Textual-Visual Logic Challenge: Understanding and Reasoning in Text-to-Image Generation Appendix

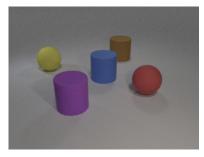
Peixi Xiong¹, Michael Kozuch¹, and Nilesh Jain¹

Intel Labs, USA {peixi.xiong, michael.a.kozuch, nilesh.jain}@intel.com

1 Dataset Overview

1.1 Dataset Specifications

Ground Truth Image



Text Prompts

Add a **blue cylinder** at the center . Add a **red sphere** in front of **it** on the right . Add a **purple cylinder** in front of **it** on the left and in front of **blue cylinder** on the left . Add a **brown cylinder** behind **it** on the right and behind **red sphere** on the left and behind **blue cylinder** on the right . Add a **yellow sphere** in front of **it** on the left and behind **purple cylinder** on the left and behind **red sphere** on the left and behind **blue cylinder**

Scene Graph

"image_index": 9, "image_filename": "TVLOGIC_detail_000009.png", "split": "detail", "objects": ["rotation": 322.6178642312034, "shape": "cylinder", "color": "blue", "3d_coords": [160, 101, 11.217621803283691], "pixel_coords": [160, 101, 11.217621803283691], "size": "large", "material": "rubber" },, ; "relationships": { "front": [[1, 2], [2], [], [0, 1, 2, 4], [0, 1, 2]], "behind": [[3, 4], [0, 3, 4], [0, 1, 3, 4], [], [3]], "left": [[2, 4], [0, 2, 3, 4], [4], [0, 2, 4], []], "right": [[1, 3], [], [0, 1, 3], [1], [0, 1, 2, 3]] }, "directions": { "below": [-0.0, -0.0, -1.0], ""get", [0 565211325279201] 0.7544003552005003

"left": [-0.6563112735748291, -0.7544902563095093, 0.0], "above": [0.0, 0.0, 1.0], "front": [0.754490315914154, -0.6563112735748291, -0.0], "behind": [-0.754490315914154, 0.6563112735748291, 0.0], "right": [0.6563112735748291, 0.7544902563095093, -0.0]

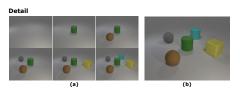
, "manipulation": None

Fig. 1: Overview of Our Annotated Dataset. For each text prompt, the dataset includes the corresponding ground truth image and a detailed scene graph. This graph serves as an enhanced reference by indicating the spatial location of each entity and mapping out their interrelations. It is important to note that the scene graph is not employed in either the training or evaluation phases of any experiments.

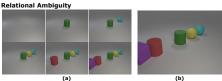
3

Figure 1 illustrates the format of the scene graph attached to our dataset annotations. The graph encompasses essential information, including the file

$\mathbf{2}$ P. Xiong et al.



Place a green cylinder at the middle the green cylinder . Place a gray spl to the left behind it . Insert a yellow Position a **brown sphere** to the left, in front of ere to the left, behind the **green cylinder** and ube to the right in front of it, to the right behind ft behind it . Insert a yellow cube to the right in front of it, to the right on sphere, and to the right in front of the green cylinder. Position a to the left behind it, to the right behind the gray sphere, to the right on sphere, and to the right behind the green cylinder. the brow the hr (c)



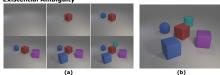
en cylinder in the middle . Place a to the right and behind it. yellow sphere in the space between the green cylinder and the wind yellow sphere in the space between the green cylinder and the cyan . Position a red cylinder to its left and in front, also to the left in front of the here, and to the left in front of the green cylinder . Position a purple cube t and in front, also to the left in front of the yellow sphere, to the left in fron sphere, and to the left in front of the green left and in front, also to the left in front of the cyan sphere, and to the left in front of the of the green cylinder (c)

Inference

(a) (b) Place a gray sphere along with two cyan and two brown cubes. Position a gray sphere at the middle. Have a cyan cube set up on its right, right in front of it. Situate a brown cube in front of it on its left side and to the right of the gray sphere - Position a brown cube to the left at its backside, at the left of the gray sphere and both behind and to the left of the cyan cube that is situated to the right of the gray sphere. The cyan cube should be positioned on the left, right in front of and to the left of the other cyan cube, to the left of the gray sphere. It also needs to be obtin in front of and to the left of the brown cube which is placed on the left in fro of the cyan cube and to the right of the gray sphere.

(c) Existential Ambiguity

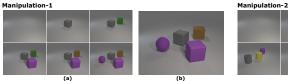
(a)

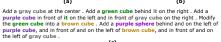


Add a red cube at the center . Add a blue re behind it on the left . Add a In fort of its due tendent and a finite of particle being to on the right and being blue on the right. The blue cube is in front of **fue cube** on the right and being blue on the right. The blue cube is in front of **blue sphere** on the right and being blue blue on the left. Add a cyan cylinder behing it on the right and behing burgle on the left and behing blue sphere on the right and behing reduce be on the right ube



(c)

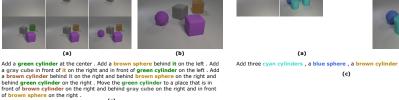








(c)





(b)

(c)

Fig. 2: Overview of Our Dataset. For each text prompt, in addition to the final ground truth image, we also attach a sequence of intermediate results to offer better reference points across certain categories. These sequences of images are not employed in either the training or evaluation processes.

index and name, detailed descriptions of the entities, and their interrelations. For each entity, attributes such as shape, color, size, and material are specified, along with spatial details like coordinates and rotation angles. Relationships between entities are articulated based on their relative positions in 3D space and are categorized by the type of relation. Additionally, to aid in the interpretation of

3

spatial orientations—left, up, and behind within the scene—reference directions are provided. This structured approach to annotation ensures a comprehensive and precise representation of the scene, facilitating an improved understanding and analysis for our Textual-Visual Logic challenge.

As indicated in Figure 2, the dataset includes not only the final ground truth images for each text prompt but also features a sequence of images associated with each annotation. This incorporation of intermediate results provides a valuable reference for further analysis, especially for tasks related to visual reasoning.

1.2 Generation Methodology

Figure 3 illustrates the dataset generation process, highlighting the creation of scene images through Blender [2] under predefined illumination conditions and camera directions, with added minor randomness to illumination for enhanced generalizability. In this dataset, each object—restricted to one of three shapes (cube, sphere, cylinder) and one of eight colors—maintains a consistent material and size. The initial image in each sequence positions the object at the image center, while subsequent images feature objects in random, yet visible and non-overlapping positions. Both the final image and the entire sequence are included in our dataset to provide comprehensive references.

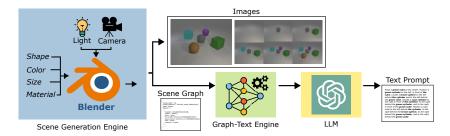


Fig. 3: Overview of the Annotation Generation Process: Blender and ChatGPT-4 are utilized in this process.

Annotations come with a scene graph delineating the relationships and positions of objects relative to the image center, structured through directed edges that represent left-right and front-back spatial relationships. To generate initial instructions, we employ a graph-text engine using simple text templates, such as "Add a [object color] [object shape] [relative position: depth] on the [relative position: horizontal]. " To increase variation in natural language, we utilize a Large Language Model, specifically ChatGPT-4 [4], to refine these instructions. For the refined prompt proposals, a human audit is conducted to filter out prompts that are mismatched in meaning, lack diversity in natural language, or contain other inaccuracies. The resulting text prompts, enriched with natural language diversity, are then incorporated into the dataset. 4 P. Xiong et al.

2 Experiment Settings and Details

2.1 Training Details and Evaluation Metrics

In our proposed model, a multi-head attention block comprising 8 heads is employed for co-attention learning, with parameters m and n within the multimodality fusion module set to 2 and 3, respectively. This configuration is based on the optimal outcomes derived from a series of tests. Relation tokens are generated using the dependency parser and POS tags from NLTK [1]. The learning rate for the Adam optimizer is established at 10^{-4} , with beta values set to (0.9, 0.999). Our model undergoes distributed training across 4 NVIDIA A100 GPUs.

The loss for the discriminator is defined as:

$$\mathcal{L}_{\mathcal{D}} = \mathcal{L}_{uNet}^{D} + \frac{1}{2} (\mathcal{L}_{info-G}^{D} + \mathcal{L}_{txt-G}^{D})$$
(1)

Each term in the equation is defined as:

$$\mathcal{L}_{uNet}^{D} = -\mathbb{E}[min(0, -1 + \mathbf{d}_{uNet}^{real})] -\mathbb{E}[min(0, -1 - \mathbf{d}_{uNet}^{fake})], \mathcal{L}_{info-G}^{D} = \mathbb{E}[d_{info-G}], \mathcal{L}_{txt-G}^{D} = -\mathbb{E}[min(0, -1 + d_{txt-G}^{real})] -\mathbb{E}[min(0, -1 - d_{txt-G}^{fake})] -\mathbb{E}[min(0, -1 - d_{txt-G}^{unpair})],$$

$$(2)$$

The loss for the generator is defined as:

$$\mathcal{L}_G = \mathcal{L}_{uNet}^G + \mathcal{L}_{txt-G}^G \tag{3}$$

Each term related to the generator loss is defined as:

$$\mathcal{L}^{G}_{uNet} = -\mathbb{E}[\mathbf{d}^{fake}_{uNet}],$$

$$\mathcal{L}^{G}_{txt-G} = -\mathbb{E}[d^{fake}_{txt-G}]$$
(4)

To address concerns regarding the object detector's performance as an evaluation metric, we complement it with reference metrics—namely, Average Precision (AP), Average Recall (AR), and F1 scores for the fully annotated image. These reference metrics are derived from a subset of the test split, based on detection results and their corresponding annotated scene graphs, yielding an AP of 0.98428, an AR of 0.97463, and an F1 score of 0.9794.

2.2 Experiment Settings

Figure 4 illustrates one of our experimental setups, where ChatGPT4 [4] and Blender [2] are utilized together to translate text prompts into visual 3D scenes. Initially, ChatGPT processes the prompts, converting the textual descriptions

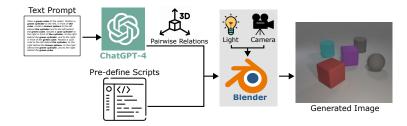


Fig. 4: Overview of the Experiment Setup for ChatGPT+Blender Integration.

into 3D coordinates. These coordinates are then integrated into predefined Blender scripts, leading to the generation of images that are rendered with predetermined view directions and lighting conditions. To optimize the performance of this process, we experimented with different settings of GPT prompts and selected the configuration that achieved the best results on the validation split.

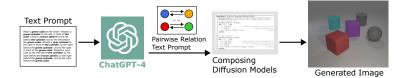


Fig. 5: Overview of the Experiment Setup for ChatGPT+Comp Integration.

Figure 5 presents another setting in our experiments. Due to the limitation of composing diffusion models [3]—their inability to process complex, logic-rich text prompts directly—these models require inputs to be simplified into pairwise relations. By employing ChatGPT to break down complex prompts into these simpler, executable forms, the models can generate visual content that reflects the specifications provided in the original text prompts. Similar to the above setting, to ensure the effectiveness of this approach, we tested various configurations of the text prompts with ChatGPT and adopted the method that yielded the most accurate visual representations according to our validation split.

3 Limitations

Figure 6 highlights certain limitations of our proposed baseline model through various failure cases. One notable issue arises with text prompts that contain variant syntax or rare vocabulary, leading to the model's difficulty in comprehending relationships and subsequently resulting in accumulated errors. For instance, in the first row, the term "rear" is not commonly used, and the sentence structure deviates from the typical format where the verb often leads. Similarly,

6 P. Xiong et al.



[Text] Place a brown sphere at the midpoint . Subsequently , position a purple cube to its rear left . To the front right of the purple cube and rear right of the brown sphere position a gray cylinder . A yellow sphere will then be situated to the left of the gray cylinder , the right of the purple cube , and the right of the brown sphere . Lastly , a blue cube goes to the left , positioned behind the yellow sphere , in front of the gray cylinder and the purple cube , and to the left of the brown sphere .

[Text] Add a yellow sphere at the center . Add a blue sphere in front of it on the right . Add a red sphere behind it on the left and behind yellow sphere on the left . Add a blue sphere in front of it on the left and behind yellow sphere on the left , and behind and on the left of the blue sphere which is in front of yellow sphere on the right . Add a blue sphere in front of it on the right and in front of red sphere on the right and in front of yellow sphere on the right , and in front of and on the left of the blue sphere which is in front of yellow sphere on the right .

[Text] Add a blue cube at the center . Add a brown cylinder in front of it on the left . Add a purple sphere behind it on the right and behind the blue cube on the right . Add a brown sphere between the brown cylinder and yellow sphere . Add a yellow sphere behind it on the right and in front of the purple sphere on the left . Add a yellow sphere behind the purple sphere on the left and behind the brown cylinder on the right and in front of the blue cube on the left .

[Text] Add two brown cylinders , a gray cylinder , two gray spheres .

[Text] Add two gray spheres , two red spheres , and a gray cube .

Ground Truth Ours

Fig. 6: Failure Cases of Our Proposed Model.

as text prompts become lengthier, the model tends to inaccurately generate entities. An example of this can be seen in row 2, where a *blue sphere*, the intended last entity, is mistakenly generated as a *blue cylinder*. This issue can be partially solved by enlarging the dataset scale or involving more input variation.

Moreover, our task encounters common challenges prevalent in the visuallanguage domain. In row 3, the model struggles with interpreting the concept of "between", indicating difficulty in correlating linguistic tokens with their visual spatial representations. Additionally, rows 4 and 5 exemplify the persistent challenge of aligning numerical descriptions with visual representations, a task that remains complex due to the inherent differences in two domain features. These challenges arise largely from ambiguities present in both modalities. These examples highlight the difficulties faced in integrating language understanding with visual generation, marking areas for future improvement and research.

References

- Bird, S., Klein, E., Loper, E.: Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit. O'Reilly Media (2009), https://books. google.com/books?id=KGlbfiiP1i4C 4
- 2. Blender Foundation: Blender a free and open source 3d creation suite. https: //www.blender.org/ (2023), accessed: 2024-02-29 3, 4
- Liu, N., Li, S., Du, Y., Torralba, A., Tenenbaum, J.B.: Compositional visual generation with composable diffusion models. In: European Conference on Computer Vision. pp. 423–439. Springer (2022) 5

4. OpenAI: Gpt-4: Openai's generative pre-trained transformer 4. https://openai. com/ (2023), accessed: 2024-02-29 3, 4