DIFNet: Boosting Visual Information Flow for Image Captioning

Mingrui Wu[1*], Xuying Zhang[1*], Xiaoshuai Sun[124†], Yiyi Zhou[1],
Chao Chen[3], Jiaxin Gu[3], Xing Sun[3], Rongrong Ji[124]
[1]Media Analytics and Computing Lab, School of Informatics, Xiamen University, 361005, China.
[2]Institute of Artificial Intelligence, Xiamen University. [3]Youtu Lab, Tencent.
[4]Fujian Engineering Research Center of Trusted Artificial Intelligence Analysis and Application,
Xiamen University, 361005, China.
mingrui0001@gmail.com, zhangxuying@stu.xmu.edu.cn, xssun@xmu.edu.cn, zhouyiyi@xmu.edu.cn,
{aaroncccchen, jiaxingu}@tencent.com, winfred.sun@gmail.com, rrji@xmu.edu.cn

Abstract

Current Image Captioning (IC) methods predict textual words sequentially based on the input visual information from the visual feature extractor and the partially generated sentence information. However, for most cases, the partially generated sentence may dominate the target word prediction due to the insufficiency of visual information, making the generated descriptions irrelevant to the content of the given image. In this paper, we propose a Dual Information Flow Network *(**DIFNet**[1]) to address this issue, which takes segmentation feature as another visual information source to enhance the contribution of visual information for prediction. To maximize the use of two information flows, we also propose an effective feature fusion module termed Iterative Independent Layer Normalization (**IILN**) which can condense the most relevant inputs while retraining modality-specific information in each flow. Experiments show that our method is able to enhance the dependence of prediction on visual information, making word prediction more focused on the visual content, and thus achieves new state-of-the-art performance on the MSCOCO dataset,* e.g.*, 136.2 CIDEr on COCO Karpathy test split.*

1. Introduction

Image captioning is a task of generating a description in natural language based on a given image. It needs a model to understand the given image from multiple aspects, including identifying objects, actions, as well as relationships, and generate a language description for that image.

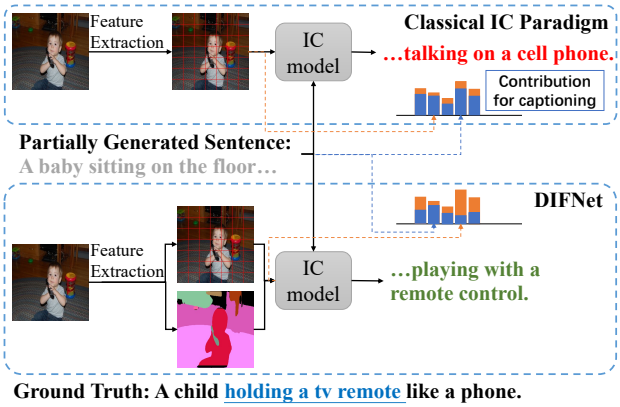

Figure 1. Comparison between the popular captioning paradigm (top) and the proposed *Dual Information FLow Network* (DIFNet). (bottom). Compared with existing methods, DIFNet introduces the visual representation of the dual information flow to facilitate reliable and accurate image understanding.

Inspired by the development of neural machine translation, the encoder-decoder framework has been widely used in image captioning tasks. The encoder takes a set of visual features (such as grid feature [10]) extracted by an offline CNN-based network as input and further encodes them into visual-language space. Then the decoder uses the visual information provided by the encoder and the partially generated caption to predict the next word. Most existing approaches [5, 9, 22] follow this paradigm to build their captioning networks, , as shown in Fig.1 (top). However, they suffer from a main drawback: the visual information from the visual feature extractor is insufficient and sometimes inaccurate. Although the research of feature extractors has made great progress [15, 25], key visual information, such as action and depth information, may still be ignored, even using the powerful visual-language pre-trained models [8]. The above drawback leads to an insufficient visual infor-

*Equal Contribution
†Corresponding Author
[1]Source code is available at: https://github.com/mrwu-mac/DIFNet

mation flow for the decoder, forcing the decoder to rely excessively on partially generated captions to predict the rest words in order to ensure the fluency of the generated description. This issue ultimately makes the generated descriptions irrelevant to the actual visual content, as shown in Fig. 1 (top), the baseline model generates incorrect phrase "talking on a cell phone" because the 'remote control' feature is hard to be captured by only grid feature.

To overcome these shortcomings, recent works [15, 19, 31, 37] introduce high-level visual cues, such as concepts, to supplement visual information. However, due to semantic inconsistency [17] and spatial misalignment, an additional fusion module is required to align these cues with visual features, which is inefficient and difficult to be combined with IC models with grid features. In contrast, this paper considers a new type of cues, *i.e.* the segmentation map, where region semantics are naturally aligned with grid features. As shown in Fig. 1 (bottom), segmentation map can be regarded as spatial semantic guidance and provide a coarse-grained context for grid features to facilitate image understanding. On the one hand, its pixel-level category information helps correct categories that are misjudged due to unreliable information in the grid features. On the other hand, its spatial information also helps to infer the underlying semantic and spatial relationships.

Motivated by this, we propose a *Dual Information Flow Network* (DIFNet), which takes the segmentation feature as another visual information source to supplement grid features, thereby enhancing the contribution of visual information for reliable prediction. Since it is easy to integrate grid features and segmentation features, we only need a simple fusion method. To maximize the benefit of two visual information flows, we propose an effective feature fusion module named *Iterative Independent Layer Normalization* (IILN), which can condense the most relevant inputs by a common LN layer while retraining modality-specific information in each flow via private LN layer. Note that certain visual information that is difficult to be captured might be directly filtered out by the attention layer, we adopt additional skip connections to further enhance the flow of information within and between the encoder and decoder.

We evaluate our method on the MSCOCO benchmark for image captioning, where the effectiveness of our proposals is well validated. In particular, our proposed model achieves the new state-of-the-art performance of MSCOCO. DIFNet achieves 136.2 CIDEr score on the COCO Karpathy test split under the setting of single-model. To gain more insights, we apply Layerwise Relevance Propagation (LRP) [4] to estimate how the visual information and partially caption contexts contribute to prediction, whose results demonstrate that our proposed model can enhance the contribution of visual information for prediction.

Our contributions are:

- We propose a Dual Information Flow Network (DIFNet), which takes the segmentation feature as an additional visual information source. DIFNet can enhance the contribution of visual content for prediction.
- We propose a feature fusion module termed Iterative Independent Layer Normalization (IILN), which can condense the most relevant inputs by a common LN layer while retraining modality-specific information in each flow via the private LN layer.
- Experiments show that our method can enhance the dependence of prediction on visual information and achieve significant performance improvements over the state-of-the-art on MSCOCO benchmark.

2. Related Work

Image Captioning. The encoder-decoder framework has been widely adopted by image captioning models [2, 5, 9]. However, most previous methods follow a single-stream pipeline and design architecture typically by increasing model complexity. Recent works [15, 31, 37] introduced concepts, attributes, and tags to enhance visual semantics, but they are hard to be aligned with the visual features [17]. Instead of using concepts, attributes, and tags, we use the segmentation feature as the second information stream to enhance the visual representation.

Panoptic Segmentation. Panoptic segmentation task [13, 33] unifies the instance segmentation task and the semantic segmentation task. It can identify the semantic class of a pixel while providing instance boundaries for classes like 'person' in a given image. To take advantage of segmentation cues, HIP [36] constructs a hierarchy parsing architecture to associate the instance-level, region-level, and image-level features for image captioning. Different from the HIP, we use a segmentation map to construct structured visual semantic representation, which retains the spatial structure information of the original image and is easier to be fused with the grid feature.

Multimodal Fusion. A lot of works have been done towards multimodal fusion [17, 20, 21, 30]. Early methods used simple aggregation operations (*e.g.* concatenation [21]) to combine multimodal sub-networks. Recent methods use a cross-modal attention mechanism [17] to align data from different modalities while still retaining the sub-networks of all modalities. In order to reduce the burden of computing power brought about by maintaining multiple subnets, some works [30] use sharing parameters and privatizes the normalization layer to maintain specific patterns. In relation to these works, we design an Iterative Independent Layer Normalization module for multimodal fusion and interaction.

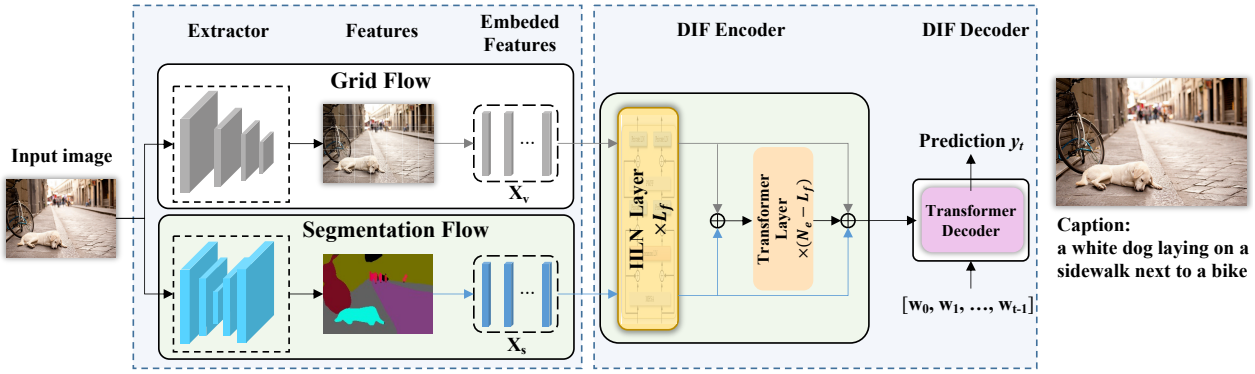

Figure 2. The overview of our DIFNet architecture. The grid features and segmentation features are first extracted along the Grid Flow and Segmentation Flow. Next, segmentation features and grid features are fused together by our proposed IILN module to enrich the information for visual inference. Besides, extra skip connections are explored to further enhance the information flow within and between the encoder and decoder.

3. Preliminaries

We first provide the image captioning problem definition. Given an image I, which can be described by a sentence A, where $A = \{w_1, w_2, \ldots, w_L\}$ consisting of L words. Let V denote grid visual features [10] which extracted from image I by an offline visual feature extractor, where $V = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N\}$ consisting of N grids and $\mathbf{v}_i \in \mathbb{R}^{D_v}$. Similar to the most of the existing captioning system [5, 9], our work is based on encoder-decoder Transformer [27] which encodes the grid features V into a sequence of continuous representations Z and then decodes it paired with the previously generated words to generate the output y_t. The model produces one word in the sentence at each time step in an auto-regressive manner [6]. This standard paradigm can be formulated as:

$$y_t = \mathcal{F}_l\left(E_v\left(V\right), w_0, w_1, w_2, \ldots, w_{t-1}\right), \tag{1}$$

where E_v is vision encoder and $\mathcal{F}_l$ is language decoder, w_0 is a start symbol.

3.1. The Transformer Architecture

The Transformer is a sequence transduction model. To process 2D inputs, we need to convert them into a series of 1D tokens, as follows:

$$U' = Flatten(Pool(U)), \tag{2}$$

where U is vision feature(original grid feature $O \in \mathbb{R}^{H \times W \times D_v}$ or segmentation feature S(will be discussed in Sec. 4.1)), $U' = \{\mathbf{u}'_1, \mathbf{u}'_2, \ldots, \mathbf{u}'_N\}$ is input vision feature sequence(such as V), $Pool$ is the AdaptiveAvgPool2d which output size is $H' \times W'$. Then we use a linear projection mapping each token to $\mathbb{R}^{d_{model}}$, as follows,

$$X = LN(\sigma(W_1 U' + b_1))), \tag{3}$$

where $\sigma(\cdot)$ is ReLU activation function, LN is Layer Normalization [3], $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ consisting of $N(N = H' \times W')$ tokens.

Then a encoder consisting of a stack of N_e transformer layers is used to map X into Z. Each transformer layer has two sub-layers, Multi-Head Self-Attention (MHSA) and Position-Wise Feed-Forward (PWFF) networks [27], each of two sub-layers around a residual connection [7] and layer normalization. We denote a transformer layer, $Z^{l+1} = Transformer(Z^l)$ as

$$\begin{aligned} M &= LN(MHSA(Z^l) + Z^l), \\ Z^{l+1} &= LN(PWFF(M) + M), \end{aligned} \tag{4}$$

where LN is Layer Normalization.

The decoder is composed of a sequence of N_d transformer layers and each layer inserts a third sub-layer in the middle of MHSA and PWFF, which takes the output of the encoder and the output of the MHSA as input, more details refer to Transformer [27].

4. Method

In this section, we describe our proposed DIFNet, which uses segmentation features and additional skip connections to enhance visual information flow. Figure 2 gives the overview of the DIFNet. We begin by describing the introduction of segmentation features(Sec. 4.1). Then we investigate the VSA fusion method and describe our IILN fusion method(Sec. 4.2) to fuse segmentation features with grid features. Next, we discuss the use of extra skip connection for visual information flow enhancement(Sec. 4.3). The training details are presented in Sec. 4.4.

4.1. Segmentation Feature

Panoptic segmentation map contains the semantic category information of each pixel and discriminative instance

information. As a result, panoptic segmentation map can be regarded as a high-level visual semantic cue and provides a coarse-grained context. To simply and effectively fit the grid features, we only extract the semantic segmentation map instead of the panoptic segmentation map from the semantic segmentation head of a panoptic segmentation network and then convert them into semantic feature vector S, where $S \in \mathbb{R}^{H\times W\times C}$, C, H and W are the class number, height, and width respectively. Each dimension of the semantic feature vector S is a bit-map which denotes a semantic class. After additionally integrating the segmentation feature, our paradigm can be formulated as:

$$y_t = \mathcal{F}_l\left(E_v\left(V, S\right), w_0, w_1, w_2, \ldots, w_{t-1}\right), \quad (5)$$

where E_v is vision encoder and $\mathcal{F}_l$ is language decoder.

4.2. Fusing Grids with Segmentations

In this section, we show how to integrate the two input representations in Transformer. We first investigate a fusion strategy VSA, and then present our fusion method IILN.

Fusion via Vanilla Self-Attention. We begin by discussing a Vanilla Self-Attention (VSA) [20] fusion method, which simply uses the transformer layers for encoding and fusing the two input sequences. Given the grid input sequence $X_v \in \mathbb{R}^{N\times d_{model}}$ and the segmentation input sequence $X_s \in \mathbb{R}^{N\times d_{model}}$, we first encode them with the transformer layers respectively, which allows each representation to have its own parameters. Then an element-wise sum is applied to integrate them into a single sequence $Z_{vs} \in \mathbb{R}^{N\times d_{model}}$. In order to exchange information and form a common representation, the transformer layers are used to encode Z_{vs}. This can be formulated as follows,

$$\begin{cases} Z_v^{l+1} = Transformer(Z_v^l), & \\ Z_s^{l+1} = Transformer(Z_s^l), & \text{if } l < L_f; \\ Z_{vs}^{l+1} = Transformer(Z_v^l + Z_s^l), & \text{if } l == L_f; \\ Z_{vs}^{l+1} = Transformer(Z_{vs}^l), & \text{otherwise.} \end{cases} \quad (6)$$

where l is from 0 to N_e, Z_v^0 and Z_s^0 are X_v and X_s, respectively. L_f denotes the layer after which to aggregate the two sequences, L_f=0 corresponds to 'early fusion', $0 < L_f < N_e$ corresponds to 'mid fusion' and $L_f = N_e$ corresponds to 'late fusion'. When $L_f = N_e$, there is no Transformer layer after the aggregation operation. Generally, the 'early fusion' cannot retain the specific patterns of the two representations, the 'late fusion' cannot effectively exchange information between the two representations, the 'late fusion' and the 'mid fusion' are both introduce a large number of parameters.

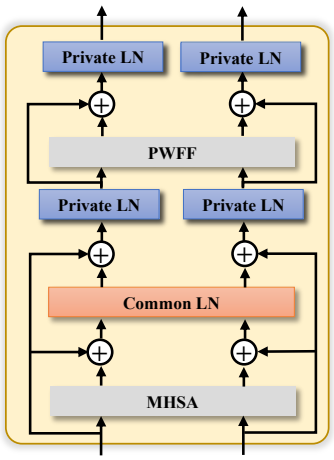

Figure 3. The Iterative Independent LN (IILN) module.

Fusion via Iterative Independent LN. We propose Iterative Independent Layer Normalization (IILN) to overcome the aforementioned problems. The transformer encoder layers are equipped with IILN when $l < L_f$, as shown in Figure 2 encoder part. The schema of IILN is shown in Figure 3. Inspired by [30], we first share the parameters of the MHSA layer and the PWFF layer to avoid the increase of network parameters. And we adopt a common LN layer to obtain a single distribution which includes the common information crossing two representations,

$$\begin{aligned} M_v &= LN(MHSA(Z_v^l;\theta_{vs}) + Z_v^l;\alpha_{vs},\beta_{vs}), \\ M_s &= LN(MHSA(Z_s^l;\theta_{vs}) + Z_s^l;\alpha_{vs},\beta_{vs}), \end{aligned} \quad (7)$$

where θ is model parameter of MHSA and PWFF, α and β are learnable scale and shift parameters. Then two private LN layer are applied to affine the single distribution into two pattern-specific distributions which integrate the private information(through the affine transformation of the private LN layer and the residual connection) of each representation and common information(through the common LN layer) of two representations,

$$\begin{aligned} M_v &= LN(M_v + Z_v;\alpha_v,\beta_v), \\ M_s &= LN(M_s + Z_s;\alpha_s,\beta_s). \end{aligned} \quad (8)$$

Finally, PWFF and two private LN are applied to further enhance two representations,

$$\begin{aligned} Z_v^{l+1} &= LN(PWFF(M_v;\theta_{vs}) + M_v;\alpha_v,\beta_v), \\ Z_s^{l+1} &= LN(PWFF(M_s;\theta_{vs}) + M_s;\alpha_s,\beta_s). \end{aligned} \quad (9)$$

We also apply iteration [17] with the proper number of iterations T on IILN to integrate more information into each representation. After IILN, the distributions of the two representations will be closer to each other while simultaneously maintaining modality-specific information.

4.3. Extra Skip Connection

The unique nature of the attention mechanism enables it to filter out irrelevant information. However, it may also filter out some weak but potentially useful information. To enhance the information flow within and between encoder and decoder to protect some fragile visual information from being filtered out by the attention layer, we add extra skip connections on them.

Similar to recursive skip connection [18], we first add an extra skip connection on the MHSA of the transformer, which can be formulated as,

$$M = LN(LN(MHSA(Z) + Z) + Z), \qquad (10)$$

it can incorporate more information that may have been filtered out by MHSA.

The information obtained from the IILN layer may not be effectively retained after passing through multiple transformer layers. We add the skip connection from the output of L_fth encoder layer to the output of encoder to force useful information from various flows to flow directly into the decoder:

$$Z = Z_{vs}^{N_e} + Z_v^{L_f} + Z_s^{L_f}. \qquad (11)$$

After combining IILN with skip connection, the visual information from the encoder is maximized.

4.4. Training Details

We follow a standard two-stage training strategy in image captioning [5, 9, 26]: pre-train the model with cross-entropy loss(XE) and finetune the model with reinforcement learning. For the reinforcement learning stage, we follow the training approach of the M^2 Transformer [5], the whole model is optimized with CIDEr reward which is the most popular metric of image captioning.

5. Experiments

5.1. Experimental setup

Datasets and Evaluation Metrics. We evaluate our method on the most popular image captioning benchmark, MS-COCO dataset [16]. The whole MS-COCO dataset contains 123,287 images, which includes 82,783 training images, 40,504 validation images, and 40,775 testing images, each of which corresponded with 5 different captions. We adopt the Karpathy splits [11], where 5,000 images are used for validation, 5,000 images for testing, and the rest images for training. All results are evaluated on the COCO Karpathy test split. Following the standard evaluation protocol, we employ the captioning metrics: BLEU at K(B@K) [23], METEOR(M) [14], ROUGE(R) [14], CIDEr(C) [28], and SPICE(S) [1].

Grid Feature Backbone	Seg. Feature Backbone	B@1	B@4	M	R	C	S
X101	✗	80.9	38.7	29.1	58.7	131.7	22.8
X101	R50	81.2	39.3	29.6	59.1	133.8 (**2.1**↑)	23.3
X101	R101	81.5	39.2	29.5	59.0	135.1 (**3.4**↑)	23.3
X152	✗	81.1	39.6	29.4	59.1	133.3	23.2
X152	R101	81.9	39.7	29.7	59.2	136.3 (**3.0**↑)	23.5
X152	GT	82.5	40.4	29.8	59.6	137.3 (**4.0**↑)	23.8

Table 1. Impact of feature quality. 'X' and 'R' denote ResNeXt and ResNet respectively.

Feature Size	Seg. Feature	FLOPS	B@1	B@4	M	R	C	S
7x7	✗	0.76G	80.9	38.7	29.1	58.7	131.7	22.8
	✓	0.92G	81.5	39.2	29.5	59.0	135.1	23.3
10x10	✗	1.41G	81.0	38.8	29.1	58.5	131.3	23.0
	✓	1.74G	81.9	40.0	29.7	59.2	135.2	23.5
14x14	✗	2.66G	80.9	38.9	29.2	58.5	131.4	22.9
	✓	3.33G	81.7	39.5	29.7	59.2	134.5	23.3

Table 2. Impact of feature size.

Implementation Details. For the visual representations of the input image, we follow the operation in [10] to extract gird features. For segmentation representation, we take UPSNet [33] as the segmentation feature extractor. We convert the segmentation map that came from the semantic segmentation head to semantic feature vector S whose size is $H \times W \times 133$, the dimension of 133 is logit corresponding to COCO classes.

Our model is implemented in PyTorch [24]. All the models are trained/tested on a single NVIDIA GTX1080Ti 10GB GPU. We use Aadm [12] with $\beta_1 = 0.9$, $\beta_2 = 0.98$ to optimize model training. We employ the Vanilla Transformer [27] with extra skip connection on multi-head attention as our baseline model. We follow the implementation of M^2 Transformer [5] to train our model. Specifically, the number of layers in transformer encoder and decoder are N_e = 3 and N_d = 3 respectively. More information on network setups are given in Appendix.

5.2. Analysis of Segmentation Feature

We perform ablation experiments in Sec. 5.2 and 5.3. Unless otherwise specified, the backbone of the grid feature extractor is ResNeXt-101 [32], the backbone of the segmentation feature extractor is ResNet-101 [7], the feature size is 7x7, the fusion method is VSA (L_f=1), all transformer models use recursive skip connection on the multi-head self-attention, and all listed perplexities are based on ten words piece. The parameter and GFLOPs analysis is based on the Transformer pipeline excluding the feature extraction phase. All ablation results are evaluated on COCO "Karpathy" test split.

Models	Seg. Feature	B@1	B@4	M	R	C	S
Transformer	✗	80.9	38.7	29.1	58.7	131.7	22.8
	✓	**81.5**	**39.2**	**29.5**	**59.0**	**135.1 (3.4↑)**	**23.3**
M^2 Transformer	✗	80.1	38.6	29.3	58.6	129.2	23.1
	✓	**81.0**	**39.0**	**29.5**	**58.9**	**132.7 (3.5↑)**	**23.2**
M^2 Transformer*	✗	80.9	38.7	29.0	58.4	131.2	22.5
	✓	-	-	-	-	-	-
AoA Transformer	✗	80.8	39.2	29.2	58.7	131.5	22.7
	✓	**81.2**	**39.4**	**29.5**	**58.9**	**134.2 (2.7↑)**	**23.3**

Table 3. Impact on different Transformer models. The '*' indicates using the original warm-up learning rate policy. The '-' indicates failure for the unacceptable metric results.

Fusion Method	L_f	Params	FLOPs	B@1	B@4	M	R	C	S
MIA	-	29.1M	0.92G	81.4	39.1	29.3	58.8	133.6	23.1
	1	35.1M	1.24G	81.3	39.5	29.5	59.0	133.0	23.2
VSA	0	**32.1M**	0.77G	81.2	39.4	**29.6**	**59.2**	134.6	23.2
	1	35.1M	0.92G	**81.5**	39.2	29.5	59.0	**135.1**	**23.3**
	2	38.1M	1.08G	81.4	**39.5**	29.5	59.1	134.3	23.2
	3	41.1M	1.24G	81.4	39.2	29.4	58.9	133.8	23.2
IILN	1	**32.1M**	**1.24G**	**81.5**	**39.5**	**29.6**	**59.1**	**135.0**	**23.3**
	2	**32.1M**	1.71G	81.4	39.4	29.4	59.0	134.6	23.3
	3	**32.1M**	2.18G	81.4	39.5	29.5	59.1	134.5	23.3

Table 4. Impact of fusion method at different fusion layer L_f. The '-' indicates we use original MIA without Transformer layer behind for fusion.

Impact of feature quality. We compare variants of DIFNet equipped with different quality visual features and segmentation features. For visual features, we adopt the features extracted by ResNeXt-101 and ResNeXt-152 backbone respectively. For the segmentation feature, we use the features extracted by UPSNet equipped with ResNet-50, ResNet-101 backbone, and ground-truth (GT) respectively. Results are shown in Table 1. Compared to the model only with grid features, the introduction of segmentation features can significantly improve model performance. With the growth of the quality of the two features, the performance of the model can be further improved.

Impact of feature size. We compare models with different input feature sizes $H' \times W' \in \{7 \times 7, 10 \times 10, 14 \times 14\}$, results shown in Table 2. Compared to the model with feature size 7×7, the larger size does not provide a significant performance boost while increasing the huge computational cost. This is potential because the large size makes it difficult for self-attention to decide which grids need to attend.

Impact on different Transformer models. To show the generality of segmentation features, we compare different Transformer models with the segmentation features, including our baseline Transformer, M^2 Transformer, and AoA Transformer. For M^2 Transformer, we use original architecture and provide two versions of results, one uses our epoch decay schedule, the other applies their original warm-up learning rate policy. For AoA Transformer, we replace transformer encoder layers of our baseline Transformer with AoA Refine modules and use the same decoder as our baseline, the reason for this setting is that the encoder is mainly related to the segmentation feature.

Results are shown in Table 3. We can see that the performance of all the aforementioned models has been significantly improved after the introduction of segmentation features. Note that, for the M^2 Transformer model with the original warm-up learning rate strategy, we got unacceptable metric results due to the unstable training process after the introduction of the segmentation feature.

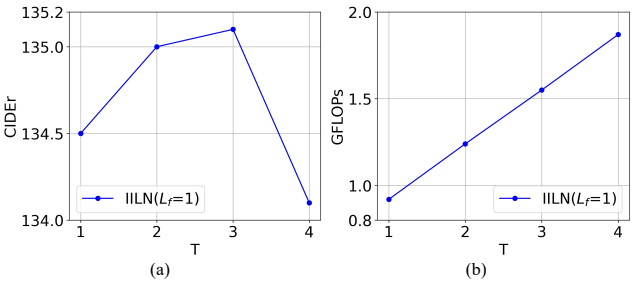

Figure 4. Impact of iteration times T. IILN with T=3 performs similarly compared to the one with T=2, indicating that T=2 is enough for information integration.

Impact of fusion method. We explore different methods to fuse grid features and segmentation features, such as MIA [17], VSA, and our IILN(T=2). And we investigate the impact of varying the fusion layer $L_f = 0, 1, 2, 3$ on the VSA and IILN. Results are shown in Table 4. We can see that the best performance is achieved at L_f=1. Further increasing L_f not only reduces the performance, but also increases the parameters and calculation costs. Compared with MIA and VSA fusion model, IILN can maintain the same number of parameters without significant performance degradation.

We also compare the IILN(L_f=1) with different iteration times T. As shown in Fig. 4, the computational cost increases linearly with T, while the best performance is achieved at around T=3. As T continues to increase, performance begins to decline, which may be caused by too many iterations causing the over-smoothing problem [17]. And IILN with T=3 performs similarly compared to the one with T=2, suggesting that T=2 already integrates enough information into each feature.

5.3. Analysis of Information Flow

Impact of extra skip connection. We investigate the models with extra skip connections. We first conduct exper-

Models	B@1	B@4	M	R	C	S
Vanilla Transformer	80.8	38.9	29.0	58.5	129.3	22.7
+MHSA,PWFF	80.8	38.8	29.2	58.7	131.5 (2.2↑)	22.8
+Dec: MHSA	80.6	38.7	29.1	58.6	130.9 (1.6↑)	22.8
+MHSA	**80.9**	38.7	29.1	**58.7**	**131.7 (2.4↑)**	**22.8**
MIA	81.3	39.5	29.5	59.0	133.0	23.2
MIA + skip	81.2	**39.5**	**29.6**	**59.2**	**134.5 (1.5↑)**	**23.3**
VSA	81.5	39.2	29.5	59.0	135.1	23.3
VSA + skip	**81.8**	**39.9**	**29.5**	**59.2**	**135.4 (0.3↑)**	23.2
IILN	81.5	39.5	29.6	59.1	135.0	23.3
IILN(w/o com_LN) + skip	81.7	40.0	29.7	59.3	135.3 (0.3↑)	23.4
IILN + skip	**81.7**	**40.0**	**29.7**	**59.4**	**136.2 (1.2↑)**	23.2

Table 5. Impact of extra skip connection.

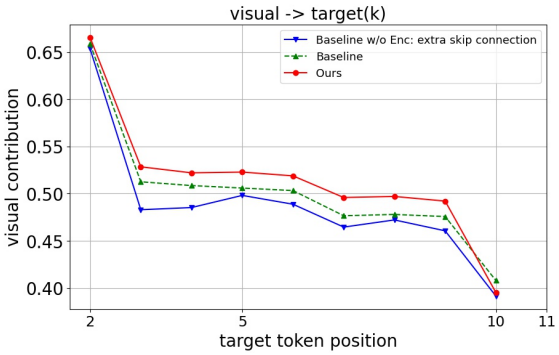

Figure 5. Contribution of visual information. DIFNet significantly improves the visual contribution, making the final prediction faithful to the visual content.

iments on the Vanilla Transformer with only grid features to explore where to add extra skip connection: (i) MHSA and PWFF; (ii) MHSA; (iii) only MHSA in Decoder. Then, we add an extra skip connection from the output of L_f encoder layer to the output of encoder for MIA, VSA, and IILN.

Results are shown in Table 5. Overall, the models with extra skip connections can significantly improve performance. For Vanilla Transformer, the best performance is achieved by adding the extra skip connection to MHSA, which is higher than the performance of adding extra skip connection on both MHSA and PWFF, we believe this is because it retains a lot of useful information filtered by MHSA. In addition, we can see that the IILN fusion model with skip connection gets the most benefit. To find out why it works, we also conduct an experiment to replace the common LN layer with two private LN layers. We can see that it performs similar to VSA with skip connection, which is potential because the common LN layer makes the fusion representation distribution and each flow representation distribution close to each other, so that they are easier to aggregate before being input to the decoder.

Contribution of Visual Information. We use $\alpha\beta$-LRP [4, 29] to evaluate the contribution of visual information to the prediction at each time step. More information

Figure 6. Examples of captions generated by the baseline model and our model, along with the ground-truth sentence.

about LRP are given in Appendix. We compare the models: (i) Baseline without adding extra skip connection in Encoder; (ii) Baseline; (iii) Our model. Note that at each time step, the sum of the contribution of the visual information and the context of the partial caption is equal to 1.

Results are shown in Figure 5. We can see that the visual contribution of baseline without adding extra skip connection into encoder is mostly below 0.5, which indicates that the contribution to the prediction is dominated by the caption context. When adding extra skip connections into the encoder, the situation has improved. Results show that our model further improves the visual contribution, making the final prediction faithful to the visual content.

5.4. Visualizations

Fig. 6 shows representative examples of captions generated by the baseline model and our model. Compared to the baseline model which generates some phrases that violate visual content, our model can generate the description consistent with the visual content of the image. For example, in Fig. 6 (a), because the feature of the scissor is difficult to capture, the baseline model tends to generate the

Models	B@1	B@4	M	R	C	S
SCST [26]	-	34.3	26.7	55.7	114.0	-
UpDown [2]	79.8	36.3	27.7	56.9	120.1	21.4
GCN-LSTM [35]	80.5	38.2	28.5	58.5	128.3	22.0
SGAE [34]	80.8	38.4	28.4	58.6	127.8	22.1
AoANet [9]	80.2	38.9	29.2	58.8	129.8	22.4
M^2 Transformer [5]	80.8	39.1	29.2	58.6	131.2	22.6
X-Transformer [22]	80.9	39.7	29.5	59.1	132.8	**23.4**
DLCT [19]	81.4	39.8	29.5	59.1	133.8	23.0
RSTNet [38]	81.1	39.3	29.4	58.8	133.3	23.0
Ours	**81.7**	**40.0**	**29.7**	**59.4**	**136.2**	23.2

Table 6. Comparison with state-of-the-art methods.

phrase "holding a spoon" according to the language context "baby sitting at a table", which appears more frequently in the dataset. However, DIFNet uses segmentation map information to correct visual semantics, thereby generating the description that is strongly related to the image. In addition, as shown in Fig. 6 (e)(f), DIFNet can infer the underlying semantic and spatial relationships compared to the baseline.

Fig. 7 shows a failure example of captions generated by the baseline model and our model. The "two adult elephants" is misjudged as "an adult elephant" by our DIFNet, which may be because the segmentation feature we used in our implementation is semantic segmentation instead of panoptic segmentation map, which makes it hard to distinguish instances of the same category. And the model also failed to correctly distinguish which instance the related grids should be aligned with.

5.5. Comparison to State-of-the-art

We compare our DIFNet with state-of-the-art methods on the COCO 'Karpathy' test split. The compared models include: SCST [26], UpDown [2], GCN-LSTM [35], SGAE [34], AoANet [9], M^2 Transformer [5], X-Transformer [22], DLCT [19], RSTNet [38]. The visual feature extractor is all ResNet-101 or ResNeXt-101. The backbone of the segmentation feature extractor of our model is ResNet-101, the feature size is 7×7, the fusion method is IILN(L_f=1, T=2) with skip connection.

The results are shown in Table 6. Our model achieves significant performance gain over existing methods. Our model outperforms the DLCT [19] by 2.4% on CIDEr. Note that DLCT uses the grid feature and the region feature at the same time, but these two features are difficult to effectively align and there is a lot of information redundancy.

5.6. Discussions

Limited by resources, we still have two aspects to explore in the future. On the one hand, it is possible to perform multi-task, *e.g.* panoptic segmentation and captioning, in an end-to-end manner to reduce computational complexity.

Baseline: a baby elephant standing between two adult elephants
DIFNet: a baby elephant standing next to an adult elephant
Ground Truth :
Two adult elephants are surrounding a baby elephant
a baby elephant kneeling in front of two bigger elephants
A baby elephant and it's parents eat fruit.

Figure 7. A failure case of captions generated by baseline model and our model. Presented images are input image (left) and panoptic segmentation map (right), respectively.

On the other hand, we believe that a variety of visual representations, such as object detection, action recognition, depth estimation, *etc.*, could be integrated together to facilitate flexible and accurate visual-lingual understanding.

6. Conclusion

In this work, we present DIFNet to generate caption sequence faithful to the given image. We first use the segmentation feature to enhance grid visual representation by the Iterative Independent LN (IILN) fusion module to maximize the use of two information flows. We also use additional skip connections to enhance the flow of information within and between encoder and decoder to protect some fragile visual information. Experiments show that the various transformer variants with the segmentation features get better performance, and DIFNet with segmentation features outperforms state-of-the-art methods. Comprehensive ablation studies reveal several key factors that lead to this success and show that dual information flow is highly effective in boosting the dependence of prediction on visual content.

Acknowledgement

This work was supported by the National Science Fund for Distinguished Young Scholars (No.62025603), the National Natural Science Foundation of China (No. U21B2037, No. 62176222, No. 62176223, No. 62176226, No. 62072386, No. 62072387, No. 62072389, and No. 62002305), Guangdong Basic and Applied Basic Research FoundationNo.2019B1515120049), and the Natural Science Foundation of Fujian Province of China (No.2021J01002).

References

[1] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, 2016. 5

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086, 2018. 2, 8

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016. 3

[4] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015. 2, 7

[5] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10578–10587, 2020. 1, 2, 3, 5, 8

[6] Alex Graves. Generating sequences with recurrent neural networks, 2014. 3

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3, 5

[8] Lisa Anne Hendricks and Aida Nematzadeh. Probing image-language transformers for verb understanding, 2021. 1

[9] Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. Attention on attention for image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4634–4643, 2019. 1, 2, 3, 5, 8

[10] Huaizu Jiang, Ishan Misra, Marcus Rohrbach, Erik Learned-Miller, and Xinlei Chen. In defense of grid features for visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10267–10276, 2020. 1, 3, 5

[11] Andrej Karpathy and Fei-Fei Li. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, pages 3128–3137. IEEE Computer Society, 2015. 5

[12] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[13] Alexander Kirillov, Ross Girshick, Kaiming He, and Piotr Dollár. Panoptic feature pyramid networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6399–6408, 2019. 2

[14] Alon Lavie and Abhaya Agarwal. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *SMT*, StatMT '07, page 228–231, USA, 2007. ACL. 5

[15] Xiujun Li, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020. 1, 2

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014. 5

[17] Fenglin Liu, Yuanxin Liu, Xuancheng Ren, Xiaodong He, and Xu Sun. Aligning visual regions and textual concepts for semantic-grounded image representations, 2019. 2, 4, 6

[18] Fenglin Liu, Xuancheng Ren, Zhiyuan Zhang, Xu Sun, and Yuexian Zou. Rethinking skip connection with layer normalization in transformers and resnets, 2021. 5

[19] Yunpeng Luo, Jiayi Ji, Xiaoshuai Sun, Liujuan Cao, Yongjian Wu, Feiyue Huang, Chia-Wen Lin, and Rongrong Ji. Dual-level collaborative transformer for image captioning, 2021. 2, 8

[20] Arsha Nagrani, Shan Yang, Anurag Arnab, Aren Jansen, Cordelia Schmid, and Chen Sun. Attention bottlenecks for multimodal fusion, 2021. 2, 4

[21] Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. Multimodal deep learning. In *ICML*, 2011. 2

[22] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10971–10980, 2020. 1, 8

[23] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002. 5

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019. 5

[25] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. 1

[26] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7008–7024, 2017. 5, 8

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 3, 5

[28] R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, pages 4566–4575, 2015. 5

[29] Elena Voita, Rico Sennrich, and Ivan Titov. Analyzing the source and target contributions to predictions in neural machine translation, 2021. 7

[30] Yikai Wang, Fuchun Sun, Ming Lu, and Anbang Yao. Learning deep multimodal feature representation with asymmetric multi-layer fusion. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 3902–3910, 2020. 2, 4

[31] Qi Wu, Chunhua Shen, Lingqiao Liu, Anthony Dick, and Anton Van Den Hengel. What value do explicit high level concepts have in vision to language problems? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 203–212, 2016. 2

[32] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 5

[33] Yuwen Xiong, Renjie Liao, Hengshuang Zhao, Rui Hu, Min Bai, Ersin Yumer, and Raquel Urtasun. Upsnet: A unified panoptic segmentation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8818–8826, 2019. 2, 5

[34] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10685–10694, 2019. 8

[35] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699, 2018. 8

[36] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Hierarchy parsing for image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2621–2629, 2019. 2

[37] Ting Yao, Yingwei Pan, Yehao Li, Zhaofan Qiu, and Tao Mei. Boosting image captioning with attributes. In *Proceedings of the IEEE international conference on computer vision*, pages 4894–4902, 2017. 2

[38] Xuying Zhang, Xiaoshuai Sun, Yunpeng Luo, Jiayi Ji, Yiyi Zhou, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Rstnet: Captioning with adaptive attention on visual and non-visual words. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15465–15474, 2021. 8