It's not about the Journey; It's about the Destination: Following Soft Paths under Question-Guidance for Visual Reasoning

Monica Haurilet Alina Roitberg Rainer Stiefelhagen
Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany
{haurilet, alina.roitberg, rainer.stiefelhagen}@kit.edu

Abstract

Visual Reasoning remains a challenging task, as it has to deal with long-range and multi-step object relationships in the scene. We present a new model for Visual Reasoning, aimed at capturing the interplay among individual objects in the image represented as a scene graph. As not all graph components are relevant for the query, we introduce the concept of a question-based visual guide*, which constrains the potential solution space by learning an optimal traversal scheme. The final* destination nodes *alone are then used to produce the answer. We show, that finding relevant semantic structures facilitates generalization to new tasks by introducing a novel problem of knowledge transfer: training on one question type and answering questions from a different domain without any training data. Furthermore, we achieve state-of-the-art results for Visual Reasoning on multiple query types and diverse image and video datasets.*

1. Introduction

Interpreting and answering subsequent questions about the semantic relationships of the complex and noisy environment is a key trait of our cognition. Extraordinary progress linked to the rise of deep learning in the core vision tasks [36, 23, 9, 26] (*e.g.* object recognition) has created a solid basis for the new research direction of *higher level* visual reasoning. Going beyond the conventional recognition, Visual Reasoning [37, 48] decides about the necessary future actions [16], which is crucial for artificial intelligence applications. The compositional structure of our world makes this task especially hard, as merely recognizing individual building blocks at a lower level is not enough. Such models require precise *relational reasoning* about the entities present in the scene and their interactions with each other.

Visual Reasoning tasks are often posed in the form of Visual Question Answering (VQA) [37, 48, 16], which lies in the intersection of vision and language and attempts to answer a specific question about the scene. Complex semantic associations between both, language query and the visual scene entities (Figure 1) are characteristic to this task.

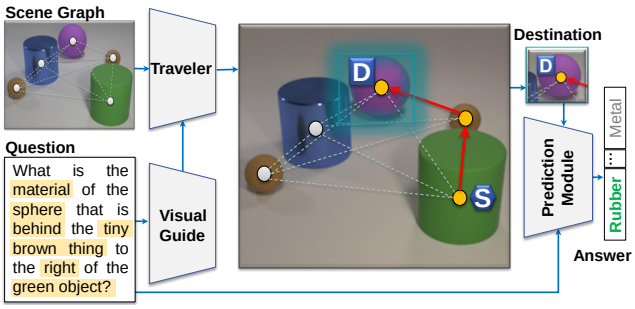


Figure 1: Visual reasoning example, where object interplay is crucial for the correct answer and an overview of our graph neural network-based approach. The *visual guide* learns to give question-dependent directions to follow on the scene graph. The final answer is then produced solely from the embeddings of the reached *destination* nodes.

Despite the exceedingly structured nature of the visual information needed to answer open-ended questions, the majority of previous works focus on spatial feature maps obtained from a pre-trained CNN and further combined with an attention mechanism on parts of the image [47, 50, 39]. While pre-trained CNNs offer excellent object embeddings, they face problems in relational reasoning about their large-scale interactions. An excellent way to model such multi-step associations in an image are *scene graphs* [46], where the nodes represent the object- and the connecting edges specify their relationship embeddings. We notice that even though the relations between objects are indispensable for the complete scene understanding, only a portion of the graph is relevant for answering a specific question. We therefore leverage the visual graph in a *selective* way through a question-dependent *visual guide*.

We aim at unifying graph-based inference with *question-specific visual guidance*, in order to identify paths with relevant information flow and present a new model for Visual

Reasoning. Given an image-question pair, we first use the *visual guide* to create question-specific directions to follow in the graph. Next, the *graph traveler* traverses the visual graph guided by these directions and computes the probability distributions over the nodes being the final destination. Finally, we compute our answer prediction solely from the expected destination node as visual representation for our *prediction module*. While conventional graphical models for VQA follow the graph-refinement paradigm (*i.e*. refined embeddings of *all* components are used for the prediction), we maintain the original node representations, identify the key paths and answer the question solely from the expected final *destination* nodes, hence: *It's not about the journey; It's about the destination*.

We demonstrate the effectiveness of our model on three well-known datasets for different visual reasoning tasks: question-answering on video data (COG [48]), compositional reasoning on 3D synthetic images (CLEVR [15]) as well as diagram question-answering, with real-life figures extracted from textbooks (AI2D [17]), which is much noisier while having less training data. Our model consistently outperforms previous approaches on the AI2D and COG benchmarks and shows strong performance on the CLEVR dataset.

As our model operates on semantic structures inside the scene graph, it has two beneficial properties: interpretability and generalization to new tasks. An ablation study illustrates that we can easily shed light on the internal choices our model made to produce the answer by following the final *soft path*. To evaluate the generalization capabilities, we propose a new task of knowledge transfer for VQA, by splitting the training and test set based on question *types* (*e.g*. *query attributes* questions for training and *counting* for testing). Through knowledge obtained from training on one kind of questions, our model is able to derive the answers for queries which type it has never seen before.

2. Related Work

Graph Neural Networks. Current models are conventionally formed through convolution operations in a *local* neighborhood and address long-range dependencies merely through large receptive fields. Rich structure of the scene can be targeted in a more efficient way through *graphs*, which have been utilized in a wide range of applications, such as language [24], social interaction [21, 38, 49], knowledge representation [3, 29, 42] and chemistry [33]. This is achieved by either generating graphs directly from the CNN feature maps [25, 18, 46] or by combining the existing graph representations with the previously acquired knowledge base [45, 4].

We distinguish three groups of knowledge-base guided algorithms: approaches using graph-refinement through the network either for better node representation [20, 43, 5, 41], or for refining the edges [40, 37], and the graph traversal approaches [45, 4]. The first group performs feature pooling for the node itself and its neighborhood (*e.g*. through a recurrent neural network (RNN) [41]). In contrast, the second group combines the *edges e.g*. through average-, sum-pooling [37] or a weighted combination [20]. Unfortunately, a graph representation of an image which has been strongly modified *e.g*. through an RNN looses its interpretability for the human eye. Our proposed approach falls into the third category, as the graph representation built once at the beginning remains fixed throughout the process. The questions are subsequently answered by *exploring various paths* of the graph without any further feature refinement (*e.g*. depending on the question). The decision is based solely on the *destination* node embeddings and the reason, *why* our model has favored one answer over the other can be easily understood through the found graph trails.

To the best of our knowledge, we are first to present a model based on graph *traversal* for Visual Reasoning. Of particular relevance are recent works of Xiong *et al*. [45] and Go *et al*. [4] in the field of *language-based* question answering. The authors represent text-based knowledge as a graph and perform training with the REINFORCE [44] paradigm in order to traverse it. However, these procedures are constrained by the query paths being *discrete*. In comparison, our model is trained on *visual* entities and follows *soft paths*, as we obtain a *continuous* confidence over the nodes in each step (*i.e*. as opposed to the paths weighted either by 0 or 1 in previous work).

Visual Question Answering (VQA). VQA has rapidly gained popularity over the past years [1, 52, 22, 10], mostly being addressed through image feature maps extracted with a pre-trained CNN and subsequent question-related attention module [50, 51]. In general, the ways of addressing this problem can be divided into four categories: 1) Global embedding methods [31, 1, 28, 34, 35, 30] that use a joint embedding of the global image representation and the question to produce an answer; 2) Models that attend to parts of the image are able to improve performance [50, 51, 6, 47]; 3) Compositional models [2, 13, 16] use a modular representation of the neural networks; 4) Graph-based VQA models [37, 41, 11, 17, 18], where a graph representation of the image or the question is used to produce the answer.

The latter category has emerged recently and is by design well-suited for relational reasoning, as object connections are explicitly represented through the edges. Such approaches mostly follow the graph-refinement paradigm. Teney *et al*. [41] refine the features of each node using an RNN by pooling based on the similarity of to the current node. In [17], an RNN is applied on the edges which are subsequently filtered through a question-based atten-

tion, while in [18] an end-to-end version is proposed, where the edges are learned inside the model. Finally, the models in [37, 11] represent the graphs as an unordered set of edges using weighted average to get a fixed image representation for answering the question.

Our model falls into the graph neural network category, leveraging the object- and their relationships embedding as the scene graph components. Other than previous graph-based approaches for VQA [41, 17, 18, 37, 11], our model is not based on graph-refinement. While conventional methods refine the embeddings of *all* graph components and use them to compute the final answer, we hold the original node representations, identify the key paths through the question-based *visual guide* and answer the question solely from the final *destination* nodes.

3. Visual Reasoning via Guided Soft Paths

We present a new model for visual reasoning that deals with the composite object relationships in the scene as a graph traversal problem. The challenge is that the space of potential paths in a visual graph is very large. When asked '*What is the material of the sphere that is to the left of the tiny brown thing behind the green object?*' (Figure 1), a human would immediately look for the green object, thereafter, at the tiny brown sphere, then, select the sphere left of it. Likewise, our idea is to greatly constrain the solution space by learning the optimal graph traversal strategies based on question-specific decisions.

Conceptually, our visual reasoning model is composed of three main components: 1) the visual guide, 2) the graph traveler and 3) the prediction module. The *visual guide* takes as input the question and produces direction embeddings. The *graph traveler* follows these directions and computes the *soft paths* – probability distributions over the nodes of being in the route to the nodes that include relevant information to produce the answer. The final decision is made by the *prediction module*, which exploits the found *destinations* as weights for the graph nodes and infers the final answer. We want to highlight, that the prediction module operates exclusively on the destination node representations, dismissing the preceding components of the paths. While the *visual guide* and the *prediction module* can be viewed as individual neural networks connected by the *graph traveler*, they are optimized jointly in an end-to-end training fashion.

An overview of our model is illustrated in Figure 2. Next, we give a general definition of our model's building blocks (Section 3.1); provide a mathematical foundation for computing the *soft paths* (Section 3.2); and, finally, we present our complete graph-based neural architecture for Visual Reasoning (Section 3.3).

3.1. Data Structures

Graph. We define as a visual graph $G = (V, F, R)$ a structure with the following properties:

1. V – a set of N vertices representing the object instances present in the image.

2. $F \in \mathbb{R}^{N \times D}$ – a D-dimensional representation for each of the N visual nodes. These can be one-hot vectors representing the object instance or features extracted from a pre-trained CNN.

3. $R \in \mathbb{R}^{N \times N \times E}$ – an E-dimensional relation representation for each pair of nodes $(n, m) \in V \times V$. One way to define the representation R is a one-hot embedding of predicates (*e.g.* 'on top', 'holding'), which can be obtained as in [27], or features extracted from a CNN on the image crop surrounding both objects. A simpler method is to represent each edge by concatenating the node pair representations F.

Path. We call an ordered set of nodes of length T in graph G a *path*:

$$\tau = [n_1^\tau, n_2^\tau, \ldots, n_T^\tau].$$

We note that this definition of path assumes a *discrete* assignment of each node in each time step t.

Soft Path. A soft path does not return *discrete* associations of each of a node with the path but softens its inclusion. Formally, for each time step t and node n in graph G we have an association score $p^t(n) \in [0, 1]$. As we aim to model a probability distribution, we require that the sum over all nodes in time step t in the graph is one:

$$\sum_{n \in V} p^t(n) = 1.$$

Thus, a soft path is described by the two dimensional array $\tau = [p^1(V), p^2(V), \ldots, p^T(V)]$, where we use: $p^t : \mathbb{R}^N \rightarrow [0, 1]^N$ element-wise on each node.

Starting Node. The starting node of path τ is the node at the first time step: n_1^τ. In case of a soft path it is defined by a probability distribution over all nodes n.

Destination. A destination n is a node in path τ that occurs in time step T, while for the soft paths it is equal to the probability in the last time step.

3.2. Reaching the Destinations

Our model is built upon the assumption that by traversing the scene graph in a controlled way, we are able to identify

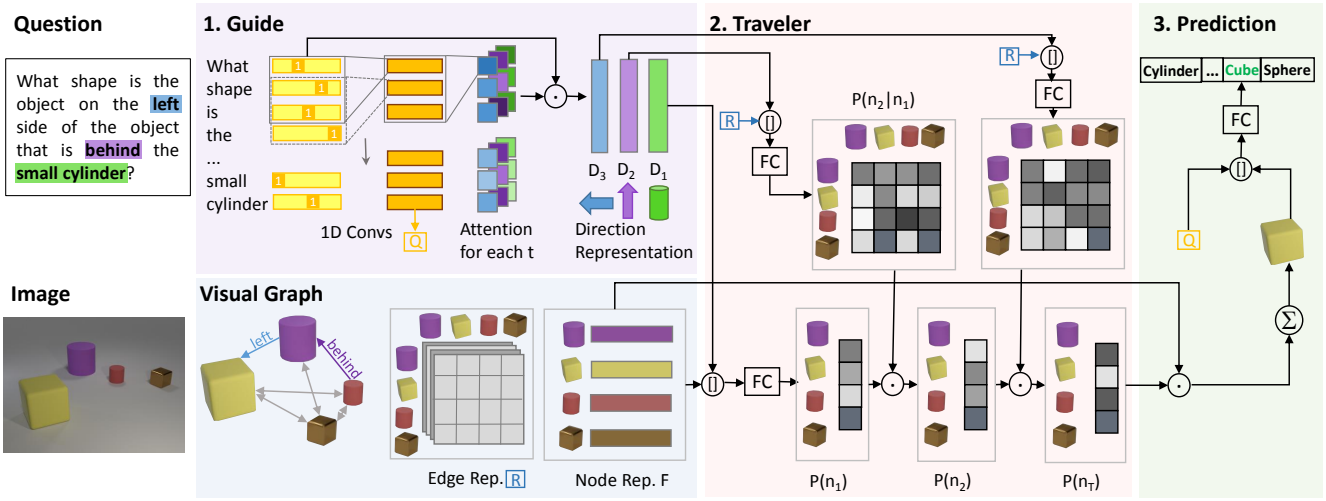


Figure 2: Proposed graph neural network architecture which learns traversal strategies for the scene graph (simplified for path length $T = 3$). While the visual *guide*, the graph *traveler* and the *prediction module*, are individual neural network components, they are optimized jointly in an end-to-end fashion. The visual guide takes as input the question and provides direction embeddings for the traveler to follow. Prediction module gives the final answer based only on the question and the destination nodes embeddings, the predecessors are therefore dismissed: it's not about the journey; it's about the *destination*.

the information relevant for the specific question. We therefore compute the probability of the node n being a destination, which is equal to the sum of the probabilities of all paths ending in n:

$$P(n_T = n) = \sum_{\tau} P(\tau) \cdot 1[n_T^{\tau} = n]. \tag{1}$$

According to the marginalization rule, the probability of the path τ is then equal to:

$$P(\tau) = P(n_1, \ldots, n_T) = P(n_1) \cdot \prod_{t=2}^{T} P(n_t | n_{t-1}, \ldots, n_1) \tag{2}$$

Our approach models a discrete Markov Chain (*i.e.* we assume the Markov Property) with the set of states equal to the nodes V in our graph G. We obtain the probability of each path as:

$$P(\tau) \approx P(n_1) \cdot \prod_{t=2}^{T} P(n_t | n_{t-1}). \tag{3}$$

In case of $t = 1$, it is straight forward to compute the probability of the nodes in the path (*i.e.* $P(n_1 = n)$). For $t > 1$ we have to consider the transition probabilities $P(n_t|n_{t-1})$. Since the number of possible path options grows exponentially with the path length, we further reformulate this calculation for time steps larger than one. We iteratively transform the path probability to the probability of each node lying in each time step *e.g.*:

$$\begin{aligned} P(\tau) &= P(n_1) \cdot P(n_2|n_1) \cdot \prod_{t=3}^{T} P(n_t|n_{t-1}) \\ &= P(n_2) \cdot \prod_{t=3}^{T} P(n_t|n_{t-1}) \end{aligned} \tag{4}$$

Thus, the new estimation lies in the calculation of each probability $P(n_t)$. For this, we make use of the function $\tau^t(n)$ which computes the probability of each node n being in the path in an iterative way using the formulation:

$$\tau^t(n) = \sum_{m \in N} P^t(n|m) \cdot \tau^{t-1}(m). \tag{5}$$

We stop the calculation at time step T and the final values become the probability of each node being the destination *i.e.* the node has information relevant for the question. Next, we show the models for obtaining the start- and transition probabilities.

3.3. Neural Graph Architecture

In conventional graph neural networks for VQA, node features F change depending on their neighbors in each training time step, becoming a mixture of the initial and foreign object representations (*i.e.* graph-refinement) [17, 41, 37]. In comparison, our model keeps the semantic node representations and focuses on the network *topology*, learning to find relationships of the scene entities relevant for the current question (see Figure 2). We can easily shed light

upon the choices of our model, as we retain the initial interpretation of its nodes and highlight the key links between them.

1. Visual Guide. The *visual guide* considers the static graph as a map to be traversed using the question as the reference. That is, the guide takes as input the question, embeds it *e.g.* using an LSTM [12] or a one dimensional CNN with self-attention [8] and produces direction embeddings D for the traveler to follow on the graph. In case of an LSTM, we represent the question as the final hidden state, while we use weighted average over the feature maps in case of a CNN. Predicted directions at a time step t are then obtained through learned fully connected layers: $D^t = W_D^t \cdot H + b_D^t$, where $W_D^t \in \mathbb{R}^{|D^t| \times |H|}$ with the size of the direction embeddings $|D^t|$ chosen empirically.

2. Graph Traveler. The *graph traveler* traverses the visual graph based on the directions suggested by the *guide*. Thus, it produces prior probabilities (*i.e.* the confidence of each node being the first one visited) and computes the transition probabilities (*i.e.* confidence of traversing one node to the next).

For the first node of a path, we obtain the confidence by training a fully connected layer on top of the node representations F from the visual graph and the first direction D^1 given by the *guide*:

$$P_\theta(n_1) = softmax(W_{p_1} \cdot [D^1, F] + b_{p_1}), \tag{6}$$

where θ is the collection of all the learnable parameters in the model and the $softmax$ function normalizes over the nodes:

$$softmax(X)_i = exp(x_i) / \sum_{j \in V} exp(x_j). \tag{7}$$

In case of the transition probabilities, we make use of the edge features R between each pair of nodes:

$$P_\theta(n_t | n_{t-1}) = softmax_{source}(W_{p_t} \cdot [D^t, R] + b_{p_t}). \tag{8}$$

Here, the $softmax$ operation normalizes over the rows, such as the sum over the outputs is equal to one:

$$\sum_{n \in V} P_\theta(n|m) = 1. \tag{9}$$

In the last time step T the *graph traveler* computes the probability of a node being the final destination $\tau^T(n)$ (as introduced in Equation 5).

3. Prediction Module. The *prediction module* differentiates between the problem types and generates the answer leveraging the probability distribution over the destinations (see step 3 in Figure 2). In case of query-type questions (*i.e.* questions about the shape, color *etc*. of an object), the solution is determined from the *destination* nodes *i.e.* soft path probabilities $\tau(n)$ at time step T as:

$$g^H = \sum_{n \in V} \tau^T(n) \cdot F_n, \tag{10}$$

where F_n is the nth row of the matrix F (*i.e.* the feature representation of each node in V). We concatenate this visual global representation g^H with the question embedding Q. Then, a fully connected layer is used to produce the final prediction over all possible answers. For *existence* questions, we answer the question with 'yes', in case that any of the destinations has a probability over 0.5. In the task *counting*, we estimate the number of destinations that round to one. For tasks, where sum of the final soft path probabilities may be larger than one, as multiple destinations could be applicable (*e.g. counting* or *existence*), we use sigmoid function instead of softmax for edge normalization.

Dataset	Type	# Imgs	# Inst	# Q
COG	Videos	11M	9.6	44M
AI2D	Diagrams	5K	9.1	15K
CLEVR	3D-Synthetic	100K	6.5	700K

Table 1: Visual Reasoning benchmarks used to evaluate our model (by task type, number of images/videos, average amount of instances per example and number of questions).

Model Configuration. We train the network end-to-end by minimizing the cross entropy using Adam [19] with an initial learning rate of 0.00025 without any weight or learning rate decay. We choose a maximal path length T empirically on the validation data. The question-based guide uses multiple 1D convolution layers with 32 hidden units, while the final fully connected layers of the graph traveler have the size of 128 (we include a detailed description of the parameters in the supplemental material).

4. Evaluation

We perform comprehensive studies on three challenging datasets for Visual Reasoning with diverse query types (overview in Table 1). All datasets cover visual examples, task queries with the ground-truth solutions (open-ended or multiple choice form), as well as annotations for the scene graph. In Section 4.1, we evaluate our model on video sequences, then, in the task of diagram question answering (Section 4.2) and on highly compositional reasoning problems on 3D synthetic images (Section 4.3). We further discuss how different path lengths T impact the performance (Section 4.4), evaluate how well our model generalizes to previously unseen tasks (Section 4.5) and, finally, visualize concrete examples of soft paths (Section 4.6).

4.1. Visual Reasoning on Videos

Dataset. In this section, we use the COG [48] dataset as a test bed for both, spatial and temporal reasoning. The dataset comprises of over 11 Million questions on videos. While the videos are of synthetic 2D scenes, it specifically targets temporal memory and logical deductive reasoning about video input, being difficult for humans [48]. The task is to deduce the correct answer while taking into account changes of the scene in three different query types: pointing, yes/no, conditional and attribute-related questions. Higher number of scene entities is also characteristic for the dataset.

Results. We demonstrate the effectiveness of our model in Table 2. Additionally to the original Working Memory [48] approach, we compare our model to three baselines: 1) random performance, 2) a question-only model consisting of a 1D CNN over the question words followed by fully-connected layers, and 3) a graph-based approach, where instead of computing the answer from the destination nodes of the found paths, we use a joint embedding of the question and all of the nodes in the graph as input and use fully-connected layers to make a prediction.

Approach	Atts.	Condit.	Point	Yes/No	All
Baselines					
Random	1.9	8.4	17.5	50.0	26.6
Question-only	1.6	2.3	19.4	49.7	27.4
Memory Networks					
Work. Memory† [48]	–	–	–	–	93.7
Graph-based Methods					
Question+Nodes	73.7	63.5	92.5	57.9	63.3
Ours	**99.2**	**98.4**	**100.0**	**95.0**	**97.2**

Table 2: Results for visual reasoning on videos on the test set of COG for different tasks: pointing, existence, conditional questions and questions about object attributes. † Best model selected from 50 trained models.

Our model yields the best recognition rates in all query types. The distinction from the natural-language-based benchmarks becomes obvious, as the *question-only* approach exceeds the random baseline by less than 1%. *Visual* reasoning is therefore decisive for this benchmark. The yes/no questions have been the major source of our model's unreliability. Our analysis of these confusions indicates occasional difficulties in case of 'and' connections in the question (*e.g.* 'Shape of last magenta object equal shape of last lavender object and shape of now mint object equal shape of last olive object?'). Nonetheless, our model achieves excellent performance of 100% for pointing questions, and establishes new state-of-the-art overall accuracy of 97.2%.

4.2. Diagram Question Answering

Dataset. Next, we evaluate our approach on real-life images in the diagram understanding task. AI2D [17] dataset contains images extracted from school textbooks of various subjects and evaluates understanding of causal relations in these figures. As middle school pupils are required to learn from such diagrams, reason and answer questions about them, this dataset represents an excellent realistic testbed for visual reasoning. As we are dealing with real-life data, AI2D is smaller and noisier than other datasets we used for testing, with 666 lessons of total 5K diagrams and 15K questions.

Approach	All
Baselines	
Random	25.00
Classical VQA Methods	
VQA [1]	32.90
Graph Neural Networks	
DQA-Net [DSDP] [17]	38.47
DQA-Net [DGGN] [18]	39.73
DQA-Net [18]	41.55
Ours	**43.45**

Table 3: Diagram Question Answering results on real images extracted from school textbooks (AI2D dataset) [17]

Results. In Table 3, we compare our model with a multitude of published approaches, including three graph-based methods. As AI2D is evaluated in multiple choice form with four possible options, random choice performance is 25%. Overall, there is a clear benefit of using structured approaches. Our graph-traversal based model consistently outperforms state-of-the-art graph neural networks and therefore confirms the effectiveness of focusing on traversal schemes and the found destination nodes, instead of the message-passing paradigm.

4.3. VQA on 3D Synthetic Images

Dataset. Compositional Language and Elementary Visual Reasoning dataset (CLEVR) [15] is a widely used diagnostic benchmark for compositional understanding of 3D scenes for different tasks, such as counting, finding attributes of objects based on their relations with other instances and comparison between object attributes. Long reasoning chains, demanding memory-related tasks and absence of question-based biases are distinctive for this benchmark. Although it is comprised of synthetic scenes, conventional VQA models often face significant difficulties on CLEVR as they tend to focus on the dataset bias [16, 37, 7].

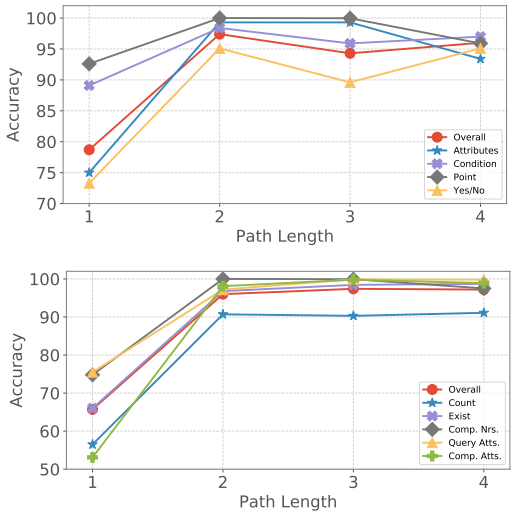


Figure 3: Performance for different maximal path lengths T on validation set of COG (top) and CLEVR (bottom).

Results. We report results on all five problem types of the CLEVR benchmark: counting, existence, query attributes and questions about comparing numbers and attributes of objects. A high number of novel methods have been recently proposed to tackle CLEVR reasoning tasks, which we group based on their way of addressing object relations and compare to our model in Table 4.

Approach	Reference	Count	Exist	Comp. Nrs.	Query Attrs.	Comp. Attrs.	All
Human [16]	–	86.7	96.6	86.5	95.0	96.0	92.6
Qtype [16]	–	34.6	50.2	51.0	36.0	51.3	41.8
Classical VQA Methods							
LSTM [16]	–	41.7	61.1	69.8	36.8	51.8	46.8
CNN [16]	–	43.7	65.2	67.1	49.3	53.0	52.3
CNN+SA [37]	*ECCV'16*	64.4	82.7	77.4	82.6	75.4	76.6
QGHC [7]	*ECCV'18*	91.2	78.1	79.2	89.7	86.8	86.3
FiLM [32]	*AAAI'18*	94.3	99.1	96.8	99.1	99.1	97.7
Compositional Models							
N2NMN* [13]	*ICCV'17*	68.5	85.7	84.9	90.0	88.7	83.7
PG(9K)* [16]	*ICCV'17*	79.7	89.7	79.1	92.6	96.0	88.6
PG(700K)* [16]	*ICCV'17*	92.7	97.1	98.7	98.1	98.9	96.9
Memory Networks							
Work. Mem. [48]	*ECCV'18*	91.7	99.0	95.5	98.5	98.8	96.8
MAC† [14]	*ICLR'18*	**97.1**	**99.3**	96.8	99.1	99.1	**98.9**
Graph Neural Networks							
CNN+RN‡ [37]	*NIPS'17*	90.1	97.8	93.6	97.9	97.1	95.5
Ours	–	91.3	98.6	**99.6**	**99.5**	**99.8**	97.5

Table 4: Visual reasoning results for different tasks on the CLEVR test set [15]. (*) denotes the use of extra supervision in form of program labels, ‡ denotes the use of data augmentation, † denotes the use of pre-trained models.

We achieve state-of-the-art accuracy of over 99% on three tasks (comparing numbers and two attribute-related problems) and report a strong overall performance (97.5%), surpassing humans (92.6%) and the recent graph-based method based on edge representation sum [37] (95.5%).

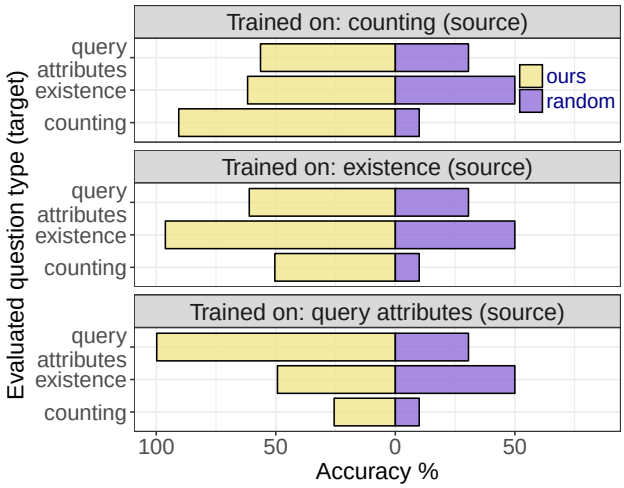


Figure 4: Generalization to unseen tasks: our model is trained on one query type is evaluated on a different task.

4.4. Impact of the path length on performance

As we explicitly focus on *relations* in the scene, we compare variants of our model to measure the effect of different restrictions of the *soft path* at length T. Figure 3 illustrates changes of accuracy in relation to T for different COG and CLEVR tasks. The model benefits immensely from considering paths of length two or more, *e.g.* for the *query attributes* task, percentage of correct answers rises from 53.1% ($T = 1$) to 98.1% ($T = 2$), further improving to 99.8% ($T = 3$), confirming the significance of causal connections in the scene. Starting at $T = 4$ for CLEVR and $T = 3$ for COG, we observe a slight decline in overall performance, which we link to the extend of chained questions in the datasets. For example, in a question 'What is the material of the sphere behind the tiny brown thing to the right of the green object?' (Figure 1) the reasoning chain consists of two pairwise relationship. In general, enforcing longer paths than necessary for the question is not a problem in our architecture, as it permits self-loops. However, the option of including more nodes than required might result in higher level of noise, as the overall search space becomes larger. This slight accuracy drop should be viewed with caution, as it is also connected to the nature of the questions in the dataset *i.e.* it is expected to increase with the amount of entities mentioned in the question. Nonetheless, when further increasing the path length to a higher path length the performance stabilizes *e.g.* for COG the model achieves 95.6% for $T = 8$.

4.5. Performance on unseen tasks

Humans have an impressive ability to address new tasks of increasing difficulty by transferring solutions from familiar problems. Similarly, our motivation for focusing on the scene *structure*, is to develop a model which processes

queries by decomposing them into granular tasks, which then could be easily re-used to answer questions our model has never seen before.

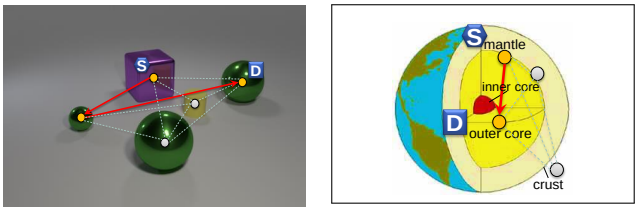
To evaluate our assumption, we propose a new challenging benchmark for visual reasoning on problems not previously seen during training. We regard three tasks from the CLEVR dataset: *query attributes*, *existence* and object *counting*. In our proposed evaluation setup, the model is trained on one of these tasks and is intended to solve another one. Consider the *existence* task, where we output 'yes' if in the last time step T there is at least one destination node with probability over 0.5 (see Section 3.3). As the node representations are not refined throughout the process, we can extend our model to *counting* without additional training, by merely using the *counting* prediction module version, *i.e*. summing the number of destinations with an activation over 0.5, as described in Section 3.3. For the *query attribute* task, we select the node with the maximal activation.

We report the performance of our model on previously unseen tasks in Figure 4. Our approach successfully applies the knowledge it had acquired from *counting* or *existence* to previously unseen query types. These two tasks are especially re-usable as they involve a universal granular question: whether objects are present in the scene, or not. In case of learning on the attribute-based questions, we assume that the destinations are always available (as we question specific attributes of the node and not their presence). Re-usability of the learned information is therefore lower. Training on the *counting* queries turned out to be most beneficial for solving new problems. We assume, this is due to *counting* being a more composite task as it covers both, checking for object presence and determining, whether the objects have certain properties (*e.g*. 'What number of brown balls are the same size as the metal object?'). Our model trained on the *counting* task was able to solve the query attribute problem in 56.4% of times, surpassing random chance (30.6%) by 25.8%.

Obviously, solving previously unseen tasks is per design a much harder problem than conventional supervised Visual Reasoning and the recognition rates are considerably lower. Apart from the lack of supervision, language expressions not present during training pose an additional challenge (*e.g*. 'how many' if the model was trained on the *existence* task and evaluated on *counting*). Still, our model consistently outperforms the random chance baseline, being able to address new tasks without costly annotations of training examples.

4.6. Qualitative Results

An important property of our model is the ability to trace back the underlying reasoning behind the final answer. In Figure 5, we revisit the final soft paths of our model on two examples from CLEVR and AI2D benchmarks. We visualize the nodes with the maximal probability at each time step t: *e.g*. in the left image the starting node 'S' points at the violet cube. Edges which belong to the path are marked with red arrows, starting with the source node 'S' and ending in the final destinations, which are the only graph components used as input in the prediction modules. In case of CLEVR (left), we have a very long and strongly compositional question on which we produce a path of length 3: traversing from the cube to the small sphere until we finally reach the destination: the large sphere in the right side of the image. In case of AI2D textbook diagram question (right), our model solves the query 'What is between mantle and inner core' with a soft path of length 2 by starting at the mantle and, next, choosing the destination and also the correct answer: 'outer core'.



There is a green metal object that is behind the thing on the left side of the metal block; what shape is it? Answer: Sphere

What is between mantle and inner core? Answer: Outer Core

Figure 5: Example visualizations of the final soft path. Orange circles mark the highest activation at each time step t.

5. Conclusion

We presented a new approach for compositional visual reasoning, where we employ a graph neural network architecture to tackle far-reaching relationships in the scene. Our framework learns how to traverse the graph in a controlled way and, then answers the question based on the reached destination nodes of the found paths. Our model exceeds state-of-the-art methods on two challenging datasets for Visual Reasoning: on Videos (COG) and Diagram Question-Answering (AI2D), as well in the three tasks on the 3D synthetic data (CLEVR). At the same time, our model is highly interpretable as the graph trails directly shed light on the underlying reasoning, showing that our model breaks complex instructions into smaller tasks. Furthermore, we demonstrate the positive impact of focusing on relevant semantic *structures* on the ability to reuse the acquired knowledge for novel tasks. In this new benchmark setting, our model was trained on a certain question type (*e.g*. *existence*) and could successfully handle tasks of a different kind (*e.g*. *counting*) without any further training. Our experiments show encouraging evidence that modern visual recognition approaches could benefit further from structured methods especially in high-level understanding of global causal relations.

References

[1] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. L. Zitnick, D. Parikh, and D. Batra. Vqa: Visual question answering. *International Journal of Computer Vision*, 2017. 2, 6

[2] J. Andreas, M. Rohrbach, T. Darrell, and D. Klein. Deep compositional question answering with neural module networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[3] G. Bouchard, S. Singh, and T. Trouillon. On approximate reasoning capabilities of low-rank vector spaces. *AAAI Spring Syposium on Knowledge Representation and Reasoning (KRR): Integrating Symbolic and Neural Approaches*, 2015. 2

[4] R. Das, S. Dhuliawala, M. Zaheer, L. Vilnis, I. Durugkar, A. Krishnamurthy, A. Smola, and A. McCallum. Go for a walk and arrive at the answer: Reasoning over paths in knowledge bases using reinforcement learning. *International Conference on Learning Representations*, 2017. 2

[5] K. Do, T. Tran, T. Nguyen, and S. Venkatesh. Attentional multilabel learning over graphs-a message passing approach. *arXiv preprint arXiv:1804.00293*, 2018. 2

[6] A. Fukui, D. H. Park, D. Yang, A. Rohrbach, T. Darrell, and M. Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. *Conference on Empirical Methods in Natural Language Processing*, 2016. 2

[7] P. Gao, P. Lu, H. Li, S. Li, Y. Li, S. Hoi, and X. Wang. Question-guided hybrid convolution for visual question answering. *IEEE conference on computer vision and pattern recognition*, 2018. 6, 7

[8] J. Gehring, M. Auli, D. Grangier, and Y. N. Dauphin. A convolutional encoder model for neural machine translation. *arXiv preprint arXiv:1611.02344*, 2016. 5

[9] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 1

[10] D. Gurari, Q. Li, A. J. Stangl, A. Guo, C. Lin, K. Grauman, J. Luo, and J. P. Bigham. Vizwiz grand challenge: Answering visual questions from blind people. *IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[11] M. Haurilet, Z. Al-halah, and R. Stiefelhagen. Moqa - a multi-modal question answering model. In *Workshop on Shortcomings in Vision and Language*, 2018. 2, 3

[12] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 5

[13] R. Hu, J. Andreas, M. Rohrbach, T. Darrell, and K. Saenko. Learning to reason: End-to-end module networks for visual question answering. *IEEE International Conference on Computer Vision*, 2017. 2, 7

[14] D. A. Hudson and C. D. Manning. Compositional attention networks for machine reasoning. *International Conference on Learning Representations*, 2018. 7

[15] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1988–1997. IEEE, 2017. 2, 6, 7

[16] J. Johnson, B. Hariharan, L. van der Maaten, J. Hoffman, L. Fei-Fei, C. L. Zitnick, and R. B. Girshick. Inferring and executing programs for visual reasoning. In *IEEE International Conference on Computer Vision*, pages 3008–3017, 2017. 1, 2, 6, 7

[17] A. Kembhavi, M. Salvato, E. Kolve, M. Seo, H. Hajishirzi, and A. Farhadi. A diagram is worth a dozen images. In *European Conference on Computer Vision*, 2016. 2, 3, 4, 6

[18] D. Kim, Y. Yoo, J. Kim, S. Lee, and N. Kwak. Dynamic graph generation network: Generating relational knowledge from diagrams. *Conference on Computer Vision and Pattern Recognition*, 2017. 2, 3, 6

[19] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[20] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *International Conference on Learning Representations*, 2017. 2

[21] S. Kok and P. Domingos. Statistical predicate invention. In *Proceedings of the 24th international conference on Machine learning*, pages 433–440. ACM, 2007. 2

[22] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L. Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. 2016. 2

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 1

[24] M. Kuhlmann and S. Oepen. Towards a catalogue of linguistic graph banks. *Computational Linguistics*, 42(4):819–827, 2016. 2

[25] Y. Li, O. Vinyals, C. Dyer, R. Pascanu, and P. Battaglia. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324*, 2018. 2

[26] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1

[27] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision*, 2016. 3

[28] L. Ma, Z. Lu, and H. Li. Learning to answer questions from image using convolutional neural network. In *Association for the Advancement of Artificial Intelligence*, volume 3, page 16, 2016. 2

[29] F. Mahdisoltani, J. Biega, and F. M. Suchanek. Yago3: A knowledge base from multilingual wikipedias. In *Conference on Innovative Data Systems Research*, 2013. 2

[30] M. Malinowski and M. Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *Advances in neural information processing systems*, pages 1682–1690, 2014. 2

[31] M. Malinowski, M. Rohrbach, and M. Fritz. Ask your neurons: A deep learning approach to visual question answering. *International Journal of Computer Vision*, 2017. 2

[32] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. Film: Visual reasoning with a general conditioning layer. *Association for the Advancement of Artificial Intelligence*, 2017. 7

[33] P. Radivojac, W. T. Clark, T. R. Oron, A. M. Schnoes, T. Wittkop, A. Sokolov, K. Graim, C. Funk, K. Verspoor, A. Ben-Hur, et al. A large-scale evaluation of computational protein function prediction. *Nature methods*, 10(3):221, 2013. 2

[34] M. Ren, R. Kiros, and R. Zemel. Exploring models and data for image question answering. In *Advances in neural information processing systems*, pages 2953–2961, 2015. 2

[35] M. Ren, R. Kiros, and R. Zemel. Exploring models and data for image question answering. In *Advances in neural information processing systems*, pages 2953–2961, 2015. 2

[36] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 1

[37] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, 2017. 1, 2, 3, 4, 6, 7

[38] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad. Collective classification in network data. *AI magazine*, 29(3):93, 2008. 2

[39] K. J. Shih, S. Singh, and D. Hoiem. Where to look: Focus regions for visual question answering. In *IEEE conference on computer vision and pattern recognition*, pages 4613–4621, 2016. 1

[40] M. Simonovsky and N. Komodakis. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[41] D. Teney, L. Liu, and A. v. d. Hengel. Graph-structured representations for visual question answering. *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2, 3, 4

[42] K. Toutanova, D. Chen, P. Pantel, H. Poon, P. Choudhury, and M. Gamon. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509, 2015. 2

[43] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. Graph attention networks. *International Conference on Learning Representations*, 2017. 2

[44] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992. 2

[45] W. Xiong, T. Hoang, and W. Y. Wang. Deeppath: A reinforcement learning method for knowledge graph reasoning. *Empirical Methods in Natural Language Processing*, 2017. 2

[46] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing. *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 2

[47] H. Xu and K. Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering. In *European Conference on Computer Vision*, pages 451–466. Springer, 2016. 1, 2

[48] G. R. Yang, I. Ganichev, X.-J. Wang, J. Shlens, and D. Sussillo. A dataset and architecture for visual reasoning with a working memory. *European Conference on Computer Vision*, 2018. 1, 2, 6, 7

[49] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha. Like like alike: joint friendship and interest propagation in social networks. In *Proceedings of the 20th international conference on World wide web*, pages 537–546. ACM, 2011. 2

[50] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked attention networks for image question answering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1, 2

[51] D. Yu, J. Fu, T. Mei, and Y. Rui. Multi-level attention networks for visual question answering. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2

[52] Y. Zhu, O. Groth, M. Bernstein, and L. Fei-Fei. Visual7w: Grounded question answering in images. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2