Explore and Tell: Embodied Visual Captioning in 3D Environments

Anwen Hu[1], Shizhe Chen[2], Liang Zhang[1], Qin Jin[1]*
[1]School of Information, Renmin University of China
[2]INRIA
{anwenhu,zhangliang00,qjin}@ruc.edu.cn
shizhe.chen@inria.fr

Abstract

While current visual captioning models have achieved impressive performance, they often assume that the image is well-captured and provides a complete view of the scene. In real-world scenarios, however, a single image may not offer a good viewpoint, hindering fine-grained scene understanding. To overcome this limitation, we propose a novel task called Embodied Captioning, which equips visual captioning models with navigation capabilities, enabling them to actively explore the scene and reduce visual ambiguity from suboptimal viewpoints. Specifically, starting at a random viewpoint, an agent must navigate the environment to gather information from different viewpoints and generate a comprehensive paragraph describing all objects in the scene. To support this task, we build the ET-Cap dataset with Kubric simulator, consisting of 10K 3D scenes with cluttered objects and three annotated paragraphs per scene. We propose a Cascade Embodied Captioning model (CaBOT), which comprises of a navigator and a captioner, to tackle this task. The navigator predicts which actions to take in the environment, while the captioner generates a paragraph description based on the whole navigation trajectory. Extensive experiments demonstrate that our model outperforms other carefully designed baselines. Our dataset, codes and models are available at https://aim3-ruc.github.io/ExploreAndTell.

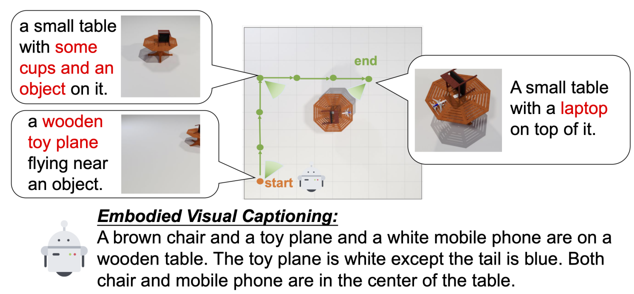

Figure 1. Existing visual captioning models generate a sentence to describe a single image, which often fail when the image is not well-captured at good viewpoints (the captions next to the images are generated by the state-of-the-art BLIP model [23]). To generate more accurate and comprehensive visual descriptions, we propose a new Embodied Captioning task which allows agents to navigate the environment to reduce visual ambiguity of the scene.

1. Introduction

Visual captioning [14, 22, 32, 18, 9] is an essential vision-and-language task which aims to generate natural language descriptions of visual contents. In recent years, many captioning models have been developed and achieved significant improvements in describing major objects and relationships in an image [41, 4, 28, 25, 46, 23, 39]. However, the current models typically rely on well-captured images that provide a good viewpoint of the scene. Unfortunately, in real-world scenarios, capturing such images may not always be feasible. As illustrated in Fig. 1, the initial position to capture an image may not provide a complete view of the scene, and a single image may not be sufficient to capture all objects within it, potentially leading to incomplete or inaccurate visual captions. To address this limitation, it is essential for visual caption models to navigate the environment actively and gather information from multiple viewpoints in order to generate more comprehensive and accurate visual captions.

In this paper, we propose a novel task called Embodied Captioning, which integrates such navigation ability into visual captioning. The task requires an agent, which starts at a random viewpoint in a 3D environment, to navigate the environment to reduce visual ambiguity of the scene, and finally generate a comprehensive paragraph that describes all objects in the scene. Different from existing navigation tasks [5, 15, 44, 30, 24], our Embodied Captioning task does not explicitly define a target location to navigate. Instead, we have an implicit target which is to accurately recognize all objects along with their attributes and relationships in

*Corresponding Author.

the scene as soon as possible.

To support the Embodied Captioning task, we build a high-quality 3D dataset ET-Cap with manually annotated paragraph descriptions. Leveraging high-quality 3D object assets from ShapeNet [11] and GSO [16], we use Kubric [17] to construct 10,000 scenes. For each scene, three annotators are provided with 20 images from different viewpoints of the scene and asked to write a detailed paragraph to describe all visible instances. We also require the annotators to select images with good viewpoints among all the image candidates. Although our dataset is based on synthetic scenes with limited obstacles, it still presents significant challenges for Embodied Captioning. The agent only receives an RGB image of a restricted field of view at each step without any localization information, such as the location of the agent and objects. Therefore, a model must be equipped with long-term memories of previous visual observations and actions in order to efficiently explore the environment, accurately recognize objects, and generate fine-grained scene descriptions.

To address these challenges, we propose a Cascade Embodied Captioning model (CaBOT), which consists of a History-aware Navigator and a Trajectory-aware Captioner. The navigator leverages histories of both observed images and performed actions to predict the next action via a transformer model. The captioner is fed with all images in the predicted trajectory and utilizes a bi-level cross attention over spatial and temporal dimensions of the image sequence to generate a paragraph. The proposed CaBOT model outperforms carefully designed baselines for the challenging Embodied Captioning task. Nevertheless, there is still much room to improve compared to human performance, such as joint modeling the navigation and captioning process.

In summary, our contributions are three-fold:

- We propose a novel and challenging Embodied Captioning task which requires agents to explore in 3D environments to generate better visual descriptions.
- A high-quality dataset is constructed to benchmark the Embodied Captioning task, with 10K synthetic 3D scenes and 24K manually annotated good viewpoints and 30K paragraph descriptions.
- We present a Cascade Embodied Captioning model which incorporates navigation histories for captioning, providing a strong starting point for future work.

2. Related Work

Visual Captioning. Visual captioning aims to describe visual contents with natural language. A variety of visual captioning tasks have been proposed such as image captioning [14, 22, 32, 18, 9], video captioning [40, 40, 48, 38, 37] and 3D captioning [12]. In image captioning, a model should describe major objects, attributes, relations [14, 43, 22, 2, 13] or even named entities [9] and scene texts [32, 21] in the image. Most of the image captioning datasets [27, 22, 32, 9] use high-quality web images. Such dataset bias results in poor generalization when an image is not well taken as shown in VizWiz-Caption dataset [18] which contains photos taken by the blind. Video captioning tasks [40, 48, 38, 37] take a video as input, and focus more on the actions or events. 3D captioning task [12] is designed to describe a 3D indoor scene with almost complete point clouds as input. Much progress has been made to improve the captioning performance such as attention mechanism [41, 4, 20], transformer architecture[28, 45, 33] and pretraining framework [47, 23, 25, 42, 39]. In this work, we aim to push the standard visual image captioning tasks one step further, which do not passively receive an visual input, but should actively explore in the environment to obtain better visual observations to describe.

Embodied Vision and Language Tasks. Growing research attentions have been paid to embodied vision and language tasks [5, 15, 44, 30, 34, 24] in recent years, which require agents to perform actions to achieve various goals specified in natural language. Vision-and-Language Navigation [5] requires an agent to follow natural language instructions to navigate to a target place. Embodied visual referring expression [30, 24] provides object-oriented high-level instructions and requires both navigation and object grounding. In embodied question answering [15, 44], a question is provided to the agent such as 'What room is the microwave located in?', and the agent should firstly find relevant visual information and then generate the answer. Compared with the above tasks, our proposed Embodied Captioning does not provide explicit goals for navigation. In contrast, there is only an implicit goal that the agent should generate more informative captions after its navigation.

The most similar work to ours is embodied scene description (ESD) [8, 7, 34]. Our work is distinguished from ESD in two main aspects. Firstly, the main goal of navigation in ESD is to enlarge the explored area or to find a single viewpoint that captures more objects, while our navigation goal is more entangled with visual captioning to enable to generate accurate and comprehensive captions as soon as possible. Secondly, existing ESD work simply uses pretrained image captioning models and does not have an in-domain dataset to benchmark the task with systematic evaluation. In this work, we construct a high-quality dataset with human-annotated viewpoints and paragraph descriptions and we also provide multiple strong baselines.

3. ET-Cap Dataset

We build a high-quality synthetic dataset ET-Cap with human annotations to teach agents how to **E**xplore and **T**ell in 3D environments.

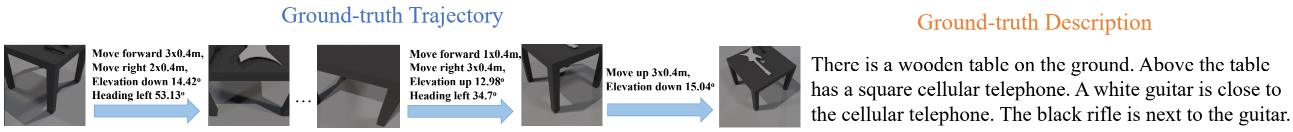

Figure 2. An example of ground-truth trajectory and corresponding description in ET-Cap.

3.1. 3D Scenes Simulation

3D scenes in ET-Cap are constructed with Kubric [17], a recently proposed framework that enables the generation of photo-realistic scenes. We use Kubric to place multiple objects in a scene and render images. 3D object assets come from ShapeNet [11] and Google Scanned Objects [16].

Each scene in ET-Cap is constructed following three steps: *Instance Resizing*, *Instance Selection*, and *Instance Placing*. Specifically, in order to make the scene more realistic, we first resize 3D object assets to their common sizes in real life. Since the indoor environment is often organized by arranging small objects around big furniture, we then randomly choose a big-size furniture instance (called a base instance) and multiple relatively smaller instances (called placing instances). Finally, to ensure the diversity of instance arrangement, we place instances by performing a physic simulation. The base instance is first placed at the center of the scene. The other instances are then dropped above the base instance, leading to diverse poses of instances across scenes because of collision. More details can be found in the supplementary material.

Navigation Space. We discretize the whole environment as a 3D grid world with the total size of $8 \times 8 \times 4\ m^3$ (length×width×height) and each grid size of $0.4 \times 0.4 \times 0.4\ m^3$. There are on average 4,838 navigable grids in a scene removing the grids occupied by instances. Agents in the 3D grid world can perform five types of actions relative to their current pose: `forward-backward move` (stop, forward, backward), `left-right move` (stop, left, right), `up-down move` (stop, up, down), `heading rotate` (stop, left, right) and `elevation rotate` (stop, up, down). The maximum distance traveled in each direction is 1.6m. Each action type involves a combination of a direction classification and a magnitude regression task, *e.g.*, move forward for $0.8m$, rotate horizontally to the left for 30 degrees.

3.2. Manual Annotation

After the automatic scene construction, we recruit expert annotators from Appen Platform[1] to select good viewpoints and write detailed descriptions for each scene.

Viewpoint Selection. Explicitly annotating trajectories for Embodied Captioning is time-consuming, subjective, and less flexible due to dependency on initial positions. Therefore, we ask annotators to select good viewpoints for each scene at which the scene can be well-captured. This allows us to automatically generate trajectories from random initial positions to those good viewpoints (Section 3.3). To be specific, as good viewpoints are always a certain distance away from instances in both horizontal and vertical directions, we first sample 20 candidate viewpoints in a highly probable spatial range of the grid world and then set the camera to look at the center of the scene to render images. To help annotators understand the scene, we provide category labels of instances in the scene. The annotators can select *multiple* good viewpoints as long as the image provides a clear view of (almost) all objects in the scene.

Paragraph Annotation. We ask annotators to write descriptions about objects in the scene, mentioning categories, attributes, and spatial relationships of instances. To ensure that the description is comprehensive, annotators are required to write a paragraph in which the number of sentences roughly matches the number of instances.

For each scene, there are three independent expert annotators recruited to select viewpoints and write paragraphs. Besides, an extra expert inspector is arranged to check the annotation quality. More details about the annotation platform, annotation cost, and annotation samples can be found in the supplementary material.

3.3. Trajectory Generation

We generate ground-truth trajectories based on the annotated good viewpoints and ground-truth 3D environment information in each scene. Specifically, we take a good viewpoint as the target position and randomly select three start positions. In contrast to previous navigation works [5, 15, 34] which use a fixed move step size of 0.25m, we allow our agent to move more flexibly and cover longer distances, with a maximum length of 1.6m, to reduce trajectory length. We employ the Dijkstra algorithm to generate the shortest movement path in the 3D space from the start to the target position. Then, we generate an optimal rotation action of the camera after each move step of the agent. To ensure that most instances are visible, we directly set the camera to look at the center of the scene, as we can use the perfect agent's location and object locations for ground-truth trajectory generation. According to the camera view, position changes are converted to three types of move ac-

[1] https://www.appen.com/

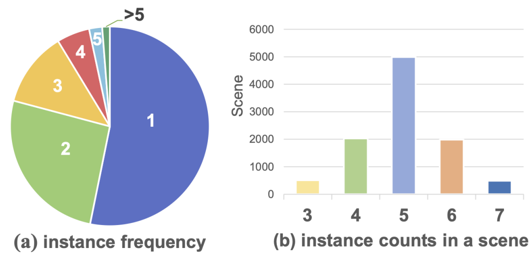

Figure 3. Instance frequency distribution (a) and distribution of the number of instances in a scene (b).

Table 1. Statistics of captions in our ET-Cap compared to other visual captioning datasets.

Statistics	COCO [14]	ImPG [22]	Nr3D [1]	ScanR [12]	ET-Cap
# of words	11.3	68.5	12.4	20.2	56.3
# of sentences	1.8	6.7	1.7	3.0	5.9
adjectives (%)	6.54	9.84	8.73	9.38	17.01
nouns (%)	32.30	25.13	26.65	24.12	24.62
spatial words (%)	9.72	8.51	11.87	10.15	10.84

tions. We merge the move action and the camera rotation action at each step, resulting in five ground-truth actions at each step: 3 movements with 3 step lengths, a camera heading rotation with direction and angle, and a camera elevation rotation with direction and angle. Fig. 2 shows an example of a ground-truth trajectory and its paragraph description.

3.4. Dataset Statistics

Scene Statistics. Our ET-Cap dataset consists of 10,000 scenes and 27,520 unique instances, which include 7,517 base instances and 20,003 placing instances across 10 and 47 object categories respectively. The detailed category distribution is provided in the supplementary material. Notably, about 53% of the instances appear only once in the dataset, as shown in Fig. 3(a), which can prevent models from overfitting the instances. As shown in Fig. 3(b), the number of instances in a scene varies from 3 to 7.

Trajectory Statistics. We generate 72,594 trajectories with an average of 6.13 viewpoints and a mean length of 6.3m. The ground-truth trajectories exhibit no significant bias in the action direction. The 'move down' direction is relatively less frequent than the 'move up' direction because most good viewpoints are in higher positions than the height of the objects, which is consistent with our intuition.

Caption Statistics. We collect 30,000 captions in total. A caption contains an average of 5.9 sentences and 50.2 words. The average number of adjectives, nouns and spatial relation words (e.g. 'left', 'right', 'center') in a caption are 9.6, 13.8, and 6.1, respectively. As shown in Tab. 1, our captions are more descriptive than those in existing visual captioning datasets, with a much higher ratio of adjectives. They also contain slightly more spatial words compared to traditional image captions [14, 22] and competitive to 3D object referring expression datasets [1, 12].

Table 2. Statistics of ET-Cap training, validation and test sets.

Split	Subset	Scene	Caption	Trajectory
Training	-	8,324	24,972	60,474
Validation	Common	300	900	2,178
	Novel Instance	300	900	2,176
	Novel Category	238	714	1,638
	Total	838	2,514	6,012
Test	Common	300	900	2,178
	Novel Instance	300	900	2,223
	Novel Category	238	714	1,707
	Total	838	2,514	6,108

Dataset Splits. We split our ET-Cap into training, validation, and test sets based on scenes. Tab. 2 shows the statistics of each set. To test the generalization ability of agents, we further divide the validation and test sets into three subsets: a) *Common subset*, in which instances in the scene have all been seen in the training set; b) *Novel Instance subset*, in which the categories are all covered in the training set, but more than 50% of the instances in the scene do not appear in the training set; c) *Novel Category subset*, in which each scene contains at least 1 novel object category that has not been seen in the training set.

4. Cascade Embodied Captioning Model

In the Embodied Captioning task, the agent is initialized at a bad viewpoint and can only receive an RGB image at each step. It should navigate in the 3D environment to comprehend the scene and generate a paragraph describing all objects at the end of the navigation. We propose a **Ca**scade em**BO**died cap**T**ioning (CaBOT) model to tackle the task. As shown in Fig. 4, CaBOT is composed of a History-aware Navigator and a Trajectory-aware Captioner. The navigator utilizes an episodic history to explore the environment, while the captioner generates paragraphs using the whole trajectory produced by the navigator.

4.1. History-aware Navigator

Given the observed RGB image at each step, the navigator must generate 5 types of actions: forward-backward movement, left-right movement, up-down movement, elevation rotation, and heading rotation. For each action type, it needs to predict both the direction and magnitude. Once the predictions are made, the agent will move to a new position and adjust its camera orientation accordingly. It stops until all the values of the types of actions are 'stop'.

Our navigator consists of a backbone to extract image features where we use a ResNet [19], a region encoder to enhance spatial modeling within each image, a historical vision encoder to enhance temporal modeling across images, and a historical camera-informed decoder to predict actions.

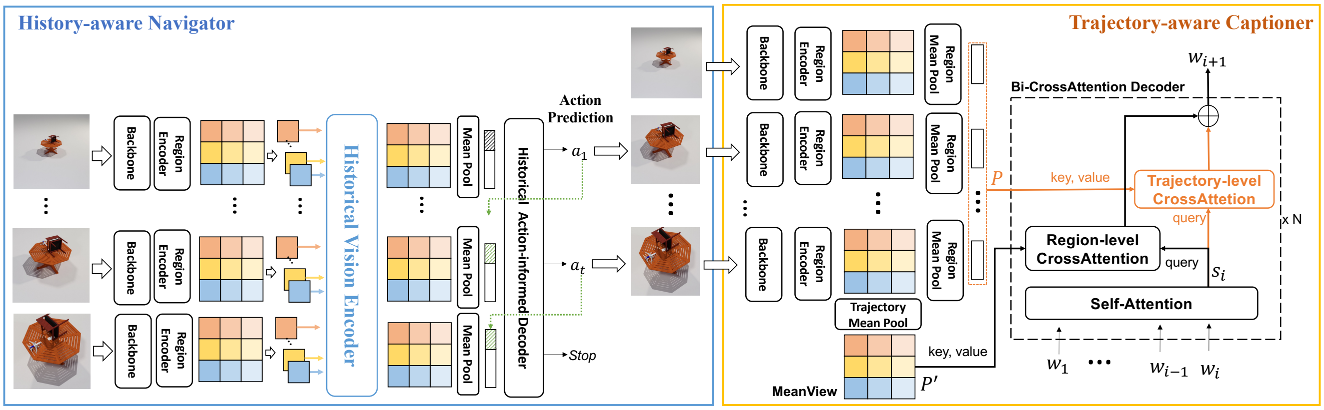

Figure 4. The overall architecture of **Ca**scade em**BO**died cap**T**ioning (CaBOT) model. It consists of a History-aware Navigator and a Trajectory-aware Captioner. The navigator navigates from a start viewpoint in a 3D environment. The captioner generates a paragraph to describe the scene according to the trajectory of navigator.

Region Encoder. During navigation, to know which direction is possible to capture more visual information of instances, it is necessary to compare the region difference of the current image. Therefore, we apply a Transformer layer [35] with self-attention over the backbone for region-level encoding. For the image v_t rendered from the t^{th} viewpoint, its region-level features are denoted as $H_t = [h_t^1, h_t^2, ..., h_t^R]$, where R is the region sequence length.

Historical Vision Encoder. Besides spatial relation within the current image, historical vision information also helps to decide the next actions. For example, observing the change of the central region from blank to one containing partial instances helps to confirm continued movement in the same direction. Thus, we further apply a Historical Vision Encoder to learn the temporal relation of the same region across images at different steps as follows:

$$\tilde{h_t^r} = \text{Trans}_<([h_1^r, .., h_{t-1}^r, h_t^r]), 1 \leq r \leq R, \quad (1)$$

where $\text{Trans}_<(\cdot)$ is a Transformer layer with causal-masked self-attention mechanism, which means only previous image information can be attended.

Historical Action-informed Decoder. In addition to historical vision information, historical action information (direction and magnitude) is also helpful for deciding the next actions because it records the viewpoint change during previous navigation. Therefore, we design a Historical Action-informed Decoder leveraging both vision and action information to predict the next actions as follows:

$$c_t = W^c a_{t-1} + b^c, \quad (2)$$
$$o_t = [\tfrac{1}{R}\textstyle\sum_{r=1}^{R} \tilde{h_t^r}; c_t], \quad (3)$$
$$\bar{o_t} = \text{Trans}_<([o_1, .., o_{t-1}, o_t]), \quad (4)$$

where a_{t-1} is a 10D vector consisting of the five independent action classes and magnitudes in the previous step. W^c and b^c are trainable parameters, $[;]$ means concatenation.

With $\bar{o}_t$ as the input, we apply linear layers to predict the next action a_t including the direction and magnitude for each of the five types of actions. We use a softmax function for the direction classification and a sigmoid function for magnitude regression at the end of the linear layer.

4.2. Trajectory-aware Captioner

The Trajectory-aware Captioner takes images $V = [v_1, v_1, ..., v_T]$ observed in the navigation trajectory as input, and outputs a paragraph to describe the scene. Due to the importance of accurate object recognition for visual captioning [4, 25], we train a DETR [10] model to detect objects (more details can be found in our supplementary material) and use its backbone ResNet [19] to initialize the backbone in our captioner. Similar to the navigator, the captioner also applies a Region Encoder which converts the image sequence into $\mathcal{H} = [H_1, ..., H_T]$, where $H_t = [h_t^1, h_t^2, ..., h_t^R], 1 \leq t \leq T$, T is the trajectory length, and R is the region sequence length.

Bi-CrossAttention Decoder In order to gather visual information from different steps in the navigation trajectory, we propose a Bi-CrossAttention Decoder to leverage all images at both the region level and trajectory level for captioning.

We first perform a region-level mean pooling and a trajectory-level mean pooling on $\mathcal{H}$ as follows:

$$p_t = \tfrac{1}{R}\textstyle\sum_{r=1}^{R} h_t^r, 1 \leq t \leq T, \quad (5)$$
$$p'_r = \tfrac{1}{T}\textstyle\sum_{t=1}^{T} h_t^r, 1 \leq r \leq R, \quad (6)$$

$P = [p_1, .., p_T]$ and $P' = [p'_1, .., p'_R]$ are pooled trajectory-level and region-level visual features, respectively.

At the i^{th} decoding step of paragraph generation, the decoder first performs causal-masked self-attention on previous words to keep text coherence. Then it applies two cross-attention layers to select relevant visual information at the region level and trajectory level as follows:

$$s_i = \text{Trans}_<([w_1, w_2, ..., w_i]), \tag{7}$$

$$\alpha = \text{Softmax}([s_i^\intercal p'_1, ..., s_i^\intercal p'_R]), \tag{8}$$

$$\beta = \text{Softmax}([s_i^\intercal p_1, ..., s_i^\intercal p_T]), \tag{9}$$

$$y_i = \textstyle\sum_{r=1}^{R} \alpha_r p'_r + \sum_{t=1}^{T} \beta_t p_t, \tag{10}$$

where s_i is the output of casual-masked self-attention for word w_i. Projection matrices in attention layers are omitted for equation simplicity. y_i is finally added with s_i and used to predict the next word w_{i+1} with a classification layer.

4.3. Training and Inference

We train the History-aware Navigator and the Trajectory-aware Captioner separately. Both modules leverage images in ground-truth trajectories as the input for imitation / supervised learning. The loss function of the navigator is the summation of each type of action. Assume $\mathcal{L}^{cls}$ is the cross-entropy loss for action direction classification and $\mathcal{L}^{reg}$ is the L1 loss for action magnitude regression, the navigation loss is: $\mathcal{L} = \mathcal{L}^{cls} + 2.0\mathcal{L}^{reg}$. The captioner (including its backbone) is optimized by the cross-entropy loss of word prediction [4, 25, 41]. More implementation details can be found in the supplementary material.

At the inference phase, CaBOT performs Embodied Captioning in a cascade manner. Given a starting viewpoint, the navigator first generates a trajectory. The captioner then leverages images from the predicted trajectory to generate a paragraph description.

5. Experiments

5.1. Evaluation Metrics

Captioning evaluation. We employ common captioning metrics to evaluate the quality of generated paragraphs in our Embodied Captioning task, including BLEU4 [29], METEOR [6], ROUGE-L [26], CIDEr [36] and SPICE [3]. The CIDEr and SPICE scores are emphasized because they have been demonstrated to be aligned with human evaluations better for image captions. CIDEr puts more weight on rare words, while SPICE better measures the semantic similarity of objects, object-attribute pairs, and object-relation triplets. We compare each generated paragraph against three human-written descriptions as references. To promote efficient exploration of the scene, we factor in trajectory length when evaluating descriptions. Specifically, we calculate a weight $r^l = L_{gt}/\max(L_{gt}, L_{pred})$, where L_{gt} and L_{pred} are the lengths of the ground-truth and predicted trajectories, respectively. We then multiply conventional captioning metrics with r^l to obtain length-penalized metrics, denoted as $*^l$ for each captioning metric.

Navigation evaluation. We propose three types of evaluation metrics to separately measure the navigation quality. The first one **N**avigation **E**rror (NE) is commonly adopted in previous embodied navigation works [5, 15]. It measures the minimum Manhattan distance in meters between a predicted viewpoint and the manually selected good viewpoints in a scene. However, as the selected good viewpoints are only a subset of good viewpoints, NE is a pessimistic metric of the navigation quality. Therefore, we introduce the other two metrics that directly evaluate the quality of the image rendered at a predicted viewpoint: **I**mage **S**imilarity (IS) and **S**egmentation Similarity (SS). IS measures the visual similarity using a CLIP [31] vision encoder:

$$\text{IS}(v, \mathcal{V}) = \max_{v' \in \mathcal{V}} \frac{\cos(\text{CLIP}(v), \text{CLIP}(v')) - \gamma}{1 - \gamma}, \tag{11}$$

where $\mathcal{V}$ are images in good viewpoints, v refers to an image in a predicted viewpoint, $\text{CLIP}(\cdot)$ refers to the CLIP [31] vision encoder clip-ViT-B-32, γ is set to 0.7 to reduce the influence of background. The SS metric measures the similarity of semantic segmentation between v and $\mathcal{V}$:

$$\text{SS}(v, \mathcal{V}) = \max_{v' \in \mathcal{V}} \frac{1}{N_C} \sum_{c \in C} \min(\frac{A(v, c)}{A(v', c)}, 1.0), \tag{12}$$

where C refers to the set of categories covered in v', N_C refers to the number of categories in C, $A(\cdot, c)$ computes the area ratio of category c in the image.

As images in the whole trajectory might provide useful visual information about the scene, we evaluate the quality of the whole trajectory rather than only the stop viewpoint or best viewpoint. Specifically, for all the navigation metrics, we evaluate the score at each step in the trajectory and take the average as the final score. Similar to caption evaluation, we introduce trajectory length into navigation evaluation. We multiply NE by $1/r^l$ to obtain NE^l, and multiply IS and SS score by r^l to obtain IS^l and SS^l, respectively.

5.2. Captioning Ablation with Oracle Trajectory

In this section, we evaluate the effectiveness of the proposed captioner given ground-truth trajectories. The compared baselines include a template-based method and different variants of our proposed captioner. In the template-based method, we carefully design paragraph templates and insert automatically detected objects from images in the trajectory into the template (more details can be found in the supplementary material). In the variants of our model, we ablate the contributions of region-level cross-attention, trajectory-level cross-attention, and backbone initialization.

Tab. 3 presents the comparison results. Firstly, all models outperform the template-based method, indicating the rule-based method is not competent enough for describing the 3D scene in detail. Secondly, as the end view in an oracle trajectory is the image in good viewpoints annotated by humans, r2 achieves comparable performance with r3.

Table 3. Captioning performance with oracle trajectories.‘ ‘single’ means only utilizing a single view at the end of the trajectory for region-level cross-attention, and ‘multi’ means using the average of multi-view images in the whole trajectory. ‘Human’ denotes using one human-written caption as the candidate and the other two human-written captions as references.

	Region-level CrossAtt	Trajectory-Level CrossAtt	DETR Init Backbone	BLEU1	BLEU2	BLEU3	BLEU4	METEOR	ROUGE-L	CIDEr	SPICE
r1	Template-based			56.79	38.32	23.55	13.22	21.10	38.44	26.41	18.25
r2	single	×	×	62.88	47.83	35.54	25.46	20.76	44.99	29.34	14.22
r3	multi	×	×	62.35	46.76	34.15	24.74	22.33	43.34	28.35	16.25
r4	×	✓	×	66.36	50.49	36.89	26.24	23.81	44.59	39.49	18.28
r5	multi	✓	×	66.26	51.00	37.86	27.58	24.21	45.72	42.10	19.22
r6	multi	✓	✓	**66.79**	**51.44**	**38.02**	**27.62**	**24.50**	**46.00**	**45.24**	**20.45**
r7	Human			73.47	55.73	41.20	29.84	29.84	49.24	96.72	39.73

Table 4. Embodied Captioning performance using different navigators and our proposed captioner (r6 in Tab. 3).

	Action history	Historical Vision Encoder	Navigation Evaluation						Captioning Evaluation			
			NE↓	IS↑	SS↑	NE^l↓	IS^l↑	SS^l↑	CIDEr↑	SPICE↑	$CIDEr^l$↑	$SPICE^l$↑
r1	Rule-based		4.42	45.25	30.31	11.24	17.79	11.53	33.90	16.54	15.80	7.69
r2	×	×	3.81	69.85	69.62	5.23	56.02	55.99	40.59	19.35	33.35	15.96
r3	×	✓	3.81	69.76	**69.92**	5.00	57.70	57.75	**40.91**	**19.45**	34.44	16.45
r4	✓	✓	**3.78**	**70.15**	69.58	**4.89**	**58.54**	**58.13**	40.83	19.43	**34.88**	**16.62**

However, it’s still worse than r4, showing the importance of merging visual information at the trajectory level for scene description. Additionally, by combing both region-level and trajectory-level cross attention, the captioner achieves better performance (r5 vs r4), indicating region-level features and trajectory-level features are complementary for paragraph generation. Furthermore, by initializing the backbone with ResNet from the object detection model, the captioner performs best (r6), showing knowledge about object detection is also fundamental to describing 3D environments. Finally, there is a large gap (e.g., 51.48 on CIDEr) between model performance and human performance, indicating there is still a large room to improve on our dataset.

5.3. Embodied Captioning Results

Main results. In this section, we combine the best captioning model (r6) in Tab. 3 with automatically predicted trajectories by a navigator. We compare the proposed navigator with its variants and a carefully designed rule-based method. The rule-based method primarily determines the direction of movement by analyzing the distribution of object areas within the current field of view. Specifically, to decide movement directions, we adopt a cropping strategy that splits the current image into upper and lower parts of equal size. If the upper/lower part contains more instances, the agent will move up/down accordingly. We also divide the image into equal-sized left, middle, and right parts. The agent will move forward/left/right depending on where most instance area is located. Moreover, if the agent moves forward, we will examine whether this results in fewer instances being visible. If so, the agent will move backward. Once the agent reaches a new position, we make it turn around by four 90^o turns to the left in order to determine the best orientation. The rule-based approach stops when the maximum trajectory length is reached.

Tab. 4 presents both the navigation performance and the captioning performance. Firstly, rule-based navigation is shown to be inferior to variants of CaBOT navigator, which suggests that navigating for better scene descriptions generation in ET-Cap is not a naive task. Secondly, with historical vision information during decoding, the navigator achieves better performance (r3 vs r2), which indicates that visual difference across time at the same image region is helpful for navigation. Besides, introducing historical action encoding further improves navigation performance (r4 vs r3), suggesting that the previous action trajectory is also beneficial to predict the next action. Finally, better navigation leads to better scene description, verifying that navigation ability is crucial for effectively embodied captioning.

Generalization analysis. We evaluate the generalization ability of CaBOT on three subsets of the test set. As shown in Tab. 5, CaBOT achieves similar performance on Common Set and Novel Instance Set for all three test settings. It shows that novel instances cause little impact on navigation and captioning ability. We attribute this instance generalization ability to the diverse training data, where most instances appear only one time, as shown in Fig. 3 (a). As for Novel Category Set, there is a small performance drop for navigation but an obvious performance drop for captioning. The navigator can be confused by an instance of novel categories, but may still find good viewpoints by observing other instances. However, for description generation, the captioner struggles to correctly describe instances whose categories have never been seen during training.

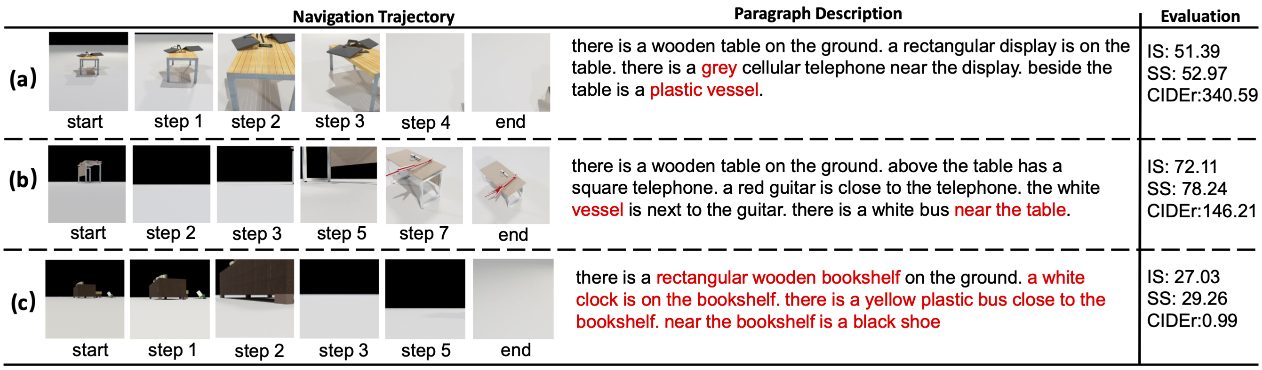

Figure 5. Qualitative results of CaBOT on Embodied Captioning test set. Wrong descriptions are marked in red.

Table 5. Breakdown analysis on different test subsets.

Subset	Navigation			OracleCap		EmboCap	
	$NE^l\downarrow$	$IS^l\uparrow$	$SS^l\uparrow$	$C^l\uparrow$	$S^l\uparrow$	$C^l\uparrow$	$S^l\uparrow$
Common	**4.73**	58.76	58.17	48.91	21.68	37.60	17.60
Novel Instance	4.88	**58.94**	**58.90**	**51.08**	**22.72**	**39.10**	**18.45**
Novel Category	5.11	57.73	57.08	34.02	15.91	26.63	12.98

Potential benefits from joint navigation and captioning. Although our proposed model separates navigation and caption generation, we believe that simultaneously performing navigation and captioning would lead to better performance in a more efficient manner. The captioner can inform the navigator of objects, attributes, or relations that it is uncertain of, allowing the navigator to explore the environment more efficiently in order to resolve the uncertainty and improve the quality of visual captioning. We conduct an exploratory experiment to demonstrate the potential benefits of joint navigation and captioning. We utilize the CaBOT captioner to generate scene descriptions at each navigation step predicted by the CaBOT navigation, and select the partial trajectory with the best CIDEr performance against the ground-truth captions. As shown in Tab. 6, this method achieves a significant improvement of +25.55 on the CIDEr and +28.02 on the CIDEr^l. It indicates that imperfect navigation could bring more noise and deteriorate visual captioning, and thus it is necessary to guide the navigation according to the captioning ability.

5.4. Qualitative Analysis

Fig. 5 presents some embodied captioning results of our CaBOT on the test set. Case (a) shows that CaBOT can gradually approach instances and try to wrap around to find better viewpoints. When starting from a relatively lower position, CaBOT can also explore to find a higher viewpoint, as shown in case (b). CaBOT stops at a good viewpoint in case (b) but a bad viewpoint in case (a). However, it generates acceptable descriptions in both cases, which indicates that the overall quality of the trajectory matters more than the end viewpoints. We also show a failure case in case (c).

Table 6. Exploratory experiments of joint navigation and captioning, where we simultaneously generate captions at each navigation step and stop navigation according to the captioning performance.

Method	CIDEr	SPICE	CIDEr^l	SPICE^l
Cascade	40.83	19.43	34.88	16.62
Joint	66.38	21.51	62.90	20.41

When the agent reaches a bad viewpoint where no instance can be seen, it is less robust to find good viewpoints though historical visual and action information are leveraged. We consider this is mainly due to unseen states in testing, as we only utilize ground-truth trajectories for imitation learning. This problem can be relieved by data augmentation or reinforcement learning, and we leave it to future work.

6. Conclusion

In this work, we present a novel and challenging task called Embodied Captioning, where an agent should navigate in a 3D environment to gather visual information and use natural language to comprehensively describe objects in the scene. To support this task, we build the ET-Cap dataset with manually annotated good viewpoints and paragraph descriptions for 10,000 synthetic scenes. We propose a Cascaded Embodied Captioning (CaBOT) model, which utilizes both visual and action history to perform navigation and then generates captions by leveraging the whole navigation trajectory. Experiments demonstrate the effectiveness of CaBOT and show a promising direction of joint modeling navigation and captioning.

There is no ethical or privacy issue with the ET-Cap dataset as it uses publicly released simulated assets. As a result, the dataset is also limited in the simple simulated scenes. The proposed Embodied Captioning task has the potential to benefit a wide range of applications, such as assisting visually impaired people.

Acknowledgements. This work was partially supported by the National Key R&D Program of China (No.2020AAA0108600) and the National Natural Science Foundation of China (No. 62072462).

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas J. Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *ECCV (1)*, volume 12346 of *Lecture Notes in Computer Science*, pages 422–440. Springer, 2020. 4

[2] Harsh Agrawal, Karan Desai, Yufei Wang, Xinlei Chen, Rishabh Jain, Mark Johnson, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. nocaps: novel object captioning at scale. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8948–8957, 2019. 2

[3] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*, pages 382–398. Springer, 2016. 6

[4] 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*, pages 6077–6086. Computer Vision Foundation / IEEE Computer Society, 2018. 1, 2, 5, 6

[5] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian D. Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*, pages 3674–3683. Computer Vision Foundation / IEEE Computer Society, 2018. 1, 2, 3, 6

[6] Satanjeev Banerjee and Alon Lavie. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In *IEEvaluation@ACL*, pages 65–72. Association for Computational Linguistics, 2005. 6

[7] Roberto Bigazzi, Marcella Cornia, Silvia Cascianelli, Lorenzo Baraldi, Rita Cucchiara, et al. Embodied agents for efficient exploration and smart scene description. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2023. 2

[8] Roberto Bigazzi, Federico Landi, Marcella Cornia, Silvia Cascianelli, Lorenzo Baraldi, and Rita Cucchiara. Explore and explain: self-supervised navigation and recounting. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 1152–1159. IEEE, 2021. 2

[9] Ali Furkan Biten, Lluís Gómez, Marçal Rusiñol, and Dimosthenis Karatzas. acute; good news, everyone! context driven entity-aware captioning for news images. In *CVPR*, pages 12466–12475. Computer Vision Foundation / IEEE, 2019. 1, 2

[10] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV (1)*, volume 12346 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2020. 5

[11] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *CoRR*, abs/1512.03012, 2015. 2, 3

[12] Dave Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X. Chang. Scan2cap: Context-aware dense captioning in RGB-D scans. In *CVPR*, pages 3193–3203. Computer Vision Foundation / IEEE, 2021. 2, 4

[13] Shizhe Chen, Qin Jin, Peng Wang, and Qi Wu. Say as you wish: Fine-grained control of image caption generation with abstract scene graphs. In *CVPR*, pages 9959–9968. Computer Vision Foundation / IEEE, 2020. 2

[14] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO captions: Data collection and evaluation server. *CoRR*, abs/1504.00325, 2015. 1, 2, 4

[15] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied question answering. In *CVPR*, pages 1–10. Computer Vision Foundation / IEEE Computer Society, 2018. 1, 2, 3, 6

[16] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *ICRA*, pages 2553–2560. IEEE, 2022. 2, 3

[17] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J. Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam H. Laradji, Hsueh-Ti Derek Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, A. Cengiz Öztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: A scalable dataset generator. In *CVPR*, pages 3739–3751. IEEE, 2022. 2, 3

[18] Danna Gurari, Yinan Zhao, Meng Zhang, and Nilavra Bhattacharya. Captioning images taken by people who are blind. In *ECCV (17)*, volume 12362 of *Lecture Notes in Computer Science*, pages 417–434. Springer, 2020. 1, 2

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778. IEEE Computer Society, 2016. 4, 5

[20] Anwen Hu, Shizhe Chen, and Qin Jin. ICECAP: information concentrated entity-aware image captioning. In *ACM Multimedia*, pages 4217–4225. ACM, 2020. 2

[21] Anwen Hu, Shizhe Chen, and Qin Jin. Question-controlled text-aware image captioning. In *ACM Multimedia*, pages 3097–3105. ACM, 2021. 2

[22] Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. In *CVPR*, pages 3337–3345. IEEE Computer Society, 2017. 1, 2, 4

[23] 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*, volume 162 of *Proceedings of Machine Learning Research*, pages 12888–12900. PMLR, 2022. 1, 2

[24] Xinghang Li, Di Guo, Huaping Liu, and Fuchun Sun. REVE-CE: remote embodied visual referring expression in contin-

uous environment. *IEEE Robotics Autom. Lett.*, 7(2):1494–1501, 2022. 1, 2

[25] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV (30)*, volume 12375 of *Lecture Notes in Computer Science*, pages 121–137. Springer, 2020. 1, 2, 5, 6

[26] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 6

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[28] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10971–10980, 2020. 1, 2

[29] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 6

[30] Yuankai Qi, Qi Wu, Peter Anderson, Xin Wang, William Yang Wang, Chunhua Shen, and Anton van den Hengel. REVERIE: remote embodied visual referring expression in real indoor environments. In *CVPR*, pages 9979–9988. Computer Vision Foundation / IEEE, 2020. 1, 2

[31] 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*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021. 6

[32] Oleksii Sidorov, Ronghang Hu, Marcus Rohrbach, and Amanpreet Singh. Textcaps: A dataset for image captioning with reading comprehension. In *ECCV (2)*, volume 12347 of *Lecture Notes in Computer Science*, pages 742–758. Springer, 2020. 1, 2

[33] Yuqing Song, Shizhe Chen, and Qin Jin. Towards diverse paragraph captioning for untrimmed videos. In *CVPR*, pages 11245–11254. Computer Vision Foundation / IEEE, 2021. 2

[34] Sinan Tan, Huaping Liu, Di Guo, Xinyu Zhang, and Fuchun Sun. Towards embodied scene description. In *Robotics: Science and Systems*, 2020. 2, 3

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 5

[36] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015. 6

[37] Weiying Wang, Yongcheng Wang, Shizhe Chen, and Qin Jin. Youmakeup: A large-scale domain-specific multimodal dataset for fine-grained semantic comprehension. In *EMNLP/IJCNLP (1)*, pages 5132–5142. Association for Computational Linguistics, 2019. 2

[38] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, pages 4580–4590. IEEE, 2019. 2

[39] Zirui Wang, Jiahui Yu, Adams Wei Yu, Zihang Dai, Yulia Tsvetkov, and Yuan Cao. Simvlm: Simple visual language model pretraining with weak supervision. In *ICLR*. OpenReview.net, 2022. 1, 2

[40] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, pages 5288–5296. IEEE Computer Society, 2016. 2

[41] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2048–2057. JMLR.org, 2015. 1, 2, 6

[42] Zhengyuan Yang, Yijuan Lu, Jianfeng Wang, Xi Yin, Dinei Florêncio, Lijuan Wang, Cha Zhang, Lei Zhang, and Jiebo Luo. TAP: text-aware pre-training for text-vqa and text-caption. In *CVPR*, pages 8751–8761. Computer Vision Foundation / IEEE, 2021. 2

[43] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. 2

[44] Licheng Yu, Xinlei Chen, Georgia Gkioxari, Mohit Bansal, Tamara L. Berg, and Dhruv Batra. Multi-target embodied question answering. In *CVPR*, pages 6309–6318. Computer Vision Foundation / IEEE, 2019. 1, 2

[45] Zhihao Yuan, Xu Yan, Yinghong Liao, Yao Guo, Guanbin Li, Shuguang Cui, and Zhen Li. X -trans2cap: Cross-modal knowledge transfer using transformer for 3d dense captioning. In *CVPR*, pages 8553–8563. IEEE, 2022. 2

[46] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5579–5588, 2021. 1

[47] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J. Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and VQA. In *AAAI*, pages 13041–13049. AAAI Press, 2020. 2

[48] Luowei Zhou, Chenliang Xu, and Jason J. Corso. Towards automatic learning of procedures from web instructional videos. In *AAAI*, pages 7590–7598. AAAI Press, 2018. 2