

“© 2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

# Fine-grained and Semantic-guided Visual Attention for Image Captioning

Zongjian Zhang      Qiang Wu  
University of Technology Sydney  
Zongjian.Zhang@student.uts.edu.au,  
Qiang.Wu@uts.edu.au

Yang Wang      Fang Chen  
Data61-CSIRO  
{Yang.Wang, Fang.Chen}@data61.csiro.au

## Abstract

Soft-attention is regarded as one of the representative methods for image captioning. Based on the end-to-end CNN-LSTM framework, it tries to link the relevant visual information on the image with the semantic representation in the text (i.e. captioning) for the first time. In recent years, there are several state-of-the-art methods published, which are motivated by this approach and include more elegant fine-tune operation. However, due to the constraints of CNN architecture, the given image is only segmented to fixed-resolution grid at a coarse level. The overall visual feature created for each grid cell indiscriminately fuses all inside objects and/or their portions. There is no semantic link among grid cells, although an object may be segmented into different grid cells. In addition, the large-area stuff (e.g. sky and beach) cannot be represented in the current methods. To tackle the problems above, this paper proposes a new model based on the FCN-LSTM framework which can segment the input image into a fine-grained grid. Moreover, the visual feature representing each grid cell is contributed only by the principal object or its portion in the corresponding cell. By adopting the pixel-wise labels (i.e. semantic segmentation), the visual representations of different grid cells are correlated to each other. In this way, a mechanism of fine-grained and semantic-guided visual attention is created, which can better link the relevant visual information with each semantic meaning inside the text through LSTM. Without using the elegant fine-tune, the comprehensive experiments show promising performance consistently across different evaluation metrics.

## 1. Introduction

Image captioning is an important AI-complete task of scene understanding, which is an ultimate goal of artificial intelligence. Through the automatic generation of captions based on a comprehensive understanding of the real-world scene, it can benefit the human-machine interaction, autonomous/assistant driving, and intelligent navigation for

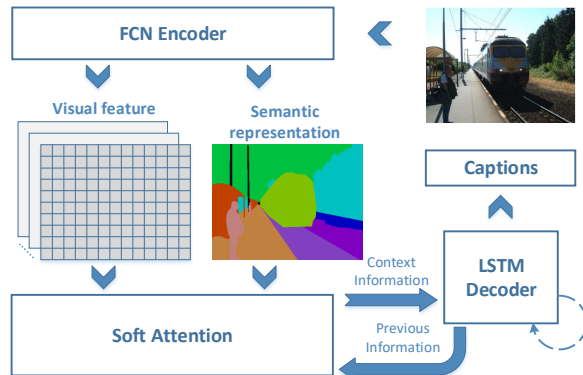


Figure 1. The overview of our proposed framework.

visually impaired people. Moreover, this task bridges together Computer Vision and Natural Language Processing [12]. An accurate description requires a comprehensive understanding of objects, stuff, and their mutual relations/interactions at all different image regions, which are then selectively attended to due to their semantic relations to each generated word. Such visual attention mechanism has attracted lots of research interests, leading to a large performance gain [3, 5, 14, 24, 25, 28].

Most state-of-the-art spatial visual attention models are based on the CNN-LSTM framework in an end-to-end trainable way [5, 14, 24]. CNN plays a role of image encoder, responsible for understanding visual information and encoding them into region-wise features at different grid locations. As a caption decoder, LSTM is responsible for understanding all words that have been generated, and generating the following word at each time step. As an agent between CNN and LSTM, the attention mechanism makes joint inferences and adaptively attends to those semantically related image regions by generating a distinct attention weight for each region. Based on this weight map, a visual context feature is firstly summarized through the weighted sum of all region features encoded by CNN, and then fed into the LSTM for language inference.

However, to the best of our knowledge, current soft-attention-based approaches only use CNN as the image

encoder to create the attention module. Their underlying CNN-LSTM framework has four limitations in providing an accurate attention mechanism. 1) Due to the constraints of current CNN architecture, the attention mechanism has a fixed low grid resolution (e.g. VGG [20] supports grid dimension 14x14 and ResNet [8] supports grid dimension 7x7) in the soft-attention framework. It is impossible to lift it to fine-grained level. 2) The representation of each grid cell is indiscriminately a mix of visual information of all objects and/or their portions inside this cell. So, it lacks the semantic correspondence related to the most salient visual cue within the grid cell. 3) Due to the lack of mutual reference information across grid cells, those different grid cells containing partial visual information from the same objects cannot be correlated to each other. The semantic visual guidance just does not exist across grid cells. 4) Because of object-oriented CNN encoder, existing soft-attention framework is not able to recognize and describe large-area stuff, like sky, beach, and grass. Hence, the context information cannot be well represented based on only object information.

In this paper, we propose a novel model based on the FCN-LSTM framework that augments the spatial visual attention, inspired by the Soft-Attention framework [24]. It leverages the spatially dense and semantically abundant outputs of Fully Convolutional Network (FCN) to solve above-mentioned limitations. FCN is particularly designed for Semantic Segmentation task to do dense pixel-level predictions [4, 19]. Therefore, it naturally excels in generating both visual features and semantic labels in the form of a spatial grid at a fine-grained level, which theoretically can reach up to the pixel level. Thus, 1) this enables our model to have a fine-grained visual attention at a high grid resolution, given the same-size image. It can attend to relevant object regions more accurately, and hence extract a more precise context feature with fewer noises. Moreover, the grid resolution of our attention module can be flexibly adjusted. 2) Based on pixel-level semantic labels, our model can represent each grid cell based on the dominating area which is associated with an object or its portion inside the cell. This saliency-related semantic correspondence can be kept when the resolution is adjusted. 3) Guided by the semantic labels of all grid cells, our model can grasp the semantic layout across grid cells, and efficiently associate the grid cells containing different portions of the same object. In this way, wrong inferences can be mitigated. 4) As the FCN encoder is both object-oriented and stuff-oriented, our model can extract a better representation of context information by attending to large-area stuff, such as sky and beach. So, the contextual inference is more comprehensive. 5) Semantic context feature can be also summarized from semantic labels to form the joint context feature with visual context feature. This joint context feature

can provide a stronger context information to LSTM decoder. Specifically, our FCN-LSTM model is designed with fine-grained and semantic-guided attention mechanism, and demonstrates state-of-the-art performances on MSCOCO dataset on metrics BLEU@N, METEOR, and CIDEr.

This paper is organized into five sections. The first section is the introduction, followed by the section of related works. In section three, the method of our model is described in detail. Section four specifies the experiment details. The last section is the conclusion.

## 2. Related Works

Most state-of-the-art models for image captioning are based on Deep Neural Networks [4, 8, 9, 19, 20]. The best one is the encoder-decoder neural framework [1, 3, 5, 6, 7, 15, 14, 16, 23, 24, 25, 26, 27, 28] inspired by the Machine Translation [1]. In this mainstream framework, CNN encoder is responsible for extracting image features at the highest semantic level, which are then fed into RNN decoder to generate the natural language caption in a sequential word-by-word way. Attention mechanism bridges CNN encoder and RNN decoder together efficiently, by enabling the RNN decoder to adaptively attend to, via a weight map, only those image features that are semantically related to the word to be generated at a certain time-step. So far, the attention mechanism has been researched in two manifolds. They all try to establish alignments between visual information and word information in a LSTM style. The major difference lies in the outputs of the encoder.

**Grid-wise visual feature without semantic label.** This type of attention focuses on which spatial regions to attend to [24, 14, 5]. The features of regions at different locations are extracted by the CNN-encoder from its last convolutional layer, and fed into the attention model for relativity inference. This kind of attention mechanism is generally integrated into an end-to-end trainable encoder-decoder framework, and trained implicitly without any explicit supervision. However, all these spatial attention models have a fixed low grid resolution, which is difficult to be converted to high grid resolution. Moreover, they are only object-oriented due to the use of CNN-encoder, and not able to recognize large area stuff as sky and beach. Another problem is that it has a lack of representation on the connections among the grid cells on the image.

**Attribute-based visual representation.** This kind of attention chooses which semantic concepts to be focused [28, 29]. The image feature is represented by a vector of confidences on all concepts, which is a mixture of objects, stuff, attributes, interactions, relations, etc. Although involving abundant semantic concepts, these semantic attention models suffer from lack of the significant spatial layout.

To the best of our knowledge, our FCN-LSTM model is the first work to propose a novel attention mechanism

that combines the grid-wise visual representation with grid-wise semantic label at a fine-grained resolution. Moreover, our model can grasp the semantic connections among all objects and stuff in the image.

### 3. Method

We first describe the overall FCN-LSTM framework for our captioning model in Sec 3.1, and then further introduce our fine-grained semantic-guided attention modules in Sec 3.2.

#### 3.1. FCN-LSTM Framework for Image Captioning

Similar to the mainstream CNN-LSTM framework, our novel FCN-LSTM framework is also a variant of Encoder-Decoder framework for image captioning. It can be regarded as a translation from vision to language. The FCN-encoder firstly extracts both visual representations and semantic labels from the input image at the pixel level, then the LSTM-decoder generates caption word-by-word based on joint understanding over these visual and semantic information. Given an image and its corresponding caption, the FCN-LSTM model directly maximizes the probability of word sequence:

$$\theta^* = \arg \max_{\theta} \sum_{(\mathbf{I}, \mathbf{y})} \log p(\mathbf{y} | \mathbf{I}; \theta) \quad (1)$$

where  $\theta$  are the parameters of the model,  $\mathbf{I}$  is the image, and  $\mathbf{y} = \{y_1, y_2, \dots, y_t\}$  is the word sequence of corresponding caption. Based on chain rule, the log likelihood of the joint probability distribution over  $\mathbf{y}$  is comprised of  $T$  conditional probabilities:

$$\log p(\mathbf{y}) = \sum_{t=1}^T \log p(y_t | y_{t-1}, \dots, y_1, \mathbf{I}) \quad (2)$$

where  $T$  is the total length of the caption. Here, the dependency on model parameters  $\theta$  is removed for convenience. During training phase,  $(\mathbf{I}, \mathbf{y})$  is a training image-caption pair, and the overall optimization objective is the sum of log probabilities over all training pairs in the training set. During testing phase, only image  $\mathbf{I}$  is fed into the model for caption generation. Specifically, our FCN-LSTM framework consists of three parts: FCN-encoder, LSTM-decoder, and soft-attention (Figure 1). It firstly uses FCN-encoder to extract both spatial visual features and semantic representations from image at pixel level. Then, the fine-grained and semantic-guided soft-attention summarizes all outputs of FCN-encoder into a joint context feature for LSTM-decoder to generate captions.

**FCN-encoder.** Particularly designed for the Semantic Segmentation task, Fully Convolutional Network (FCN) can directly perform pixel-wise classification. To encode

image, our framework employs the FCN to directly extract both visual feature and semantic label for each different pixel in the image. First of all, the  $N \times N$  size image  $\mathbf{I}$  can be represented by the spatial visual features:

$$V = FCN_v(\mathbf{I}) = \{v_1, v_2, \dots, v_k\} \quad (3)$$

where  $k = N^2$  is the number of image pixels. Each feature  $v_i \in R^d$  is a  $d$  dimensional representation corresponding to an image pixel. Specifically, the visual features are taken from the second last layer of FCN. This is similar to what CNN encoder does in the CNN-LSTM framework. Differently, the image  $\mathbf{I}$  also has a corresponding spatial semantic representations:

$$S = FCN_s(\mathbf{I}) = \{s_1, s_2, \dots, s_k\} \quad (4)$$

where  $s_i$  is a semantic label for each pixel indicating which object or stuff it may belong to. Note that the concatenation of 2-D image pixels into 1-D form does not break the spatial correspondence.

**LSTM-decoder.** As each conditional probability in Equation 2 can be naturally modeled based on Recurrent Neural Network (RNN), our model adopts the Long-Short Term Memory (LSTM) as the caption decoder. At time  $t$ , the previous conditional variable-length word sequence  $\{y_1, y_2, \dots, y_{t-1}\}$  and image  $\mathbf{I}$  are represented by a fixed-length hidden state  $h_t$  of LSTM as following:

$$x_t = W_e y_{t-1} \quad (5)$$

$$h_t = LSTM(x_t, h_{t-1}, c_t) \quad (6)$$

Here,  $y_{t-1}$  is the output word at time  $t - 1$ . As the current new input,  $x_t$  is word embedding of  $y_{t-1}$  based on embedding matrix  $W_e$ . Each word  $y_i$  is simply encoded as the one-hot vector.  $h_{t-1}$  is the hidden state representing the conditional word sequence  $\{y_1, y_2, \dots, y_{t-2}\}$  and image  $\mathbf{I}$ .  $c_t$  is the context feature extracted from image at time  $t$ , via the attention mechanism. This context feature represents the dynamic combination of visual and semantic information from image  $\mathbf{I}$ .

Specifically, the detailed definition of LSTM model is as follows:

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1} + W_{ic}c_t + b_i) \quad (7)$$

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1} + W_{fc}c_t + b_f) \quad (8)$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1} + W_{oc}c_t + b_o) \quad (9)$$

$$g_t = \tanh(W_{gx}x_t + W_{gh}h_{t-1} + W_{gc}c_t + b_g) \quad (10)$$

$$m_t = f_t \odot m_t + i_t \odot g_t \quad (11)$$

$$h_t = m_t \odot o_t \quad (12)$$

Here,  $i_t$ ,  $f_t$ ,  $o_t$ ,  $g_t$ ,  $m_t$ ,  $h_t$  are the input gate, forget gate, output gate, modulated input, memory, and hidden state of

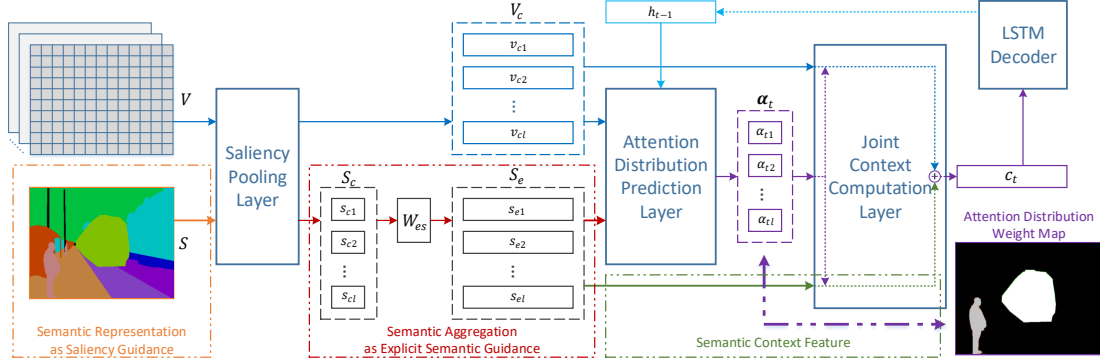


Figure 2. The detailed structure of our fine-grained and semantic-guided attention model.

the LSTM at time  $t$  respectively. Moreover, the operation  $\sigma$ ,  $\tanh$ ,  $\odot$  are sigmoid, hyper tangent, and element-wise multiplication respectively.

Finally, the probability of generating word  $y_t$  at time  $t$  is modeled based on input (previous word), hidden state, and context feature:

$$p(y_t|y_{t-1}, \dots, y_1, \mathbf{I}) = f(h_t, x_t, c_t) \quad (13)$$

$$= \text{softmax}(W \tanh(W_h h_t + W_c c_t + x_t + b_h) + b)$$

**Soft-attention.** Traditionally, the soft-attention mechanism [1] selectively attends to relevant regions in the image with reference to previously generated words, and generates an attention distribution in the form of a weight map over all regions. A higher attention weight indicates that the region has a higher relevance (or importance) to the generation of next word, and vice versa. Then, based on the attention distribution, the information of relevant regions is summarized together, and fed into the LSTM-decoder as the above-mentioned context feature  $c_t$ . Therefore, this attention mechanism serves as an agent between FCN-encoder and LSTM-decoder by sending needed information from the former to the latter. A better attention mechanism provides a more accurate context feature to the LSTM-decoder, which can then generate a more accurate word for a higher quality caption.

Our novel soft-attention mechanism is enhanced by both the fine-grained resolution and the semantic guidance on the basis of this novel FCN-LSTM framework. It can attend to relevant regions more precisely based on a high-resolution weight map. Moreover, the region-wise semantic guidance provides a global view of semantic relations among all regions. Both will make the context feature  $c_t$  more accurate. Details are further described in Section 3.2.

### 3.2. Fine-grained and Semantic-guided Attention

Illustrated by Figure 2, our attention mechanism comprises three layers: saliency pooling layer, attention distribution prediction layer, and joint context computation layer. It requires three inputs:  $V$  and  $S$  from FCN-encoder, and

$h_{t-1}$  from LSTM-decoder.  $V$  is the spatial visual features through which the attention model attends to relevant regions locally.  $S$  is the spatial semantic representations related to pixel-wise semantic labels. It serves as a guidance to provide the attention model a global view. The pixel-wise nature of  $V$  and  $S$  contributes to the fine-grained attention.  $h_{t-1}$  is the hidden state of LSTM at time  $t - 1$ , which contains previously generated words and their corresponding relevant image information. After the extraction of compact visual features  $V_c$  and semantic representations  $S_c$ , it predicts the attention distribution weight map  $\alpha_t$  over all regions. Then, the context feature  $c_t$  is computed by adding two weighted sums of visual features  $V_c$  and semantic features  $S_e$ . Correspondingly, the attention model is specifically defined as:

$$c_t = f_{att}(h_{t-1}, V, S) \quad (14)$$

which will be specifically described in the following five parts:

**Fine-grained Grid Attention.** Our fine-grained grid attention depends on pixel-wise nature of FCN-encoder, which is demanded by the Semantic Segmentation task. Therefore, the fine-grained grid resolution is determined by the resolution of FCN-encoder’s grid output, and hence can reach up to pixel level. Practically, most FCN-encoders can only reach a certain small-patch level, and each grid cell corresponds to a small patch ( $n \times n$  pixels) in the image. Due to this, all regions of relevant objects/stuff can be attended to with a high spatial accuracy, as smaller patch can distinguish the object/stuff boundary more precisely. Particularly, along the object boundary, the grid patch contains pixels of both this object and its neighbors (including other objects and stuff). Using smaller grid patch (i.e. fine-grained grid) can mitigate the noisy information created by neighbor objects and stuff. Hence, the context feature will be more accurate because of less irrelevant information.

**Saliency Pooling Layer.** Ideally, FCN may ultimately provide pixel-level labeling and pixel-level visual feature, which then may be fed into rest layers of the neural network

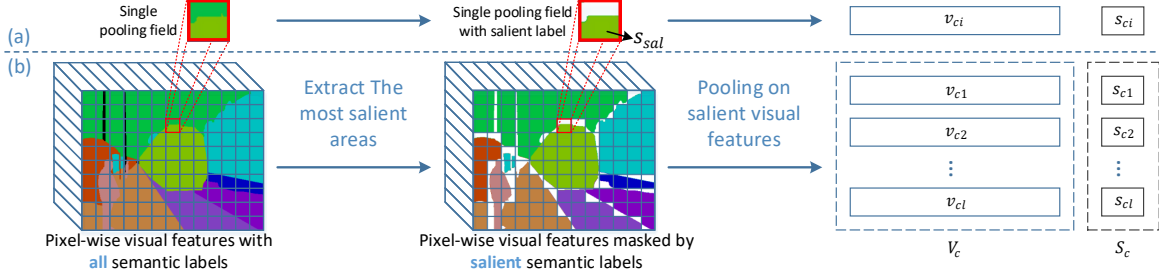


Figure 3. An illustration of saliency pooling layer for a single field (a) and the entire image (b).

for further processing. However, in practice, due to the constraint of computation power, the rest layers can process a limited number of patches although the patch can be of fine-grained size because of the nature of FCN. That is, the visual feature and semantic representation of pixels inside a patch of given size have to be pooled together. Normally, this process on visual feature can be carried out through a common average pooling which simply sums the visual features of all pixels inside patch equally. In this paper, we propose Saliency Pooling which only pools the visual features of salient pixels. The salient pixels are defined as those pixels whose pixel labels generated by FCN dominate inside the patch. The pooling process can be modeled as:

$$(V_c, S_c) = Pooling_{sp}(V, S) \quad (15)$$

$$V_c = \{v_{c1}, v_{c2}, \dots, v_{cl}\} \quad (16)$$

$$S_c = \{s_{c1}, s_{c2}, \dots, s_{cl}\} \quad (17)$$

Displayed in Figure 3, it pools visual features  $V$  of the original grid resolution to a compact visual features  $V_c$  at an acceptable lower level (i.e.  $M_c \times M_c$ ), under the guidance of semantic representations  $S$ .  $S_c$  is the compact semantic representation. Let  $s_{sal_i}$  denote the labels of pixels which dominate the area inside the patch  $i$ , where  $i = 1, 2, \dots, l$ . Then,  $s_{ci}$  in Equation 17 can be defined as:

$$s_{ci} = s_{sal_i} \quad (18)$$

Correspondingly, the number of grid locations is reduced to  $l = M_c^2$ . Each  $v_{ci}$  is a brief visual feature pooled from those original visual features inside the pooling field  $i$ . Saliency pooling layer generates the visual feature  $v_{ci}$  in Equation 16 based on salient pixels only. In the Equation 19 below,  $v_c$  is a generic representation of any patch  $v_{ci}$ , where  $i = 1, 2, \dots, l$ .

$$v_c = \frac{1}{w^2} \sum_{j=1}^{w^2} v_j \cdot f_{sal}(s_j) \quad (19)$$

$$f_{sal}(s_j) = \begin{cases} 1, & s_j = s_{sal} \\ 0, & s_j \neq s_{sal} \end{cases} \quad (20)$$

where  $j$  stands for the relevant location of each pixel inside the  $w \times w$  pooling field.  $w \times w$  is the size of patch where

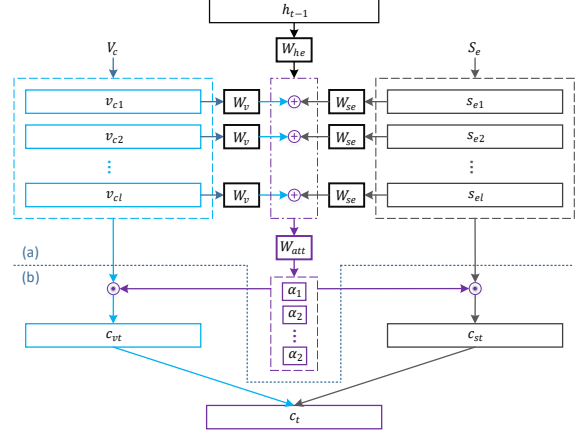


Figure 4. An illustration of attention distribution prediction layer (a) and joint context computation layer (b).

pooling processing is carried out.  $v_j$  is the visual feature of each pixel.  $w^2$  represents the number of pixels inside the pooling field.

From Equation 19, it may be seen that if  $f_{sal}(s_j)$  is enforced to be 1, saliency pooling is equivalent to the common average pooling. As seen in Figure 3, the output of saliency pooling layers are the salient visual features on the patches (i.e. pooling visual feature on salient pixels in the patch) and salient pixel labels of the patches.

**Attention Distribution Prediction Layer.** In Figure 2, the inputs of Attention Distribution Prediction Layer include visual feature pooling (i.e. the proposed saliency pooling or the simple common average pooling), explicit semantic guidance, and the hidden state  $h_{t-1}$  feedback from LSTM. In the existing CNN-LSTM framework [5, 14, 24], there is no such explicit semantic guidance.

Similar to the word embedding for LSTM-decoder, the compact semantic representations  $S_c$  are a map of semantic label words, which are encoded as the one-hot vector. So, they need to be embedded into dense semantic features via the embedding matrix  $W_{es}$ .

$$S_e = W_{es}S_c = \{s_{e1}, s_{e2}, \dots, s_{ek}\} \quad (21)$$

The attention prediction model is specifically designed as a two-layer perception. The first layer is mainly respon-

sible for feature fusion. From different feature spaces, the hidden state  $h_{t-1}$ , compact visual features  $V_c$ , and dense semantic features  $S_e$  are mapped into a shared feature space by embedding matrices  $W_{he}, W_v$ , and  $W_{se}$  respectively. As the hidden state  $h_{t-1}$  does not have the spatial dimension, an all-one vector  $\hat{\mathbf{1}}$  is used to extend its spatial dimension by simple copying. Then, these three embedded features are merged via the element-wise sum and fed into the hyperbolic tangent activation function. The overall process can be illustrated as in Figure 4. The fused feature  $z_t$  is fed into the second layer with a softmax function, to generate the attention weights over the  $l$  grid regions.

$$z_t = \tanh(W_{he}h_{t-1}\hat{\mathbf{1}} + W_vV_c + W_{se}S_e + b_z) \quad (22)$$

$$\alpha_t = \text{softmax}(W_{att}z_t + b_{att}) = \{\alpha_{t1}, \alpha_{t2}, \dots, \alpha_{tl}\} \quad (23)$$

where  $\alpha_{ti}$  represents the attention distribution for grid location  $i = 1, 2, \dots, l$  at time  $t$ .

Note that the semantic aggregation serves as the explicit guidance, as the semantic meanings of labels are fully used for guiding the prediction model. Specifically, the semantic labels are firstly embedded into dense semantic features, and then mapped into a shared feature space so as to guide the layer to predict the attention distribution explicitly.

**Joint Context Computation Layer.** Based on the attention weights, the visual context feature  $c_{vt}$  is computed as the weighted sum of compact visual features  $V_c$ , and the semantic context feature  $c_{st}$  is calculated as the weighted sum of dense semantic features  $S_e$ . See Figure 4 (b). Then, the joint context feature  $c_t$  is computed as the element-wise sum of  $c_{vt}$  and  $c_{st}$ , and fed into LSTM-decoder for word generation.

$$c_{vt} = \alpha_t \cdot V_c = \sum_{i=1}^l \alpha_{ti}v_{ci} \quad (24)$$

$$c_{st} = \alpha_t \cdot S_e = \sum_{i=1}^l \alpha_{ti}s_{ei} \quad (25)$$

$$c_t = c_{vt} + c_{st} \quad (26)$$

In Equation 26, without considering  $c_{st}$ ,  $c_t$  will become the aggregated visual feature based on attention distribution only. That is,  $c_t$  only presents the visual context instead of joint context. The initial states of LSTM-decoder are provided in supplementary material.

## 4. Experiment

Experiments are designed to demonstrate two advantages of our model: the fine-grained grid attention, and semantic guidance. The saliency guidance is used in saliency pooling layer. The explicit semantic guidance is adopted in attention distribution prediction layer. The joint context computation layer summarizes the context of semantic guidance for LSTM-decoder.

Table 1. Performances compared with the state-of-the-art models on MSCOCO test split via all metrics

Method	B@1	B@2	B@3	B@4	MTR	CIDEr
NIC v1 [22]	0.666	0.461	0.329	0.246	-	-
DeepVS [11]	0.625	0.450	0.321	0.230	0.195	0.660
emb-gLSTM [10]	0.670	0.491	0.358	0.264	0.227	-
m-RNN [17]	0.670	0.490	0.350	0.250	-	-
SCA-VGG-1layer [5]	-	-	-	0.281	0.235	0.847
Soft-Attention [24]	0.707	0.492	0.344	0.243	0.239	0.773
Hard-Attention [24]	<b>0.718</b>	0.504	0.357	0.250	0.230	-
Our model	0.712	<b>0.514</b>	<b>0.368</b>	<b>0.265</b>	<b>0.247</b>	<b>0.882</b>

### 4.1. Datasets and Metrics

Our experiments use two datasets. **MSCOCO** is the largest dataset for image captioning, with 82,783 training images, 40,504 validation images, and 40,775 testing images. For the offline evaluation, we use the same data split as [24, 28], containing 5000 images for validation and test respectively. **COCO-Stuff** is a more semantic-complete dataset for Semantic Segmentation. In total, it has 10,000 images sampled from MSCOCO training images, and annotations for 80 objects, 91 stuff, and 1 unknown background. Our DeepLab encoder is pre-trained on MSCOCO 80-object dataset and then finetuned on this COCO-stuff dataset. We use BLEU (B@1, B@2, B@3, B@4) [18], METEOR [2], and CIDEr [21] as evaluation metrics. Their scores are calculated via the COCO captioning evaluation tool [13].

### 4.2. Settings

**FCN-encoder.** A elegantly designed DeepLab [4], designed based on VGG-16 [20], is used as the FCN-encoder. The spatial visual features are extracted as the mean of four sets of spatial visual features with different Field-Of-View(FOV) from the outputs of the second last layer. Its dimension is  $81 \times 81, 1024d$ . The spatial semantic representations are extracted from the outputs of the final layer, which has dimension of  $81 \times 81, 1d$ .

**LSTM-encoder.** A single-layer LSTM with hidden size of 1024 is used in our model. The dimension of word embedding is 1024.

**Attention model.** The output size of Saliency Pooling Layer is set as  $14 \times 14, 1024d$  and  $27 \times 27, 1024d$  respectively.  $14 \times 14$  is selected to make comparisons with the Soft-Attention model [24].  $27 \times 27$  is the highest resolution we can achieve to demonstrate the improvements of fine-grained attention.





















### 4.3. Quantative Analysis.

The proposed method is motivated by soft-attention [24] which is based on the spatial visual attention idea. Thus, it is necessary to compare the performance of the proposed method against this method. In our implementation, we do not carry out fine-tuning by re-training the visual encoder

Table 2. Performances of our ablated models on MSCOCO test split on all metrics

Attention Resolution	Semantic Guidance	B@1	B@2	B@3	B@4	METEOR	CIDEr
14 x 14 (Soft-Att)	Case 1 - Average Pooling	0.707	0.492	0.344	0.243	0.239	0.773
	Case 2 - Saliency Pooling	0.703	0.501	0.354	0.251	0.240	0.827
	Case 3 - Explicit Guidance	0.705	0.504	0.358	0.255	0.241	0.846
	Case 4 - Joint Context Feature	0.708	0.507	0.360	0.257	0.242	0.846
27 x 27	Case 1 - Average Pooling	0.707	0.505	0.357	0.254	0.241	0.837
	Case 2 - Saliency Pooling	0.708	0.507	0.359	0.256	0.241	0.839
	Case 3 - Explicit Guidance	0.709	0.508	0.361	0.258	0.242	0.844
	Case 4 - Joint Context Feature	<b>0.712</b>	<b>0.514</b>	<b>0.368</b>	<b>0.265</b>	<b>0.247</b>	<b>0.882</b>

Table 3. Qualitative Analysis on the Advantages Provided by Higher Attention Resolution. In the illustration, each image provides two attention maps corresponding to two most meaningful nouns in the captions under two different attention resolutions. Attention maps with blue/red boundaries correspond to the words highlighted by blue/red respectively.

Image 1	14 × 14 Attention	27 × 27 Attention	Image 2	14 × 14 Attention	27 × 27 Attention
					
					
	Caption (14 × 14): A boy is playing <b>baseball</b> on a field. Caption (27 × 27): A young <b>boy</b> is playing <b>soccer</b> on a field.			Caption (14 × 14): A baseball <b>player</b> swinging a <b>bat</b> at a ball. Caption (27 × 27): A baseball <b>player</b> is swinging his <b>bat</b> .	
Image 3	14 × 14 Attention	27 × 27 Attention	Image 4	14 × 14 Attention	27 × 27 Attention
					
					
	Caption (14 × 14): A fire <b>hydrant</b> on the side of the <b>street</b> . Caption (27 × 27): A yellow fire <b>hydrant</b> sitting on the side of a <b>street</b> .			Caption (14 × 14): A <b>man</b> riding a wave on top of a <b>surfboard</b> . Caption (27 × 27): A <b>man</b> riding a <b>surfboard</b> on a wave in the ocean.	

on the large captioning dataset like Adaptive-Attention [14], MSM [27] and ATT-FCN [28] do. To maintain a fair comparison in order to show the performance boosted by fine-grain and semantic-guided attention, this paper only compares with those approaches using a VGG-based encoder like our method, such as DeepVS [11], NIC v1 [22], emb-gLSTM [10], m-RNN [17], SCA-VGG-1layer [5], and hard-attention [24].


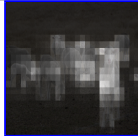
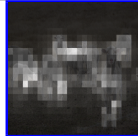


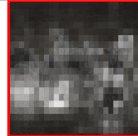






Table 1 shows results on the test split of MSCOCO dataset. Our model significantly outperforms all chosen state-of-the-art models over all metrics. Particularly, when compared with the base-line model Soft-Attention [24], our model improves the CIDEr score from 0.773 to 0.882, and the METEOR score from 0.239 to 0.247, and the B@4 from 0.243 to 0.265.

To better demonstrate the performance improvements contributed by the key components in our framework, the

intermediate results along with framework are shown in Table 2. In this table, two fine-grained attention resolutions are adopted. For semantic guidance, there are four different cases according to framework design in Figure 2. **Case 1** - Common average pooling scheme is adopted in pooling layer. Then, the aggregated visual features are fed into the rest layers without considering semantic information at all. This is the base-line scheme of the proposed framework. **Case 2** - Instead of common average pooling, the framework adopts saliency pooling in the pooling layer. However, the semantic aggregation for explicit semantic guidance is not fed into rest layers of the framework. **Case 3** - Saliency pooling results along with explicit semantic guidance are both fed into attention distribution prediction layer. However, explicit semantic guidance is not fed into the last context computation. **Case 4** - All components are fully adopted as illustrated in Figure 2. The Soft-Attention [24]



Table 4. Qualitative Analysis on the Advantages Provided by Semantic Guidance. The analysis is carried out on  $27 \times 27$  attention. The attention maps of different color boundaries (i.e. blue, green and red) correspond to the different words (highlighted by blue, green or red) in the captions. The words by red color are not discovered without semantic guidance, instead, which are captured by the proposed methods by using semantic guidance.

Original Image	More Precise Attention Provided by Semantic Guidance				New Meaningful Words Discovered by Semantic Guidance
	No Semantic Guidance	Semantic Guidance	No Semantic Guidance	Semantic Guidance	
					
Image 1	Caption (No semantic guidance): A herd of <b>sheep</b> standing on top of a <b>grass</b> covered field. Caption (Semantic guidance): A herd of <b>sheep</b> <b>gazing</b> on a dry <b>grass</b> field.				
					
Image 2	Caption (No semantic guidance): A <b>man</b> in a suit and tie holding a <b>umbrella</b> . Caption (Semantic guidance): A <b>woman</b> walking down a <b>street</b> holding an <b>umbrella</b> .				

model belongs to Case 1 at the  $14 \times 14$  resolution.

By comparing **Case 1** under  $14 \times 14$  and  $27 \times 27$  attention resolutions, merely increasing the attention resolution significantly improves the performance (e.g. 0.064 in CIDEr and 0.011 in B@4). Moreover, **Case 1** under  $27 \times 27$  already outperforms most methods in Table 1. For both attention resolutions, integrating saliency pooling, explicit guidance, and joint context feature one by one can all lead to better performances. Although very modest, the improvements of adding each component are steady and consistent, and can be demonstrated as sense-making by below insightful qualitative analysis. Furthermore, the accumulated improvement of all semantic guidances (**Case 4** v.s. **Case 1** under  $27 \times 27$ ) is also significant. The CIDEr is boosted by 0.045, B@4 by 0.011, and METEOR by 0.006. Our best model under  $27 \times 27$  resolution and with full semantic guidance (**Case 4**) has the best performance, and can beat nearly all models in Table 1.

#### 4.4. Qualitative Analysis

We further visualize the improved attention maps and captions by increasing the attention resolution from  $14 \times 14$  to  $27 \times 27$ . Table 3 shows that attention with higher resolution can capture related regions more accurately. In image 2, the  $27 \times 27$  attention model can attend to the ‘bat’ regions accurately, whereas the  $14 \times 14$  attention model attends to wrong regions. In image 1, the  $14 \times 14$  attention model generates the wrong word ‘baseball’ due to inaccurate attention. All blue-color words, such as ‘boy’, ‘player’, ‘hydrant’ and ‘man’, have more accurate attention maps under  $27 \times 27$  resolution. Moreover, stuff region like ‘street’ in image can also be correctly located. In the meantime, it is noticed that the overall quality of captions under higher

attention resolution is improved, which is more meaningful. More experimental results of visualizations are provided in supplementary material.

We also analyze the improvements on semantic comprehension that are brought by the semantic guidance in Table 4. Obviously, the semantic guidance helps the model attend to large-area objects/stuffs, such as ‘grass’ in image 1. Besides the improvement on the completeness and/or correctness of the attention maps, the semantic guidance can also discover new meanings to make the caption more meaningful. For image 2, ‘street’ is not captured without using semantic guidance. After introducing semantic guidance in the proposed method, they are exposed in the new captions. In image 1, the word ‘gazing’ is more precise than ‘standing’, and attentions have correctly focused onto those regions where sheep are eating grass. Therefore, adding semantic guidance can greatly increase the caption quality. More experimental results of visualizations are provided in supplementary material.

#### 5. Conclusion

In this paper, we propose a fine-grained and semantic-guided attention mechanism over a novel end-to-end FCN-LSTM framework for image captioning for the first time. Our model achieves state-of-the-art performances on MSCOCO dataset, compared with models having a VGG-based encoder. Moreover, our model is more of a framework that can be easily adapted to all Soft-Attention-based approaches. The results show that our model has a huge potential for a comprehensive attention on the abstract visual relation. Moreover, our framework can have a broad application in other tasks, like Image QA.

## References

- [1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2014.
- [2] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *ACL*, 2005.
- [3] K. Chen, J. Wang, L.-C. Chen, H. Gao, W. Xu, and R. Nevatia. Abc-cnn: An attention based convolutional neural network for visual question answering. In *CVPR*, 2016.
- [4] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. In *PAMI*, 2017.
- [5] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T. Chua. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In *CVPR*, 2017.
- [6] X. Chen and C. L. Zitnick. Minds eye: A recurrent visual representation for image caption generation. In *CVPR*, 2015.
- [7] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollar, J. Gao, X. He, M. Mitchell, J. C. Platt, and et al. From captions to visual concepts and back. In *CVPR*, 2015.
- [8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.
- [9] S. Hochreiter and J. Schmidhuber. Long short-term memory. In *Neural Computation*, 1997.
- [10] X. Jia, E. Gavves, B. Fernando, and T. Tuytelaars. Guiding the longshort term memory model for image caption generation. In *ICCV*, 2015.
- [11] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.
- [12] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, and et al. Visual genome: Connecting language and vision using crowd-sourced dense image annotations. In *IJCV*, 2016.
- [13] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.
- [14] 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.
- [15] J. Lu, J. Yang, D. Batra, , and D. Parikh. Hierarchical question-image co-attention for visual question answering. In *NIPS*, 2016.
- [16] M. Malinowski, M. Rohrbach, and M. Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *ICCV*, 2015.
- [17] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). In *ICLR*, 2015.
- [18] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002.
- [19] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *PAMI*, 39(4):640–651, 2017.
- [20] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.
- [21] R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015.
- [22] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015.
- [23] Q. Wu, C. Shen, L. Liu, A. Dick, and A. v. d. Hengel. What value do explicit high level concepts have in vision to language problems? In *CVPR*, 2016.
- [24] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015.
- [25] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked attention networks for image question answering. In *CVPR*, 2016.
- [26] Z. Yang, Y. Yuan, Y. Wu, R. Salakhutdinov, and W. W. Cohen. Encode, review, and decode: Reviewer module for caption generation. In *NIPS*, 2016.
- [27] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei. Boosting image captioning with attributes. In *arXiv preprint arXiv:1611.01646*, 2016.
- [28] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo. Image captioning with semantic attention. In *CVPR*, 2016.
- [29] L. Zhou, C. Xu, P. Koch, and J. J. Corso. Watch what you just said: Image captioning with text-conditional attention. In *arXiv preprint arXiv:1606.04621*, 2016.