Towards Embodied Scene Description

Sinan Tan
Tsinghua University

Huaping Liu*
Tsinghua University

Di Guo
Tsinghua University

Fuchun Sun
Tsinghua University

***Abstract*—Embodiment is an important characteristic for all intelligent agents (creatures and robots), while existing scene description tasks mainly focus on analyzing images passively and the semantic understanding of the scenario is separated from the interaction between the agent and the environment. In this work, we propose the *Embodied Scene Description*, which exploits the embodiment ability of the agent to find an optimal viewpoint in its environment for scene description tasks. A learning framework with the paradigms of imitation learning and reinforcement learning is established to teach the intelligent agent to generate corresponding sensorimotor activities. The proposed framework is tested on both the AI2Thor dataset and a real world robotic platform demonstrating the effectiveness and extendability of the developed method.**

I. INTRODUCTION

When a visually impaired person enters a new room, he can easily take pictures of his surroundings using the smartphone and the built-in advanced computer vision modules are able to provide some scattered semantic information of these pictures. For example, the smartphone can detect certain classes of objects in the image with the help of an object detector and speak them out to the visually impaired person. However, such information is likely to make people confusing and uncomfortable due to its disorder and disorganization. A better way is to generate higher level semantic description such as natural language sentences or even paragraphs to describe the image. At present, great progress has been made in the areas of *Image Captioning*[1][2], *Dense Captioning*[3], and *Image Paragraphing* [4], and it has been becoming more and more mature with the booming of deep learning techniques[5]. See Fig.1 for some typical scene description tasks. It is believed that such semantic description will be an indispensable approach for the visually impaired people to perceive the environment[6]. In this case, a further question arouses – what is the next step?

In fact, no matter how accurate the semantic description is, it can only provide information that exists in the current image, but not tell the user what to do next. The semantic understanding of the scenario is separated from the interaction between the agent and the environment. When a visually impaired person enters a room, the first photo captured is likely to be a bare wall or window. At this time, he usually has to move the smartphone randomly expecting to capture a more meaningful image from another viewpoint. In this situation, it is more useful to tell him where to look next rather than just to provide him the vague description of the current scene (e.g. *there is window on the wall*). On the other hand, a notorious problem of the semantic description is that it is very

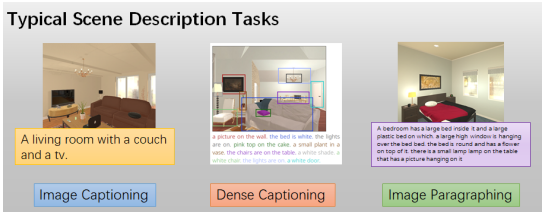

Fig. 1. Typical scene description tasks: *Image Captioning*, *Dense Captioning*, and *Image Paragraphing*.

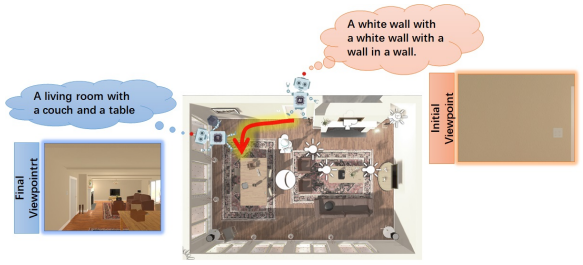

Fig. 2. An intuitive *Embodied Scene Description* demonstration. Here we take the *Image Captioning* as an example, while the idea is applicable to other tasks such as *Dense Captioning*, *Image Paragraphing*, etc. At first glance, the agent captures the initial image (rendered with *pink*). Since this image is non-informative, the generated caption provides very limited information about the scene. However, the agent may explore the environment by itself to find a better viewpoint to capture a new image (rendered with *blue*). The generated caption yields more informative and suitable results.

sensitive to the camera viewpoint[7]. Although the content of the captured image may seem good, a deviation in camera viewpoint will lead to a totally wrong semantic description result. Under this circumstance, it is important to tell the visually impaired person how to adjust the position of the camera and even his body (e.g. move left, right, forwards, backwards) to get more meaningful and accurate scene description for the current scenario. Unfortunately, existing scene description work[8][3][4] and the free APP software[6] do not take this point into account. The reason is that all of them ignore the embodiment which is a very important characteristic of all intelligent agents (creatures and robots). The embodiment concept asserts that the intelligence emerges by interactions between the agent and the environment. Without embodiment, the semantic understanding of the scenario is separated from actions. It is a difficult problem which mainly involves the key issues of semantic scene description, description evaluation, and action instruction generation.

In this work, we propose the *Embodied Scene Description* problem, which exploits the embodiment ability of the agent

to find an optimal viewpoint in its environment for scene description tasks (e.g. *Image Captioning*, *Dense Captioning*, *Image Paragraphing*, etc). The main idea is illustrated in Fig.2. In addition to the visually impaired person, this problem is also extensively applicable to mobile robots. For example, it can facilitate the robot with many tasks such as actively exploring the unknown environment, quickly acquiring meaningful scene, and automatic photo taking.

To tackle this newly proposed problem, we establish a framework that makes use of existing image description models to guide the agent to explore an embodied environment. We encourage the agent to actively explore the environment and capture scenarios with good semantic description. It is noted that we consider the following two aspects when defining a good semantic description: (1) there should be sufficient visual objects detected in the scene and (2) these visual objects are able to compose a complete and reasonable semantic description. Since both the object detector and the semantic description may make mistakes, the combination of the two aspects is supposed to yield more reliable results. Having the definition of a good scene, we can build a learning framework with the paradigms of imitation learning and reinforcement learning to teach the intelligent agent to generate corresponding sensorimotor activities to explore the environment actively. It is worth noting that this work is different from a type of the embodied QA task[9][10], which is driven by finding answer to the question. In our work, the agent implements the task of environment exploration entirely with intrinsic motivation.

The main contributions are summarized as follows:

1) We propose a new framework for the *Embodied Scene Description* problem, which exploits the embodiment characteristic of the intelligent agent to explore the environment to find the best viewpoint for scene description in an embodied environment.
2) We develop a learning framework with the paradigms of imitation learning and reinforcement learning to help the agent to acquire the intelligence to generate sensorimotor activities.
3) We testify the proposed method on AI2Thor dataset and evaluate its effectiveness using the quantitative and qualitative performance indexes.
4) We implement the proposed method on a robotic platform, which shows promising experimental results in real physical environment.

II. Related Work

The deep learning methods have brought great success in many computer vision tasks such as object recognition[5] and detection[11]. Moreover, many research studies have began to investigate a higher level task of semantic scene description with natural language. The proposed work focuses on the embodiment task of finding an optimal viewpoint for these scene description tasks.

Refs.[12][1] are some early-stage works that propose to use a combination of CNN and LSTM model to generate image captions. These image caption models are further improved by integrating different visual and semantic attention mechanisms[2][13][8]. Due to the fact that information expressed in a single sentence is limited when describing an image[14], researchers begin to investigate some more complex models to bridge the gap between images and human language. Therefore, *Dense Captioning*[3] is proposed, which describes an image with multiple sentences. Each sentence is corresponding to an area within a bounding box in the image. It is further improved by *Image Paragraphing*[4], which is able to generate a long paragraph to describe an image instead of a single sentence. Ref.[15] proposes a better model for *Image Paragraphing*, which utilizes the attention and copying mechanisms, as well as the adversarial training technique.

With the recent rapid development in computer vision and many traditional computer vision problems being addressed, the problem of embodied exploration has gradually emerged[16]. In the embodied exploration, an embodied agent actively explores the environment to have a better understanding of the scene[17]. Contrary to traditional computer vision tasks, which mainly focus on analyzing static images passively, embodied exploration requires the agent both understands the content of the current image and takes proper actions accordingly to explore the environment. In most cases, the agent needs to make decisions based on observed image sequences instead of a single image[18].

Ref.[19] develops the target-driven visual navigation, where the agent tries to find an object that is given by an RGB image in an indoor scenario. The model is improved in [20] by incorporating the semantic segmentation information. The embodied visual recognition task proposed in [21] aims to address the problem of navigating in an embodied environment to find an object which might be occluded at first glance. Refs.[22][23] investigate the look-around behavior through active observation completion. Recently, language understanding and active vision are tightly coupled. In [24], the authors propose the task of visual-and-language navigation, where the agent is expected to follow the given language instructions, and use the collected vision information to navigate through the indoor scene. Refs.[9][10] develop embodied question answering and interactive question answering tasks, where an agent is spawned at a random location in a 3D environment and explore to answer a given question. Such tasks have attracted many attentions from the computer vision communities[25][26][27][28]. Although more and more work has taken the embodiment into consideration, the investigated tasks mainly focus on object search, scene recognition, and question answering. The problem of scene description in an embodied environment has not be investigated yet.

To solve the embodied perception problem, the deep reinforcement learning has become the most popular method for its ability to integrate the perception and action modules seamlessly. However, many scholars have pointed out that the end-to-end training for such complex tasks is rather difficult to converge[29]. To tackle this problem, some hybrid learning methods are proposed, such as sidekick policy learning which allows the agent to learn via an easier auxiliary task[30].

In addition, some work prefers to use the imitation learning method[31][32] for pre-training and use reinforcement learning for fine-tuning[33]. In this work, we resort to such methodology to solve the proposed *embodied scene description* task.

III. Problem Formulation

The goal of this work is to develop a method to help the agent to rapidly find a proper viewpoint to capture a scene for generating the high-quality semantic scene description. Concretely speaking, we denote the image captured by the agent as $\boldsymbol{I}_t$ and the corresponding description as $\mathcal{U}(\boldsymbol{I}_t)$, where t is the time instant. Please note that the operator $\mathcal{U}(\cdot)$ denotes the description generation procedure, which can be easily implemented by existing work, such as *Image Captioning*, *Dense Captioning*, *Image Paragraphing*, and so on.

Fig.2 gives an intuitive introduction of the *Embodied Image Description* problem. At time instant $t = 0$, the captured $\boldsymbol{I}_0$ may contain *wall* only and the produced caption *a white wall with a white wall with a wall in a wall* is non-informative. Then the agent exploits its embodied capability to select an action to explore the room and get a new image. Such procedure is iterated until the agent captures an image containing plenty of objects and produces the informative caption *A living room with a couch and a table*. The problem is therefore formulated as to develop an appropriate policy π to help the agent to search a high-quality scene description about the scene. At each step t, the developed policy is used for the agent to take action a_t to acquire the observed image $\boldsymbol{I}_t$.

Though our general idea is to learn action policies for an agent to locate a target scene in indoor environments using only visual inputs, the target scene is not specified by the user. This significantly differs from the work in [19][20] which requires a pre-specified target image.

IV. Navigation Model

The proposed navigation model is shown in Fig.3. The action the agent would take in one step can be relevant to all its previous actions and observations. Therefore, we model it using the LSTM network, which is very commonly used for sequence modeling [1][2]. With the learned policy, the agent is expected to take as few steps as possible to approach the target scene from a random starting position.

A. State Representation

In our implementation, we use a small ResNet-18 [5] as a feature extractor, which is trained from scratch, jointly with the navigation model.

Since the image semantic segmentation results can improve the generalization performance[20], we also use the class segmentation map to help image description generation. To this end, we modify the number of input channels of the original ResNet-18 from 3 to 6. The added 3 channels are used to deal with the class segmentation map. Fig.4 shows how those 6 channels of the input fed to ResNet are generated. We use PSPNet [34] to predict the class segmentation map for a given image.

Furthermore, since we need to train the model with thousands of images in one batch (100 scenes times approximately 30 steps at most for the demonstration trajectories of those scenes, which means about 3000 images in one batch), we shrink the original ResNet-18 to a smaller network with 10 convolution layers. There are 4 kinds of residue building blocks in the original ResNet18, and each of them is repeated twice, leading to 16 convolution layers in residue blocks with parameters (and there are another 2 convolution layers in ResNet18). We use each kind of those residue building blocks only once, yielding a model with only $18 - (16 - 8) = 10$ weighted layers. Besides, the output channels for all convolution layers are also halved (e.g. 512 output channels are shrunk to 256 channels for the final output layer).

Since the description result $\mathcal{U}(\boldsymbol{I}_t)$ can directly show how well the scene description model performs for the current frame, we extract the Bag-of-Words (BoW) feature $\boldsymbol{L}_t$ for all appeared words (after removing stop words) of the output of the 2D image understanding model.

Finally, we combine those multiple features to form the state representation. Denoting the class segmentation map of $\boldsymbol{I}_t$ as $\tilde{\boldsymbol{I}}_t$, the state vector can be represented as

$$\boldsymbol{s}_t = [ResNet(\boldsymbol{I}_t, \tilde{\boldsymbol{I}}_t); W_L \boldsymbol{L}_t] \tag{1}$$

where $ResNet$ denotes the feature extraction module mentioned above. W_L is a trainable parameter for language embedding.

B. Action Space

As illustrated in Fig.5, for one step, we permit the agent to perform the following two kinds of discrete actions in the plane:

1) *Move:* The agent can take nine basic actions which correspond to 8 directions and *no move*. The move step is set to a fixed value Δ_m and the set of the movement actions is denoted as $\mathcal{A}_M$. In this work, we set $\Delta_m = 0.25$m.
2) *Rotation:* The agent can rotate for a fixed interval of Δ_r. In this work, we set $\Delta_r = 45^\circ$ and therefore the set of the rotation actions $\mathcal{A}_R$ contains 8 action atoms.

For each step, the complete action space of the agent is $\mathcal{A} = \mathcal{A}_M \times \mathcal{A}_R$ and the agent is permitted to take the move actions firstly and the rotation action secondly. Please note that if the agent selects the action *no move* from $\mathcal{A}_M$ and 0 from $\mathcal{A}_R$, then the exploration is completed and the obtained image with description is reported as the final result.

In practical environment, the agent has motion limitation and may encounter obstacles or dead corner. Thus the selected action may not be realized. To solve this problem, the agent may use its sensors to detect the feasible region and construct the available action set $\mathcal{A}_t \subset \mathcal{A}$ for the t-th step.

V. Matching between scene image and description

Since the goal of the navigation module is to guide the agent to find the scene which is good for both the image itself and the semantic description, we should design a matching score

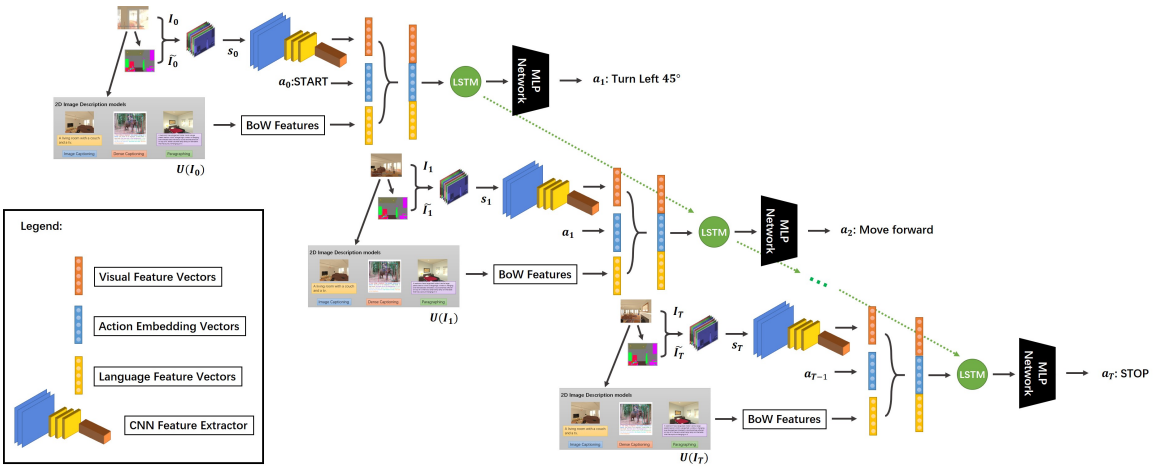

Fig. 3. The proposed navigation model.

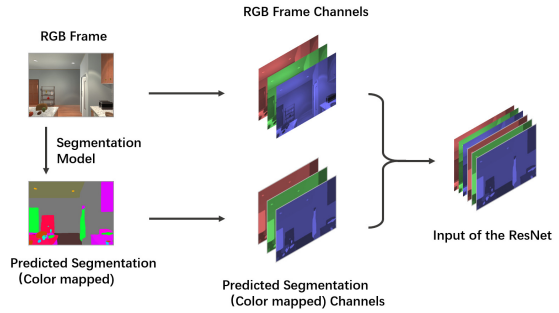

Fig. 4. Demonstration of the visual input feed into the ResNet.

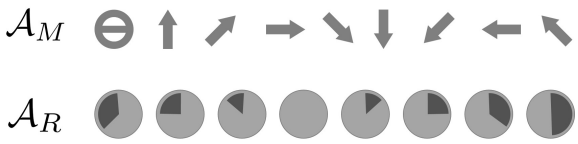

Fig. 5. A representative action space. Please note that some actions (such as *move left*) can be easily implemented in the simulation environment, but cannot be realized by some mobile agents, due to the non-holonomic constraints.

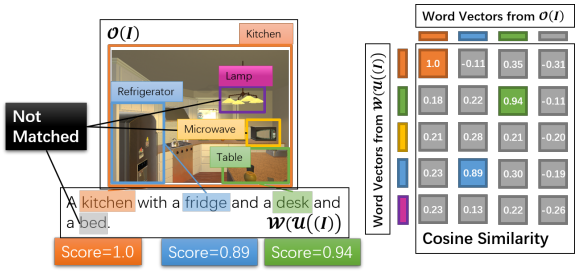

Fig. 6. Demonstration of the proposed scoring function.

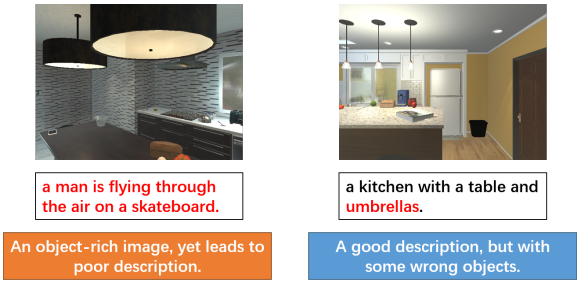

Fig. 7. Some examples which show the mis-match between images and the corresponding description.

between the scene image and its description. This is indeed not a trivial task because the visual object detection results may contain noises and the image-text translator is far from perfectness. On one hand, an object-rich image is preferred but may lead to poor or non-informative description. On the other hand, a good description may include some wrong objects which do not appear in the image at all. See Fig.7 for some examples.

To tackle this difficulty, we apply the off-the-shelf object detectors on the image $\boldsymbol{I}$ to find the visual objects, and extract the object nouns in the text description $\mathcal{U}(\boldsymbol{I})$. The matching score $score(\boldsymbol{I})$ is designed according to their connections. By matching those words with all of the detected objects for one image, we can quantitatively measure how "good" a viewpoint is.

We denote all appeared words in the category labels of all detected objects in the image I as $\mathcal{O}(\boldsymbol{I}) = \{o_1, o_2, \cdots, o_n\}$, and all appeared noun words in the output of the description model $\mathcal{U}(\boldsymbol{I})$ as $\mathcal{W}(\mathcal{U}(\boldsymbol{I})) = \{w_1, w_2, \cdots, w_m\}$, where n and m are the numbers of the detected visual objects in the image $\boldsymbol{I}$ and the extracted noun words in the description $\mathcal{U}(\boldsymbol{I})$. Since the vocabularies adopted by the visual object detector and the semantic description may be different, the same object may be expressed by different words (such as *desk* in the image and *table* in the description). We resort to the Word2Vec

[35] embedding to semantically vectorize these words. For the object category label o_i in $\mathcal{O}(\boldsymbol{I})$ and the noun word w_j in $\mathcal{W}(\mathcal{U}(\boldsymbol{I}))$, we can define their similarly as

$$R(o_i, w_j) = k(o_i, w_j) cos\langle o_i, w_j \rangle$$

where $\cos\langle o_i, w_j \rangle$ is the cosine similarity between the word vectors of the two words. The value $k(o_i, w_j)$ is related to the confidence score of the word and the bounding box. Fig.6 is an intuitive demonstration of the matching-based score function.

The determination of the confidence $k(o_i, w_j)$ is dependent of the adopted description model. For example, if the adopted scene description task is *Image Captioning* or *Image Paragraphing*, since there is no specific information provided for the confidence score by the image description model, we just set $k(o_i, w_j) = 1$. For *Dense Captioning* task, we have the confidence score and bounding box provided by the dense captioning model, therefore we can set $k(o_i, w_j) = IoU(BB(o_i), BB(w_j)) \cdot C(w_j)$, where $BB(\cdot)$ is the corresponding bounding box and $C(\cdot)$ is the confidence score for the bounding box provided by the dense captioning model.

Based on the definition of $R(o_i, w_j)$, we can formulate the calculation of the similarity $sim(\boldsymbol{I}, U(\boldsymbol{I}))$ as the maximum matching problem between the sets $\mathcal{O}(\boldsymbol{I})$ and $\mathcal{W}(\mathcal{U}(\boldsymbol{I}))$. Such a problem can be easily solved using the Hungarian algorithm. Please note that this similarity value is normalized to [0,1].

Finally, we combine the similarity between image-description and the richness of objects to define the following viewpoint scoring function:

$$score(\boldsymbol{I}) = sim(\boldsymbol{I}, \mathcal{U}(\boldsymbol{I})) + \lambda \frac{|\mathcal{O}(\boldsymbol{I})|}{N} \tag{2}$$

where λ is a penalty parameter; the symbol $|\cdot|$ denotes the number of atoms in a set and N represents the number of all possible objects. The second term encourages the agent to search the object-rich scene. It is very useful to some description tasks such as *Image Captioning*, which usually contains few words.

VI. Learning for Embodied Scene Description

A natural method to train the model presented in the previous section is the reinforcement learning algorithm. However, training such a complex model using end-to-end reinforcement learning from scratch is very hard to converge[30]. Therefore, we first use demonstrations to develop imitation learning method to train the embodied scene description model from scratch, and then fine-tune this model with reinforcement learning. Such methodology has been extensively used for several difficult tasks[10].

A. Imitation Learning

The goal of imitation learning for sequential prediction problems is to train the agent to mimic expert behavior for some tasks. To develop the imitation learning algorithm, we have to annotate some scenes with the pre-trained caption model and generate demonstrations for the agent. Therefore, for a specific scene $\mathcal{S}$, we discretize it with grids of a fixed size of Δ_m, and fixed angle Δ_r as is stated in the *Action Space* section. For each possible position (x, y) with rotation ϕ, the corresponding viewpoint can be represented as the tuple (x, y, ϕ). We denote all these possible discrete viewpoints in the given scene $\mathcal{S}$ as $\mathcal{S}_D$. With some abuse of notation, we use $score(x, y, \phi)$ to represent the score of the image which is captured at this viewpoint.

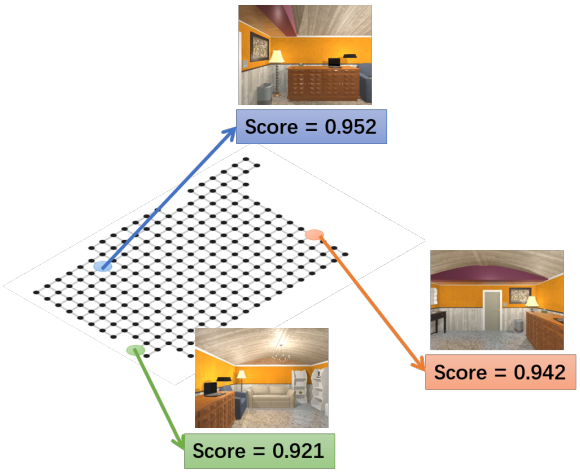

Fig. 8. Demonstration of selecting the target locations.

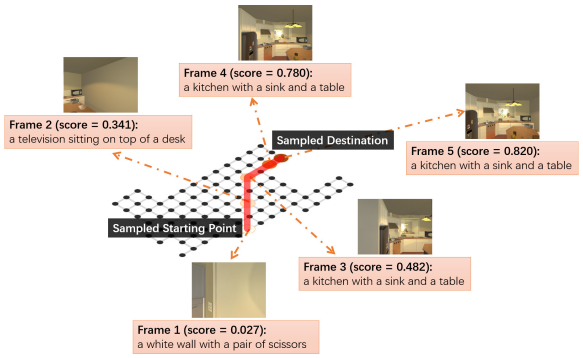

Fig. 9. Demonstration of the generated shortest path.

To produce the demonstration trajectories for a scene, we first find a special viewpoint (x^*, y^*, ϕ^*):

$$(x^*, y^*, \phi^*) = \underset{(x,y,\phi) \in \mathcal{S}_D}{\arg\max}\ score(x, y, \phi) \tag{3}$$

which achieves the highest score $s_{max} = score(x^*, y^*, \phi^*)$. Then we randomly sample one item from the set of candidate locations of which the score is in the interval of $[\gamma s_{max}, s_{max}]$ as the target location (demonstrated in Fig.8). The parameter γ is set to 0.95 to prevent over-fitting. Finally, the shortest path between one randomly selected initial point and the target point, demonstrated in Fig.9, can be obtained using the all-pairs shortest path table generated by the Floyd-Warshall algorithm[36]. This path is used as the demonstration trajectory. Using the multiple demonstrations from various scenes, we can develop supervised imitation learning to train the feature extractor and the navigation model together. The loss

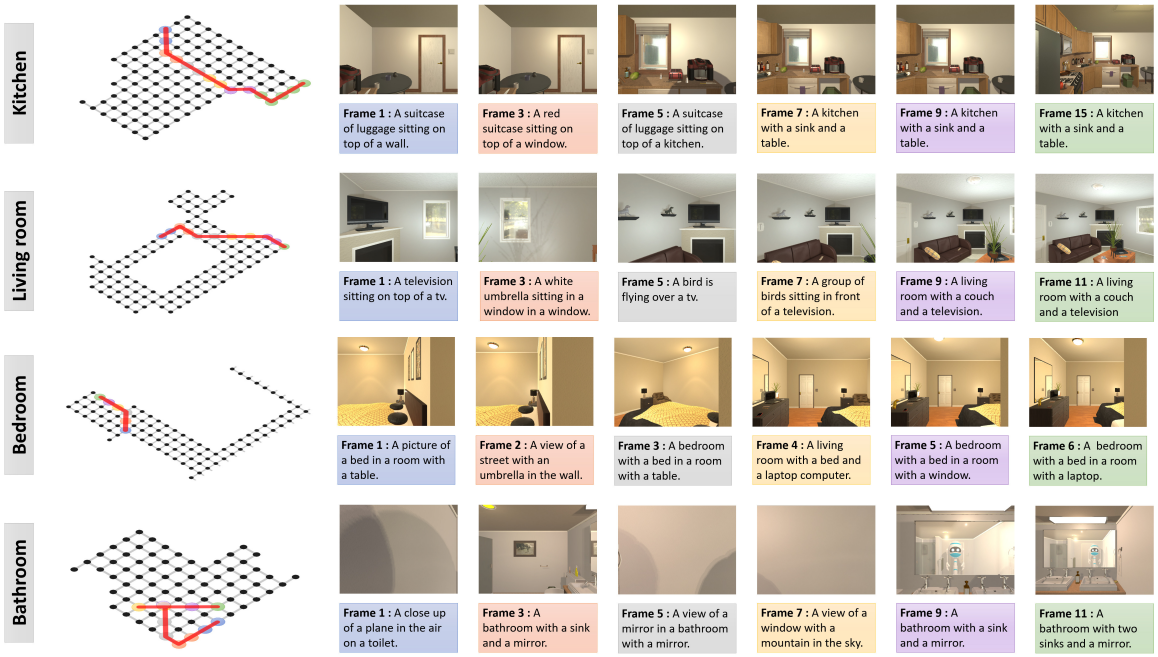

Fig. 10. Four representative examples for the scenes *Living Room*, *Kitchen*, *Bedroom* and *Bathroom*. The trajectories of the agent are shown as red curves in the left panel.

function is defined as follows

$$\mathcal{L}_\theta = \sum_{k=1}^{K}\sum_{t=1}^{T_k} -\log \pi_\theta(\hat{a}_{k,t}|\hat{s}_{k,0}, \hat{a}_{k,0}, \hat{s}_{k,1}, \hat{a}_{k,1}, \cdots, \hat{s}_{k,t}), \tag{4}$$

where K is the number of demonstration trajectories used for training in one batch, T_k is the length of the k-th trajectory, $\hat{s}_{k,t}$ and $\hat{a}_{k,t}$ are the annotated observation and action, and θ denotes all of the parameters to be optimized. During the training phase, we assume a map of the environment is available and give the agent access to information about the shortest paths to some targets.

B. Fine-Tuning with Reinforcement Learning

After pre-training the navigation model with imitation learning, we then try to further improve its performance using the REINFORCE algorithm. The key to fine-tune the model is to design the reward function for the generated trajectory. Generally speaking, we hope to get high-quality caption within a short period of time and therefore a score can be designed as $p_t = score(\boldsymbol{I}_t) - \rho t$, where ρ is used to balance the scales of the two terms and is set to 0.01.

According to the above definition, the immediate reward is set as the incremental of the score $r(s_t, a_t) = p_t - p_{t-1}$ and the cumulative reward which is used to fine-tune the model is as

$$R(s_t, a_t) = r(s_t, a_t) + \sum_{t'=t+1}^{T} \beta^{t'-t} r(s_t, a_t), \tag{5}$$

where the discounted parameter β is set to 0.99 and T is the prescribed maximum steps and is set to 40.

Based on the above-defined reward function, we use REINFORCE algorithm to fine-tune the policy network. To reduce the variance of the reward and improve the stability, we record the moving average of the reward and minus the reward by that moving average in practical training. We use SGD optimizer with a learning rate of 10^{-3}.

VII. Experiment Results

The proposed framework is able to generalized to various semantic description tasks such as *Image Captioning*, *Dense Captioning*, *Image Paragraphing*, and so on. Considering that the *Image Captioning* provides a single-sentence description, which is more intuitive and convenient for practical applications, we focus on the scene description task of *Image Captioning* in this section for performance evaluations. The experimental results on the tasks of *Dense Captioning* and *Image Paragraphing* are illustrated in the supplementary materials.

A. Dataset

For there isn't any existing dataset for the proposed *Embodied Scene Description* task, we generate a new dataset with the AI2Thor dataset. A pretrained caption model is used to generate captions for each scene from different viewpoints.

The AI2Thor dataset contains 120 scenarios belonging to four categories: *Living Room*, *Kitchen*, *Bedroom* and *Bathroom*. Each category has 30 rooms. For each category, we use 25 rooms as the training set, and 5 rooms as the validation/test set. The layout of the room is discretized with grids. For those rooms used as validation/test set, one fixed point in every 4×4 grid is regarded as a point in validation set and the

TABLE I
PERFORMANCE COMPARISON

	NoS	SoL^*	SoL	BLEU-1	BLEU-2	BLEU-3	BLEU-4	Meteor	ROUGE_L	CIDEr
Random	26.78	0.3015	0.3017	0.6176	0.5088	0.4252	0.3686	0.2598	0.6086	1.7135
IL (RGB)	21.38	0.7398	0.7430	0.8471	0.7997	0.7537	0.7144	0.4633	0.8293	4.8382
IL (Segm.)	19.76	0.7524	0.7525	0.8607	0.8113	0.7611	0.7154	0.4653	0.8368	4.6627
IL (RGB+Segm.)	15.43	0.7777	**0.7734**	0.8741	0.8334	0.7902	0.7502	0.4910	0.8625	4.9376
RL (RGB+Segm.)	18.38	0.4490	0.4531	0.7228	0.6399	0.5682	0.5139	0.3401	0.7106	2.8099
IL+RL(RGB+Segm.)	15.10	**0.7813**	0.7724	**0.8752**	**0.8345**	**0.7906**	**0.7502**	**0.4913**	**0.8626**	**4.9482**

* denotes the evaluations on the validation set.

rest 15 points belong to the test set. The image caption model proposed in [8] is adopted for its satisfying performance. The model is trained with the MSCOCO captioning dataset[37].

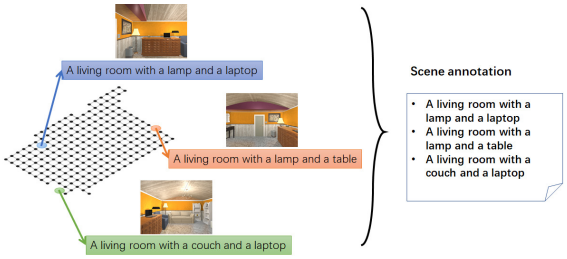

Fig. 11. The annotation for the scene using selected three representative viewpoints.

B. Evaluation Metrics

To evaluate the performance of the *embodied scene description* task, we resort to the score function defined in Eq.(2) to calculate the score for each location in the room. Concretely speaking, for each scene, we select all of the locations whose scores are in the interval of $[\gamma s_{max}, s_{max}]$ and generate their corresponding captions. The generated captions are combined together to act as the ground truth annotation of the scene (Fig.11). Based on this ground truth annotation, the following metrics are designed:

1) *Number of Steps (NoS):* The number of steps the agent takes before stopping.
2) *Score of the Last Image(SoL):* The score of the location which triggers the *Stop* action.
3) *Natural Language Metrics:* With the generated ground truth annotation for each scene, metrics for natural language tasks can be used for evaluating the proposed task. We select several metrics including *BLEU-1*, *BLEU-2*, *BLEU-3*, and *BLEU-4* which are based on the n-gram precision[38], *Meteor* which considers the word-level alignment, *ROUGE_L* which is based on the longest common sub-sequence[39], and CIDEr[40].

C. Result Analysis

The performance of the proposed framework for the scene description of image captioning is illustrated in Table I. Comprehensive comparisons are conducted with several different settings. The full implementation of the proposed framework is denoted as *IL+RL (RGB+Segm.)*. *IL (RGB)* and *IL (Segm.)* only use the imitation learning for the RGB image and segmentation map respectively. *IL (RGB+Segm.)* uses both the RGB image and semantic segmentation map under the same imitation learning framework without reinforcement learning. We also investigate the performance of the reinforcement learning only framework which is trained from scratch and it is denoted as *RL (RGB+Segm.)*. The baseline method is to randomly select each action.

The detailed results over all test samples are summarized in Table I, from which we have the following observations:

- *The combination of IL+RL demonstrates good performance:* The results show that IL+RL method which contains both the pre-training process using imitation learning and fine-tuning process using reinforcement learning achieves the best performance according to all of the language-related metrics including BLEU, Meteor, ROUGE_L and CIDEr. This verifies that the proposed method indeed helps the agent to find good viewpoints to get high-quality captions with an average of 15.10 steps that is satisfying among all the methods.
- *The RL only method works poorly.* The RL only method, though using the same information with IL+RL, yields very poor results, which is just better than the baseline random method. It demonstrates that the pre-training process using the imitation learning is helpful.
- *Both RGB and semantic segmentation information are important.* It can be seen that with the same imitation learning framework, using both the RGB image and semantic segmentation information has better performance than that with single modal information. The main reason is that RGB image is supposed to provide more details, while the segmentation map provides higher-level semantic information. In addition, IL(RGB) and IL(Segm.) methods take more steps before the stop action is triggered.

Although the obtained results are promising, we notice that the fine-tuning process using reinforcement learning only slightly improves the performance. This is in accordance with the results shown in existing literature [10][31]. However, we believe the fine-tuning step could play more important roles when some more sophisticated strategies are utilized.

D. Representative Examples

In Fig.10, we list four representative examples for different scenario categories. The agent is able to navigate in the room

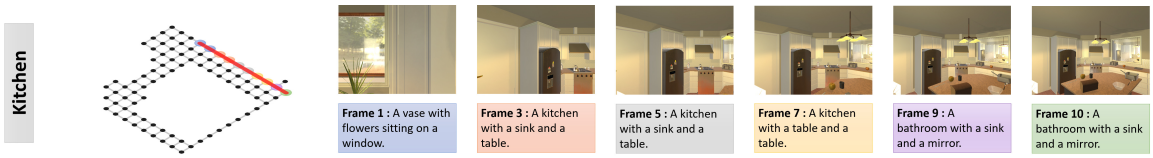

Fig. 12. Failure case. Though an object-rich scene is finally discovered, the adopted caption model does not work well.

Fig. 13. LEFT: The developed robotic platform. A Kinect is equipped on the top of it and we carefully adjust its position to ensure the optical center of the camera in Kinect is aligned with the center of the platform. RIGHT: Two representative real working scenes for the agent. For each scene, we show the the-third-person view (left) and the first-person view (right).

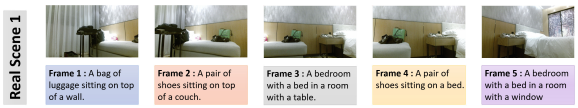

Fig. 14. Real Scene 1: The robot turns around from the wall corner to the bed and finally correctly recognizes the bedroom scene.

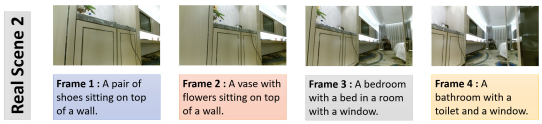

Fig. 15. Real Scene 2: The robot adjusts its position to observe the room. But the caption model mistakenly recognizes the scene as a bathroom.

and finally find a good viewpoint to describe the scene. For example, for a living room which is illustrated in the second row of Fig.10, the agent can only see the television initially, and then it continuously explores in the room until it reaches a position where a good view of the living room is obtained. For a bathroom which is illustrated in the last row of Fig.10, the agent starts at a location where only the wall is visible. With the help of the navigation model, it gradually discovers the mirror and the sinks. Finally, it generates a description that contains major objects in the bathroom.

It is noted that the performance of *embodied scene description* is also strongly dependent of the scene description task (*Image Captioning* in this task). In Fig.12 we show a failure case. One possible reason is that the caption model generates improper caption for the scene even though the agent is actually find a good viewpoint for the scene description.

We also perform extensive experimental validation on some other typical scene description tasks such as *Dense Captioning* and *Image Paragraphing*. The results are shown in the Appendix and the video.

E. Real-World Experiments

Our model is trained on the AI2Thor dataset, rendered with Unity in high quality real-time realistic computer graphics, which makes the difference between the real world and simulation environments minimal. Therefore, it is possible for a simulation-to-real transfer and applying our model to real-world robots and scenes. The learned policy is able to provide action instructions to the robot.

As shown in Fig.13, a mobile robot equipped with a Kinect camera is used in the real world experiment. The mobile robot is able to rotate 360 degrees around itself and move forwards/backwards flexibly, which allows for implementing actions in the action space. The Kinect camera is mounted on the top of the mobile robot and is used to collect egocentric images in real time. The robot is placed in an unseen hotel room for a simulation-to-real experiment. Although the layout of the room and the viewpoint of the camera are significantly different from those in the simulation environment, promising results are obtained to validate the effectiveness of the trained model. In Real Scene 1 (Fig.14), the robot firstly faces to a corner of the room. With the generated instructions, the robot moves around the room until it recognizes that it is a bedroom scene. In Real Scene 2 (Fig.15), the robot starts with a scene where it faces to the door of a cabinet and it is difficult to obtain much useful semantic description. Then, with generated action instructions, the robot adjusts its positions gradually and stops at a position where it can get a full view of the room. It reflects that the learned model is capable of transferring the semantic knowledge learned in simulation environment to real world environment. However, it can be seen that although the robot moves in a reasonable path, the captions generated are indeed wrong. It is because that the caption model used is not robust and accurate enough. More details can be found in the attached video.

VIII. Conclusions

In this work, we propose the new *Embodied Scene Description* problem, in which the agent exploits its embodiment ability to find an optimal viewpoint in its environment for scene description tasks. A learning framework with the paradigms of imitation learning and reinforcement learning is established to teach the agent to generate corresponding sensorimotor activities. The trained model is evaluated in both the simulation and real world environment demonstrating that the agent is able to actively explore the environment for good scene description.

This work only takes one single frame into consideration for each step. Image sequences collected during the exploration process is believed to reveal more information for a better scene description. It will be also useful to leverage the attention, preference, and 3D relationship between objects to further actively understand the scenario. In the future, we plan to integrate this feature into smartphones and intelligent glasses, which are supposed to assist visually impaired person for a better living.

REFERENCES

[1] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3156–3164.

[2] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.

[3] J. Johnson, A. Karpathy, and L. Fei-Fei, "Densecap: Fully convolutional localization networks for dense captioning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4565–4574.

[4] J. Krause, J. Johnson, R. Krishna, and L. Fei-Fei, "A hierarchical approach for generating descriptive image paragraphs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 317–325.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[6] "https://news.microsoft.com/features/bonjour-bienvenidos-seeing-ai-expands-to-5-new-languages/."

[7] D. H. Park, T. Darrell, and A. Rohrbach, "Robust change captioning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4624–4633.

[8] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6077–6086.

[9] D. Gordon, A. Kembhavi, M. Rastegari, J. Redmon, D. Fox, and A. Farhadi, "Iqa: Visual question answering in interactive environments," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4089–4098.

[10] A. Das, S. Datta, G. Gkioxari, S. Lee, D. Parikh, and D. Batra, "Embodied question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 2054–2063.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[12] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2625–2634.

[13] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proceedings of*

the IEEE conference on computer vision and pattern recognition, 2016, pp. 4651–4659.

[14] J. Song, Y. Guo, L. Gao, X. Li, A. Hanjalic, and H. T. Shen, "From deterministic to generative: Multimodal stochastic rnns for video captioning," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 10, pp. 3047–3058, 2018.

[15] X. Liang, Z. Hu, H. Zhang, C. Gan, and E. P. Xing, "Recurrent topic-transition gan for visual paragraph generation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3362–3371.

[16] K. Chen, J. P. de Vicente, G. Sepulveda, F. Xia, A. Soto, M. Vazquez, and S. Savarese, "A behavioral approach to visual navigation with graph localization networks," *Robotics: Science and Systems*, 2019.

[17] H. Li, Q. Zhang, and D. Zhao, "Deep reinforcement learning-based automatic exploration for navigation in unknown environment," *IEEE transactions on neural networks and learning systems*, 2019.

[18] F. Sadeghi, "Divis: Domain invariant visual servoing for collision-free goal reaching," *Robotics: Science and Systems*, 2019.

[19] Y. Zhu, R. Mottaghi, E. Kolve, J. J. Lim, A. Gupta, L. Fei-Fei, and A. Farhadi, "Target-driven visual navigation in indoor scenes using deep reinforcement learning," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 3357–3364.

[20] X. Ye, Z. Lin, J.-Y. Lee, J. Zhang, S. Zheng, and Y. Yang, "Gaple: Generalizable approaching policy learning for robotic object searching in indoor environment," *IEEE Robotics and Automation Letters*, 2019.

[21] J. Yang, Z. Ren, M. Xu, X. Chen, D. Crandall, D. Parikh, and D. Batra, "Embodied visual recognition," *arXiv preprint arXiv:1904.04404*, 2019.

[22] D. Jayaraman and K. Grauman, "Learning to look around: Intelligently exploring unseen environments for unknown tasks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1238–1247.

[23] S. K. Ramakrishnan, D. Jayaraman, and K. Grauman, "Emergence of exploratory look-around behaviors through active observation completion," *Science Robotics*, vol. 4, no. 30, p. eaaw6326, 2019.

[24] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. van den Hengel, "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3674–3683.

[25] L. Yu, X. Chen, G. Gkioxari, M. Bansal, T. L. Berg, and D. Batra, "Multi-target embodied question answering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6309–6318.

[26] E. Wijmans, S. Datta, O. Maksymets, A. Das, G. Gkioxari, S. Lee, I. Essa, D. Parikh, and D. Batra, "Embodied question answering in photorealistic environments with point cloud perception," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6659–6668.

[27] Y. Wu, L. Jiang, and Y. Yang, "Revisiting embodiedqa: A simple baseline and beyond," *arXiv preprint arXiv:1904.04166*, 2019.

[28] A. Das, G. Gkioxari, S. Lee, D. Parikh, and D. Batra, "Neural modular control for embodied question answering," *arXiv preprint arXiv:1810.11181*, 2018.

[29] D. Jayaraman and K. Grauman, "End-to-end policy learning for active visual categorization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 7, pp. 1601–1614, 2018.

[30] S. K. Ramakrishnan and K. Grauman, "Sidekick policy learning for active visual exploration," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 413–430.

[31] X. Wang, Q. Huang, A. Celikyilmaz, J. Gao, D. Shen, Y.-F. Wang, W. Y. Wang, and L. Zhang, "Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6629–6638.

[32] G. Li, M. Mueller, V. Casser, N. Smith, D. L. Michels, and B. Ghanem, "Oil: Observational imitation learning," *Robotics: Science and Systems*, 2019.

[33] T. Haarnoja, S. Ha, A. Zhou, J. Tan, G. Tucker, and S. Levine, "Learning to walk via deep reinforcement learning," *Robotics: Science and Systems*, 2018.

[34] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2881–2890.

[35] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[36] S. Hougardy, "The floyd–warshall algorithm on graphs with negative cycles," *Information Processing Letters*, vol. 110, no. 8-9, pp. 279–281, 2010.

[37] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: Lessons learned from the 2015 mscoco image captioning challenge," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 4, pp. 652–663, 2016.

[38] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[39] M. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proceedings of the ninth workshop on statistical machine translation*, 2014, pp. 376–380.

[40] R. Vedantam, C. Lawrence Zitnick, and D. Parikh,

"Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4566–4575.