Visual Fact Checker: Enabling High-Fidelity Detailed Caption Generation

Yunhao Ge Xiaohui Zeng Jacob Samuel Huffman Tsung-Yi Lin Ming-Yu Liu Yin Cui

NVIDIA

https://research.nvidia.com/labs/dir/vfc/

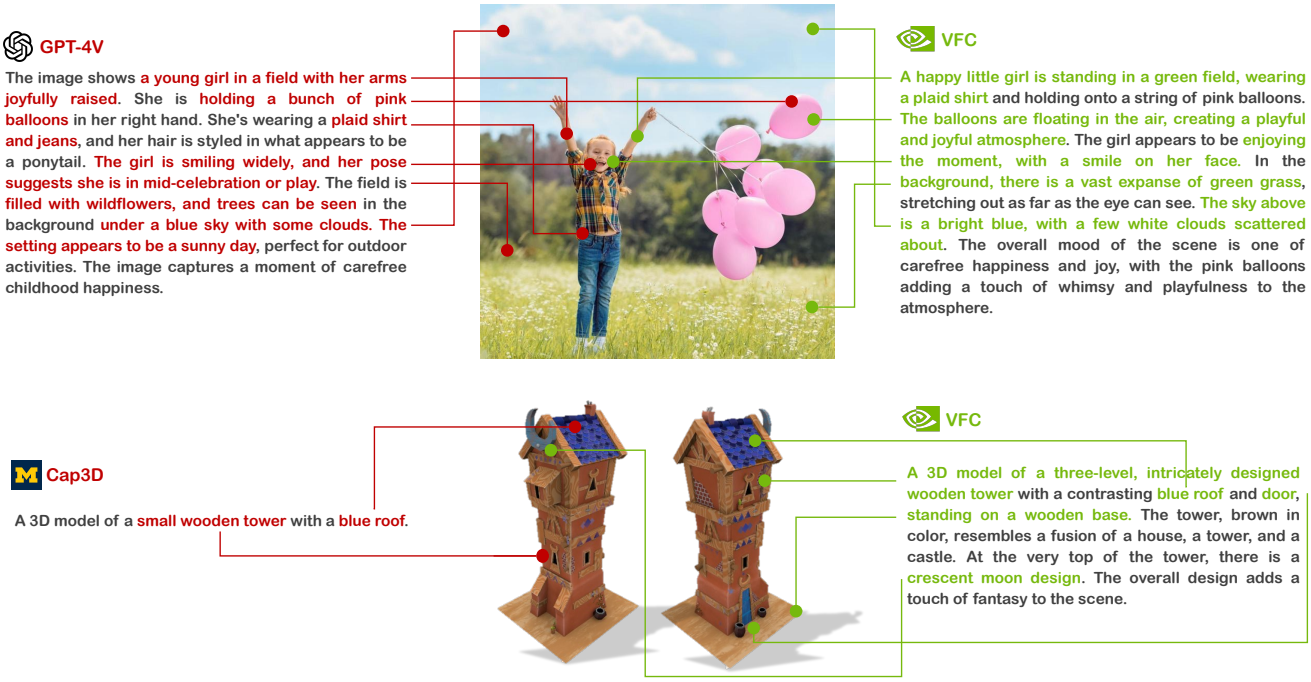

Figure 1. Comparison of VisualFactChecker (VFC) with GPT-4V and Cap3D. VFC can generate high-fidelity detailed captions that closely match GPT-4V's quality for 2D images and offer significantly more details for 3D objects than Cap3D. VFC used a pre-trained Llama-2 as the LLM when generating the caption for the above 2D image.

Abstract

Existing automatic captioning methods for visual content face challenges such as lack of detail, content hallucination, and poor instruction following. In this work, we propose VisualFactChecker (VFC), a flexible training-free pipeline that generates high-fidelity and detailed captions for both 2D images and 3D objects. VFC consists of three steps: 1) proposal, where image-to-text captioning models propose multiple initial captions; 2) verification, where a large language model (LLM) utilizes tools such as object detection and VQA models to fact-check proposed captions; 3) captioning, where an LLM generates the final caption by summarizing caption proposals and the fact check verification results. In this step, VFC can flexibly generate captions in various styles following complex instructions. We conduct comprehensive captioning evaluations using four metrics: 1) CLIP-Score for image-text *similarity; 2) CLIP-Image-Score for measuring the* image-image *similarity between the original and the reconstructed image generated by a text-to-image model using the caption. 3) human study on Amazon Mechanical Turk; 4) GPT-4V for* fine-grained *evaluation. Evaluation results show that VFC outperforms state-of-the-art open-sourced captioning methods for 2D images on the COCO dataset and 3D assets on the Objaverse dataset. Our study demonstrates that by combining open-source models into a pipeline, we can attain captioning capability comparable to proprietary models such as GPT-4V, despite being over 10× smaller in model size.*

1. Introduction

Image captioning is a pivotal challenge in computer vision and natural language processing. Its central goal is to encapsulate visual data within a textual description, which requires a nuanced understanding of both modalities. The recent advent of multimodal large language models (MM-LLMs), such as GPT-4V [26], and text-to-image generation models, such as DALLE-3 [3], has marked significant progress in this field. These proprietary models could leverage expansive human-labeled data and enormous computing resources to learn to generate detailed and contextually appropriate image descriptions. On the other hand, existing open-sourced captioning methods in the community still face significant challenges. Methods such as BLIP-2 [17] and OFA [35] often yield overly succinct captions that neglect essential visual information. Conversely, systems like Mini-GPT4 [39], InstructBLIP [8], and LLaVA [20, 21] can suffer from hallucination, producing long descriptions that do not align with the actual content of the images.

In light of this, we propose VisualFactChecker (VFC), a flexible training-free pipeline designed to produce accurate and comprehensive captions for both 2D images and 3D objects. Fig. 1 shows examples of captions generated by VFC and their comparisons with captions generated by GPT-4V [26] and Cap3D [23]. Captions generated by VFC are faithful textural representations of the visual contents. This can also be verified by reconstructing images and 3d objects from captions using state-of-the-art text-to-image and text-to-3d models, as shown in Fig. 2.

VFC focuses on tackling hallucinations and insufficient details in generated captions and is structured around three core components: **Proposer**, serving as the system's "eye", creating detailed caption proposals as preliminary captions by using image-to-text captioning models; **Large Language Model**, acting as the "brain", calling and summarizing information from other components, and leveraging its advanced generalization capabilities to steer the captioning process following specified captioning instructions; **Detector and VQA models**, functioning as "tools" utilized by the LLM for fact-checking caption proposals, ensuring the fidelity of the final generated caption. VFC is versatile and effectively handles captioning for both 2D images and 3D objects through a unified pipeline. Fig. 3 shows an overview of the pipeline. The details of each component and their interplay are explained in Sec. 3.

To comprehensively evaluate the generated captions, other than leveraging the commonly used CLIP-Score that primarily gauges the image-caption similarity, we propose a new metric: the CLIP-Image-Score. This metric assesses the similarity between the input image and a reconstructed image created by a text-to-image model from the caption, offering a complementary measure. Furthermore, we conducted a human study on Amazon Mechanical Turk for caption evaluation. Finally, we also performed a fine-grained evaluation by asking GPT-4V to compare and judge captions with detailed reasoning. The combination of CLIP-Score, CLIP-Image-Score, GPT-4V, and human study provides a more robust evaluation of captions.

We summarize our main contributions as follows: 1) We propose VisualFactChecker (VFC), a training-free pipeline to generate high-fidelity detailed 2D and 3D captions, effectively mitigating the challenge of hallucination in long captions. (2) CLIP-Image-Score: A novel caption evaluation metric that measures the similarity between the input image and a reconstructed image from the caption. (3) Our evaluation shows that VisualFactChecker achieves state-of-the-art results in 2D and 3D captioning tasks compared with open-sourced models. (4) Our work shows that using an LLM to chain open-source models can achieve captioning capability on par with proprietary models such as GPT-4V.

2. Related Work

2.1. Image Captioning

Image captioning has made significant progress with the advent of deep learning. Pioneering works [2, 10, 14] primarily focus on integrating deep neural networks for enhanced image understanding and language generation.

Recent strides have been made with the introduction of Multimodal-Large Language Models (MM-LLMs), which are trained on extensive vision and language data. The general approach involves leveraging a pre-trained large language model (LLM) and a vision encoder with a projector to align with the LLM's embeddings, thus enhancing visual understanding. Several models have emerged as significant contributors in this domain. BLIP [16], BLIP-2 [17], OFA [35], Flamingo [1], Kosmos-2 [27], MiniGPT-4 [39], InstructBLIP [8], LLaVA [20, 21] have demonstrated impressive performance in single-view image captioning tasks. However, they exhibit varying limitations. For instance, BLIP-2 and OFA often generate overly concise captions, while others, like InstructBLIP, can produce detailed captions that often include inaccurate or hallucinatory content. Our method aims to address these limitations by combining different models into a pipeline via an LLM, striking a better balance between accuracy and detailedness in generated captions while mitigating hallucinations.

2.2. Large Language Models for Captioning

Recent advancements in large language models (LLMs) like GPT-3 [5], LAMDA [30], PALM [7], Llama [32], GPT-4 [26] have demonstrated exceptional zero-shot capabilities in language analysis and summarization tasks. This proficiency has naturally extended to the multimodal domain, particularly in image-language contexts, where LLMs can summarize multimodal information in a zero-shot manner.

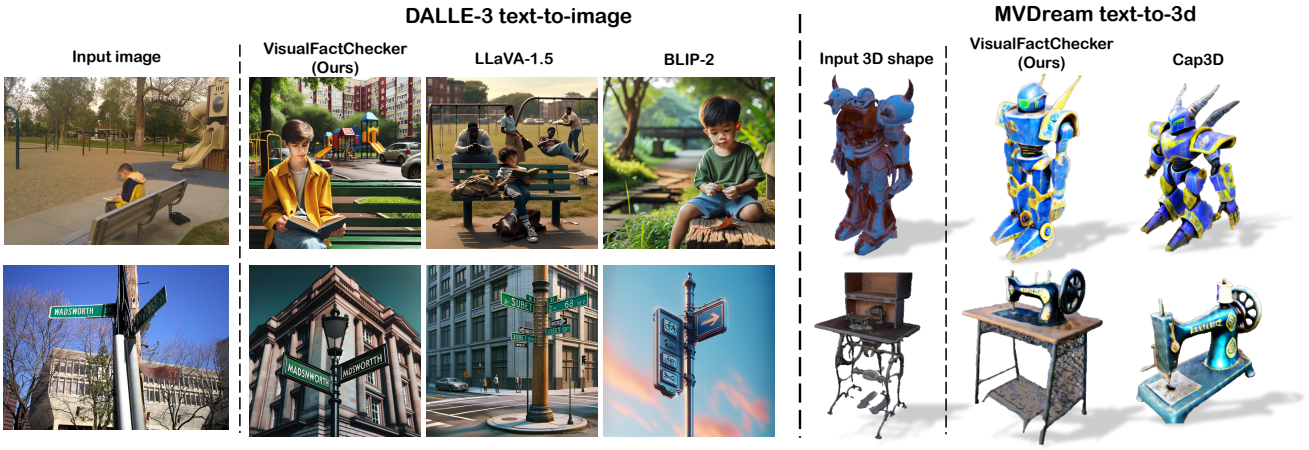

Figure 2. We use DALLE-3 [3] as a text-to-image model to reconstruct 2D images using generated captions from different captioning methods (BLIP-2, LLaVA-1.5 and ours). Similarly, we use MVDream [29] as a text-to-3D model to reconstruct 3D objects using different 3D captions (generated by Cap3D [23] and ours). From the results, we can see that the reconstructed images or 3D objects using BLIP-2 or Cap3D captions are less similar than the input ones, suggesting their captions may not contain sufficient information or incorrectly describe the visual contents; the reconstructed images using LLaVA-1.5 captions contain objects or scenes that are not present in the original images (top: people in the background, bottom: pedestrians and cars on the street), suggesting there might be hallucinations in LLaVA-1.5 captions. Images or 3D objects reconstructed using our captions are more similar to the inputs.

Vision-blind LLMs are prominent in multimodal applications, often utilizing language-only prefixes generated by pre-trained tools. Clipcap [25] demonstrates this by using a continuous embedding as a prompt for a GPT-style language model, achieving notable performance in single-viewpoint image captioning. Similarly, Promptcap [13] and PNP-VQA [31] leverage natural language prompts with GPT models to excel in visual question answering.

Recent methods have employed LLMs to generate image captions by summarizing initial captions or keywords from Vision-Language models. For instance, Socratic models [37] use a CLIP-based model to extract key tags from images, followed by GPT-3 with specialized prompts to create stylized captions. ChatCaptioner [38] builds upon this by integrating ChatGPT and BLIP-2 [17] in a conversational approach for question-answering about the image, and summarizing them into a caption. Visual Clues [36] uses similar tags to generate a paragraph-caption. IC3 [6] and LLM-Fusion [4] use LLMs to summarize captions from existing models augmented with temperature-based sampling. Cap3D [24] extends this concept to 3D object.

Our method differentiates itself in two critical ways: First, we focus on reducing hallucinations in captions by employing visual grounding tools, such as object detection, to fact-check captions for enhanced accuracy. Second, our pipeline can be used for captioning both 2D images and 3D objects. Unlike previous methods that rely on a single captioning model, we integrate multiple captioning sources from different models, ensuring a more comprehensive coverage of visual content to generate captions.

2.3. Hallucination in MM-LLM

There are two popular topics on the hallucination of MM-LLMs. (1) Hallucination evaluation: Detection approaches such as Gunjal *et al*. [11] train classification models to identify hallucination. They focus on distinguishing between accurate and hallucinated content. Ground truth comparison methods [18, 34] compare model outputs with ground truth data to detect hallucinations. These techniques assess the alignment of generated captions with actual image content. (2) Mitigation Strategies [22]: Data optimization methods such as Liu *et al*. [19] address hallucination by creating negative instances in training datasets to reduce model overconfidence. Iterative generation methods such as Wang *et al*. [33] adopt an iterative process for caption generation, where brief answers are generated in succession and amalgamated, aiming to improve accuracy and relevance.

Our VisualFactChecker is a training-free pipeline mitigating hallucination in image captioning. Our method utilizes visual grounding tools for improved accuracy, thereby actively reducing the hallucination and offering high-fidelity captions for both 2D images and 3D objects.

3. Visual Fact Checker

This section introduces the key components of VisualFactChecker as shown in Fig. 3 in detail and explains their interplay in generating accurate and detailed captions. The following sections delve into specifics. First, we detail the pipeline for 2D image captioning (Sec. 3.1), with Fig. 3 (top) illustrating this process. Then, we explore how our

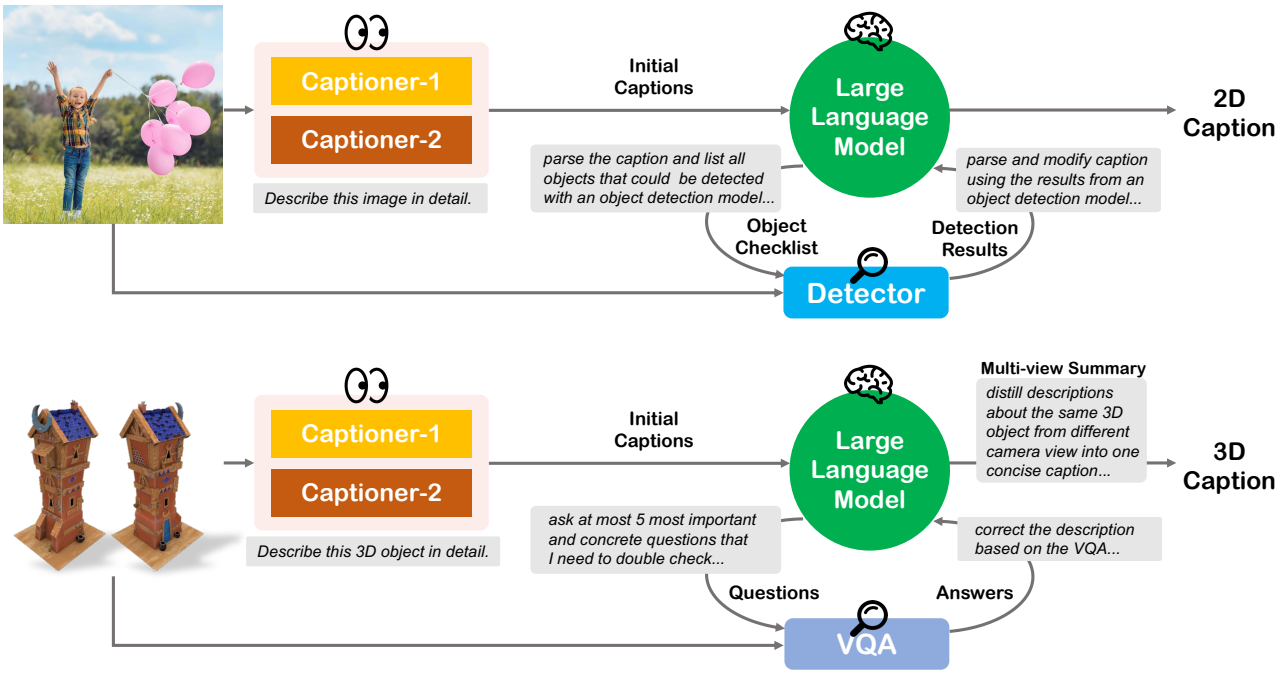

Figure 3. Pipeline of the VisualFactChecker for captioning 2D images (top) and 3D objects (bottom). The process begins with the input being captioned by two multimodal captioning models (Captioner-1 and Captioner-2) to generate preliminary captions. These captions are then verified using a Large Language Model (LLM) to call object detection (Detector) and VQA models for fact-checking the captions. Finally, the LLM incorporates all the results and summarizes the final caption by following instructions.

approach is adapted for 3D object captioning as shown in Fig. 3 (bottom), underscoring both shared methodologies and unique aspects relevant to 3D contexts (Sec. 3.2).

3.1. 2D Image Captioning

The caption generation takes three steps: 1) proposal, 2) verification, and 3) captioning. Each step is detailed below.

Proposal: The Proposal step serves as the cornerstone of the captioning process that generates initial captions. This is achieved through the utilization of advanced image-to-text models, specifically "LLaVA" and "Kosmos2". These models are trained on expansive datasets, enabling them to comprehend and interpret visual content effectively. By analyzing the input image, they suggest various preliminary captions, each reflecting different facets and interpretations of the image (Fig. 3 top). The rationale behind using multiple image-to-text multimodal LLMs lies in the complexity of adequately capturing an image's essence in a single attempt. Since an image can be accurately described in numerous ways, different models bring unique perspectives, thereby encompassing a broader range of information present in the image. Although the initial captions proposed may not possess perfect fidelity, the primary objective at this stage is to generate captions that are as comprehensive as possible. Fig. 3 displays the specific prompts we used for each step, with more details in Appendix A.

Verification and Captioning: The goal of the verification step is to scrutinize and rectify any inaccuracies or hallucinations in the captions during the proposal step. It employs a combination of a Large Language Model (LLM) and grounding tools, including an open-vocabulary object detection model and/or a visual question answering (VQA) model. Here the LLM can be GPT-4 or Llama2. As shown in Fig. 3 (top), the process involves the following steps: Step 1: LLM first summarizes the initial detailed descriptions from different MM-LLMs into a single, detailed caption. While this caption is comprehensive, it may not always be accurate. Step 2: The LLM then analyzes this synthesized caption, identifying all objects that could be verified by object detection and summarizing an object checklist. In 2D image captioning, the focus is on eliminating hallucinations, particularly descriptions of non-existent objects in the image. Identifying these objects is crucial for the subsequent verification process. Step 3: Taking the object checklist as input, an open-vocabulary object detection model examines candidate objects in the checklist and determines their presence in the image. This step is pivotal in validating the existence of objects mentioned in the caption, thus supporting the fidelity of the caption.

After verification, we go to the last captioning step: Based on the object detection results, the LLM revises the summarized single detailed caption. Each object described

in the caption is cross-checked; if detected, it remains unchanged, while undetected objects are considered potential hallucinations and are removed from the caption. This step results in a final caption that is both detailed and reliable. The underlying assumption is that the detection model, serving as an object grounding expert, provides more reliable results than a general image descriptor.

In the verification and captioning steps, the LLM plays a pivotal role as a "brain". It starts by parsing the initial caption and identifying key objects for detailed examination. The LLM then meticulously assesses whether each object mentioned actually appears in the image based on detection results. Following this thorough analysis, it refines and revises the initial captions, transforming them into final versions that are both coherent and richly detailed. The LLM is instrumental in guaranteeing linguistic fluency, ensuring that the captions not only accurately represent the image but also maintain the necessary level of detail for high-fidelity captioning. Moreover, the LLM can follow complex instructions to write the captions in a specified style, such as a caption that only mentions the foreground objects without mentioning the background. Fig. 3 displays the specific prompts used for each step.

3.2. 3D Object Captioning

The 3D object captioning process follows a similar structural pipeline to that of 2D images, with a few key distinctions in certain steps, as depicted in Fig. 3 (bottom). In 3D captioning, an object may present multiple views, each offering unique information. The comprehensive caption for a 3D object is derived by integrating the perspectives from all these views. For each view, VisualFactChecker is employed to create a detailed, high-fidelity description. Subsequently, the LLM (GPT-4 or Llama-2) is used to amalgamate the information from all views, producing a unified caption for the 3D object. In particular, for each view's captioning, we have the same three-step approach akin to 2D image captioning. In the proposal step, LLaVA-1.5 and InstructBLIP are utilized for generating initial detailed descriptions. We opt out of using Kosmos2 for single 3D objects due to its less effective performance in providing detailed descriptions, possibly linked to its reliance on an implicit detection model. Additionally, a slightly modified prompt is used (see Fig. 3 bottom), which incorporates 3D-specific considerations. In the verification and captioning step, we primarily address hallucinations related to the attributes of 3D objects, such as shape and color. To mitigate these inaccuracies, rather than enumerating potential objects, we employ the LLM to generate five critical questions that could influence a text-to-3D generation model in reconstructing the 3D model. Following this, we utilize VQA models (specifically LLaVA-1.5) to respond to these questions based on the input 3D object view image. Subsequently, the LLM amends

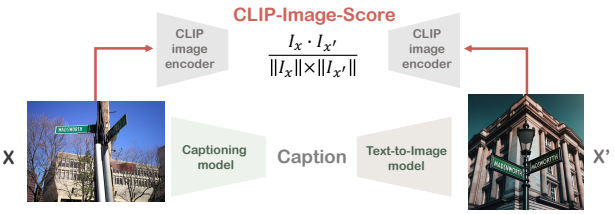

Figure 4. The CLIP-Image-Score pipeline evaluates caption accuracy by encoding an original image X into a feature representation I_X using a CLIP image encoder. A captioning model generates a caption that is then input into a text-to-image model to reconstruct an image X', which is encoded to $I_{X'}$. The score is computed by assessing the cosine similarity between I_X and $I_{X'}$, providing a measure of the caption's fidelity and hallucination detection.

the initial caption in accordance with the answers provided by the VQA model. We operate under the assumption that answering targeted questions results in fewer hallucinations compared to generating a general description. Once the caption for each individual view is complete, the LLM synthesizes these multiple perspectives into a singular, comprehensive caption for the entire 3D object. The prompts used for the LLM at each stage are detailed in Appendix A.

4. CLIP-Image-Score

Accurate evaluation of caption correctness and detailedness is paramount in determining the performance of an image captioning model. Traditional metrics like the CLIP-Score [12] have served as a standard for measuring the alignment between generated captions and their corresponding images. However, our CLIP-score may lack the sensitivity needed to detect the specific issue of hallucination within captions.

We present the CLIP-Image-Score, an alternative metric specifically developed to reflect the subtleties of caption quality. This metric is different from CLIP-Score by introducing an additional reconstruction step. Specifically, the CLIP-Image-score evaluates the similarity between the original image and a reconstructed version of the image generated by a fixed text-to-image model using the caption as a prompt. By comparing the raw image to its reconstructed image, the metric is able to detect discrepancies indicative of hallucination, thus providing a different perspective of the caption quality assessment. The underlying principle of the CLIP-Image-Score is the recognition that multiple "correct" captions may exist for a single image. However, it's only when a caption is both "detail" and "correct" that the reconstructed image closely resembles the original. Moreover, any hallucinations present in the caption become evident in the reconstructed image. Fig. 2 presents examples of such reconstructions. For instance, consider the results from LLaVA-1.5 shown in the third column. The cap-

tion generated for the first image falsely mentions "several other people in the background". This error is clearly reflected in the image reconstructed by the text-to-image generator. In essence, comparing the two images indirectly ensures alignment between the image and its caption, thereby providing a complementary method to assess the quality of the caption than directly comparing the image and caption.

The CLIP-Image-Score evaluation process is depicted in the following steps:

- **Caption Generation**: An original image X is input into a captioning model, which generates a caption.
- **Caption-to-Image Reconstruction**: This generated caption is then used as input for a text-to-image model, which creates a reconstructed image X' that visually represents the textual description.
- **Raw Image Encoding**: The original image X is processed through a CLIP image encoder, translating the visual content into an encoded representation I_X.
- **Reconstructed Image Encoding**: The reconstructed image is also processed through the CLIP image encoder to obtain its encoded representation $I_{X'}$.
- **Score Calculation**: Finally, the encoded representations of the original and reconstructed images are compared to calculate the CLIP-Image-Score. The score is given by the cosine similarity, which assesses the congruence between I_X and $I_{X'}$:

$$\text{CLIP-Image-Score} = \frac{I_X \cdot I_{X'}}{\|I_X\| \times \|I_{X'}\|} \tag{1}$$

Most notably, CLIP-Image-Score offers a sensitive measure for detecting hallucinations. In scenarios where the generated caption includes elements that are not in the original image, the reconstructed image will also likely contain these discrepancies. By comparing the original and reconstructed images, the CLIP-Image-Score can effectively highlight these differences, offering a clearer insight into the fidelity and accuracy of the generated caption.

Furthermore, CLIP-Image-Score turns a cross-modality comparison into a more intuitive comparison in the same image modality (as shown in Fig. 4). CLIP-Image-Score represents a new complementary perspective for image captioning evaluation. By leveraging the capabilities of text-to-image models and focusing on the congruence between the original and reconstructed images, it provides an accurate assessment of caption quality, particularly in identifying and measuring hallucinations, thereby enhancing the overall reliability of caption generation systems.

5. Experiments

This section presents a thorough evaluation of captioning models across both 2D and 3D visual content, employing a variety of datasets and methodologies. Table 1 provides a summary of our comprehensive evaluation experiments.

Eval	Input pairs for evaluation		Method	Reference
2D	Raw image	Caption	CLIP-Score	Table 2
			Human evaluation	Fig. 6
			CPT4V evaluation	Fig. 7
	Raw image	Image(recon)	CLIP-Image-Score	Table 2
3D	Multi-view (raw)	Caption	CLIP-Score	Table 3
			GPT4V evaluation	Fig. 7
	Multi-view (raw)	Multi-view (recon)	CLIP-Image-Score	Table 3

Table 1. Summary of evaluation methods and results.

5.1. Overall: CLIP-Score and CLIP-Image-Score

2D image captioning. Dataset: Our evaluation utilized 5,000 COCO test images from the Karpathy split. **Baseline methods**: We benchmarked against state-of-the-art captioning models, including BLIP-2 [17], InstructBLIP [8], and LLaVA-1.5 [20]. The evaluation focused on each model's ability to produce accurate, detailed, and coherent captions that effectively encapsulate the essence of the images. **Evaluation Metric**: We employed two metrics: CLIP-Score [12] and CLIP-Image-Score (Sec. 4). The CLIP-Score, a prevalent metric in image caption quality assessment, involves processing the raw image through the CLIP image encoder and the caption through the CLIP text encoder. The resultant embeddings are then compared for cosine similarity, with a higher score indicating greater semantic resemblance between the image and the caption. For our analysis, we first calculated the CLIP-Score for each image-caption pair, then averaged these scores across all 50,000 text/image pairs, scaling the result by a factor of 100. Table 2 displays the comparative performance of various image captioning methods on the 5,000 COCO test set images. The results demonstrate that our VisualFactChecker surpasses all baseline methods in performance.

Captioning Method	CLIP-Score (%) ↑	CLIP-Image-Score (%) ↑
Human Label (COCO GT)	30.36 (-2.54)	71.21 (-2.40)
BLIP2	30.11 (-2.79)	70.79 (-2.82)
InstructBLIP	31.45 (-1.45)	72.95 (-0.66)
LLaVA-1.5	32.08 (-0.82)	73.24 (-0.37)
Kosmos-2	32.32 (-0.58)	73.28 (-0.33)
VisualFactChecker (Ours)	**32.90**	**73.61**

Table 2. Image captioning comparison with different metrics on 5000 COCO test set in Karpathy split, we use raw image and caption as input pairs for evaluation.

As outlined in Sec. 4, the CLIP-Image-Score provides a complementary view to assess the quality of image captions. This metric is derived by comparing the cosine similarity between the CLIP embeddings of two images: the original image and a reconstructed image, which is generated using the provided caption through a text-to-image generation model. A higher CLIP-Image-Score signifies a more accurate and effective image caption. For this process, Stable Diffusion XL (SDXL) [28] is utilized as the designated text-to-image model to reconstruct images based on

Captioning Method	CLIP-Score (%) ↑	CLIP-Image-Score (%) ↑
Cap3D	33.44 (-0.57)	79.88 (-0.44)
VisualFactChecker (Ours)	**34.01**	**80.32**

Table 3. 3D object captioning comparison with different metrics on 1000 objects in Objaverse. For CLIP-Score, we use the average score of two views for evaluation. For CLIP-Image-Score, we use an off-the-shelf text-to-3D model, MVDream, to generate 3D models from 3D captions. We compare two views of the raw object and the same views of generated 3D object for evaluation.

the generated captions. Table 2 presents the CLIP-Image-Scores obtained for the 5000 images in the COCO test set, where our method outperforms all baseline methods.

3D object captioning. Dataset: 1,000 3D objects sampled from Objaverse dataset [9]. **Baseline methods**: We use state-of-the-art 3D object captioning model Cap3D [23] as the baseline. Cap3D uses 8 view images to generate the final object caption, our VisualFactChecker uses only 2 views to generate the object caption. **Evaluation Metric**: CLIP-Score and CLIP-Image-Score on multiple views rendered from 3D objects. To evaluate the similarity of a 3D object and the generated caption, we evaluate the similarity of the caption with the multi-view images used to generate the caption. Specifically, we evaluate the similarity of the generated caption with the two views that were used to generate the caption and use the average score to represent the CLIP-Score. Table. 3 shows the performance of 3D object captioning methods on 1,000 3D objects from Objaverse dataset. VisualFactChecker outperforms Cap3D.

We also use CLIP-Image-Score to evaluate the 3D caption quality. CLIP-Image-Score needs reconstructed images to compare with the raw images. We treat the two views that were used to generate the 3D object caption as the raw image. To obtain the reconstructed image, we use an off-the-shelf text-to-3D generation model, MVDream, to generate a 3D object given the generated 3D object caption. We then render the same two views of images based on the generated 3D object, and we calculate the CLIP-Image-Score between the raw image and the rendered image. Table. 3 shows the CLIP-Image-Score on 1000 objects in Objaverse dataset.

5.2. Per Image Evaluation: Wining Rate

CLIP-Score and CLIP-Image-Score indicate an overall performance comparison, which shows an average score among all 5000 images. The average score may be dominated by a small group of images that have extremely high or low scores. To zoom in and show a more detailed comparison, we try to answer the following question: Given an image, what is the probability that one method performs better than another method on caption generation? To answer this question, we need to go over each image and calculate the winning rate for a pair of methods.

Specifically, for each image, we compare the CLIP-Score of our VisualFactChecker caption against the captions generated from different baselines respectively, and calculate the wining probability of our method and the baselines. Fig. 5 shows the results, for example, we can see that in the pair-wise comparison, our VisualFactChecker performs better (higher CLIP-Score) than LLaVA-1.5 on 64.4% of 5000 images (3220 images).

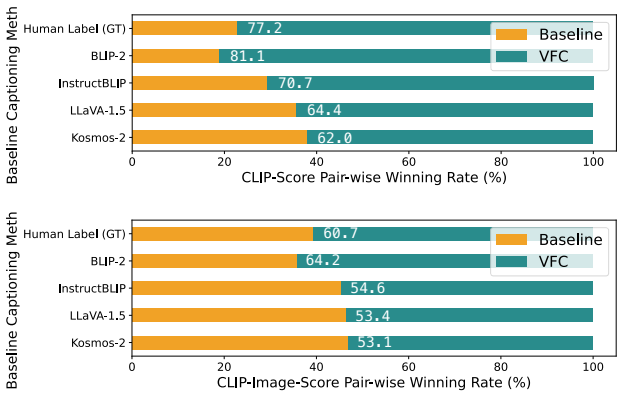

Figure 5. 2D image captioning comparison with pair-wise winning rate. VisualFactChecker (VFC) outperforms all baseline methods on both CLIP-Score (top) and CLIP-Image-Score (bottom).

Calculating the winning rate over all images provides a more detailed analysis that zooms in on the comparison of each image, which shows a complementary view than overall average CLIP-Score.

5.3. Fine-grained Evaluation: Human and GPT-4V

The CLIP-Score and CLIP-Image-Score offer a general comparison of overall performance. A pairwise per-image winning rate provides a more specific analysis, evaluating performance on individual images. However, the research highlighted in related studies [15] indicates that the CLIP-Score may not be ideally suited for image-to-image comparison tasks. Furthermore, relying on a single score fails to provide a nuanced comparison across criteria, such as accuracy and level of detail. We use Human evaluation and GPT-4V to provide a more fine-grained evaluation.

Human evaluation using Amazon Mechanical Turk (AMT). We employed a pairwise comparison strategy. From the COCO dataset, we randomly selected 100 images out of 5000. For each image, our caption was compared against 5 baseline captions respectively. To reduce variance, each comparison was done by 3 different AMT workers and we used their majority voting as the final selection. This resulted in a total of 1500 comparisons collected on AMT. AMT UI is shown in the appendix. The workers were presented with two competing captions — one from a baseline method and one from our VisualFactChecker, in randomized order. They were instructed to select the better caption describing the image based on 3 aspects: correctness, de-

tailness, and fluency. Results in Fig. 6 show our captions are more preferred by humans. The human evaluation instruction and web UI is shown in Appendix B.

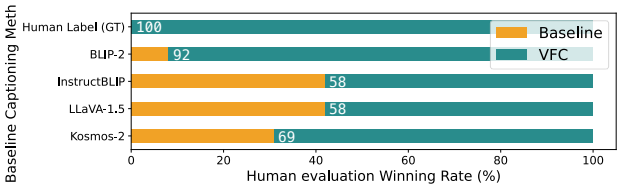

Figure 6. Amazon Mechanical Turk human evaluation results.

GPT-4V evaluation. Our study applied GPT-4V for evaluating captions in a manner akin to the caption evaluation process used in DALLE-3. We use the same randomly selected 100 images from COCO as in Human evaluation. For each image, we considered the captions generated by 5 baseline methods alongside the caption produced by our VisualFactChecker. We then presented GPT-4V with the raw image, our reference caption, and the four baseline captions. Our designed prompt instructed GPT-4V to compare each baseline caption against our reference caption, focusing on two primary aspects: correctness and detail. GPT-4V was tasked with providing a pairwise, detailed comparison for each pair, including justifications for its assessments. Based on these comparative insights, GPT-4V classified each baseline method caption as either "better" or "worse" than our VisualFactChecker. Fig. 5 shows the comprehensive results. More details about the GPT-4V evaluation prompt and examples are shown in Appendix B.

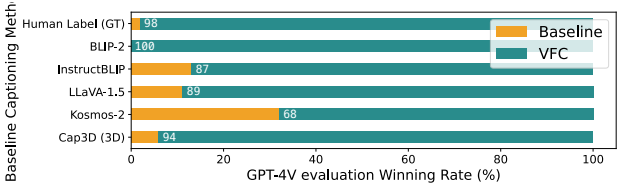

Figure 7. GPT-4V evaluation results. Our captions are significantly better than baselines.

5.4. Ablation Study

In our ablation study, we explore the impact of various components on performance. For 2D captioning tasks, we assess the efficacy of initial captioning models, LLaVA-1.5 and Kosmos-2, using the CLIP-Score metric for the captions they generate on the same 5000 COCO test images. Additionally, we ablate our method's performance in the absence of the verification (fact checker) step, which aims to mitigate hallucinations through detection grounding. Table 4 shows the detailed results. Likewise, in the context of 3D object captioning, we evaluate the individual contributions of initial captioners, namely LLaVA-1.5 and InstructBLIP on the same 1000 Objaverse 3D objects. We further investigate the performance of our methodology without the fact checker, which in this case operates by leveraging a VQA model to reduce hallucinations. Table 4 shows the detailed results. These results highlight the significance of fact checker in our approach.

	Methods or Steps	CLIP-Score
2D	LLaVA-1.5	32.08 (-0.33)
	Kosmos-2	32.32 (-0.09)
	VisualFactChecker (w/o fact check)	32.41
	VisualFactChecker	**32.90 (+0.49)**
3D	LLaVA-1.5	32.05 (-0.66)
	InstructBLIP	32.51 (-0.20)
	VisualFactChecker (w/o fact check)	32.71
	VisualFactChecker	**34.01 (+1.30)**

Table 4. Ablation study on captioning 2D images (5000 COCO test dataset) and 3D objects (1000 Objaverse).

5.5. Qualitative Results and Prompt Following

Other than quantitative evaluation results, we show more qualitative examples of VisualFactChecker for 2D and 3D captions in Appendix C.

By leveraging an LLM, VisualFactChecker can follow complex instructions to write captions in various styles. Examples are shown in Appendix D.

6. Conclusion

We propose the VisualFactChecker (VFC), a training-free pipeline to generate high-fidelity and detailed captions. By utilizing an LLM to chain multimodal models and object detection and VQA models, VFC reduces hallucination in long captions. We conducted a comprehensive caption evaluation using different metrics, including 1) image-text similarity using CLIP-Score, 2) image-reconstructed image similarity using our proposed CLIP-Image-Score, 3) human study, and 4) fine-grained evaluation using GPT-4V. Compared with open-sourced captioning models, our method achieves state-of-the-art in both 2D and 3D captioning. Our work shows combining open-sourced models into a pipeline can significantly close the captioning performance gap with proprietary models like GPT-4V. In the future, we plan to improve our pipeline further by including more components for fact-checking and making it more automatic in deciding which components to use.

Acknowledgments We would like to thank Siddharth Gururani for helping with our human evaluation using Amazon Mechanical Turk; Haochen Wang for his help in preprocessing 3D data. We also thank Qinsheng Zhang, Yogesh Balaji, and Yen-Chen Lin for their helpful discussion.

References

[1] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022. 2

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, 2018. 2

[3] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, and Yunxin Jiao. Improving image generation with better captions. Technical report, OpenAI, 2023. 2, 3, 11

[4] Simone Bianco, Luigi Celona, Marco Donzella, and Paolo Napoletano. Improving image captioning descriptiveness by ranking and llm-based fusion. *arXiv preprint arXiv:2306.11593*, 2023. 3

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020. 2

[6] David M Chan, Austin Myers, Sudheendra Vijayanarasimhan, David A Ross, and John Canny. Ic3: Image captioning by committee consensus. *arXiv preprint arXiv:2302.01328*, 2023. 3

[7] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022. 2

[8] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*, 2023. 2, 6

[9] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *CVPR*, 2023. 7

[10] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015. 2

[11] Anisha Gunjal, Jihan Yin, and Erhan Bas. Detecting and preventing hallucinations in large vision language models. *arXiv preprint arXiv:2308.06394*, 2023. 3

[12] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021. 5, 6

[13] Yushi Hu, Hang Hua, Zhengyuan Yang, Weijia Shi, Noah A Smith, and Jiebo Luo. Promptcap: Prompt-guided task-aware image captioning. *arXiv preprint arXiv:2211.09699*, 2022. 3

[14] Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. Attention on attention for image captioning. In *ICCV*, 2019. 2

[15] Klemen Kotar, Stephen Tian, Hong-Xing Yu, Daniel LK Yamins, and Jiajun Wu. Are these the same apple? comparing images based on object intrinsics. *arXiv preprint arXiv:2311.00750*, 2023. 7

[16] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 2

[17] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. *arXiv preprint arXiv:2301.12597*, 2023. 2, 3, 6

[18] Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. *arXiv preprint arXiv:2305.10355*, 2023. 3

[19] Fuxiao Liu, Kevin Lin, Linjie Li, Jianfeng Wang, Yaser Yacoob, and Lijuan Wang. Aligning large multi-modal model with robust instruction tuning. *arXiv preprint arXiv:2306.14565*, 2023. 3

[20] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023. 2, 6

[21] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023. 2

[22] Jiaying Lu, Jinmeng Rao, Kezhen Chen, Xiaoyuan Guo, Yawen Zhang, Baochen Sun, Carl Yang, and Jie Yang. Evaluation and mitigation of agnosia in multimodal large language models. *arXiv preprint arXiv:2309.04041*, 2023. 3

[23] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. In *NeurIPS*, 2023. 2, 3, 7, 17

[24] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. *arXiv preprint arXiv:2306.07279*, 2023. 3

[25] Ron Mokady, Amir Hertz, and Amit H Bermano. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*, 2021. 3

[26] OpenAI. Gpt-4 technical report, 2023. 2

[27] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023. 2

[28] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 6

[29] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 3, 17

[30] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia

Jin, Taylor Bos, Leslie Baker, Yu Du, et al. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*, 2022. 2

[31] Anthony Meng Huat Tiong, Junnan Li, Boyang Li, Silvio Savarese, and Steven CH Hoi. Plug-and-play vqa: Zero-shot vqa by conjoining large pretrained models with zero training. *arXiv preprint arXiv:2210.08773*, 2022. 3

[32] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2

[33] Bin Wang, Fan Wu, Xiao Han, Jiahui Peng, Huaping Zhong, Pan Zhang, Xiaoyi Dong, Weijia Li, Wei Li, Jiaqi Wang, et al. Vigc: Visual instruction generation and correction. *arXiv preprint arXiv:2308.12714*, 2023. 3

[34] Junyang Wang, Yiyang Zhou, Guohai Xu, Pengcheng Shi, Chenlin Zhao, Haiyang Xu, Qinghao Ye, Ming Yan, Ji Zhang, Jihua Zhu, et al. Evaluation and analysis of hallucination in large vision-language models. *arXiv preprint arXiv:2308.15126*, 2023. 3

[35] Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *ICML*, 2022. 2

[36] Yujia Xie, Luowei Zhou, Xiyang Dai, Lu Yuan, Nguyen Bach, Ce Liu, and Michael Zeng. Visual clues: Bridging vision and language foundations for image paragraph captioning. In *NeurIPS*, 2022. 3

[37] Andy Zeng, Maria Attarian, Brian Ichter, Krzysztof Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*, 2022. 3

[38] Deyao Zhu, Jun Chen, Kilichbek Haydarov, Xiaoqian Shen, Wenxuan Zhang, and Mohamed Elhoseiny. Chatgpt asks, blip-2 answers: Automatic questioning towards enriched visual descriptions. *arXiv preprint arXiv:2303.06594*, 2023. 3

[39] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 2