Context-Aware Visual Policy Network for Fine-Grained Image Captioning

Zheng-Jun Zha, *Member, IEEE* Daqing Liu, Hanwang Zhang, *Member, IEEE*, Yongdong Zhang, *Member, IEEE*, and Feng Wu, *Fellow, IEEE*

Abstract—With the maturity of visual detection techniques, we are more ambitious in describing visual content with open-vocabulary, fine-grained and free-form language, *i.e.*, the task of image captioning. In particular, we are interested in generating longer, richer and more fine-grained sentences and paragraphs as image descriptions. Image captioning can be translated to the task of sequential language prediction given visual content, where the output sequence forms natural language description with plausible grammar. However, existing image captioning methods focus only on language policy while not visual policy, and thus fail to capture visual context that are crucial for compositional reasoning such as object relationships (*e.g.*, "man riding horse") and visual comparisons (*e.g.*, "small(er) cat"). This issue is especially severe when generating longer sequences such as a paragraph. To fill the gap, we propose a Context-Aware Visual Policy network (CAVP) for fine-grained image-to-language generation: image sentence captioning and image paragraph captioning. During captioning, CAVP explicitly considers the previous visual attentions as context, and decides whether the context is used for the current word/sentence generation given the current visual attention. Compared against traditional visual attention mechanism that only fixes a single visual region at each step, CAVP can attend to complex visual compositions over time. The whole image captioning model — CAVP and its subsequent language policy network — can be efficiently optimized end-to-end by using an actor-critic policy gradient method. We have demonstrated the effectiveness of CAVP by state-of-the-art performances on MS-COCO and Stanford captioning datasets, using various metrics and sensible visualizations of qualitative visual context.

Index Terms—Image captioning, reinforcement learning, visual context, policy network

1 Introduction

Vision and natural language machine comprehension — the ever-lasting goal in Artificial Intelligence — is rapidly evolving with the help of deep learning based AI technologies [1], [2], [3], [4]. The effective visual [2], [3], [5] and textual representations [1], [4] empower computer vision systems to migrate from fixed-vocabulary, coarse-grained, and low-level visual analysis, *e.g.*, image classification [3] and object detection [5], to open-vocabulary, fine-grained, and high-level visual description, *e.g.*, image captioning [6], [7] and visual question answering [8]. The former has become relatively mature. However, the latter is still far from satisfactory, due to the lack of reasoning capability of deep neural networks [9]. Machine reasoning requires a series of complicated decisions, including inferring task-related context, identifying its efficacy for the current on-going task, as well as modeling the relationships between the context and task. How to build machines that can reason as humans is still a very challenging task [10].

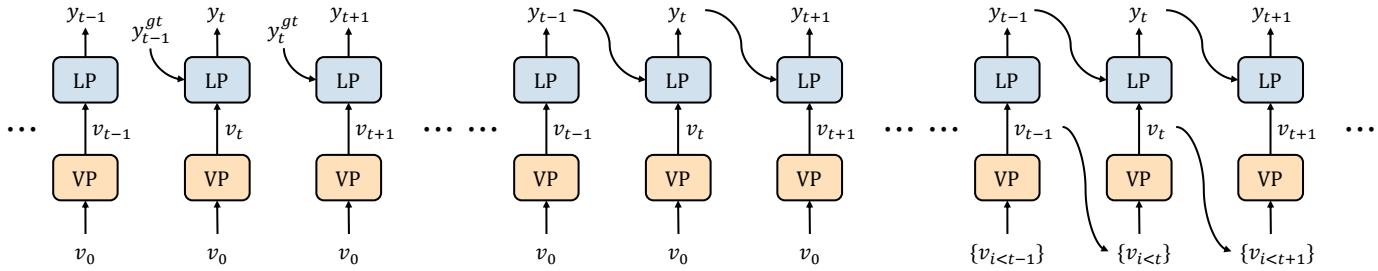
A prime example is image captioning — the task describing images with natural language — which demonstrates a machine's visual comprehension in terms of its ability of grounded natural language modeling [6], [7]. In order for this AI-complete task [11], researchers have attempted to combine the most advanced computer vision (CV) techniques like object recognition [5], relationship detection [12], and scene parsing [13], as well as the modern natural language processing (NLP) techniques such as language generative models [4], [14]. In a nutshell, the CV-end acts as an encoder and the NLP-end plays as an decoder, translating from "source" image to "target" language. Such encoder-decoder architecture is trained using human-annotated image and sentence pairs in a fully-supervised way. The decoder is supervised to maximize the posterior probability of each ground-truth word given the previous ground-truth subsequence and "source" image. Unfortunately, due to the exponentially large search space of language compositions, recent studies have shown that such conventional supervised training tends to learn data bias but not machine reasoning [15], [16], [17]. This issue is especially severe when dealing with the more challenging image paragraph captioning task, where much more fine-grained and detailed paragraphs are expected to be generated from the given image. Hence, it is arguably impossible to build a practical image-to-language system without machine reasoning.

An emerging line of endowing machine reasoning is to execute deep reinforcement learning (RL) in the sequence prediction task of image captioning [3], [18], [19], [20]. As illustrated in Figure 1a, we first frame the traditional encoder-decoder image captioning into a decision-making process, where the visual encoder can be viewed as Visual Policy (VP) that decides where to hold a gaze in the image, and the language decoder can be viewed as Language Policy (LP) that decides what the next word is. As highlighted in Figure 1b, the sequence-level RL-based framework directly

- *Z.-J. Zha, D. Liu, Y. Zhang and F. Wu are with the School of Information Science and Technology, University of Science and Technology of China. E-mail: zhazj@ustc.edu.cn*
- *H. Zhang is with the School of Computer Science and Engineering, Nanyang Technological University.*

Manuscript received April xx, xxxx; revised August xx, xxxx.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2909864, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) Traditional Framework **(b) RL-based Framework** **(c) Our Framework**

Fig. 1: The evolution of the encoder-decoder framework for image captioning. LP: language policy. VP: visual policy. v_t: visual feature at step t. y_t: predicted word at step t. y_t^{gt}: ground-truth word at step t. (a) The traditional framework focuses only on word prediction by exposing the ground-truth word y_{t-1}^{gt} as input to step t for language generation. (b) RL-based framework focuses on sequence training by directly feeding the predicted word y_{t-1} to LP at step t. (c) Our proposed framework explicitly takes historical visual actions $\{v_{i<t}\}$ as visual context at step t.

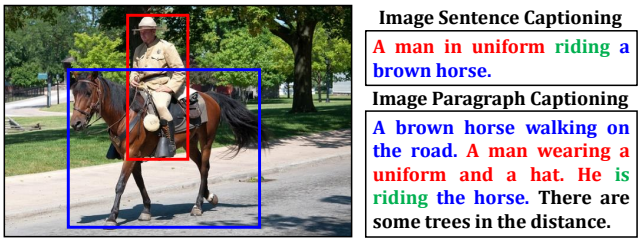


Fig. 2: The intuition of using visual context in fine-grained image captioning. The proposed CAVP is the first RL-based image captioning model which incorporates visual context into sequential visual reasoning.

injects the previously sampled word (sampling by probability distribution) to influence the next prediction. This brings the following two benefits: 1) the training supervision is delayed to the whole sequence generated. Hence, we can use non-differentiable sequence-level metrics such as CIDEr [21] and SPICE [22], which are more suitable than word-level cross entropy loss for language quality evaluation; 2) it avoids the "exposure bias" [23] by performing exploration over sequence compositions at a large scale, leading to fruitful sentences without undesirable overfitting.

However, existing RL-based framework neglects to turn VP into decision-making, *e.g.*, the input of VP is identical in every step as shown in Figure 1b. This disrespects the nature of sequence prediction, where the historical visual actions (*e.g.*, previously attended regions) should significantly influence the current visual policy. One may argue that current visual attention based models would take a hidden memory vector from LP at each time step, which encodes historical cues. However, as we will demonstrate in experiments, this strategy is not able to guide VP to concentrate on the correct regions due to that 1) the LP hidden vector is responsible to memorize linguistic context and hence lacks capacity for storing visual context; 2) it is crucial to exploit visual context to facilitate the production of fine-grained image description with complete story-line.

Motivated by the above observations, we propose a novel Context-Aware Visual Policy (CAVP) network for fine-grained image captioning. As shown in Figure 1c, CAVP allows the previous visual features, *i.e.*, the previous output of CAVP, to serve as the visual context for the current action. Different from the conventional visual attention [7], where the visual context is *implicitly* encoded in a hidden RNN state vector from LP, our visual context is *explicitly* considered in a sequence prediction process. Our motivation is in line with the cognitive evidences that the visual memory recall plays a crucial role in compositional reasoning [24]. As illustrated in Figure 2, for image sentence captioning, it is necessary to consider the related regions, *e.g.*, the previously selected "man", when generating the composition "man riding a horse". For image paragraph captioning, while generating the interaction "riding" between "man" and "horse", we should memorize the regions within blue and red bounding boxes, which had already been concentrated in generating previous sentences. The proposed CAVP explicitly models visual context in visual policy network, leading to context-aware visual feature at each time step, which is more informative and is beneficial to fine-grained image captioning.

We decompose CAVP into four sub-policy networks, which together accomplish the visual decision-making task (cf. Figure 3), each of which is an Recurrent Neural Network (RNN) controlled by shared Long Short-Term Memory (LSTM) parameters and produces a soft visual attention map. As we will show in Section 3.2, this CAVP design reduces the exponentially large search complexity to linear time. By reducing search complexity, it thus stabilizes the conventional Monte Carlo policy rollout. It is worth noting that CAVP and its subsequent language policy network can efficiently model higher-order compositions over time, *e.g.*, relationships among objects mentioned in the generated sub-sequence. Moreover, for generating a paragraph with a hierarchical structure of paragraph-sentence-word, we further develop a hierarchical CAVP network to exploit visual context at both sentence and word levels. We also design a hierarchical reward mechanism consisting of paragraph-level and sentence-level rewards.

The whole framework is trained end-to-end using an actor-critic policy gradient with a self-critic baseline [19]. It is worth mentioning that the proposed CAVP can be seamlessly integrated into any policy-based RL models [25]. We show the effectiveness of the proposed CAVP through

extensive experiments on the MS-COCO image sentence captioning benchmark [26] and Stanford image paragraph captioning dataset [27]. In particular, we significantly improve every SPICE [22] compositional scores such as object, relation, and attribute without optimizing on it. We also show promising qualitative results of visual policy reasoning over the time of generation.

2 Related Work

2.1 Image Sentence Captioning

Inspired by the recent advances in machine translation [4], existing image captioning approaches [6], [7], [28], [29], [30] typically follow an encoder-decoder framework, which can be considered as a neural machine translation task from image to text. It uses CNN-RNN architectures that encode an image as feature vectors by CNN [2], [31] and decode such vectors to a sentence by RNN [1].

More recently, attention mechanisms which allow dynamic feature vectors have been introduced to the encoder-decoder framework. Xu *et al.* [7] incorporated *soft* and *hard* attention mechanisms to automatically focus on salient objects when generating corresponding words. Chen *et al.* [30] introduced channel-wise attention besides spatial attention. Lu *et al.* [28] proposed a visual sentinel to deal with the non-visual words during captioning. Besides the spatial information comes from CNN feature maps, Anderson *et al.* [29] used an object detection network to propose salient image regions with an associated feature vector as bottom-up attention. However, these captioning approaches only focus on the current time step's visual attention and neglect to consider the visual context over time, which is crucial for language compositions. Hence, we propose to incorporate historical visual attentions to current time step as visual context.

2.2 Image Paragraph Captioning

Describing images with a coherent paragraph is challenging. A paragraph contains richer semantic content with longer and more descriptive descriptions. Moreover, a paragraph presents coherent and unified stories. Krause *et al.* [27] proposed a two-stage hierarchical recurrent neural network (RNN) to generate a generic paragraph for an image. The first RNN generates sentence topic vectors and decides how many sentences within the paragraph. The second RNN translates the topic vectors into a sentence. Liang *et al.* [32] incorporated attention mechanism into the hierarchical RNN framework to focus on dynamic salient regions while generating corresponding sentences. They also extended the model with a Generative Adversarial Network (GAN) setting, to encourage coherence among successive sentences. They proposed a GAN-based model consisting of a paragraph generator and two discriminators for personalized image paragraph captioning. Chatterjee *et al.* [33] explicitly introduced coherence vectors and global topic vectors to guide paragraph generation, pursuing the coherence among sentences. Moreover, they cast the model into a variational auto-encoder (VAE) framework to enhance the diversity of paragraphs. Despite the performance of image paragraph captioning has been steadily improved, existing approaches neglect to consider visual context over time, resulting in the lack of correlation among sentences in a paragraph. Meanwhile without reinforcement learning, they suffer from the "exposure bias" between training and sampling. To address these issues, we introduce a hierarchical CAVP model which can generate more coherent and descriptive paragraphs.

2.3 Sequential Decision-Making

Most recent captioning approaches are typically trained via maximum likelihood estimation (MLE), resulting in the "exposure bias" [23] between the training and testing phases. To mitigate it, reinforcement learning has been applied to image captioning, which introduces the notion of sequential decision-making. The idea of making a series of decisions forces the agent to take into account future sequences of actions, states, and rewards. In the case of image captioning, the state consists of visual features, preceding words and visual context, the action is choosing next word and visual representation, and the reward could be any metric of interest.

Several attempts have been made to apply sequential decision-making framework to image captioning. For example, Ranzato *et al.* [23] trained a RNN-based sequence model by policy gradient algorithm based on Monte Carlo search. The policy gradient was used to optimize a sentence-level reward. Rennie *et al.* [19] modified the classic REINFORCE algorithm [34] with a learned baseline which obtained by greedy sampling under the current model to reduce variance of the rewards. As a result, for each sampled caption, it has a sentence level value indicating how good or bad this sentence is. It assumes that each token makes the same contribution towards the sentence. Actor-Critic based method [20] was also applied to image captioning by utilizing two networks as Actor and Critic respectively. Ren *et al.* [3] recast image captioning into decision-making framework and utilized a policy network to choose the next word and a value network to evaluate the policy.

In our work, we formulate the image captioning task into a sequence training framework where each word prediction policy is based on the action performed by the proposed CAVP. Our framework is optimized using policy gradient with a self-critic value which can directly optimize non-differentiable quality metrics of interest, such as CIDEr [21].

3 Approach

In this section, we elaborate the proposed fine-grained image captioning framework. We first formulate the image captioning task into a sequential decision-making process and profile the proposed models in Section 3.1. Then, we introduce the proposed Context-Aware Visual Policy network (CAVP) in Section 3.2 and language policy network (LP) in Section 3.3. We discuss the sequence training strategy for the entire framework in Section 3.4.

3.1 Overview

We formulate the task of image captioning into a sequential decision-making process where an *agent* interacts with the *environment*, and then executes a series of *actions*, so as

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2909864, IEEE Transactions on Pattern Analysis and Machine Intelligence

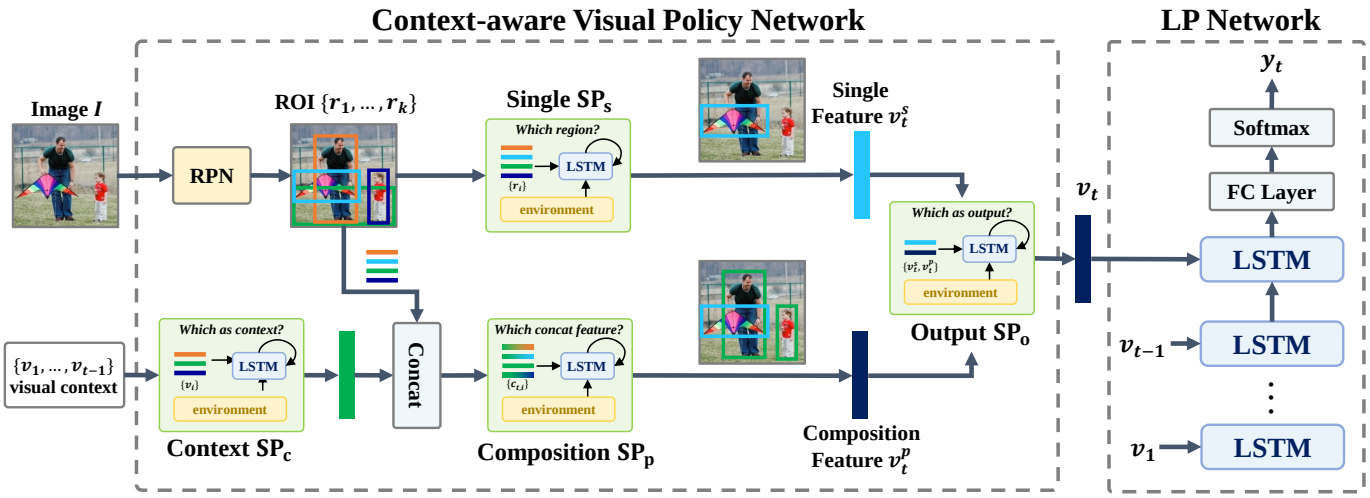


Fig. 3: Overview of the proposed RL-based image sentence captioning framework. It consists of the proposed CAVP for visual feature composition and the language policy for sentence generation. CAVP contains four sub-policy (SP) networks: Single SP, Context SP, Composition SP, and Output SP. t is the current time step and y_t is the predicted word.

to optimize the *reward* return when accomplishing a *goal*. Specifically, the *agent* is the captioning model consisting of a context-aware visual policy network (CAVP) and a language policy network (LP). The *goal* is to generate a language description (sentence or paragraph) Y for a given image I. To accomplish the goal, at each time step t, the *action* of CAVP is to generate a visual representation v_t, the *action* of LP is to predict a word y_t. The observed *state* is the image I, the visual context $\{v_1, \cdots v_{t-1}\}$, and the predicted words $\{y_1, \cdots y_{t-1}\}$ so far. The *environment* is the image I to be captioned. The *reward* could be any evaluation metric score between the ground-truth and the prediction.

Fig. 3 illustrates the overview of the proposed image sentence captioning framework. At each time step t, the CAVP takes image I and visual context $\{v_1, \cdots v_{t-1}\}$ as input to produce a visual representation v_t. The LP takes the visual representation v_t and the preceding word y_{t-1} as input to predict the next word y_t.

Fig. 4 illustrates the overview of the proposed image paragraph captioning framework. The task of generating a paragraph could be accomplished into two steps, *i.e.*, 1) producing a series of topic vectors t, and 2) translating each topic vector t_i into a sentence of words. In particular, we design a hierarchal CAVP-LP architecture for paragraph captioning. We first utilize a sentence-level CAVP-LP which takes image I and visual context $\{v_1, \cdots v_{i-1}\}$ as input and produces a visual representation v_i, a topic vector t_i and a stop probability p^i_{stop}. Then, the topic vector t_i is injected into a word-level LP to constraint the generation of each word $y_{i,j}$ of the i-th sentence.

3.2 Context-Aware Visual Policy Network

To generate a fine-grained description, we perform a series of complicated reasoning processes by decomposing the Context-Aware Visual Policy network (CAVP) into four sub-policy networks (SP): 1) a single SP to obtain the on-going task representation; 2) a context SP to infer task-related context; 3) a composition SP to model the relationship between the context and the on-going task; 4) an output SP to identify the efficacy of context. We first elaborate the four sub-policy networks of CAVP in Section 3.2.1 and then introduce the hierarchical CAVP designed for paragraph captioning in Section 3.2.2.

3.2.1 Sub-Policy Networks

In general, a sub-policy network SP encodes the observed state by an RNN with LSTM cell [1] and performs a real-valued action by soft attention selection. The selection can be considered as an approximation of Monte Carlo rollouts, reducing the sampling variance [35] caused by the diverse image regions. Specifically, at time step t, a sub-policy network as an *agent* observes a *state* s_t and performs an *action* $a_t \sim \pi(a_t|s_t;\theta)$. It then translates this *action* into a representation f as a weighted sum of a series of input features $Q_t = \{q_1, \cdots, q_d\}$, where d is the number of features. Here, without loss of generality, we first introduce the general structure of the sub-policy networks denoted as SP without any superscripts. The general formulation is given by:

$$f = \mathrm{SP}(s_t, Q_t) = \sum_{i=1}^{d} \pi(a_t = i) q_i. \tag{1}$$

To compute the probability distribution of the action, we follow the attention mechanism [7] as:

$$\pi(a_t = i) = \mathrm{softmax}(w_a^T \tanh(W_h h_t + W_q q_i)), \tag{2}$$

where $\pi(a_t = i) \in [0, 1]$, w_a, W_h and W_q are trainable parameters, and h_t is the LSTM hidden state calculated by:

$$h_t = \mathrm{LSTM}(s_t, h_{t-1}). \tag{3}$$

In this way, if we have the state s_t and the input features Q_t at each time step, the sequence decision-making is known.

Next, we elaborate the implementation of each sub-policy network by introducing the corresponding state s_t

and the input features Q_t. We use a superscript to distinguish four sub-policy networks, *i.e*, $\cdot^s$ for single SP, $\cdot^c$ for context SP, $\cdot^p$ for composition SP, and $\cdot^o$ for output SP.

Single Sub-policy Network Before Single sub-policy network, we first use Faster R-CNN [5] to extract image region features $\{r_1, \cdots, r_k\}$ from image I, where k is the number of regions. The observed state s_t^s at time step t consists of the previous LSTM hidden state h_{t-1}^l of the language policy network, concatenated with the mean-pooled region features $\bar{r} = \frac{1}{k}\sum_{i=1}^{k} r_i$, and word embedding of the preceding word y_{t-1}:

$$s_t^s = [h_{t-1}^l, \bar{r}, W_e \Pi(y_{t-1})], \tag{4}$$

where $W_e \in \mathbb{R}^{E\times\Sigma}$ is a word embedding matrix of a vocabulary learned from scratch, and Π is a one-hot encoding function. The input features at each time step are the detected region features, *i.e.*$Q_t^s = \{r_1, r_2, \cdots, r_k\}$. The output of the single sub-policy network is the single feature at time step t:

$$v_t^s = f_t^s = \mathrm{SP}^s(s_t^s, Q_t^s), \tag{5}$$

which is in turn fed into the subsequent output SP.

Context Sub-policy Network At time step t, visual context includes the historical visual outputs $\{v_1, \cdots, v_{t-1}\}$. However, not every visual context is useful for the current word generation. Therefore, we introduce the context sub-policy network SP^c to choose the most informative context and combine it with the detected region features. In particular, we define the observed state as:

$$s_t^c = [h_{t-1}^l, \bar{r}, W_e \Pi(y_{t-1})], \tag{6}$$

and the input features as $Q_t^c = \{v_1, \cdots, v_{t-1}\}$.

By the context sub-policy network, we get the visual context representation f_t^c at time step t as Eqn. 1. Then we fuse f_t^c with region features into context features $c_{t,i}$ as:

$$c_{t,i} = W_c^T [f_t^c; r_i],\ i = 1, 2, \cdots, k, \tag{7}$$

where $[\cdot;\cdot]$ indicates the concatenation of vectors and W_c^T projects context features to the original dimension as region features. The context features will be used in the composition SP.

To investigate the importance of visual context, we propose another way to represent context features, *i.e.*, only considering preceding time step $t-1$ as visual context:

$$c_{t,i} = W_c^T [v_{t-1}, r_i],\ i = 1, 2, \cdots, k, \tag{8}$$

We will discuss this approximation in Section 4.4.1.

Composition Sub-policy Network The composition sub-policy network is similar to the single sub-policy network that takes the previous hidden state of the language policy network, the mean-pooled region features, and an embedding of the preceding word as observed state:

$$s_t^p = [h_{t-1}^l, \bar{r}, W_e \Pi(y_{t-1})]. \tag{9}$$

The input features of the composition sub-policy network are the context features from the context sub-policy network:

$$Q_t^p = \{c_{t,1}, c_{t,2}, \cdots, c_{t,k}\}. \tag{10}$$

Then we take the output of composition sub-policy network as composition features at time step t:

$$v_t^p = f_t^p = \mathrm{SP}^p(s_t^p, Q_t^p). \tag{11}$$

Output Sub-policy Network After obtaining the single and compositional features from Single SP and Composition SP, we produce the visual output v_t at time step t by Output SP. We define the observed state as:

$$s_t^o = [h_{t-1}^l, \bar{r}, W_e \Pi(y_{t-1})], \tag{12}$$

and the input features as $Q_t^o = \{v_t^s, v_t^p, \bar{r}\}$. Inspired by [28], we append an extra feature $\bar{r}$ to input features for non-visual words. We take the output of Output SP as the visual feature v_t:

$$v_t = f_t^o = \mathrm{SP}^p(s_t^o, Q_t^o). \tag{13}$$

The visual feature v_t will be used in language policy network at time step t and also will be seen as visual context in subsequent time steps.

Weight Sharing We notice that the observed state of above sub-policy networks are identical as:

$$s_t^c = s_t^s = s_t^p = s_t^o = [h_{t-1}^l, \bar{r}, W_e \Pi(y_{t-1})]. \tag{14}$$

To reduce the model complexity and computational overhead of CAVP, we share the LSTM parameters among those sub-policy networks in experiments. More ablation studies of the weight sharing will be detailed in Section 4.4.1.

3.2.2 Hierarchical Context-aware Visual Policy Network

We design a hierarchical context-aware visual policy network, consisting of a sentence-level CAVP and a word-level CAVP, for image paragraph captioning. For the sake of simplicity, we denote CAVP as

$$v_t = \mathrm{CAVP}(R, S_t). \tag{15}$$

At time step t, CAVP takes a set of region features R and a sequential visual context S_t as input and produces a sequential visual representation v_t.

Sentence-level CAVP In a paragraph, each sentence should keep continuity to all the previous sentences. This requires that the model is aware of visual context. We thus construct a sentence-level CAVP with the same structure of the CAVP for image sentence captioning in Section 3.2.1. Formally, given the region features $R = \{r_1, r_2, ..., r_k\}$ and visual context $\{v_1, v_2, ..., v_{i-1}\}$, we apply the four sub-policy networks including single SP, context SP, comp. SP and output SP. The output of sentence-level CAVP will be used to guide the word-level visual policy network and feed into the sentence LSTM. Such process is given as:

$$v_i = \mathrm{CAVP}(R, \{v_1, ..., v_{i-1}\}), i = 1, 2, ..., T_s \tag{16}$$

where v_i denotes the visual representation of i-th sentence, T_s is the number of sentences.

Word-level CAVP We construct a word-level CAVP to generate words in sentences. A single sentence in a paragraph generally describes a certain region of the image. We set the visual context as the previous word-level visual representations:

$$v_{i,j} = \mathrm{CAVP}(R, \{v_1, ..., v_{i-1}\}), j = 1, 2, ..., T_w \tag{17}$$

where $v_{i,j}$ denotes the visual representation of the j-th word of the i-th sentence of a paragraph, T_w is the length of sentence. In this way, we can generate the whole paragraph by repeatedly applying the hierarchical language policy network.

3.3 Language Policy Network

We employ a language policy (LP) network towards generating a coherent image description. We first introduce the language policy network for image sentence captioning and then describe the hierarchical language policy network for image paragraph captioning.

At each time step, CAVP generates a context-aware visual representation that is most fitting to the current word. Language policy network take the visual representation and the hidden state h_t^s of Single SP as input, then use them to update LSTM hidden state:

$$h_t^l = \text{LSTM}([h_t^s, v_t], h_{t-1}^l). \tag{18}$$

To compute the distribution over all words in vocabulary, we apply a FC layer to hidden state, and after softmax layer it outputs the probability distribution of each word, given by:

$$\pi_l(y_t|y_{1:t-1}) = \text{softmax}(W_y h_t^l + b_y), \tag{19}$$

where W_y and b_y are learnable weights and biases. For a whole sentence, the distribution is calculated as the product of all time step's conditional distributions:

$$\pi_l(y_{1:T}) = \prod_{t=1}^{T} \pi_l(y_t|y_{1:t-1}). \tag{20}$$

3.3.1 Hierarchical Language Policy Network

We design a hierarchical language policy network for image paragraph captioning, consisting of a sentence-level LP and a word-level LP, which correspond to the sentence-level and word-level CAVPs, respectively. The sentence-level LP is fed by the visual representation v_i from the sentence-level CAVP, while the word-level LP takes $v_{i,j}$ by the word-level CAVP as input. The sentence-level LP is designed to produce a *topic vector* for each sentence and predict the number of sentences of a paragraph. Given a topic vector and visual representation for a sentence, the word-level LP generates each word to form the sentence.

Sentence-level LP Sentence-level language policy network consists a one-layer LSTM and two FC-layers. For each sentence in a paragraph, the LSTM receives visual representation v_i and produces hidden state h_i^s. The hidden state h_i^s is used to generate a topic vector t_i by linear projection as well as a distribution p_{stop}^i over two states {CONTINUE=0, STOP=1} by a softmax classifier. p_{stop}^i indicates whether the current sentence is the last one in the paragraph.

Word-level LP Given a topic vector from the sentence LSTM, the word LSTM is to generate the words to form the corresponding sentence. At each time step, we feed the topic vector concatenated with word embedding vector to word LSTM. The hidden state of the word LSTM is used to predict a distribution over all possible words in vocabulary by a FC-layer and a softmax classifier.

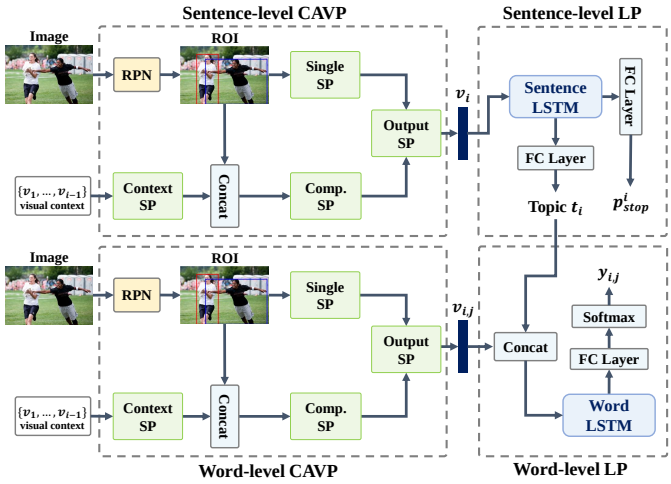


Fig. 4: Overview of the proposed hierarchical CAVP-LP framework for image paragraph captioning, consisting of a sentence-level CAVP-LP and a word-level CAVP-LP.

3.4 Sequence Training

The sequence training process consists of two phases, including pre-training by supervised learning and fine-tuning by reinforcement learning.

For pre-training, we follow the traditional captioning training strategy and optimize the cross-entropy loss between the ground-truth and the probability distribution we produce. Given a target ground-truth sequence $y_{1:T}^{gt}$ and a captioning model with parameters θ, the objective is to minimize the cross entropy loss:

$$L_S(\theta) = -\sum_{t=1}^{T} \log(\pi_l(y_t^{gt} \mid y_{1:t-1}^{gt})). \tag{21}$$

However, the "teacher-forcing" training strategy leads to "exposure bias" which means the model can hardly exposure to real sequential data beyond ground-truth dataset.

Therefore, at the fine-tune stage, we adopt the REINFORCE algorithm [34] to directly optimize the sequence-level metrics and address the exposure bias issue. Specifically, we follow the self-critical method [19]. First, we sample a greedy sequence $\hat{y}_{1:T}$ in greedy manner, *i.e.*, sampling each word with the maximum probability. Then we Monte-Carlo sample another sequence $y_{1:T}^s$, *i.e.*, sampling each word according to the probability distribution the model predicts. The objective is to minimize the negative expected relative score:

$$L_R(\theta) = -E_{y\sim\pi_l}[r(y_{1:T}^s) - r(\hat{y}_{1:T})], \tag{22}$$

where $r(\cdot)$ could be any evaluation score metric, *e.g.*, CIDEr, BLEU, or SPICE. We will discuss the influence of different metrics in Section 4.4.2.

Note that the Eqn. (22) is non-differentiable, we approximate the gradient by the REINFORCE algorithm as:

$$\nabla_\theta L_R(\theta) \approx -(r(y_{1:T}^s) - r(\hat{y}_{1:T}))\nabla_\theta \log \pi_\theta(y_{1:T}^s). \tag{23}$$

While training, this gradient tends to increase the probability of each words in the sampled captions if $r(y_{1:T}^s)$ higher than $r(\hat{y}_{1:T})$, which can been seen as the relative baseline score, and vice versa.

A brute force search of all possible contextual regions requires $\mathcal{O}(2^N)$ complexity for multinomial combinations of N image regions. For linear efficiency, we follow the "divide and rule" principle and divide the overall search process into several compositional reasoning steps by approximating the overall visual policy network as four sub-policy networks. Each sub-policy network only needs to perform specific sub-task which only requires $\mathcal{O}(N)$ complexity, *e.g.*, the single SP only needs to select one region from N image regions and the context SP only needs to select one historical visual output as the current visual context. As a result, the CAVP reduces the complexity exponentially.

3.4.1 Hierarchical Sequence Training

To train the image paragraph captioning model, we adopt the cross entropy loss from a single sentence to a paragraph containing several sentences. Given a target ground-truth paragraph $y^{gt}_{1:T_s \times T_w}$, where y has T_s sentences, and each sentence contains T_w words[1]. Besides a word prediction loss, we also add a sentence ending prediction loss, given:

$$L_S(\theta) = -\lambda_w \sum_{i=1}^{T_s \times T_w} log(\pi_l(y_i^{gt})) - \lambda_s \sum_{i=1}^{T_s} log(p^{gt}_{stop}), \quad (24)$$

where λ_w and λ_s are balancing factors.

While sampling, we run the visual policy network and sentence LSTM until the stopping probability $p_{stop} > 0.5$ or after maximum number of sentences. After trained by cross entropy loss, we also use the policy gradient method to optimize the metric score directly.

Paragraph-level Reward The straightforward extend method is following Eqn. 23, given:

$$\begin{aligned} &\nabla_\theta L_R(\theta) \approx \\ &-(r(y^s_{1:T_s \times T_w}) - r(\hat{y}_{1:T_s \times T_w}))\nabla_\theta \log \pi_l(y^s_{1:T_s \times T_w}) \end{aligned} \quad (25)$$

where $y^s_{1:T_s \times T_w}$ is a paragraph sampled according to distribution, and $\hat{y}_{1:T_s \times T_w}$ is a greedy searched paragraph description. But in this settings, sharing one reward in a whole paragraph is insensitive, while individual rewards for each word is unstable. A trade-off is using sentence-level reward.

Sentence-level Reward Since the model generate the paragraph sentence by sentence, each sentence is based on previous sentences, besides the evaluation of NLP metrics are designed for complete strings, it can't get each sentence's reward directly. To get the sentence-level reward, we design a sampling schedule. For example, to get the i-th sentence reward, we first use the previous $i-1$ ground truth sentences to guide the model, *i.e.* Teacher-Forcing, then sampling the next sentence according to word distribution as y^s_i, or greedy search the next sentence as $\hat{y}_i$. Therefore, according to Eqn. (23), given:

$$\nabla_\theta L_R(\theta) \approx -(r(y^s_{1:T_w}) - r(\hat{y}_{1:T_w}))\nabla_\theta \log \pi_l(y^s_{1:T_w}) \quad (26)$$

3.4.2 Behavior Cloning

The learning would be easier if we have some additional knowledge of the output policy. While there is no any additional knowledge in the caption datasets e.g. MS-COCO, we can use a language parser [36] as an existing expert output policy that can be used to provide additional supervision. More generally, if there is an expert output policy π^e that predicts a reasonable output policy π^o, we can first pre-train our model by behavioral cloning from π^e. This can be done by minimizing the KL-divergence $D_{KL}(\pi^e||\pi^o)$ between the expert output policy π^e and our output policy π^o, and simultaneously minimizing the captioning loss L_{XE} with expert output policy π^e. This supervised behavioral cloning from the expert output policy can provide a good set of initial parameters in our output sub-policy network. Note that the above behavioral cloning procedure is only done at cross-entropy training time to obtain a supervised initialization for our model, and the expert output policy is not used at test time.

The expert output policy is not necessarily optimal, for behavioral cloning itself is not sufficient for learning the most suitable output policy for each image. After learning a good initialization by cloning the expert output policy, our model is further trained end-to-end with gradient $\nabla_\theta L_R(\theta)$ computed using Eqn. (23), where the output policy π^o is sampled from the output policy network in our model, and the expert output policy π^e can be discarded.

1. For simplicity, we ignore the variant length of each sentence.

4 Experiments on Sentence Captioning

In this section, we first introduce the experiment settings. Then, we go through the implementation details. Finally, we report both quantitative and qualitative evaluation results, followed by detailed ablation studies.

4.1 Experiment Settings

4.1.1 Dataset

We used the most popular benchmark **MS-COCO** [26] image sentence captioning dataset, which contains 82,783 images for training and 40,504 for validation. Each image is human-annotated with 5 sentence captions. As the annotations of the official test set are not publicly available, for validating model hyperparameters and offline testing, we follow the widely used "Karpathy" splits [37] in most prior works, containing 113,287 images for training, 5,000 for validation, and 5,000 for testing. We reported the results both on "Karpathy" offline split and MS-COCO online test server.

4.1.2 Metric

The most common metrics for caption evaluation are based on n-gram similarity of reference and candidate descriptions. **BLEU** [38] is defined as the geometric mean of n-gram precision scores, with a sentence-brevity penalty. In **CIDEr** [21], n-grams in the candidate and reference sentences are weighted by term frequency-inverse document frequency weights (*i.e.* tf-idf). Then, the cosine similarity between them are computed. **METEOR** [39] is defined as the harmonic mean of precision and recall of exact, stem, synonym, and paraphrase matches between sentences. **ROUGE** [40] is a measures for automatic evaluation for summarization systems via F-measures.

All the above metrics are originally developed for the evaluation of text summaries or machine translations. It has

been shown that there exist bias between those metrics and human judgment [22]. Therefore, we further evaluated our model using **SPICE** [22] metric, which is defined over tuples that are divided into semantically meaningful categories such as objects, relations and attributes.

4.2 Implementation Details

4.2.1 Data Pre-processing

We performed standard minimal text pre-processing: first tokenizing on white space, second converting all words into lower case, then filtering out words that occur less than 5 times, finally resulting in a vocabulary of 10,369 words. Captions are trimmed to a maximum of 16 words for computational efficiency. To generate a set of image region features R, we take the final output of the region proposed network [5] and perform non-maximum suppression. In our implementation we used an IoU threshold of 0.7 for region proposal non-maximum suppression, and 0.3 for object class non-maximum suppression. To select salient image regions, we simply selected the top $k = 36$ features in each image for computation consider.

4.2.2 Parameter Settings

We set the number of hidden units of each LSTM to 1,300, the number of LSTM layers to 1, the number of hidden units in the attention mechanism we described in Eqn. (2) to 1,024, and the size of word embedding to 1000. During the supervised learning for the cross-entropy process, we use Adam optimizer [41] with base learning rate of 5e-4 and shrink it by 0.8 every 3 epochs. We start reinforcement learning after 37 epochs, we use Adam optimizer with base learning rate of 5e-5 and shrink it by 0.1 every 55 epochs. We set the batch size to 100 images and train up to 100 epochs. During inference stage, we use a beam search size of 5. While training Faster R-CNN, we follow [29] and first initialize it with ResNet-101 [31] pretrained with classification on ImageNet [42], then fine-tune it on Visual Genome [43] with attribute labels.

4.3 Comparisons to State-of-The-Arts

4.3.1 Comparing Methods

Traditional Approaches We first compared our models to classic methods including **Google NIC** [6], **Hard Attention** [7], **Adaptive Attention** [28] and **LSTM-A** [44]. These methods follow the popular encoder-decoder architecture, trained with cross-entropy loss between the predicted and ground-truth words, that is, no sequence training is applied. **RL-based Approaches** We also compared our models to the RL-based methods including **PG-SPIDEr-TAG** [18], **SCST** [19], **Embedding-Reward** [3], and **Actor-Critic** [20]. These methods use sequence training with various reward returns.

4.3.2 Quantitative Analysis

As shown in Table 1, we evaluated our model compared to multiple state-of-the-art methods. We found that almost all RL-based methods outperform traditional ones. The reason is that RL addresses the loss-evaluation mismatch problem and included the inference process in training to address the exposure bias problem. We can also find that our CAVP outperforms other non-context methods. This is because the visual context information is useful for current word generation and the policy makes better decisions. In particular, we achieved state-of-the-art performance under all metrics on "Karpathy" test split. Table 2 reports the performance comparison without any ensemble on the official MS-COCO evaluation server[1]. It is worthy to note that our approach is a single captioning model while the others are based on the ensemble of multiple captioning models.

Model	B@4	M	R	C	S
Google NIC [6]	32.1	25.7	-	99.8	-
Hard-Attention [7]	24.3	23.9	-	-	-
Adaptive [28]	33.2	26.6	54.9	108.5	19.4
LSTM-A [44]	32.5	25.1	53.8	98.6	-
PG-SPIDEr [18]	32.2	25.1	54.4	100.0	-
Actor-Critic [20]	34.4	26.7	55.8	116.2	-
EmbeddingReward [3]	30.4	25.1	52.5	93.7	-
SCST [19]	35.4	27.1	56.6	117.5	-
StackCap [45]	36.1	27.4	56.9	120.4	20.9
Up-Down [29]	36.3	27.7	56.9	120.1	21.4
Ours	**38.6**	**28.3**	**58.5**	**126.3**	**21.6**

TABLE 1: Performance comparisons on MS-COCO "Karpathy" offline split. B@n is short for BLEU-n, M is short for METEOR, R is short for ROUGE, C is short for CIDEr, and S is short for SPICE.

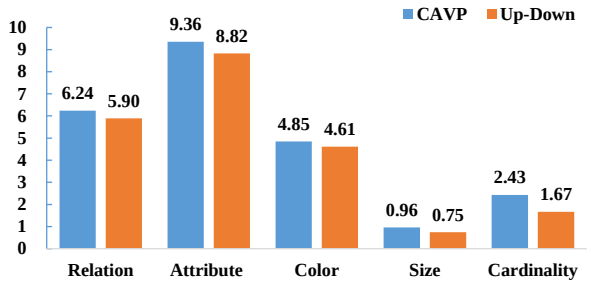


Fig. 5: The performance comparison of the CAVP model and the Up-Down method. All SPICE category scores are improved by CAVP.

To evaluate the compositional reasoning ability of our CAVP model, we also provide SPICE semantic category scores in Fig. 5. Since SPICE parses the language into scene graph and compares the graph similarity, it can provide finer-grained information such as relation, attribute, color, size, and cardinally. We can find that, our CAVP model improves all SPICE semantic category scores while comparing with Up-Down [29] model which neglects visual context. Specifically, the Relation score indicates the reasoning ability of object relationships, *e.g.*, "man riding horse". The Attribute, Color, and Size scores indicate the reasoning ability of visual comparisons, *e.g.*, "small(er) cat". Note that in most cases, the visual comparisons are implicit, for example, when we describe a cat is "small", it means the cat is relatively "smaller" than other objects.

4.3.3 Qualitative Analysis

To better reveal our CAVP model, we show some qualitative visualizations as well as the output of sub-policy network's

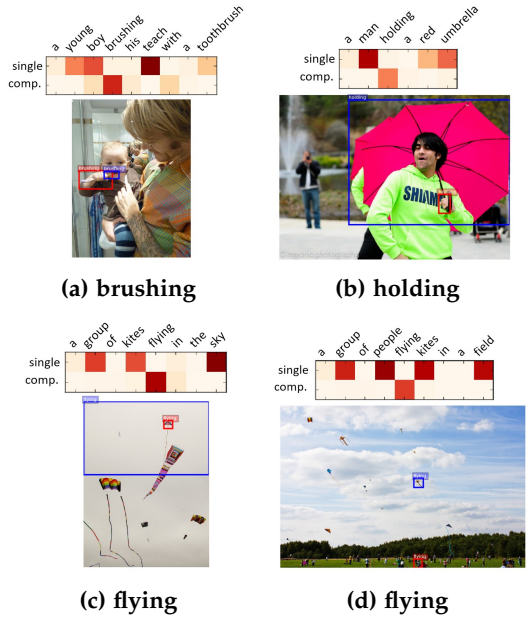
1. https://competitions.codalab.org/competitions/3221#results

	BLEU-1		BLEU-2		BLEU-3		BLEU-4		METEOR		ROUGE-L		CIDEr	
	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40	c5	c40
Google NIC [6]	71.3	89.5	54.2	80.2	40.7	69.4	30.9	58.7	25.4	34.6	53.0	68.2	94.3	94.6
MSR Captivator [46]	71.5	90.7	54.3	81.9	40.7	71.0	30.8	60.1	24.8	33.9	52.6	68.0	93.1	93.7
M-RNN [47]	71.6	89.0	54.5	79.8	40.4	68.7	29.9	57.5	24.2	32.5	52.1	66.6	91.7	93.5
Hard-Attention [7]	70.5	88.1	52.8	77.9	38.3	65.8	27.7	53.7	24.1	32.2	51.6	65.4	86.5	89.3
Adaptive [28]	74.8	92.0	58.4	84.5	44.4	74.4	33.6	63.7	26.4	35.9	55.0	70.5	104.2	105.9
PG-SPIDEr-TAG [18]	75.1	91.6	59.1	84.2	44.5	73.8	33.6	63.7	25.5	33.9	55.1	69.4	104.2	107.1
SCST:Att2all [19]	78.1	93.7	61.9	86.0	47.0	75.9	35.2	64.5	27.0	35.5	56.3	70.7	114.7	116.7
LSTM-A_3 [44]	78.7	93.7	62.7	86.7	47.6	76.5	35.6	65.2	27	35.4	56.4	70.5	116.0	118.0
Stack-Cap [45]	77.8	93.2	61.6	86.1	46.8	76.0	34.9	64.6	27.0	35.6	56.2	70.6	114.8	118.3
Up-Down [29]	**80.2**	**95.2**	64.1	88.8	49.1	79.4	36.9	68.5	27.6	36.7	57.1	72.4	117.9	120.5
Ours	80.1	94.9	**64.7**	**88.8**	**50.0**	**79.7**	**37.9**	**69.0**	**28.1**	**37.0**	**58.2**	**73.1**	**121.6**	**123.8**

TABLE 2: Highest ranking published image captioning results on the online MSCOCO test server. Except for BLUE-1 which is of little interest, our single model optimized with CIDEr, outperforms previously published works using all the other metrics.

predictions in Figure 6. Take Figure 6a as an example, after we generated "a young boy", we first focus on the visual context, *i.e.*, the boy's hand which is holding something. Then we want to find visual regions that the boy is holding, so we focus on the toothbrush in the boy's mouth. Finally, focusing on both hand and toothbrush, we generated the exactly word "brushing". In Figure 6c and 6d, although the model generated the same words, the context of the two words are different. In Figure 6c, the context is "kite" for captioning the relation between the kite and sky, while in Figure 6d, the context is "people" for captioning the action of the people. By applying the CAVP model, we can generate those captions both successfully with deep understanding of image scenes.

Besides showing a single important word of the generated sequence, we also visualize the whole policy decision across the whole sentence generation in Figure 7. Take the first sentence as an example, we notice that our context-aware model can not only focus on some single objects such as "man", "skis", and "snow", but also the compositional word "standing", connecting "man standing in snow".



(a) brushing **(b) holding**

(c) flying **(d) flying**

Fig. 6: Qualitative examples where top matrix shows the output policy network action probabilities and the bottom image shows the decision with maximum probability for composition features. The red bounding boxes are the context regions and the blue bounding boxes are the current regions which concatenated with context regions.

4.4 Ablation Studies

We extensively investigated ablation structures and settings of the CAVP model to gain insights into how and why it works.

4.4.1 *Architecture*

We investigate multiple variants of the CAVP model.

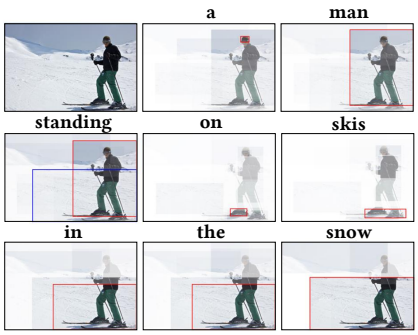
- **Up-Down [29]**: The CAVP degrades to the existing Up-Down model if we only use the single sub-policy network.
- **CAVP_scratch**: In CAVP, the context sub-policy network only takes the last visual feature as visual context and the sub-policy network is trained from scratch rather than using expert policy.
- **CAVP_cloning**: The context sub-policy network takes the last visual feature as visual context. The output sub-policy network is behavior cloned from expert policy.
- **CAVP_non-sharing**: The context sub-policy dose not share weights with the other three sub-policy networks.

Table 3 reports the performance comparison between the CAVP model and its variants on MS-COCO dataset. We

	Model	B@4	M	R	C	S
1	**Up-Down [29]**	37.5	27.7	57.9	121.9	21.0
2	**CAVP_scratch**	37.8	28.0	58.2	124.5	21.3
3	**CAVP_cloning**	38.3	27.8	58.0	124.6	21.4
4	**CAVP_non-sharing**	38.3	28.2	58.4	**126.4**	21.6
5	**CAVP**	**38.6**	**28.3**	**58.5**	126.3	**21.6**

TABLE 3: Ablation performance on MS-COCO. B@n is short for BLEU-n, M is short for METEOR, R is short for ROUGE, C is short for CIDEr, and S is short for SPICE.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2909864, IEEE Transactions on Pattern Analysis and Machine Intelligence



(a) a man standing on skis in the snow

(b) a man sitting in the grass flying a kite

Fig. 7: For each generated word, we visualized the attended image regions, outlining the region with the maximum policy probability in bounding box. The red bounding boxes are the visual context representation regions and the blue bounding boxes are the regions decided by single policy network.

Model	# of Parameters	Training	Testing
Up-Down [29]	77.6M	66	44.40
CAVP_non-sharing	108.5M	78	58.51
CAVP	83.0M	72	56.48

TABLE 4: Efficiency comparison in terms of parameter number, training time (hour) and testing time (ms/image). Experiments are performed on two Nvidia 1080Ti GPUs.

Training Metric	Evaluation Metric				
	BLEU4	ROUGE	METEOR	CIDEr	SPICE
BLEU	**38.8**	57.7	27.3	114.5	20.7
ROUGE	38.1	**59.1**	27.8	120.0	20.8
METEOR	33.6	57.6	**29.6**	113.0	22.8
CIDEr	38.3	58.4	28.2	**126.4**	21.6
SPIDEr	37.8	58.0	27.8	125.3	**23.1**

TABLE 5: Ablation performance on the MS-COCO "Karpathy" offline split with respect to various metrics as the reward.

can have the following observations: (a) The performance improvements of the other four models over Up-Down method [29] indicates the effectiveness of visual context for fine-grained image captioning. (b) The **CAVP_scratch** and **CAVP_cloning** obtain comparable performance. This shows that the off-the-shelf language parser is not very suitable to the visual-language task and the output sub-policy network can be learned from scratch without any expert policy guiding. (c) **CAVP** outperforms **CAVP_scratch** and **CAVP_cloning**. By memorizing historical visual context rather than only using the last visual feature, **CAVP** is able to generate more effective visual representations for subsequent sentence/paragraph generation.

Table 4 reports the parameter number, training and testing time costs. From the results, we can see that the **CAVP** model slightly increase the parameter number, training and testing computational overhead as compared to the existing up-down method [29]. Moreover, by sharing parameters among the four sub-policy network, **CAVP** has fewer parameters and lower computational cost than the model without parameter sharing.

4.4.2 Reward

For sequence training by policy gradient, the reward function $r(\cdot)$ can be any metrics, *e.g.* BLEU, ROUGE, METEOR, CIDEr and SPIDEr [18] (which combining the CIDEr and SPICE scores equally as the reward). Optimizing for different metrics leads to different performance. In general, as shown in Table 5, we found that optimizing for a specific metric results in the best performance on the same metric. And optimizing for CIDEr and SPIDEr gives the best overall performance, but the SPIDEr is more time consuming as the SPICE metric evaluation is very slow. Thus, we chose the CIDEr as the optimizing objective in most of our experiments.

5 Experiments on Paragraph Captioning

5.1 Experiment Settings

We conducted the experiments on the publicly available Stanford image-paragraph dataset collected by Krause *et al.* [27], which is divided into three subsets, including 14,575 images for training, 2,487 for validation and 2,489 for testing. Each image is annotated with one paragraph that contains an average of 5.7 sentences. where each sentence contains 11.9 words in average. For performance evaluation, we reported six widely used performance metrics: BLEU-{1,2,3,4}, METEOR, and CIDEr.

We performed the standard minimal textual pre-processing as in Section 4.2.1, leading to a vocabulary of 4,237 words. To generate a set of image region features, we followed the dense captioning [48] settings. In particular, we first resized each image so that so that its longest edge is 720 pixels and passed it through VGG-16 [49] network. Then, we extracted 50 region features in 4,096 dimensions. For policy network, we set LSTM size to 512, the number of hidden units in Eqn. (2) to 512, and embedding dimension to 512. We set $\lambda_w = 1.0$ and $\lambda_s = 5.0$ in Eqn. (24). During the training, we used Adam optimizer [41] with base learning rate of 5e-4 and shrank it by 0.8 every 20 epochs. We set the batch size to 64 images and trained up to 75 epochs for cross entropy loss and up to 150 epochs for RL loss. Besides, we set maximum number of sentence to 6 and maximum sentence length to 30 words.

	METEOR	CIDEr	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Sentence-Concat [27]	12.05	6.82	31.11	15.10	7.56	3.98
Template [27]	14.31	12.15	37.47	21.02	12.30	7.38
DenseCap-Concat [27]	12.66	12.51	33.18	16.92	8.54	4.54
Image-Flat [27]	12.82	11.06	34.04	19.95	12.20	7.71
Regions-Scratch [27]	13.54	11.14	37.30	21.70	13.07	8.07
Regions-Hierarchical [27]	15.95	13.52	41.90	24.11	14.23	8.69
RTT-GAN [32]	17.12	16.87	41.99	24.86	14.89	9.03
RTT-GAN* [32]	18.39	20.36	**42.06**	25.35	14.92	9.21
Hierarchical CAVP_CIDEr	16.79	20.94	41.38	25.40	14.93	9.00
Hierarchical CAVP_BLEU	16.83	**21.12**	42.01	**25.86**	**15.33**	**9.26**
Human	19.22	28.55	42.88	25.68	15.55	9.66

TABLE 6: Performance comparison on image paragraph captioning task. The proposed models outperform the state-of-the-art methods in terms of most metrics.

	METEOR	CIDEr	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Single Policy	16.29	16.36	40.11	22.31	12.39	6.86
Sentence CAVP	16.41	17.29	41.79	24.47	13.67	7.82
Hierarchical CAVP_XE	**17.14**	19.63	**42.49**	25.80	15.04	9.00
Hierarchical CAVP_CIDEr	16.79	20.94	41.38	25.40	14.93	9.00
Hierarchical CAVP_BLEU	16.83	**21.12**	42.01	**25.86**	**15.33**	**9.26**

TABLE 7: Ablation performance on image paragraph captioning task.

a stop sign is attached to a pole . there are a bunch of mountains and mountains shown from the side of the road . there is a small mountain in the distance .

a person is skiing in a snow covered area . the person is wearing a black helmet and goggles . there is loose snow all around them from him jumping . the person is holding two ski poles in their hands .

there are two zebras in the grass . they are in a very small area of grass . there is a large tree behind the zebra that is partially visible on the top of it . there is a tree trunk behind the zebra .

a man is surfing in the ocean . the surfboard is white . the man is wearing a black wet suit . there is a large wave in the water .

Fig. 8: Examples of image paragraph captioning results of our model. For each image, a paragraph description with a variable number of sentences is generated.

5.2 Comparison to State-of-the-Arts

We compared our hierarchical CAVP model to the following state-of-the-art methods. **Sentence-Concat** [27] combines five sentences sampled from a sentence captioning model trained on MS-COCO dataset. **Image-Flat** [27] directly treats a paragraph as a long sentence and applies a standard image captioning method [37]. **Template** [27] converts a structured representation of images into text via a pre-defined template. **DenseCap-Concat** [27] concatenates DenseCap [48] predictions to form a paragraph. **Region-Scratch** [27] uses a flat model, which initialized from scratch, to decode paragraph. **Region-Hierarchical** [27] uses a hierarchical structure contained a sentence RNN and a word RNN. **RTT-GAN** [32] is an recurrent topic-transition generative adversarial network coupled with an attention mechanism proposed recently. **RTT-GAN*** [32] is the version using additional training data. Moreover, we performed a **Human** evaluation by collecting an additional paragraph for 500 randomly chosen images.

Table 6 reports the performance comparison of image paragraph captioning on the Stanford image-paragraph dataset. We found that the proposed Hierarchical CAVP model optimized with either CIDEr or BLEU both outperforms the state-of-the-art methods in terms of most metrics. Note that even comparing with **RTT-GAN*** [32] which uses additional training data, the proposed model achieves better performance in terms of most metrics. Moreover, **Human** produce superior description to all the automatic methods, especially in CIDEr and METEOR which are more correlated with human judgment.

Figure 8 presents some examples generated by the proposed models. We can find that our models can generate successive sentences with a story line. For example, the paragraph for the image in the first row moves its attention from near to far. The successive sentences first focus on the nearest sign, then mountains at the side of road, and the farthest mountain in the background finally.

5.3 Ablation Studies

We conducted ablation experiments to compare the proposed model and its following variants. **Single Policy** and **Sentence CAVP** treat a paragraph as a long sentence. While **Single Policy** only uses the sentence-level single sub-policy network without any visual context, **Sentence CAVP** uses the sentence-level CAVP without hierarchical fortification. **Hierarchical CAVP_XE** is the proposed hierarchical CAVP trained by cross entropy loss.

Table 7 reports the performance comparison among the proposed modes and the variants. From the results, we can obtain the following observations. (a) The **Sentence CAVP** outperforms **Single Policy** in terms of all the metrics. This indicates that the context-aware visual policy network can generate better long sentences by exploiting visual context. (b) **Hierarchical CAVP_XE** performs better than **Sentence**

CAVP by using sentence-level and word-level visual policies augmented with visual context at both levels. (c) **Hierarchical CAVP_CIDEr** or **Hierarchical CAVP_BLEU** achieves performance improvements in terms of some metrics and causes performance degradation on the others as compared to **Hierarchical CAVP_XE**. The main reason is the lack of sufficient ground-truth paragraphs for model training. There is only one ground-truth paragraph for each image in the dataset. Given more ground-truth paragraphs, the models optimized by CIDEr or BLEU would be more superior over that by cross entropy, as shown in the evaluation of image sentence captioning, where each image has five ground-truth captions. (d) **Hierarchical CAVP_BLEU** performs better than **Hierarchical CAVP_CIDEr**. This indicates that BLEU is more stable than CIDEr when dealing with limited ground-truth and small dataset.

6 Conclusion

In this paper, we proposed a novel Context-Aware Visual Policy network (CAVP) for fine-grained image-to-language generation, including both image sentence captioning and image paragraph captioning. Superior to existing RL-based methods, the proposed CAVP based framework takes the advantage of visual context in compositional visual reasoning, which is beneficial for image captioning. Compared against traditional visual attention which only fixes a single image region at every step, CAVP can attend to complex visual compositions over time. To the best of our knowledge, CAVP is the first RL-based image captioning model which incorporates visual context into sequential visual reasoning. We conducted extensive experiments as well as ablation studies to investigate the effectiveness of CAVP. The experimental results have shown that the proposed approach can significantly boost the performances of the RL-based image captioning methods and achieves top ranking performances on MS-COCO server and Stanford image paragraph captioning dataset. We will continue our future works in two directions. First, we will integrate the visual policy and language policy into a Monte Carlo search strategy for image sentence/paragraph captioning. Second, we will also apply CAVP to other sequential decision-making tasks such as visual question answering and visual dialog.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) under Grants 61622211, 61620106009 and 61525206 as well as the Fundamental Research Funds for the Central Universities under Grant WK2100100030.

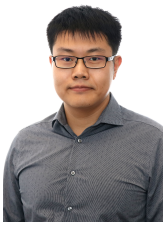
References

[1] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, 1997.
[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.
[3] Z. Ren, X. Wang, N. Zhang, X. Lv, and L.-J. Li, "Deep reinforcement learning-based image captioning with embedding reward," *NIPS Workshop*, 2017.
[4] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.
[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NIPS*, 2015.
[6] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *CVPR*, 2015.
[7] 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 *ICML*, 2015.
[8] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh, "Vqa: Visual question answering," in *ICCV*, 2015.
[9] Q. Zhang, Y. N. Wu, and S.-C. Zhu, "Interpretable convolutional neural networks," *CVPR*, 2018.
[10] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, and S. J. Gershman, "Building machines that learn and think like people," *Behavioral and Brain Sciences*, 2017.
[11] D. Geman, S. Geman, N. Hallonquist, and L. Younes, "Visual turing test for computer vision systems," *PNAS*, 2015.
[12] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua, "Visual translation embedding network for visual relation detection," in *CVPR*, 2017.
[13] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene graph generation by iterative message passing," in *CVPR*, 2017.
[14] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient." in *AAAI*, 2017.
[15] J. Johnson, B. Hariharan, L. van der Maaten, J. Hoffman, L. Fei-Fei, C. Lawrence Zitnick, and R. Girshick, "Inferring and executing programs for visual reasoning," in *ICCV*, 2017.
[16] A. Jabri, A. Joulin, and L. van der Maaten, "Revisiting visual question answering baselines," in *ECCV*, 2016.
[17] R. Hu, J. Andreas, M. Rohrbach, T. Darrell, and K. Saenko, "Learning to reason: End-to-end module networks for visual question answering," in *ICCV*, 2017.
[18] S. Liu, Z. Zhu, N. Ye, S. Guadarrama, and K. Murphy, "Improved image captioning via policy gradient optimization of spider," *ICCV*, 2017.
[19] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," *CVPR*, 2017.
[20] L. Zhang, F. Sung, F. Liu, T. Xiang, S. Gong, Y. Yang, and T. M. Hospedales, "Actor-critic sequence training for image captioning," *arXiv preprint arXiv:1706.09601*, 2017.
[21] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *CVPR*, 2015.
[22] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic propositional image caption evaluation," in *ECCV*, 2016.
[23] M. Ranzato, S. Chopra, M. Auli, and W. Zaremba, "Sequence level training with recurrent neural networks," *ICLR*, 2016.
[24] C. Stanfill and D. Waltz, "Toward memory-based reasoning," *Communications of the ACM*, 1986.
[25] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998.
[26] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.
[27] J. Krause, J. Johnson, R. Krishna, and L. Fei-Fei, "A hierarchical approach for generating descriptive image paragraphs," in *CVPR*, 2017.
[28] J. Lu, C. Xiong, D. Parikh, and R. Socher, "Knowing when to look: Adaptive attention via a visual sentinel for image captioning," in *CVPR*, 2017.
[29] 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 vqa," in *CVPR*, 2018.
[30] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua, "Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning," in *CVPR*, 2017.
[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.
[32] X. Liang, Z. Hu, H. Zhang, C. Gan, and E. P. Xing, "Recurrent topic-transition gan for visual paragraph generation," in *ICCV*, 2017.
[33] M. Chatterjee and A. G. Schwing, "Diverse and coherent paragraph generation from images," in *ECCV*, 2018.
[34] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, 1992.
[35] H. Zhang, Y. Niu, and S.-F. Chang, "Grounding referring expressions in images by variational context," in *CVPR*, 2018.

[36] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *NAACL*, 2003.
[37] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *CVPR*, 2015.
[38] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *ACL*, 2002.
[39] S. Banerjee and A. Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 2005.
[40] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *ACL Workshop*, 2004.
[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *ICLR*, 2015.
[42] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, 2015.
[43] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *IJCV*, 2017.
[44] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei, "Boosting image captioning with attributes," *ICCV*, 2017.
[45] J. Gu, J. Cai, G. Wang, and T. Chen, "Stack-captioning: Coarse-to-fine learning for image captioning," *AAAI*, 2018.
[46] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt *et al.*, "From captions to visual concepts and back," in *CVPR*, 2015.
[47] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (m-rnn)," *ICLR*, 2015.
[48] J. Johnson, A. Karpathy, and L. Fei-Fei, "Densecap: Fully convolutional localization networks for dense captioning," in *CVPR*, 2016.
[49] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.

Zheng-Jun Zha (M08) received the B.E. and Ph.D. degrees from the University of Science and Technology of China, Hefei, China, in 2004 and 2009, respectively. He is currently a Full Professor with the School of Information Science and Technology, University of Science and Technology of China, the Vice Director of National Engineering Laboratory for Brain-Inspired Intelligence Technology and Application. He was a Researcher with the Hefei Institutes of Physical Science, Chinese Academy of Sciences, from 2013 to 2015, a Senior Research Fellow with the School of Computing, National University of Singapore (NUS), from 2011 to 2013, and a Research Fellow there from 2009 to 2010. His research interests include multimedia analysis, retrieval and applications, as well as computer vision etc. He has authored or coauthored more than 100 papers in these areas with a series of publications on top journals and conferences. He was the recipient of multiple paper awards from prestigious multimedia conferences, including the Best Paper Award and Best Student Paper Award in ACM Multimedia, etc. He serves as an Associated Editor of IEEE Trans. on Circuits and Systems for Video Technology.

Daqing Liu received the B.E. degree in Automation from Chang'an University, Xi'an, China, in 2016, and currently working toward the Ph.D. degree from the Department of Automation, University of Science and Technology of China, Hefei, China. His research interests mainly include computer vision and multimedia.



Hanwang Zhang is currently an assistant professor at Nanyang Technological University, Singapore. He was a research scientist at the Department of Computer Science, Columbia University, USA. He has received the B.Eng (Hons.) degree in computer science from Zhejiang University, Hangzhou, China, in 2009, and the Ph.D. degree in computer science from the National University of Singapore in 2014. His research interest includes computer vision, multimedia, and social media. Dr. Zhang is the recipient of the Best Demo runner-up award in ACM MM 2012, the Best Student Paper award in ACM MM 2013, and the Best Paper Honorable Mention in ACM SIGIR 2016and TOMM best paper award 2018. He is also the winner of Best Ph.D. Thesis Award of School of Computing, National University of Singapore, 2014.

Yongdong Zhang (M08-SM13) received the Ph.D. degree in electronic engineering from Tianjin University, Tianjin, China, in 2002.He is currently a Professor with the School of Information Science and Technology, University of Science and Technology of China. His current research interests are in the fields of multimedia content analysis and understanding, multimedia content security, video encoding and streaming media technology. He has authored over 100 refereed journal and conference papers. He was a recipient of the Best Paper Awards in PCM 2013, ICIMCS 2013, and ICME 2010, the Best Paper Candidate in ICME 2011.He serves as an Associate Editor of IEEE Trans. on Multimedia and an Editorial Board Member of Multimedia Systems Journal.

Feng Wu (M'99-SM'06-F'13) received the B.S. degree in Electrical Engineering from Xidian University in 1992. He received the M.S. and Ph.D. degrees in Computer Science from Harbin Institute of Technology in 1996 and 1999, respectively. Now he is a professor in University of Science and Technology of China. Before that, he was a principle researcher and research manager with Microsoft Research Asia. His research interests include computational photography, image and video compression, media communication, and media analysis and synthesis. He has authored or co-authored over 200 high quality papers (including several dozens of IEEE Transaction papers and top conference papers in MOBICOM, SIGIR, CVPR and ACM MM). He has 77 granted US patents. Fifteen of his techniques have been adopted into international video coding standards. As a co-author, he received the best paper award from IEEE T-CSVT 2009, PCM 2008 and SPIE VCIP 2007. Wu has been a Fellow of IEEE. He serves as an associate editor for IEEE Transactions on Circuits and System for Video Technology, IEEE Transactions on Multimedia and several other International journals. He received the IEEE Circuits and Systems Society 2012 Best Associate Editor Award. He also served as the TPC chair for MMSP 2011, VCIP 2010 and PCM 2009, and the Special Sessions chair for ICME 2010 and ISCAS 2013.