verbs

We experimented training the MaxEnt classifier using different features to investigate how the scene and VDR features influence the probabilities of verbs. We start by training a simple model that uses a few unigram and bigram features. The feature number and complexity is gradually increased for each experiment. The complex features were created from the simpler features already present in the model, and these capture any interaction effects the features might have. Various experiments were done to find the combination of features that maximised the accuracy of the verbs predicted. Finally, we compared the optimal model against a baseline that uses the same feature set except the VDR Features.

4.4.2 Experiments

In this section, we will look at how different feature sets affect verb prediction. We will evaluate the MaxEnt model on randomly selected subsets of training data that were held off for validation. The features referred to in this section can be found in Section 4.3.2. The results from these experiments are summarised in Table 4.3

- SIMPLE: We begin by training the model on the unigram linguistic features and bigram VDR features.
- Reducing the size of vocabulary: It is a common practice in the image description literature to limit the target vocabulary in a model can generate. (Yang et al., 2011; Li et al., 2011). We clustered the target verbs using WordNet similarity measures in order to reduce the number of classes being modelled. It increased the classification accuracy by 5.4%.
- Increasing training instances: We increased the number of training instances by keeping all the repeating SOV tuples generated from the description of each scene. This resulted in a 4% improvement on the validation set.
- Trigram VDR features: The VDR trigram feature rel(subj,obj) gave an additional improvement which showed that subject-object-relation tuples convey more information than simpler bigrams.
- Object Features: Additional visual features such as pose, expression improved performance while the size of the object had no effect. This is probably because object size correlates to the object category.
- Additional Image Structure: Next, we added features such as distance and centrality of the objects and found the using distance improved the performance.

Further, using separate features for x and y distance was slightly better than using Euclidean distance. After experimenting, we found that dividing the exact distance measure by 10 produced better results, which implies that the direct pixel distance measures more add noise than useful information.

- Scene Features: Including information on scene type improved performance which suggests that the types of interactions found in indoor and outdoor scenes is different to some extent.

Additional experiments were done by removing each feature one by one. Features that did not contribute were removed and the final model used 16 features. Then were tested it against a baseline where all the features that provided structural information (VDR features and distance) were removed. The results showed that verb classification improves by approximately 10% when image structure is taken into account.

Experiment	Training Set Accuracy	Validation Set Accuracy
SIMPLE	42	37.4
VerbCat	49.1	42.8
increasing N	52	45.5
rel(subj,obj)	53.01	47.4
connected(subj,obj)	no effect	no effect
connected(subjCat,objCat)	53.06	47.75
size(o)	no effect	no effect
pose(o)	55.65	48.94
expression(o)	57.78	49.09
distance (o1,o2)	58.3	51.5
centrality(o)	no effect	no effect
sceneType	58.1	52.7
FINAL	58.1	52.4
BASELINE	46.3	42.7

Table 4.3: Results of experiments carried out to optimise the MaxEnt verb classifier using different features.

In order to examine the bottleneck in verb prediction, we carried out a few experiments using an alternative model that used support vector machines to classify verbs. However, the best performance we got using the SVM was 41% on the training set, which is lower than our MaxEnt baseline. We can conclude that the features being used are noisy and not discriminative enough for the task of predicting interactions among scene objects.

We investigated the SOV tuples used to extract features and found that the algorithm used to select the tuples was adding noise as it did not discriminate between objects that were actually involved in an interaction from objects that just happened to be nearby. For instance, consider the following two scene descriptions: *A man is sitting on the couch* and *A man is standing next to a couch*. Both the descriptions would yield SOV tuples with same subject and objects but different verbs. We realised that scenarios such as this could be leading to the generation of features that make the target classes indistinguishable from each other. This was confirmed when we analysed the confusion matrix produced by the classification of verbs by the MaxEnt model. For the verbs *sit* and *stand*, approximately 30% of the test instances were being being classified as the other class.

4.4.3 VQA

The MaEnt verb classifier was added to the VQA pipeline and we evaluated our approach on a subset of open ended questions from the Abstract Scenes VQA dataset. We compared the results against a baseline that used the same approach, but does not use VDRs for training.

4.4.3.1 Baseline

Prior work on VQA that uses the Abstract Scenes VQA dataset is very limited. Additionally, we did not evaluate our model on the whole dataset, so we cannot compare our results to the existing approaches in literature that do not use explicit image structure encoding. Therefore, to evaluate our model, we develop a baseline that represents scenes as a bag-of-objects and does not use VDRs to generate answers. The components of the baseline model work as follows:

Object Prediction: to predict an object involved in a given interaction, the probability estimate used is based on simple object co-occurrence. From the scene objects, this model chooses an object that is most likely to co-occur with the question object in the training set.

Verb Prediction: verbs are predicted using the maximum entropy model, but VDR features are not included during training. All other features based on the objects and linguistic annotations remain constant.

4.4.3.2 VQA Results

VQA evaluation in the current dataset is done automatically by comparing the generated answers to the gold standard annotations provided. An answer is considered

correct if it matches atleast three human annotated answers. Antol et al. (2015) define the following evaluation metric for VQA evaluation.

$$\text{accuracy} = \min(\frac{\text{number of matches}}{3}, 1) \tag{4.7}$$

The accuracy obtained on different question types in the validation set is presented in Table 4.4. Our model showed no significant improvement compared to the baseline for binary questions. On questions that required predicting verbs or objects (or both), our model slightly outperforms the baseline. However, questions that are directly about image structure, using VDRs showed an improvement of 54%. %

Question Type	Baseline	Our model
Is	35%	35.8%
Are	32.1%%	32.3%
What is the	28%	31.4%
What are the	25.3%	26%
Where is the	29%	73.1%
Where are the	27.7%	69.2%

Table 4.4: Accuracy on the VQA task for different question types. We compare our model that uses VDRs to a baseline that does not use a structured representation of images.

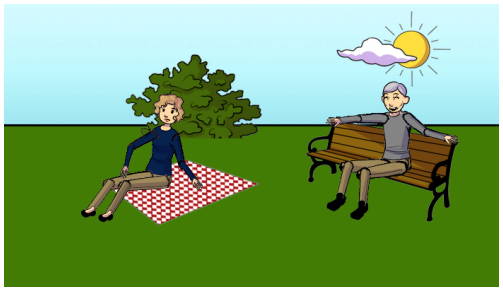
Fig 4.6 shows a random sample of answers generated for the validation test.

4.4.4 Error Analysis & Discussion

Scene structure encodings using VDRs provides enough information to accurately predict an object taking part in an interaction, with equal performance in instances where the said interaction is a spatial dependency or an action. This suggests that for questions that rely directly on image structure alone, VDRs were critical to the performance.

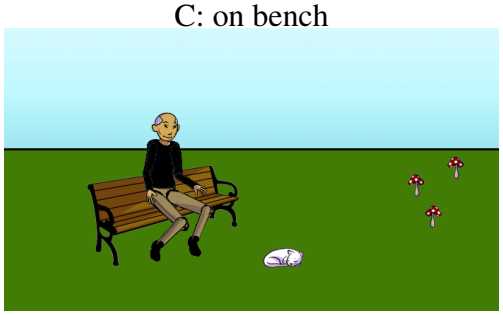
Another trend evident form the results is that the model performs lower on questions involving multiple objects *Where are....* This suggests that there is some ambiguity in the first step of the VQA pipeline that is involved with identifying question objects.

Further, we analysed the results from the question type that did not show an overall improvement. Through manual evaluation, we did find some improvements in those types— but these were, again, limited to questions that relied on image structure alone. For instance, subsets of questions in each type, such as *What is dog sleeping on?* or *Is the girl playing with the ball?*, where the task was to verify spatial relations, showed an increased accuracy as compared to the baseline.

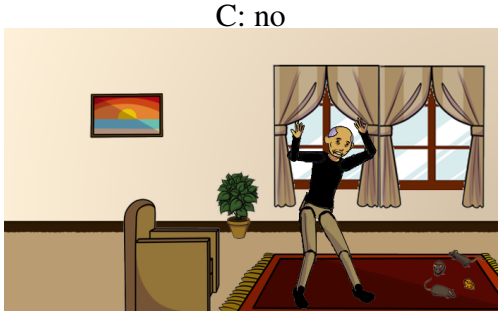


Where is the woman sitting?
A: on bench
B: bench
C: on bench

Is the man eating a hotdog?
A: no
B: yes
C: no



Is the cat running?
A: yes
B: yes
C: no



Where is the old man's hand?
A: window
B: couch
C: in air

What is the cat doing?
A: sitting
B: chasing
C: sleeping

Where is the fire?
A: below plant
B: room
C: fireplace

Figure 4.6: Sample answers generated for the VQA task. A: answer generated by our model, B: answer generated by the baseline, C; correct answer.

4.5 Conclusion

In this Chapter, we developed an approach to use image structure for the VQA task. The model included initial template matching to understand the question content and classifiers based on relative count estimates and maximum entropy for predicting the objects and verbs respectively.

Our results showed that using VDRs resulted in a significant improvement in accuracy on open ended questions that ask about the objects involved in particular relationships. However, for questions that rely on both image structure and reasoning or other attributes, the performance was poor. This implies that for a VQA model, being able to reason about image structure is critical to predicting interactions taking place in a scene.

Overall, we can confirm our hypothesis that structured representation of images improves performance for VQA. However, to use image structure effectively, we need a more powerful model that is capable of combining the information gained form the added structure representation, and use it in conjunction with other visual and linguistic features.

Chapter 5

Conclusion

We set out to investigate the role of structured image representation in Visual Question Answering. Clipart based scenes were used instead of real-world images, which allowed us to focus on designing a mechanism to represent image structure and using it for the VQA task, without any added noise from image segmentation and object recognition.

We will now summarise the main research contributions of this project and review some results that were of interest to our hypothesis. Finally, we will suggest some avenues for future research and how image structure can be exploited to gain a better understanding of scenes.

5.1 Summary of Contributions

The first part of the project involved enriching the Visual Dependency Representation proposed by Elliott and Keller (2013). We used findings from research in the field of scene perception and eye movements to propose an augmented Visual Dependency Representation. We implemented this using a heuristic based approach to generate a silver standard VDR set for the Abstract Scenes VQA dataset. The results showed that our approach is powerful enough to identify the important interactions in a scene, while being robust enough to give good results even when some of the features used were noisy.

In the second part, we tested our hypothesis by using the augmented VDRs that were generated in Chapter 3 for VQA and showed that the explicit use of image structure improves the model's understanding of scenes. We obtained a significant increase in accuracy of the answers generated on questions that rely entirely on image structure. Additionally, we found that the structured representation could be useful for all question types— provided the model has the capacity to reason using a combination of cross modal features and image structure. While automatic evaluation of the VQA task is much straightforward as compared to other tasks such as image description generation, we observed that human evaluation gives more insight into a model's performance as

compared to automatic evaluation metrics alone.

5.2 Final Remarks

We showed that a representation of image structure can be obtained automatically, even in the absence of gold standard annotations. The major advantage of this approach is that it makes structural information of scenes more readily available for use in research. It solves the problem of having to manually annotate large datasets with their structure encoding. During the generation of silver standard VDRs, we employed some simple heuristics based on literature on how humans perceive and understand images. Therefore, it follows that our approach can be applied to any dataset to identify important interactions in scenes and capture the structural dependencies.

Our experiments on verb prediction using different features demonstrated that image properties like distance and pose of the objects are most important for understanding the scenes. We also found that using more features did not necessarily improve the prediction of interactions between scene objects. Additionally, from these experiments we realised that even though the scenes in the dataset are composed of simple clipart images that use a fixed set of objects, the variety the poses of each object enable the scenes to cover a wide range of complex interactions that can be found in the real world. Therefore, we can extend our results to tasks involving real world images and expect to see a similar improvement in the model's understanding of an image when using structured image representation.

5.3 Future Work

In this report, we have established that using image structure leads us towards the goal of a more holistic scene understanding. This is a promising new avenue for research. There are several steps that can be taken to improve our current model's scene understanding capabilities. We will discuss these briefly in this section.

- the first obvious direction to take would be to combine the current findings with the state-of-the-art that use neural network based architectures that are more powerful at capturing the relations and dependencies in the data. This includes using CNNs to learn image features beyond the kind of annotations present in the dataset in combination with LSTMs to model language to obtain a multi-modal representation of features. The major shortcoming of our model was that while it could use the structural information directly, it could not use it effectively for high level reasoning in combination with other visual and linguistic features. When the use of structures image representation is combined with a deep learning approach, we can expect to see even bigger improvements, because, then the model would be able to use the structure information in conjunction with other features from the visual and linguistic domains.

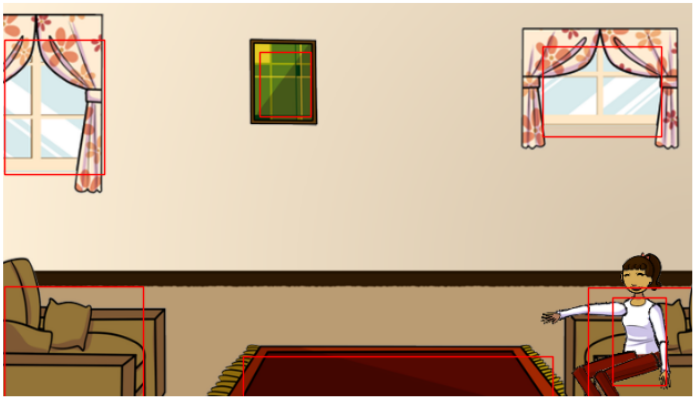
- exploring the use of natural language processing techniques to include additional annotations, for example, using sense annotations for verbs (Gella, Lapata, & Keller, 2016) or semantic role labelling (J. Zhou & Xu, 2015) for sentences. These will provide a richer understanding of the associated linguistic annotations.
- the Abstract Scenes VQA dataset contains very detailed annotations for the pose of human models. Using this information, by extending the VDRs to model not only the relation between scene objects, but also to how individual parts of the objects relate to one another, can definitely improve a model's understanding of complex interactions.
- When humans look at images, they use their existing knowledge of the world to understand the image. Similarly, an AI system should not have to rely on the input features alone. Including an external knowledge base can improve the model's reasoning capacity by giving it information that is not directly present in the scene or the associated linguistic annotations. It would help the model understand how concepts relate to each other.

Inference using image structure is a relatively unexplored field and while it can have important implications for VQA, it is by no means limited to this task. Any intelligent system that has to process visual information from the outside world can benefit from the use of structured representation of images.

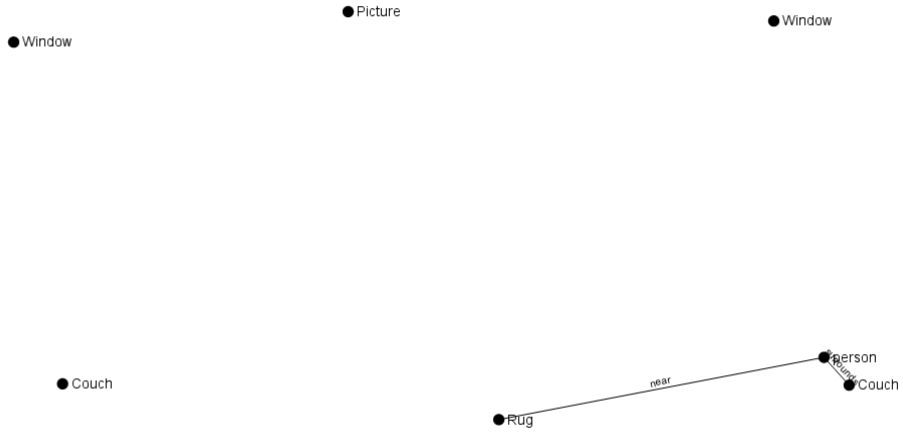
Appendix A

Sample Output: VDR Generation

Examples showing accurate outputs for different scene types:

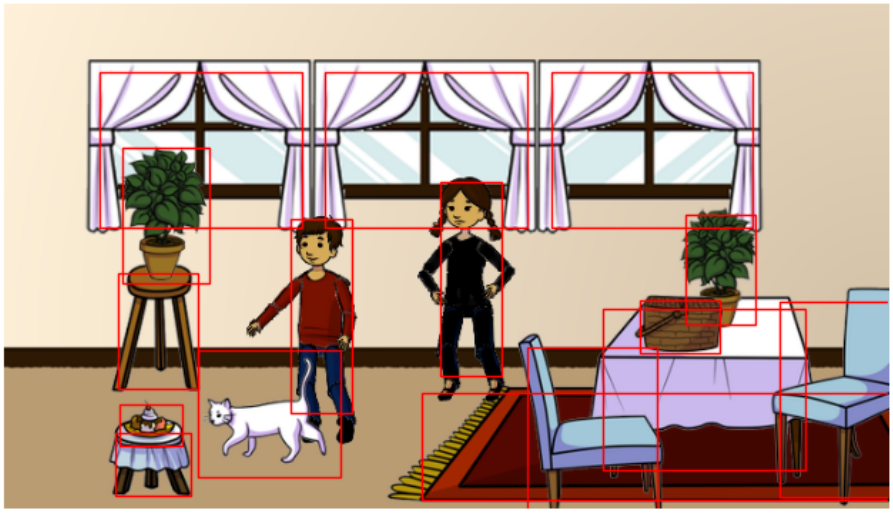


(a)

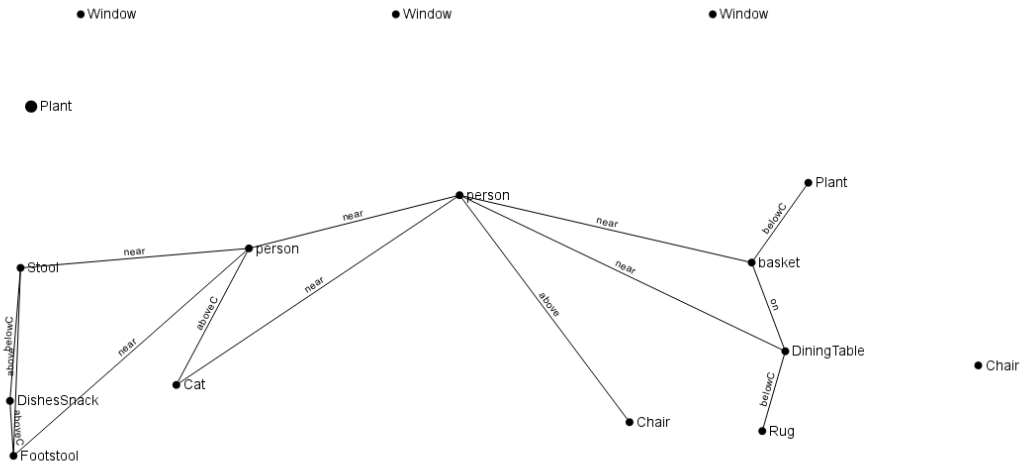


(b)

Figure A.1: An example of a VDR for a sparse scene.

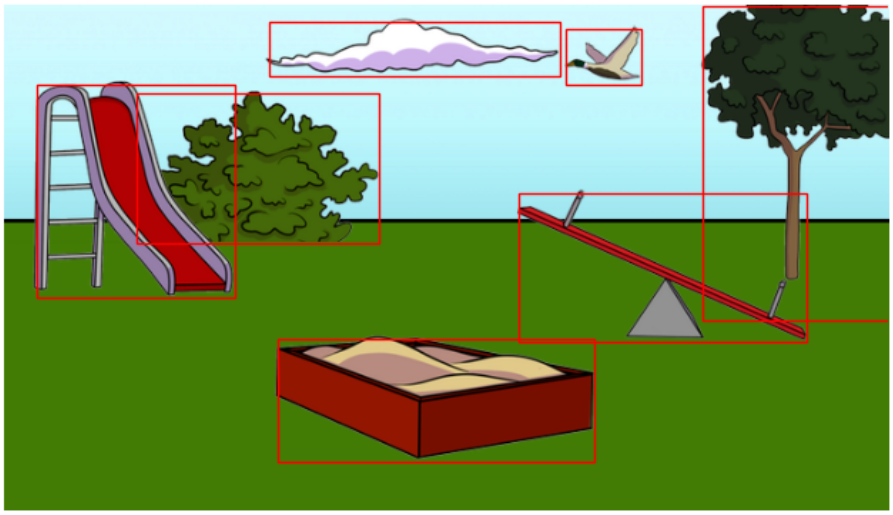


(a)

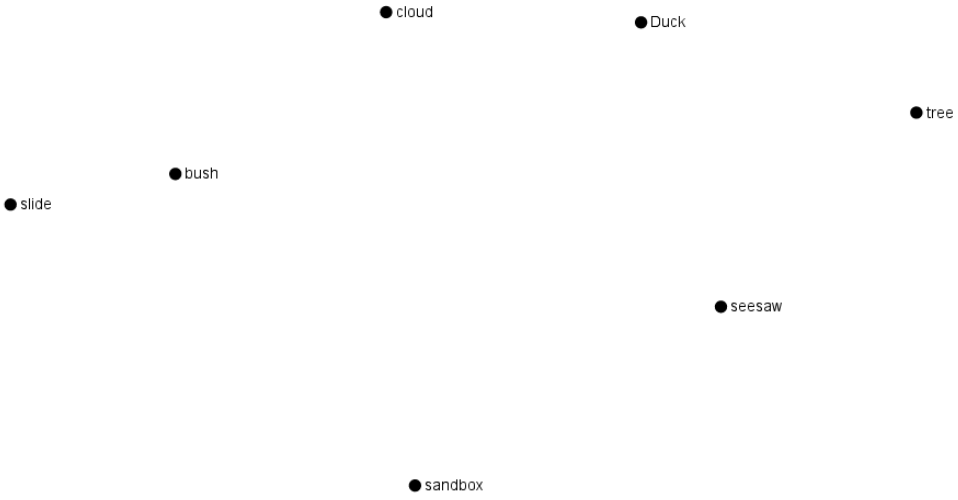


(b)

Figure A.2: An example of a dense scene that is handled well by the VDR generator.



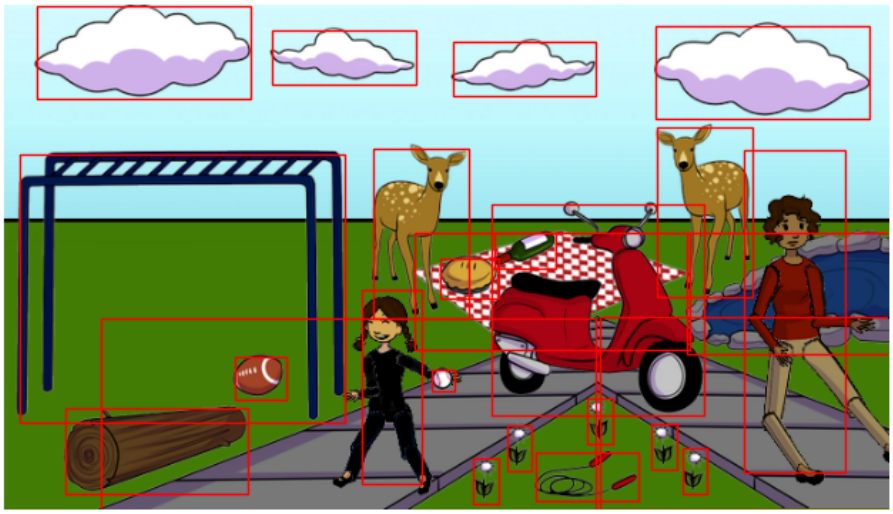
(a)



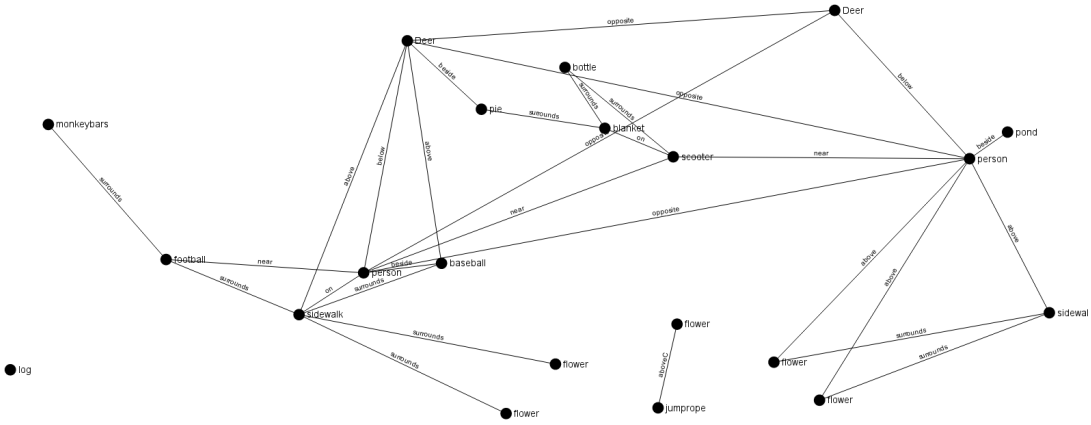
(b)

Figure A.3: An example showing a VDR that does not contain any relation because the parser did not identify any important relations in the scene. As we can see form the scene, there are no object interactions taking place.

Examples of inaccurate VDRs for different scene types:

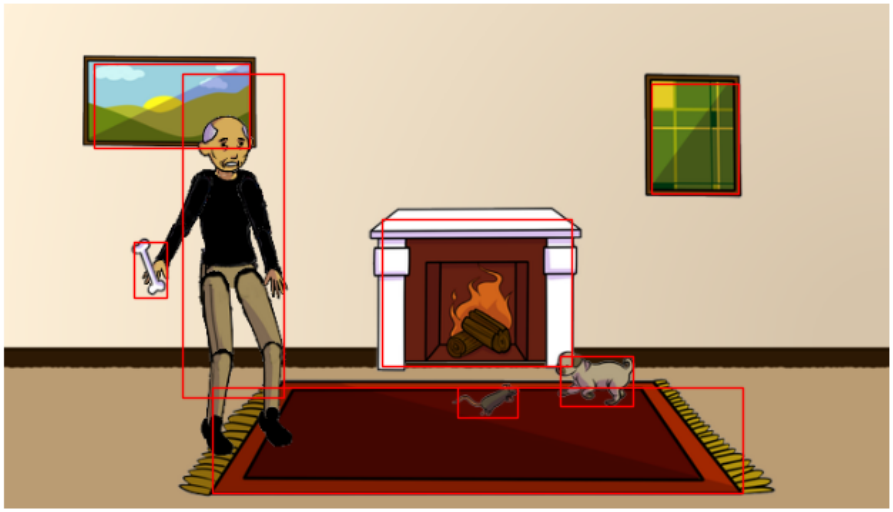


(a)

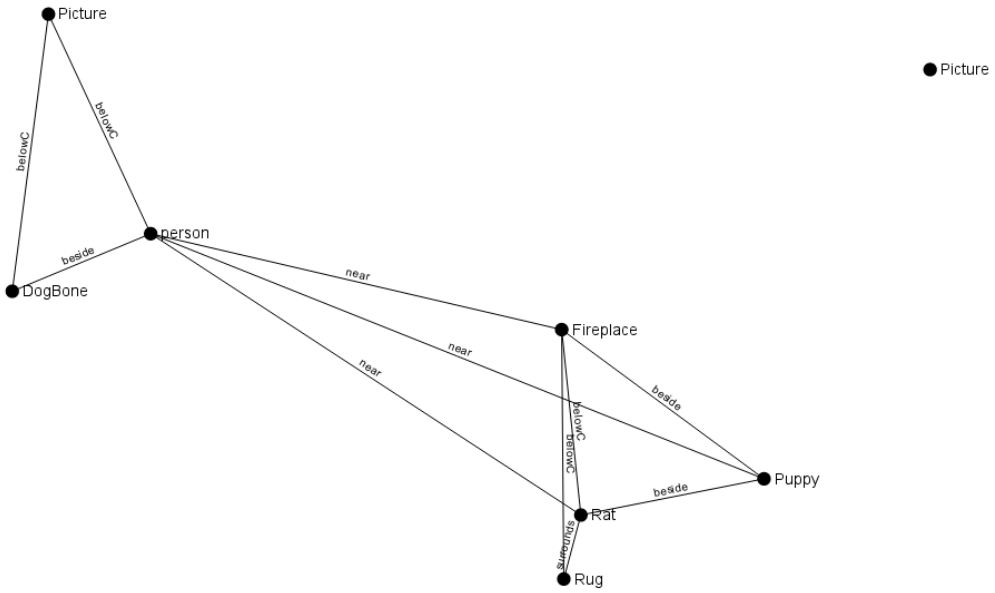


(b)

Figure A.4: An example showing a noisy VDR for a dense scene. This is because first, the bounding boxes of most scene objects are overlapping and second, all the objects are present at the same z-axis location as can be seen from the size of the nodes in the scene VDR.

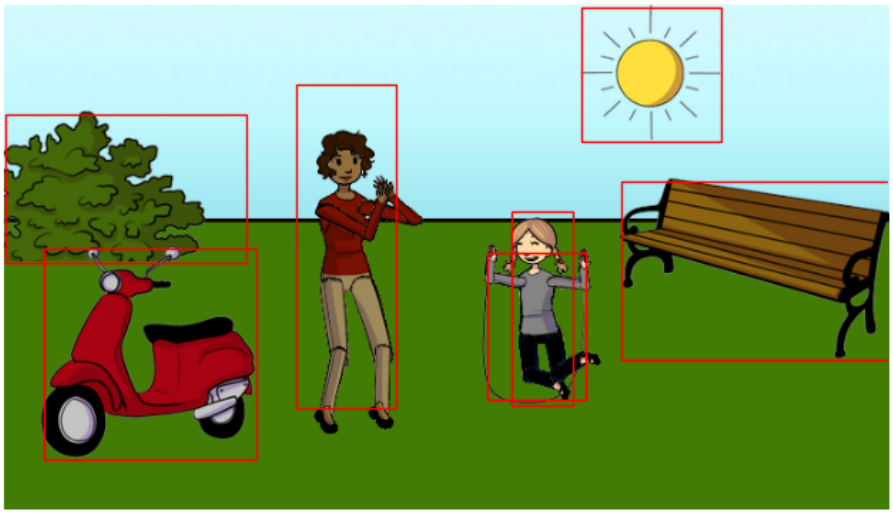


(a)

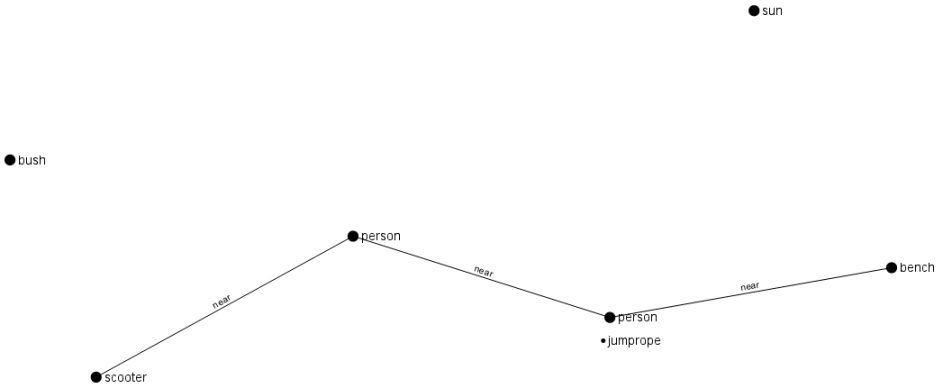


(b)

Figure A.5: An example showing a noisy VDR for a scene with average object density. Note from th size of the nodes that all objects have the same z-axis location, so there the parser cannot distinguish background and foreground objects.



(a)



(b)

Figure A.6: An example showing a VDR with a missing relation between the girl and jump rope. This occurs because of according to the z-axis annotations, the jump rope is a background object.

References

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., & Parikh, D. (2015). Vqa: Visual question answering. In *Proceedings of the ieee international conference on computer vision* (pp. 2425–2433).

Antol, S., Zitnick, C. L., & Parikh, D. (2014). Zero-shot learning via visual abstraction. In *European conference on computer vision* (pp. 401–416).

Biederman, I., Mezzanotte, R. J., & Rabinowitz, J. C. (1982). Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive psychology*, *14*(2), 143–177.

Brockmole, J. R., Castelhano, M. S., & Henderson, J. M. (2006). Contextual cueing in naturalistic scenes: Global and local contexts. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *32*(4), 699.

Clarke, A. D. F., Coco, M. I., & Keller, F. (2013). The impact of attentional, linguistic, and visual features during object naming. *Frontiers in psychology*, *4*, 927.

Elliott, D., & de Vries, A. (2015). Describing images using inferred visual dependency representations. In *Acl (1)* (pp. 42–52).

Elliott, D., & Keller, F. (2011). A treebank of visual and linguistic data.

Elliott, D., & Keller, F. (2013). Image description using visual dependency representations. In *Emnlp* (Vol. 13, pp. 1292–1302).

Elliott, D., & Keller, F. (2014). Comparing automatic evaluation measures for image description. In *Proceedings of the 52nd annual meeting of the association for computational linguistics: Short papers* (Vol. 452, p. 457).

Elliott, D., Lavrenko, V., & Keller, F. (2014). Query-by-example image retrieval using visual dependency representations. In *International conference on computational linguistics.*

Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Dollár, P., ... others (2015). From captions to visual concepts and back. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1473–1482).

Farhadi, A., Hejrati, M., Sadeghi, M. A., Young, P., Rashtchian, C., Hockenmaier, J., & Forsyth, D. (2010). Every picture tells a story: Generating sentences from images. In *European conference on computer vision* (pp. 15–29).

Fouhey, D. F., & Zitnick, C. L. (2014). Predicting object dynamics in scenes. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 2019–2026).

Gella, S., Lapata, M., & Keller, F. (2016). Unsupervised visual sense disambiguation for verbs using multimodal embeddings. *arXiv preprint arXiv:1603.09188*.

Geman, D., Geman, S., Hallonquist, N., & Younes, L. (2015). Visual turing test for computer vision systems. *Proceedings of the National Academy of Sciences*, *112*(12), 3618–3623.

Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., & Parikh, D. (2016). Making the v in vqa matter: Elevating the role of image understanding in visual question answering. *arXiv preprint arXiv:1612.00837*.

Gupta, A., Verma, Y., & Jawahar, C. (2012). Choosing linguistics over vision to describe images. In *Aaai* (p. 1).

Halligan, P. W., & Marshall, J. C. (1991). Left neglect for near but not far space in man. *Nature*, *350*(6318), 498.

Heider, F., & Simmel, M. (1944). An experimental study of apparent behavior. *The American Journal of Psychology*, *57*(2), 243–259.

Henderson, J. M., & Hollingworth, A. (1999). High-level scene perception. *Annual review of psychology*, *50*(1), 243–271.

Johnson, J., Krishna, R., Stark, M., Li, L.-J., Shamma, D., Bernstein, M., & Fei-Fei, L. (2015). Image retrieval using scene graphs. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3668–3678).

Jurafsky, D., & James, H. (2000). *Speech & language processing*. Pearson Education,.

Karpathy, A., & Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3128–3137).

Karpathy, A., Joulin, A., & Li, F. F. F. (2014). Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems* (pp. 1889–1897).

Kiros, R., Salakhutdinov, R., & Zemel, R. S. (2014). Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., ... others (2016). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *arXiv preprint arXiv:1602.07332*.

Kulkarni, G., Premraj, V., Ordonez, V., Dhar, S., Li, S., Choi, Y., ... Berg, T. L. (2013). Babytalk: Understanding and generating simple image descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(12), 2891–2903.

Li, S., Kulkarni, G., Berg, T. L., Berg, A. C., & Choi, Y. (2011). Composing simple image descriptions using web-scale n-grams. In *Proceedings of the fifteenth conference on computational natural language learning* (pp. 220–228).

Lin, D., Kong, C., Fidler, S., & Urtasun, R. (2015). Generating multi-sentence lingual descriptions of indoor scenes. *arXiv preprint arXiv:1503.00064*.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740–755).

Lu, J., Yang, J., Batra, D., & Parikh, D. (2016). Hierarchical question-image co-attention for visual question answering. In *Advances in neural information processing systems* (pp. 289–297).

Malinowski, M., & Fritz, M. (2014). A multi-world approach to question answer-

ing about real-world scenes based on uncertain input. In *Advances in neural information processing systems* (pp. 1682–1690).

Malinowski, M., Rohrbach, M., & Fritz, M. (2015). Ask your neurons: A neural-based approach to answering questions about images. In *Proceedings of the ieee international conference on computer vision* (pp. 1–9).

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., & McClosky, D. (2014). The stanford corenlp natural language processing toolkit. In *Acl (system demonstrations)* (pp. 55–60).

McDonald, R., Pereira, F., Ribarov, K., & Hajič, J. (2005). Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of the conference on human language technology and empirical methods in natural language processing* (pp. 523–530).

Ortiz, L. G. M., Wolff, C., & Lapata, M. (2015). Learning to interpret and describe abstract scenes. In *Hlt-naacl* (pp. 1505–1515).

Prest, A., Schmid, C., & Ferrari, V. (2012). Weakly supervised learning of interactions between humans and objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(3), 601–614.

Previc, F. H. (1998). The neuropsychology of 3-d space. *Psychological bulletin*, *124*(2), 123.

Rayner, K. (2012). *Eye movements and visual cognition: Scene perception and reading*. Springer Science & Business Media.

Ren, M., Kiros, R., & Zemel, R. (2015). Image question answering: A visual semantic embedding model and a new dataset. *Proc. Advances in Neural Inf. Process. Syst*, *1*(2), 5.

Rizzolatti, G., Fadiga, L., Fogassi, L., & Gallese, V. (1997). The space around us. *Science*, *277*(5323), 190–191.

Socher, R., Karpathy, A., Le, Q. V., Manning, C. D., & Ng, A. Y. (2014). Grounded compositional semantics for finding and describing images with sentences. *Transactions of the Association for Computational Linguistics*, *2*, 207–218.

Wu, Q., Teney, D., Wang, P., Shen, C., Dick, A., & Hengel, A. v. d. (2016). Visual question answering: A survey of methods and datasets. *arXiv preprint arXiv:1607.05910*.

Wu, Q., Wang, P., Shen, C., Dick, A., & van den Hengel, A. (2016). Ask me anything: Free-form visual question answering based on knowledge from external sources. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4622–4630).

Yang, Y., Teo, C. L., Daumé III, H., & Aloimonos, Y. (2011). Corpus-guided sentence generation of natural images. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 444–454).

Yao, B., & Fei-Fei, L. (2010). Grouplet: A structured image representation for recognizing human and object interactions. In *Computer vision and pattern recognition (cvpr), 2010 ieee conference on* (pp. 9–16).

Zhou, B., Tian, Y., Sukhbaatar, S., Szlam, A., & Fergus, R. (2015). Simple baseline for visual question answering. *arXiv preprint arXiv:1512.02167*.

Zhou, J., & Xu, W. (2015). End-to-end learning of semantic role labeling using recurrent neural networks. In *Acl (1)* (pp. 1127–1137).

Zhu, Y., Groth, O., Bernstein, M., & Fei-Fei, L. (2016). Visual7w: Grounded question answering in images. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4995–5004).

Zitnick, C. L., & Parikh, D. (2013). Bringing semantics into focus using visual abstraction. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 3009–3016).

Zitnick, C. L., Parikh, D., & Vanderwende, L. (2013). Learning the visual interpretation of sentences. In *Proceedings of the ieee international conference on computer vision* (pp. 1681–1688).