Look, Remember and Reason: Visual Reasoning with Grounded Rationales
–Supplemental Material–

Apratim Bhattacharyya [1]   Sunny Panchal [1]   Mingu Lee [1]   Reza Pourreza [1]

Pulkit Madan [1]   Roland Memisevic [1]

[1]Qualcomm AI Research, an initiative of Qualcomm Technologies Inc.

A. Overview

Here we provide, 1. We provide a broader overview of related work. 2. Additional architecture details. 3. Additional results across CLEVR, CATER and ACRE datasets.

B. Related Work

Large language models and reasoning. Large language models have shown strong performance on a variety of natural language processing tasks, *e.g.*, question answering, translation, summarization [3, 42]. This progress has been enabled by scaling model size [6, 21, 41], which has enhanced models' ability to learn in-context and unlock "emergent abilities" such as the ability to perform well on very challenging reasoning tasks [7, 35, 36] like commonsense reasoning [58, 47], symbolic reasoning [18, 60] and mathematical reasoning [24]. The progress in reasoning abilities has been fueled by "chain-of-thought" or "rationale" based methods which aim to mimic human reasoning processes. This has been successfully applied to improve performance on arithmetic programs [28], and commonsense reasoning [44] among others. LLMs can learn multi-step tasks like long-division [45] through rationales and manipulation of an external environment in the form of a "scratch paper". Similarly, [38] outputs intermediate steps to improve performance on computational problems. More recently, by utilizing models with a strong ability to learn in context, the possibility of generating rationales through prompting has been demonstrated in [23, 59]. Further, producing multiple chains of thought and selecting the final answer by majority vote has shown promise in [56, 59]. In this work, we aim to leverage the abilities of LLMs to reason in natural language by generating rationales and performing visual reasoning tasks.

Multi-modal language models. Analogous to the large-scale models for text, there have been breakthroughs in the development of large multimodal approaches which can deal with multi-modal, specifically visual, inputs in addition to the text. Pix2seq [5] utilize auto-regressive language models to extract low-level visual information from images. ViperGPT [51], VisProg [13] and Chameleon [31] use language-based LLMs with vision sub-modules for multi-modal tasks. Other approaches focus on joint modeling of visual and textual data. Such models include the CLIP [40] and BLIP [26] which utilize natural language instead of image-level class labels. Flamingo [2] introduces a family of language and vision models which are pre-trained on diverse vision and language tasks with a large amount of vision and text data available from the web. Recent approaches such as CM3 [1] train a multimodal LLM on a large HTML corpus for image and text generation. Other approaches [11, 25, 29, 34, 54], instead of training vision and language models from scratch on the multimodal data, incorporate pretrained LLMs as language priors. Methods like Frozen [54], leverage pretrained LLMs and train a vision encoder to encode images as a sequence of tokens which can be presented to the transformer in the same form as the text. PaLM-E [10] provides images and text as interleaved multi-modal latent vectors, allowing the model to process multiple images within any part of a sentence which serves as an input to the LLM where the model is trained end-to-end. LLaMA-Adapter [66] introduces a adapter layer with zero-init attention to enable multi-modal inputs with the LLaMA model [52]. LLaVA [29] finetunes the LLaMA model which is presented with the output of the vision encoder obtained from conversational data. A low-rank adaption to finetune a LLaMA model in a setting similar to LLaVA is further explored in mPLUG-owl [62]. FROMAGe [22] on the other hand, freezes the language model, and fine-tunes the input and output linear layers to encode multimodal interactions. In our work, we systematically study the role of low-level visual skills for visual reasoning and introduce rationales with corresponding surrogate tasks.

Attention-based models and visual reasoning. Attention-based models have been studied extensively for visual reasoning [8, 16, 17, 20, 32, 48]. Recent advances include an

object-centric encoder and a transformer reasoning module to solve RPM-like benchmarks [37], multi-hop feature modulation [50] and cascaded modulation networks [61] that use a multi-step comprehension process, neural interpreters [43] that factorize inference in a self-attention network and ALANS learner [65] that combines abstract algebra and representation theory. Calibrating concepts and operations [27] enables neural symbolic models to capture underlying data characteristics and perform hierarchical inference. In contrast to these approaches with task-specific architectures, we focus on using off-the-shelf LLMs with spatial features from a CNN for visual reasoning. We instill the ability to extract object-centric information in the network by using rationales, instead of resorting to specialized object detection modules.

C. Architecture Details

Our cross attention based top-down attention mechanism in Fig. 2 (in the main paper) exploits the rich hierarchical representation encoded in the hidden states $\mathbf{h} = \{\mathbf{h}^1, \ldots, \mathbf{h}^m\}$ of the LLM, where m is the number of self-attention layers in the LLM and $\mathbf{h}^i \in \mathbb{R}^{t \times q}$. Here, t is sequence length $t = t_v + t_s$ and q is the dimensionality of the embedding space. The first embedding layer of the LLM encodes tokens, whereas subsequent layers contain progressively richer and more information-dense representations than encode increasingly global information. Therefore, we propose to use the embedding layers higher in the hierarchy in our top-down attention mechanism to guide the information extraction process from visual inputs.

Our LRR model, employs grid-level visual features obtained from ResNet[15] based CNN, which allows us to preserve spatial information crucial for visual reasoning tasks. The adoption of a simple CNN ensures that our model is applicable across a variety of visual reasoning problems. In our approach, the CNN encodes the input image sequence $\mathbf{I} = (\mathbf{v}_1, \ldots, \mathbf{v}_{t_v})$ into $\bar{\mathbf{I}} = (\bar{\mathbf{v}}_1, \ldots, \bar{\mathbf{v}}_{t_v})$, where $\bar{\mathbf{v}}_i = \mathrm{CNN}(\mathbf{v}_i)$ and $\bar{\mathbf{v}}_i \in \mathbb{R}^{g \times q'}$. Here, g is the size of the grid and q' the dimensionality of the CNN embedding space.

To integrate and "look" for the visual information from the CNN in our LLM pipeline, we employ cross attention (CROSS-ATTN) layers at higher levels $\{k, \ldots, m\}$ of the hierarchical LLM representation space, in addition to the self-attention (SELF-ATTN) layers present in the backbone LLM (c.f. Fig. 2). The grid level features $\bar{\mathbf{I}}$ are first transformed using a multi-layer perceptron (MLP) for every cross-attention layer. For example, we learn a mapping $\mathrm{MLP}_k : \mathbb{R}^{q'} \rightarrow \mathbb{R}^q$ to transform the grid level features $\bar{\mathbf{v}}_i$ as $\bar{\mathbf{v}}_i^k = \mathrm{MLP}_k(\bar{\mathbf{v}}_i)$, for use as input to the first top-down cross attention layer. Furthermore, to preserve spatial information, we concatenate positional embeddings to each grid element $\bar{\mathbf{v}}_i^k$. The grid level image features $\bar{\mathbf{v}}_i^k$ fused with positional embeddings allow for top-down attention where the LLM guides the information extraction process using the representation

$\mathbf{h}^k$. We use the representation $\hat{\mathbf{h}}^k$ after the application of the self-attention layer to guide the visual feature extraction process in the cross-attention layer. From Fig. 2, the hidden representation $\hat{\mathbf{h}}^k$ is transformed by a linear projection to serve as the query vector (Q_s) and the visual features $\bar{\mathbf{v}}_i^k$ are linearly transformed to the keys and values (K_v, V_v) of the cross attention layer respectively,

$$
\begin{aligned}
\hat{\mathbf{h}}^k &= \text{SELF-ATTN}(\mathbf{h}^k) \\
\hat{\mathbf{v}}_i^k &= \text{CROSS-ATTN}(\hat{\mathbf{h}}^k, \bar{\mathbf{v}}_i^k) \\
\mathbf{h}^{k+1} &= \mathbf{h}^k + \hat{\mathbf{h}}^k + \hat{\mathbf{v}}_i^k \\
\mathbf{h}^{k+1} &= \text{FFN}(\mathbf{h}^{k+1}) + \mathbf{h}^{k+1}
\end{aligned}
\tag{1}
$$

where FFN denotes a feedforward layer defined the same way as in [55]. We use the hidden state $\hat{\mathbf{h}}^i$ as a query vector that encodes global semantics in the cross-attention layers with the spatial grid features $\bar{\mathbf{v}}_i^k$ as keys and values. This allows the LLM to extract information relevant to solving visual reasoning problems, including object locations and their spatial relationships in $\hat{\mathbf{v}}_i^k$. The hidden representation at level $k+1$ now includes information from both the textual ($\hat{\mathbf{h}}^k$) and visual domains ($\hat{\mathbf{v}}_i^k$) and is thus multi-modal and includes low-level visual information. This is instrumental in generating rationales for visual reasoning tasks as discussed in the following.

D. ACRE

Training details. We base our LRR models on the OPT-125M and ResNet-101 backbones. We train until convergence ($\sim$ 500k iterations) with a batch size of 1. We use the AdamW optimizer [30] with a learning rate of 1×10^{-5}, $\beta_1 = 0.9$, $\beta_2 = 0.95$ and λ (weight decay) $= 0.1$ and gradient clipping with a norm of 1.0.

Rationale construction. The key low-level visual challenge in the ACRE dataset is to identify objects in the context trials and to detect whether the blicket machine is activated. Therefore, we design the rationale with the surrogate tasks of object recognition and re-identification across the context trials. The rationale for each context trial describes the objects present and also assigns an unique integer ID to allow for re-identification. Additionally, the rationale also identifies state of the blicket machine (on/off). From Table 3 (top): Trial 1 with objects: 1(medium gray rubber cylinder) causes blicket machine to go: off. Finally, the rationale re-identifies the objects in the query image, e.g., from Table 3 (top): Will the query with objects: 1(medium gray rubber cylinder), 5(medium blue metal sphere) activate the blicket? This allows our LRR model to exploit the ("remembered") previous steps in the rationale to infer which context trials involved the objects in the query as well as the state of the blicket machine in the relevant trials. The model can

Model	Compositional					Systematic				
	All	D.R.	I.D.	S.O.	B.B.	All	D.R.	I.D.	S.O.	B.B
CNN-BERT [8]	43.7	54.0	46.8	40.5	28.7	39.9	55.9	68.2	0.0	45.5
NS-OPT [64]	69.0	92.5	76.0	88.3	13.4	67.4	94.7	88.3	82.7	16.0
ALOE [8]	91.7	97.1	90.8	96.8	78.8	93.9	97.1	71.2	98.9	94.4
OPT-125M+CLIP	83.6	95.7	70.5	87.8	67.4	83.8	95.0	68.1	87.1	74.6
OPT-125M+ViT	96.9	99.4	95.0	97.3	93.5	96.7	99.1	95.0	98.3	93.3
LRR (w/o Surrogate Re-ID)	89.7	97.6	68.3	85.4	92.3	90.2	97.5	74.7	84.3	94.2
LRR (Ours)	**99.3**	**99.8**	**98.5**	**99.5**	**98.7**	**99.0**	**99.8**	**98.4**	**99.8**	**97.6**

Table 2. Evaluation on the ACRE dataset, where, D.R. – Direct evidence, I.D. – Indirect evidence, S.O. – Screened-off and B.B. – Backward Blocked subsets.

then aggregate the information in the rationale to reason and arrive at the final answer.

Additional baselines and ablations. We compare to several baselines in Table 2. To highlight the importance of our rationale generation process, we consider a baseline without the surrogate re-identification task: LRR (w/o Surrogate Re-ID). To highlight the importance of spatial grid based features along with top-down attention, we consider (with surrogate Re-ID), 1. A OPT-125M model with visual input at the first OPT (token embedding) layer using (global) CLIP [9] embeddings, as in FORMAGe[10]. 2. A OPT-125M model with visual input at the first OPT (token embedding) layer, as in PaLM-E [10]. The visual input is patch-based, from ViT [9].

We see that without the surrogate Re-ID task, our LRR model shows weak performance. This highlights the importance of "Look, Remember, Reason" paradigm where we explicitly solve the crucial Re-ID task and "remember" the results for each context trial. The importance of spatial grid features is illustrated by the weak performance of the OPT-125M+CLIP model, which is unable to effectively capture low-level visual cues due the pooling introduced by the CLIP model. Although the OPT-125M+ViT model uses spatial grid features, it's performance is limited by the lack of top-down attention guided by the rich representations of the LLM. Furthermore, LRR model outperforms the state-of-the-art ALOE [8] model by a large margin on both the compositional (where the training and test sets contain different visual features) and systematic splits (different numbers of activated machines in the context trials) of the ACRE dataset. The gain in performance is especially significant in the backward blocked subset (B.B.) where the blicketness cannot be inferred from correlation alone due to the presence of confounding objects and the indirect subset (I.D.) where information needs to be integrated from multiple context trials. This performance advantage is due to the step-by-step reasoning enabled by our rationales which allows the model to aggregate visual information across multiple context trials.

**Qualitative examples.** We include qualitative examples in Table 3, along with our rationales. These examples illustrate that using our surrogate re-identification task, our LRR model can aggregate information from multiple context trials to arrive at the final answer – following the paradigm of "Look, Remember, Reason".

E. CLEVR

Training details. We train our LRR model with the OPT-1.3B and ResNet-101 backbones for 2 epochs (inline with prior work [48]) with a batch size of 1. We use the AdamW optimizer [30] with a learning rate of 1×10^{-5}, $\beta_1 = 0.9$, $\beta_2 = 0.95$ and λ (weight decay) = 0.1. We use gradient clipping with a norm of 1.0.

Rationale construction. As discussed in the main paper, we use the functional programs in the CLEVR dataset to construct rationales. The functional programs consist of simpler sub-routines that follow a tree-like execution structure. The sub-routines are converted into sentences using templates shown in Table 4. These templates are designed to include surrogate tasks, *e.g.*, object recognition or object localization, important for spatial reasoning in CLEVR. In detail, for the "Scene" sub-routine, the rationale requires the model to explicitly list (thus recognize) all objects in the scene including their size, color, material and shape. For the "Relate" sub-routine the rationale requires the model to explicitly list all objects at the front, behind, left or right of a target object with their spatial positions in the scene. Once these surrogate tasks have been solved, the solution remains within the context window of the model and is thus "remembered" by the model for subsequent tasks.

Additional baselines and ablations. We begin with a comparison to state-of-the-art models in Table 5. From the results in Table 5, we see that non-LLM-based methods such as MDETR [20] or NS-VQA [63], perform the best. However, such methods use a fine-tuned vision backbone (FT vision), based on DETR [4] or Mask-RCNN [14]. The advantage of a fine-tuned vision backbone on CLEVR is mainly due to improved object detection performance in the presence of occlusions. However, the use of a fine-tuned vision back-
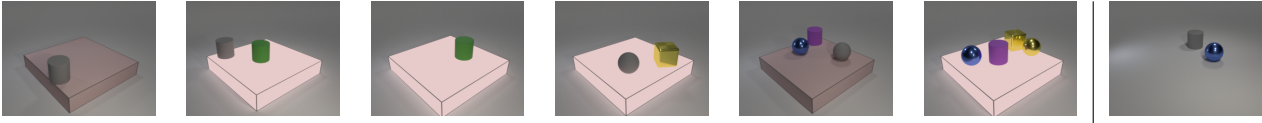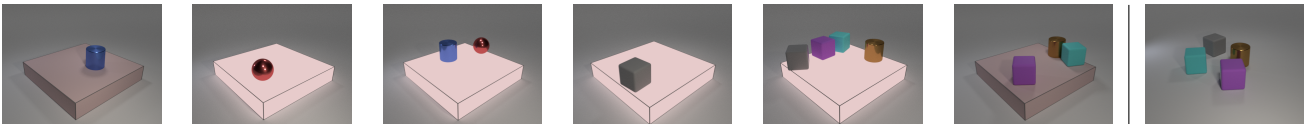
Rationale: The task is to detect whether the following combination of objects activate the blicket machine, which activates when we put a blicket on it. Next, we will conduct six trials where we put a combination of objects on the machine. Trial 1 with objects: 1(medium gray rubber cylinder) causes the blicket machine to go: off. Trial 2 with objects: 1(medium gray rubber cylinder), 2(medium green rubber cylinder) causes the blicket machine to go: on. Trial 3 with objects: 2(medium green rubber cylinder) causes the blicket machine to go: on. Trial 4 with objects: 3(medium gray rubber sphere), 4(medium yellow metal cube) causes the blicket machine to go: on. Trial 5 with objects: 5(medium blue metal sphere), 6(medium purple rubber cylinder), 3(medium gray rubber sphere) causes the blicket machine to go: off. Trial 6 with objects: 5(medium blue metal sphere), 6(medium purple rubber cylinder), 4(medium yellow metal cube), 7(medium yellow metal sphere) causes the blicket machine to go: on. Will the query with objects: 1(medium gray rubber cylinder), 5(medium blue metal sphere) activate the blicket? No.
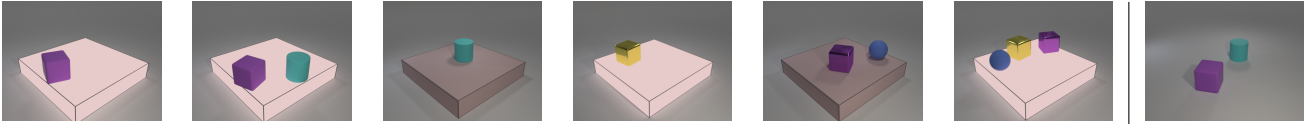
Rationale: The task is to detect whether the following combination of objects activate the blicket machine, which activates when we put a blicket on it. Next, we will conduct six trials where we put a combination of objects on the machine. Trial 1 with objects: 1(medium purple rubber cube) causes the blicket machine to go:on. Trial 2 with objects: 1(medium purple rubber cube), 2(medium cyan rubber cylinder) causes the blicket machine to go: on. Trial 3 with objects: 2(medium cyan rubber cylinder) causes the blicket machine to go: off. Trial 4 with objects: 3(medium yellow metal cube) causes the blicket machine to go: on. Trial 5 with objects: 4(medium purple metal cube), 5(medium blue rubber sphere) causes the blicket machine to go: off. Trial 6 with objects: 5(medium blue rubber sphere), 3(medium yellow metal cube), 4(medium purple metal cube) causes the blicket machine to go: on. Will the query with objects: 1(medium purple rubber cube), 2(medium cyan rubber cylinder) activate the blicket? Yes.
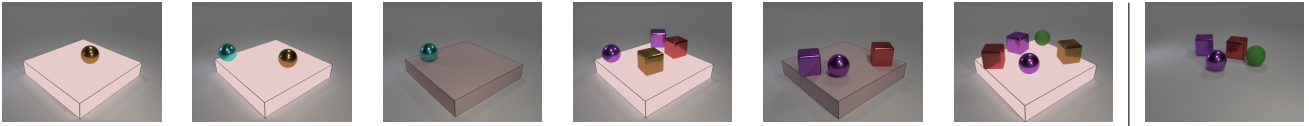
Rationale: The task is to detect whether the following combination of objects activate the blicket machine, which activates when we put a blicket on it. Next, we will conduct six trials where we put a combination of objects on the machine. Trial 1 with objects: 1(medium brown metal sphere) causes the blicket machine to go: on. Trial 2 with objects: 2(medium cyan metal sphere), 1(medium brown metal sphere) causes the blicket machine to go: on. Trial 3 with objects: 2(medium cyan metal sphere) causes the blicket machine to go: off. Trial 4 with objects: 3(medium purple metal sphere), 4(medium brown metal cube), 5(medium purple metal cube), 6(medium red metal cube) causes the blicket machine to go: on. Trial 5 with objects: 5(medium purple metal cube), 3(medium purple metal sphere), 6(medium red metal cube) causes the blicket machine to go: off. Trial 6 with objects: 6(medium red metal cube), 5(medium purple metal cube), 3(medium purple metal sphere), 7(medium green rubber sphere), 4(medium brown metal cube) causes the blicket machine to go: on. Will the query with objects: 5(medium purple metal cube), 3(medium purple metal sphere), 6(medium red metal cube), 7(medium green rubber sphere) activate the blicket? Maybe.

Rationale: The task is to detect whether the following combination of objects activate the blicket machine, which activates when we put a blicket on it. Next, we will conduct six trials where we put a combination of objects on the machine. Trial 1 with objects: 1(medium blue metal cylinder) causes the blicket machine to go: off. Trial 2 with objects: 2(medium red metal sphere) causes the blicket machine to go: on. Trial 3 with objects: 1(medium blue metal cylinder), 2(medium red metal sphere) causes the blicket machine to go: on. Trial 4 with objects: 3(medium gray rubber cube) causes the blicket machine to go: on. Trial 5 with objects: 3(medium gray rubber cube), 4(medium purple rubber cube), 5(medium cyan rubber cube), 6(medium brown metal cylinder) causes the blicket machine to go: on. Trial 6 with objects: 4(medium purple rubber cube), 6(medium brown metal cylinder), 5(medium cyan rubber cube) causes the blicket machine to go: off. Will the query with objects: 5(medium cyan rubber cube), 3(medium gray rubber cube), 4(medium purple rubber cube), 6(medium brown metal cylinder) activate the blicket? Yes.

Table 3. Example rationales generated by our LRR moddel on ACRE.

Sub-routine	Input	Rationale Template
Scene	objects	Step i, the objects in the scene are <[size,color,material,shape] objects>.
Filter	objects, attribute, value	Step j, from i, the objects of <attribute, value> are <[size,color,material,shape] objects>.
Relate	*front*, object	Step j, from i, the objects in the scene in front of <object> are <[size,color,material,shape] object> at <pos>, ..., <[size,color,material,shape] object> at <pos>.
Relate	*behind*, object	Step j, from i, the objects in the scene behind <object> are <[size,color,material,shape] object> at <pos>, ..., <[size,color,material,shape] object> at <pos>.
Relate	*left*, object	Step j, from i, the objects in the scene at the left of <object> are <[size,color,material,shape] object> at <pos>, ..., <[size,color,material,shape] object> at <pos>.
Relate	*right*, object	Step j, from i, the objects in the scene at the right of <object> are <[size,color,material,shape] object> at <pos>, ..., <[size,color,material,shape] object> at <pos>.
Intersect	objects	Step k, from i and j, the common objects are <[size,color,material,shape] object in objects>.
Union	objects	Step k, from i and j, the common objects are <[size,color,material,shape] object in objects>.
Query	*attribute*, value, object	Step j, from i, the <attribute> of the <[size,color,material,shape] object> is <value>.
Equal	objects	Step k, from i and j, there are an equal number of objects.
Less, More	objects	Step k, from i and j, there are less/more objects in step i than j.
Exist	object	Step j, from i, there is such an object / there is no such object.
Count	objects	Step j, from i, there are <number of objects> such objects.
Equal	*attribute*, objects	Step j, from i, the objects <[size,color,material,shape] object> and <[size,color,material,shape] object> have/ do not have the same <attribute>.
Unique	objects	Step j, from i, the object is unique.

Table 4. Example rationale templates for sub-routines in CLEVR (*c.f.* Figure 2 in [19]).

Method	FT vision	Overall	Count	Exist	Compare Numbers	Query Attribute	Compare Attribute
CNN+LSTM+RN [33]	✓	95.5	90.1	97.8	93.6	97.9	97.1
CNN+LSTM+FILM [39]	✓	97.6	94.3	99.3	93.4	99.3	99.3
HAN+RN [48]	✓	98.8	97.2	99.6	96.9	99.6	99.6
OCCAM [57]	✓	99.4	98.1	99.8	99.0	99.9	99.9
NS-VQA [63]	✓	**99.7**	**99.9**	**99.9**	**99.8**	99.8	99.8
MDETR [20]	✓	**99.7**	99.3	**99.9**	99.4	**99.9**	**99.9**
CNN+LSTM+SAN [19]	✗	68.5	52.2	71.1	73.5	85.3	52.3
LRR (Ours)	✗	**97.9**	**95.6**	**98.7**	**98.7**	**98.5**	**98.3**

Table 5. Evaluation on the CLEVR dataset, comparing to state-of-the-art.

bones like DETR or Mask-RCNN makes it more challenging to apply the same model architecture across diverse visual reasoning tasks, *e.g.* the moving camera split of CATER (*c.f.* Table 10). Furthermore, DETR or Mask-RCNN type backbones require bounding box annotations, which are not always available. Even without a fine-tuned vision backbone, our LRR model outperforms FiLM [39] which employs a fine-tuned ResNet-101.

Importance of rationales. We highlight the importance of rationales with surrogate tasks for visual reasoning on the CLEVR dataset in Table 7. To this end, we consider the a LRR (w/o Surrogate Spatial Reasoning) baseline, without

the surrogate object recognition and spatial reasoning tasks. We evaluate this baseline with various OPT model sizes: 125M, 1.3B and 6.7B. Additionally, we consider an *Oracle-Perception* baseline, that does not generate rationales but receives a ground truth list of objects and their corresponding positions as a prompt in addition to the query image. This baseline essentially has near perfect vision, *i.e.*, it receives all visual information essential for answering the question in the prompt. However, it lacks step-by-step reasoning capabilities.

We observe in Table 7 that without our low-level surrogate tasks, accuracy does not improve substantially even with

Method	Static Camera			Moving Camera		
	Top-1(↑)	Top-5(↑)	L1(grid;↓)	Top-1(↑)	Top-5(↑)	L1(grid;↓)
R3D LSTM [12]	60.2	81.8	1.2	28.6	63.3	1.7
R3D + NL LSTM [12]	46.2	69.9	1.5	38.6	70.2	1.5
ALOE [8]	74.0	94.0	0.44	59.7	90.1	0.69
OPNet[†] [49]	74.8	-	0.54	-	-	-
Hopper[†] [68]	73.2	93.8	0.85	-	-	-
TFC V3D Depthwise[†] [67]	79.7	95.5	0.47	-	-	-
LRR (w/o Surrogate Tracking)	61.7	82.4	0.73	49.3	65.8	1.23
LRR (Ours)	**85.1**	**96.2**	**0.23**	**75.1**	**91.9**	**0.48**

Table 6. Evaluation on the CATER dataset ([†]results reported only for static camera).

larger LLM model sizes. Further, we observe that while the Oracle-Perception baseline is provided all necessary visual information to the model via the prompt, it is unable to use this information effectively without the step-by-step reasoning process enabled by our rationales. Thus, rationales with our low-level surrogate tasks are crucial for visual reasoning on the CLEVR dataset.

Rationale types. Here, we evaluate the effectiveness of our proposed rationales that enables our LRR model to "Look, Remember, Reason". To do this, we consider alternative rationale types. First, we consider the *Object-List-Position* rationale that generates the list of objects and their corresponding positions in the scene. This enables the model to extract relevant visual features from the query image but does not enable step-by-step reasoning. Secondly, we consider rationales from the CLEVR-X dataset [46]. These rationales describe the reasoning and visual information that is necessary to answer a given question. However, they are concise and do not enable step-by-step reasoning. We report the results in Table 8. Finally, we consider a variant of our "Look, Remember, Reason" rationales that does not include the surrogate task of inferring spatial relations (w/o Spatial Relations), corresponding to the sub-routine "Relate" as detailed in Table 8.

The performance of our LRR model with Object-List-Position and CLEVR-X[46] rationales is significantly worse (*c.f*. Table 1 in the main paper). This is because the Object-List-Position rationales do not allow for step-by-step reasoning and the CLEVR-X rationales are too concise – thus not all important steps required to arrive at the final answer are included in the rationale. Finally, we see that removing surrogate tasks from the rationale as in the LRR (w/o Surrogate Spatial Relations) baseline has an adverse effect on performance. This again highlights the importance of surrogate tasks.

Qualitative examples. We include qualitative examples including rationales with surrogate tasks in Table 9.

F. CATER

Training details. We train our LRR model with the OPT-125M and ResNet-101 backbones until convergence ($\sim$ 600k iterations) with a batch size of 1. We use the AdamW optimizer [30] with a learning rate of 1×10^{-5}, $\beta_1 = 0.9$, $\beta_2 = 0.95$ and λ (weight decay) $= 0.1$. We use gradient clipping with a norm of 1.0. We train a joint model for the static and moving camera splits. In order to deal with the limited training set size, we augment the training set by *jittering* the grid positions of the cones and the snitch. In detail, jittering grid positions means that we replace the groundtruth grid position with a neighbouring (8 connected) grid position with a fixed probability (we found 0.25 to be adequate). Note that, only the input sequence to the LLM is jittered and the target sequence remains unperturbed.

Baselines and evaluation. To highlight the importance of our rationale generation process, we consider a baseline where the rationale consists only of the grid positions of the snitch at every video frame, without the surrogate task of tracking the cones, LRR (w/o Surrogate Tracking). Both our LRR model and baselines are based on the OPT-125M backbone. Our LRR model is trained jointly on both the static and moving camera splits, similar to ALOE [8]. The results are presented along with a comparison to state-of-the-art models in Table 6. Note that, OPNet [49], Hopper [68], TFC V3D Depthwise [67] and Loci [53] report results only on the static camera split. Loci [53] reports an impressive 90.7% accuracy on the static camera split, but it is not applicable to the moving camera split due to its static background and camera model.

Our LRR model outperforms TFC V3D Depthwise [67] model on the static camera and ALOE [8] on the challenging moving camera split by a large margin. The large performance gain over the LRR (w/o Surrogate Tracking) baseline shows the advantage of using surrogate tracking tasks in the rationale. This shows that our rationales help capture long-term spatio-temporal correlations by "remembering" the intermediate positions of the objects of interest within

Method	OPT Model Size	Accuracy
LRR (w/o Surrogate Spatial Reasoning)	125M	50.9
LRR (w/o Surrogate Spatial Reasoning)	1.3B	51.4
LRR (w/o Surrogate Spatial Reasoning)	6.7B	53.4
Oracle-Perception	1.3B	51.5

Table 7. Evaluation of LRR baselines without rationales.

Method	Accuracy
LRR (Object-List-Position)	69.2
LRR (CLEVR-X [46])	72.2
LRR (w/o Spatial Relations)	94.2

Table 8. Evaluation of rationale types (OPT-1.3B).

the context window of the LLM. Without the multi-target tracking surrogate task, the model does not learn to track the cones and thus fails in cases of containment.

Qualitative examples. We include qualitative examples including both the static and moving camera splits in Table 10. We see that our LRR model can successfully deal with containment in both static camera (rows 1-3; Table 10) and moving camera (row 4-5; Table 10) settings, due to our rationale that explicitly tracks cones and the snitch. Note that the example in row 5 in Table 10 from the moving camera split is especially challenging due to recursive containment.

References

[1] Armen Aghajanyan, Bernie Huang, Candace Ross, Vladimir Karpukhin, Hu Xu, Naman Goyal, Dmytro Okhonko, Mandar Joshi, Gargi Ghosh, Mike Lewis, and Luke Zettlemoyer. CM3: A causal masked multimodal model of the internet. *CoRR*, abs/2201.07520, 2022.

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob L. Menick, Sebastian Borgeaud, Andy Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karén Simonyan. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.

[3] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[5] Ting Chen, Saurabh Saxena, Lala Li, Tsung-Yi Lin, David J. Fleet, and Geoffrey E. Hinton. A unified sequence interface for vision tasks. In *NeurIPS*, 2022.

[6] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. *CoRR*, abs/2204.02311, 2022.

[7] Artur S. d'Avila Garcez and Luís C. Lamb. Neurosymbolic AI: the 3rd wave. *CoRR*, abs/2012.05876, 2020.

[8] David Ding, Felix Hill, Adam Santoro, Malcolm Reynolds, and Matt M. Botvinick. Attention over learned object embeddings enables complex visual reasoning. In *NeurIPS*, 2021.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[10] Danny Driess, Fei Xia, Mehdi S. M. Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, Wenlong Huang, Yev-
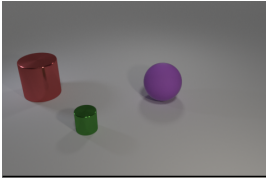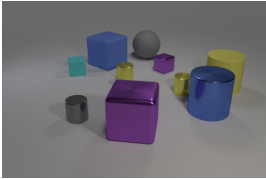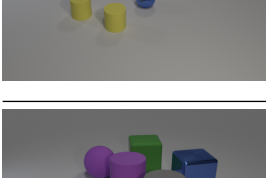
Query Image	Rationale

Question: What number of things are shiny cylinders that are in front of the large rubber thing or purple matte things?
Rationale: Step 1, the objects in the scene are large red metal cylinder, small green metal cylinder, large purple rubber sphere. Step 2, from step 1, the objects of size large, material rubber are large purple rubber sphere. Step 3, from step 2, this object is unique. Step 4, in front of the large purple rubber sphere at (58,29) the objects in the scene are small green metal cylinder at (30,43). Step 5, from step 4, the objects of material metal, shape cylinder are small green metal cylinder. Step 6, the objects in the scene are large red metal cylinder, small green metal cylinder, large purple rubber sphere. Step 7, from step 6, the objects of color purple, material rubber are large purple rubber sphere. Step 8, from steps 5 and 7, the common objects are small green metal cylinder, large purple rubber sphere. Step 9, from step 8, there are 2 such objects. The answer is, 2.

Question: How many objects are cyan cylinders or tiny things to the left of the big rubber ball?
Rationale: Step 1, the objects in the scene are small cyan rubber cube, small gray metal cylinder, large blue rubber cube, small yellow metal cylinder, large purple metal cube, large gray rubber sphere, small purple metal cube, small yellow metal cylinder, large blue metal cylinder, large yellow rubber cylinder. Step 2, from step 1, there are no objects of color cyan, shape cylinder. Step 3, the objects in the scene are small cyan rubber cube, small gray metal cylinder, large blue rubber cube, small yellow metal cylinder, large purple metal cube, large gray rubber sphere, small purple metal cube, small yellow metal cylinder, large blue metal cylinder, large yellow rubber cylinder. Step 4, from step 3, the objects of size large, material rubber, shape sphere are large gray rubber sphere. Step 5, from step 4, this object is unique. Step 6, at the left of the large gray rubber sphere at (52,15) the objects in the scene are small cyan rubber cube at (27,23), small gray metal cylinder at (27,39), large blue rubber cube at (38,17), small yellow metal cylinder at (44,25), large purple metal cube at (47,39). Step 7, from step 6, the objects of size small are small cyan rubber cube, small gray metal cylinder, small yellow metal cylinder. Step 8, from steps 2 and 7, the common objects are small cyan rubber cube, small gray metal cylinder, small yellow metal cylinder. Step 9, from step 8, there are 3 such objects. The answer is, 3.

Question: How many objects are either big red spheres or small yellow cylinders in front of the purple rubber block?
Rationale: Step 1, the objects in the scene are small yellow rubber cylinder, small purple rubber cube, small yellow rubber cylinder, small blue metal sphere. Step 2, from step 1, there are no objects of size large, color red, shape sphere. Step 3, the objects in the scene are small yellow rubber cylinder, small purple rubber cube, small yellow rubber cylinder, small blue metal sphere. Step 4, from step 3, the objects of color purple, material rubber, shape cube are small purple rubber cube. Step 5, from step 4, this object is unique. Step 6, in front of the small purple rubber cube at (36,26) the objects in the scene are small yellow rubber cylinder at (28,37), small yellow rubber cylinder at (41,41), small blue metal sphere at (52,34). Step 7, from step 6, the objects of size small, color yellow, shape cylinder are small yellow rubber cylinder, small yellow rubber cylinder. Step 8, from steps 2 and 7, the common objects are small yellow rubber cylinder, small yellow rubber cylinder. Step 9, from step 8, there are 2 such objects. The answer is, 2.

Question: Is the material of the cylinder that is behind the tiny shiny ball the same as the object that is on the right side of the big gray matte thing?
Rationale: Step 1, the objects in the scene are small cyan metal sphere, large purple rubber sphere, large purple rubber cylinder, large green rubber cube, large gray rubber cylinder, large blue metal cube. Step 2, from step 1, the objects of size small, material metal, shape sphere are small cyan metal sphere. Step 3, from step 2, this object is unique. Step 4, behind the small cyan metal sphere at (27,28) The objects in the scene are large purple rubber sphere at (35,19), large purple rubber cylinder at (45,23), large green rubber cube at (51,16), large blue metal cube at (69,22). Step 5, from step 4, the objects of shape cylinder are large purple rubber cylinder. Step 6, from step 5, this object is unique. Step 7, from step 6, the material of the large purple rubber cylinder is rubber. Step 8, the objects in the scene are small cyan metal sphere, large purple rubber sphere, large purple rubber cylinder, large green rubber cube, large gray rubber cylinder, large blue metal cube. Step 9, from step 8, the objects of size large, color gray, material rubber are large gray rubber cylinder. Step 10, from step 9, this object is unique. Step 11, at the right of the large gray rubber cylinder at (58,31) the objects in the scene are large blue metal cube at (69,22). Step 12, from step 11, this object is unique. Step 13, from step 12, the material of the large blue metal cube is metal. Step 14, from steps 7 and 13, the objects large purple rubber cylinder and large blue metal cube do not have the same material. The answer is, no.

Question: What number of big rubber things are in front of the big object that is in front of the sphere?
Rationale: Step 1, the objects in the scene are small purple metal sphere, large yellow rubber cylinder, large red metal cylinder. Step 2, from step 1, the objects of shape sphere are small purple metal sphere. Step 3, from step 2, this object is unique. Step 4, in front of the small purple metal sphere at (12,30) the objects in the scene are large yellow rubber cylinder. Step 5, from step 4, the objects of size large are large yellow rubber cylinder. Step 6, from step 5, this object is unique. Step 7, in front of the large yellow rubber cylinder at (33,30) there are no objects in the scene. Step 8, from step 7, there are no objects of size large, material rubber. Step 9, from step 8, there are 0 such objects. The answer is, 0

Table 9. Example rationales generated by our LRR model on CLEVR.

gen Chebotar, Pierre Sermanet, Daniel Duckworth, Sergey Levine, Vincent Vanhoucke, Karol Hausman, Marc Toussaint, Klaus Greff, Andy Zeng, Igor Mordatch, and Pete Florence. Palm-e: An embodied multimodal language model. *CoRR*, abs/2303.03378, 2023.

[11] Constantin Eichenberg, Sidney Black, Samuel Weinbach, Letitia Parcalabescu, and Anette Frank. MAGMA - multimodal augmentation of generative models through adapter-based finetuning. In *EMNLP*, 2022.

[12] Rohit Girdhar and Deva Ramanan. CATER: A diagnostic dataset for compositional actions & temporal reasoning. In *ICLR*, 2020.

[13] Tanmay Gupta and Aniruddha Kembhavi. Visual programming: Compositional visual reasoning without training. *CoRR*, abs/2211.11559, 2022.

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *ICCV*, 2017.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[16] Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. Learning to reason: End-to-end module networks for visual question answering. In *ICCV*, 2017.

[17] Drew A. Hudson and Christopher D. Manning. Compositional attention networks for machine reasoning. In *ICLR*, 2018.

[18] Brian Ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry
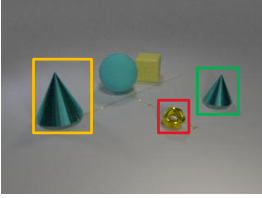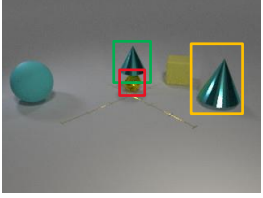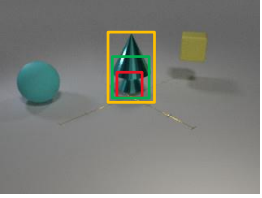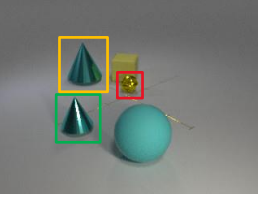
Table 10. Example rationales generated by our LRR model on CATER. The rationales contain the surrogate task of multi-target tracking. We show the predicted grid locations of the cones and the snitch below.

Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Nikhil J. Joshi, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. Do as I can, not as I say: Grounding language in robotic affordances. In *CoRL*, 2022.

[19] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, and Ross B. Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual

reasoning. In *CVPR*, 2017.

[20] Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. MDETR - modulated detection for end-to-end multi-modal understanding. In *ICCV*, 2021.

[21] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *CoRR*, abs/2001.08361, 2020.

[22] Jing Yu Koh, Ruslan Salakhutdinov, and Daniel Fried. Grounding language models to images for multimodal generation. *CoRR*, abs/2301.13823, 2023.

[23] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *NeurIPS*, 2022.

[24] Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay V. Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, Yuhuai Wu, Behnam Neyshabur, Guy Gur-Ari, and Vedant Misra. Solving quantitative reasoning problems with language models. In *NeurIPS*, 2022.

[25] Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. *CoRR*, abs/2301.12597, 2023.

[26] Junnan Li, Dongxu Li, Caiming Xiong, and Steven C. H. Hoi. BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

[27] Zhuowan Li, Elias Stengel-Eskin, Yixiao Zhang, Cihang Xie, Quan Tran, Benjamin Van Durme, and Alan L. Yuille. Calibrating concepts and operations: Towards symbolic reasoning on real images. In *ICCV*, 2021.

[28] Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In *ACL*, 2017.

[29] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *CoRR*, abs/2304.08485, 2023.

[30] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[31] Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. Chameleon: Plug-and-play compositional reasoning with large language models. *CoRR*, abs/2304.09842, 2023.

[32] Shweta Mahajan and Stefan Roth. Diverse image captioning with context-object split latent spaces. In *NeurIPS*, 2020.

[33] Mateusz Malinowski, Carl Doersch, Adam Santoro, and Peter W. Battaglia. Learning visual question answering by bootstrapping hard attention. In *ECCV*, 2018.

[34] Oscar Mañas, Pau Rodríguez López, Saba Ahmadi, Aida Nematzadeh, Yash Goyal, and Aishwarya Agrawal. MAPL: parameter-efficient adaptation of unimodal pre-trained models for vision-language few-shot prompting. In *EACL*, 2023.

[35] Gary Marcus. Deep learning: A critical appraisal. *CoRR*, abs/1801.00631, 2018.

[36] Gary Marcus. The next decade in AI: four steps towards robust artificial intelligence. *CoRR*, abs/2002.06177, 2020.

[37] Shanka Subhra Mondal, Taylor Webb, and Jonathan D. Cohen. Learning to reason over visual objects. *CoRR*, abs/2303.02260, 2023.

[38] Maxwell I. Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, Charles Sutton, and Augustus Odena. Show your work: Scratchpads for intermediate computation with language models. *CoRR*, abs/2112.00114, 2021.

[39] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018.

[40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[41] Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, H. Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant M. Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew J. Johnson, Blake A. Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Edward Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. Scaling language models: Methods, analysis & insights from training gopher. *CoRR*, abs/2112.11446, 2021.

[42] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR*, abs/1910.10683, 2019.

[43] Nasim Rahaman, Muhammad Waleed Gondal, Shruti Joshi, Peter V. Gehler, Yoshua Bengio, Francesco Locatello, and Bernhard Schölkopf. Dynamic inference with neural interpreters. In *NeurIPS*, 2021.

[44] Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. Explain yourself! leveraging language models for commonsense reasoning. In *ACL*, 2019.

[45] Gabriel Recchia. Teaching autoregressive language models complex tasks by demonstration. *CoRR*, abs/2109.02102, 2021.

[46] Leonard Salewski, A. Sophia Koepke, Hendrik P. A. Lensch, and Zeynep Akata. Clevr-x: A visual reasoning dataset for natural language explanations. In *xxAI - Beyond explainable Artificial Intelligence*. Springer, 2022.

[47] Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal V. Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Févry, Jason Alan Fries, Ryan Teehan, Teven Le Scao, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. Multitask prompted training enables zero-shot task generalization. In *ICLR*, 2022.

[48] Adam Santoro, David Raposo, David G. T. Barrett, Mateusz Malinowski, Razvan Pascanu, Peter W. Battaglia, and Tim Lillicrap. A simple neural network module for relational reasoning. In *NeurIPS*, 2017.

[49] Aviv Shamsian, Ofri Kleinfeld, Amir Globerson, and Gal Chechik. Learning object permanence from video. In *ECCV*, 2020.

[50] Florian Strub, Mathieu Seurin, Ethan Perez, Harm de Vries, Jérémie Mary, Philippe Preux, Aaron C. Courville, and Olivier Pietquin. Visual reasoning with multi-hop feature modulation. In *ECCV*, 2018.

[51] Dídac Surís, Sachit Menon, and Carl Vondrick. Vipergpt: Visual inference via python execution for reasoning. *CoRR*, abs/2303.08128, 2023.

[52] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971, 2023.

[53] Manuel Traub, Sebastian Otte, Tobias Menge, Matthias Karlbauer, Jannik Thümmel, and Martin V. Butz. Learning what and where - unsupervised disentangling location and identity tracking. In *ICLR*, 2023.

[54] Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models. In *NeurIPS*, 2021.

[55] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[56] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *CoRR*, abs/2203.11171, 2022.

[57] Zhonghao Wang, Kai Wang, Mo Yu, Jinjun Xiong, Wen-Mei Hwu, Mark Hasegawa-Johnson, and Humphrey Shi. Interpretable visual reasoning via induced symbolic space. In *ICCV*, 2021.

[58] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. *CoRR*, abs/2109.01652, 2021.

[59] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*, 2022.

[60] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *CoRR*, abs/2210.03629, 2022.

[61] Yiqun Yao, Jiaming Xu, Feng Wang, and Bo Xu. Cascaded mutual modulation for visual reasoning. In *EMNLP*, 2018.

[62] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, Chenliang Li, Yuanhong Xu, Hehong Chen, Junfeng Tian, Qian Qi, Ji Zhang, and Fei Huang. mplug-owl: Modularization empowers large language models with multimodality. *CoRR*, abs/2304.14178, 2023.

[63] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic VQA: disentangling reasoning from vision and language understanding. In *NeurIPS*, 2018.

[64] Chi Zhang, Baoxiong Jia, Mark Edmonds, Song-Chun Zhu, and Yixin Zhu. ACRE: abstract causal reasoning beyond covariation. In *CVPR*, 2021.

[65] Chi Zhang, Sirui Xie, Baoxiong Jia, Ying Nian Wu, Song-Chun Zhu, and Yixin Zhu. Learning algebraic representation for systematic generalization in abstract reasoning. In *ECCV*, 2022.

[66] Renrui Zhang, Jiaming Han, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, Peng Gao, and Yu Qiao. Llamaadapter: Efficient fine-tuning of language models with zeroinit attention. *CoRR*, abs/2303.16199, 2023.

[67] Shiwen Zhang. Tfcnet: Temporal fully connected networks for static unbiased temporal reasoning. *CoRR*, 2022.

[68] Honglu Zhou, Asim Kadav, Farley Lai, Alexandru Niculescu-Mizil, Martin Renqiang Min, Mubbasir Kapadia, and Hans Peter Graf. Hopper: Multi-hop transformer for spatiotemporal reasoning. In *ICLR*, 2021.