Global Semantic-guided Network for Text-guided Image Saliency

Yida Min, SJTU, *521030910024*

***Abstract*—Visual attention analysis and prediction are important tasks in computer vision and image processing, and many visual saliency prediction models were released recently. In practical applications, images are generally accompanied by various text descriptions, however, few studies have explored the influence of text descriptions on visual attention, let alone developed visual saliency prediction models considering text guidance. Therefore, in this paper, we mainly focus on the problem of whether and how the text-guidance influences the visual attention. Then, by using global semantic-guided network (GSGNet), we design a saliency model by using the datasets SJTUTIS and SALICON. Our proposed model significantly outperforms the state-of-the-art saliency models in terms of various evaluation metrics. The codes and models can be viewed at https://github.com/BenjaminCacok/saliency.**

***Index Terms*—Text guidance, visual attention, image saliency, Semantic information, Transformer**

I. Introduction

Visual attention analysis and prediction have long been important tasks in computer vision and image processing. Since they may give new insights about human attention mechanisms and contribute to new artificial intelligence applications. After decades of development, many visual saliency models, have been proposed with excellent results, such as SALGan, TranSalNet, SAM-VGG. These saliency prediction models only use the image information to predict the salient area of an image.

However, The effects of specific texts on the visual attention are still not clear. Human visual attention mechanism guides humans to observe scenes selectively and ignore less informative regions, enabling humans to analyze complex and diverse scenes quickly. When humans freely view images, their attention is influenced by local, global and semantic information within the visual stimulus. And Human vision attention can be categorized into two functions including scene-driven bottom-up (BU) and expectation-driven top-down (TD).

To study the influence of text-guidance on the visual attention and design a saliency prediction model considering text guidance, we propose a global semantic-guided network that refines both local and global saliency features in multi-level visual features, with the aid of global semantic information. Then we propose a channel-squeeze spatial attention (CSSA) module, which facilitates the exchange of channel information and enables the successive global feature representation through sequential channel and spatial interactions. Finally We propose a local-global fusion block (LGFB) that combines the merits of CNN and transformer to enrich local and global information for visual features at different levels. Also, we adopt the BERT encoder in natural lauguange processing (NLP) to combine image information and text information.

The framework of this paper is shown as follows. First, we will introduce the relevant backgrounds and setbacks while designing the saliency model. Second, we analyze the datasets we use and the implementation details of our models, including the metrics and the experimental setup. Third, accoding to the experimental results, we analyze the effects of different text descriptions on visual attention, and indicates that image saliency is significantly influenced by a text description. We will show the advantages of the designed model. Last, we will conclude the whole paper and the direction of future improvement is proposed.

II. Saliency Prediction Backgrounds

A. Saliency Maps

The information selection strategy of the human visual system is determined in two main ways:

- Based on a task-driven top-down approach, that is, people's own "cognitive factors", such as knowledge, expectations and goals, The human visual system (HVS) restrict the visual cognitive simulation process, and then calculate the saliency of image regions.
- The human visual system is based on a data-driven bottom-up approach, which is based only on the attention mechanism of visual signals. Without any prior information, the image features of the current scene, such as color, contrast and edge, are used to calculate the image region saliency.

Designing computational models of visual attention or saliency aims to predict where we pay attention in different scenes. Fig1 is an example of the principle of such models.

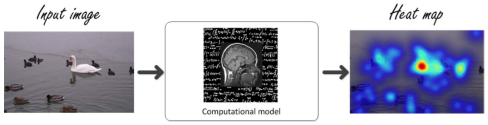


Fig. 1: principle of saliency models

Visual attention or image saliency data are usually collected by eye trackers. There are two types of maps to quantitatively represent saliency.

- Fixations: These are fixation points generated from mouse trajectories. When people look at an image, their eyes stop at certain regions in the image, which are called fixation points. In the SALICON dataset, fixation points were simulated by having participants move their mouse across the image.

- Maps: These are continuous saliency maps that are generated by clustering and blurring all preprocessed mouse click samples with a Gaussian filter. The saliency map can be viewed as a kind of heat map that represents the saliency of each pixel in the image.

In general, we use saliency maps as our ground truth to train the model. Fig2 shows the relationship of original picture, fixation map and saliency map (without texts input).

(a) original (b) fixation (c) map

Fig. 2: example of saliency map

B. Saliency Models

Image saliency prediction is an important task in computer vision, aiming to predict the areas that humans are most likely to focus on when observing images. Some major saliency prediction models include:

- Itti-Koch Model: One of the earliest saliency prediction models, based on biological principles, predicts salient areas by calculating saliency maps of features such as color, brightness, and orientation.
- GBVS (Graph-Based Visual Saliency): This model uses graph theory methods to calculate saliency maps, better handling high-level semantic information in images.
- Deep Learning Models: In recent years, deep learning technology has made significant progress in saliency prediction. For example, DeepFix and DSS models use convolutional neural networks to predict salient areas, handling more complex image content.

C. BERT Encoder

BERT (Bidirectional Encoder Representations from Transformers) is a pre-training model proposed by Google AI Research Institute in October 2018. The network architecture of BERT uses the multi-layer Transformer structure proposed in "Attention is all you need". Its main function is to pre-train deep bidirectional representations of unlabeled text by jointly conditioning on all layers of bidirectional context. This pre-trained representation can be used for various natural language processing (NLP) tasks, such as answering questions, classification, Named Entity Recognition (NER), etc.

Another important function of BERT is to reduce the need for carefully designed specific architectures for NLP tasks. We can fine-tune the pretrained BERT model with only one additional output layer to create the latest model suitable for various tasks. This allows BERT to achieve state-of-the-art performance in a series of sentence-level and character-level tasks, superior to many task-specific architectures. So we can use BERT pretrained model to implement the embedding of the texts into our pictures.

The encoder part of BERT is mainly divided into three parts: input part, multi-head attention mechanism, and feed-forward neural network (similar to the transformer encoder). BERT uses multiple encodes stacked together, where BERT-base uses 12 layers of encoder, and BERT-large uses 24 layers of encoder. The model structure of transformer encoder (i.e. BERT encoder) is shown in FIG3.

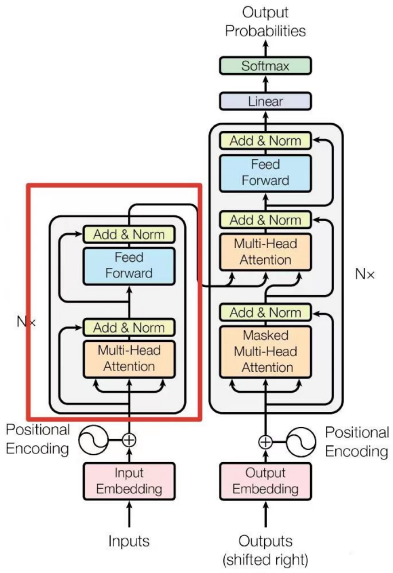


Fig. 3: transformer encoder

III. Implementation Details

In this section, we describe the details of our method. We will present the overall architecture of our proposed mode and describe the details of each part of the mode. Finally we will introduce the loss function.

A. Architecture Overview

Learned from structures like FPN and U-Net widely used in dense prediction tasks like semantic segmentation, we design a general encoder-decoder structure for the saliency prediction task, as illustrated in Fig4.

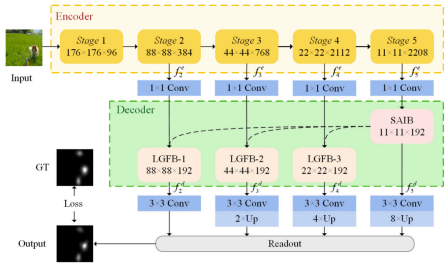


Fig. 4: general structure

The encoder part is DenseNet-161 pretrained on ImageNet, which generates low-level features at the early stages and high-level semantic features at the later stages. The backbone is divided into five stages and each stage is denoted as Stage $i, i \in [1, 2, 3, 4, 5]$. Here, we employ four stages ranging from Stage 2 to Stage 5. The multilevel features, are first fed to the corresponding 1×1 convolutional layers to reduce the channel

number to 192. The decoder part consists of a spatial attention inception block (SAIB) that enhances semantic features via a parallel CSSA structure, as well as three LGFBs that leverage these refined global semantic features to guide both local and global fusion of features from various levels using a CSSA module and CNN-based architectures. Then we utilize 3×3 convolutional layers following the decoder outputs to reduce the channel dimension to 128. Finally, the readout module produces the final saliency map of the input image by utilizing these enhanced features.

B. Channel-squeeze Spatial Attention

Spatial attention modules utilize average-pooling and max-pooling techniques to aggregate channel information of features, refining them to effectively focus on the informative parts in the spatial dimension. This indicates that by using an appropriate method for channel interaction, we can enhance the global features extracted from spatial attention maps, which aligns with the objective of self-attention to capture long-range dependencies. To explore spatial semantic information and fully leverage inter-channel relationships, we propose a Channel-Squeeze Spatial Attention (CSSA) module. Basically, the modules first learn the overall characteristic by channel compression and then refine the global spatial information through the dot-product operation.

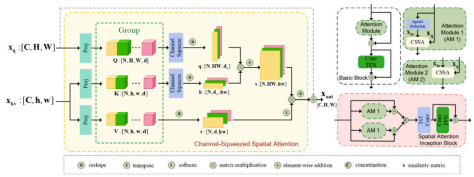


Fig. 5: the details of CSSA

For channel interaction, average pooling captures the overall spatial features, but it assigns equal importance to all channels. Max pooling only focuses on the highest values. Meanwhile, when these operations are applied to the grouped features, there is no exchange of information among groups. To tackle this issue, we apply 3D convolutions [66], which were initially introduced to extract visual features from video frames and acquire temporal feature representations.

Based on the input features operations, CSSA is applied in two basic modules. The attention module 1 employs a spatial reduction operation to decrease the spatial resolution. The features are first improved by a residual attention module and then processed by a residual feedforward neural network (FFN) module. By using different spatial reduction operations, the CSSA module can effectively extract features from multiple perspectives and enhance the feature representations. SAIB employs two attention modules to obtain multiple feature representations. Following these modules, a 1×1 convolutional layer is utilized to reduce the dimension of the concatenated features to C, which are then fed into the FFN module to obtain the enhanced features.

The details of the Channel-Squeeze Spatial Attention (CSSA) module, Spatial Attention Inception Block (SAIB) and corresponding modules are shown in Fig5.

C. Local-global Fusion Block

We propose a Local-Global Fusion Block (LGFB) that incoporates semantic information into features both locally and globally. Inspired by ShuffleNet, we employ channel split operation before two branches to reduce computational requirements. As for the local branch, since the encoder has already owned a great ability to capture local features, we apply two simple CNN-based feature embedding structures in our decoder, i.e., semantic embedding and spatial attention. Illustration of the proposed local-global fusion block is shown in Fig6.

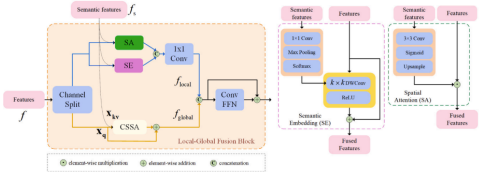


Fig. 6: the details of LGFB

- Semantic Embedding (SE): Semantic features usually have a small resolution and are full of contextual information. Therefore, features f_s are suitable to be processed as the kernels of convolutions. we modify the f_s with 1×1 convolutional layer and maxpooling operation to $k \times k$ kernels and use softmax to normalize the sum of values to 1 in the spatial domain. Then we employ a depthwise convolution with the semantic kernels to enhance the other-level features. The kernels are dynamically generated from semantic features f_s, which adjust weights with the input.
- Spatial Attention (SA): we first apply a 3×3 convolution to squeeze the dimension of global semantic features f_s to 1. Then a sigmoid function scales the values of features between 0 and 1 to obtain a spatial attention map. Finally, the attention map is upsampled and multiplied by the low-level features to highlight the informative regions.

The fused maps after SA and SE are concatenated, and a 1×1 convolutional layer is used to restore the channel dimension to half. After passing through two branches, the concatenated features of f_{local} and f_{global} are fed into a residual FFN module to interact with channel information.

D. Readout Module and Loss Function

The readout module is usually exploited to aggregate the refined features and convert the feature maps into a saliency map with the same resolution as the input.

We use the combination of Kullback-Leibler Divergence (KL) and Pearson's Correlation Coefficient (CC) as a loss function. The meaning of these two values will be disgussed in 4.1. The total loss function is defined as follows:

$$\begin{aligned} loss(P,G) =& kl(P,G) - CC(P,G) + \\ & \alpha[BCE(P_2,G_2) + BCE(P_3,G_3)] \end{aligned} \tag{1}$$

where P_i is the saliency map generated from ${f_i}^d$, G is the groundtruth map and P_i is the ground-truth map which

is resized to the same resolution as ${f_i}^d$, $i \in [2, 3]$. To balance the gradient and maintain the optimization direction controlled by saliency metrics, α is empirically set to 0.001 during the first epoch and later set to 0.

IV. Experiment and Results

In this section, we will discuss the experimental setup and the results analysis.

A. Metrics

In this experiment, we use KL, CC and NSS value for training and validating and sAUC value for testing.

- KL Divergence (Kullback-Leibler Divergence): KL divergence is used to measure the difference between two probability distributions. In saliency prediction, it can be used to measure the difference between the predicted saliency map and the real saliency map. The smaller the KL divergence, the closer the predicted saliency map is to the real saliency map.
- CC (Pearson Correlation Coefficient): The Pearson correlation coefficient is used to measure the linear correlation between two variables. In saliency prediction, it can be used to measure the linear correlation between the predicted saliency map and the real saliency map. The larger the CC value, the closer the predicted saliency map is to the real saliency map.
- NSS (Normalized Scanpath Saliency): Normalized Scanpath Saliency is used to measure the difference between the predicted saliency map and the real saliency map. The larger the NSS value, the closer the predicted saliency map is to the real saliency map.
- sAUC (Shuffled Area Under Curve): sAUC is a special AUC calculation method. When calculating AUC, it does not use all non-salient areas as negative samples, but randomly selects the same number of non-salient area samples as salient area samples. This method can eliminate the influence of objects in the image and more accurately evaluate the performance of the saliency prediction model.

B. Experimental Setup

First, the SJTUTIS dataset is divided into 5 parts according to the types of images: (1) pure pictures without text discription. (2) type 1: general scenario descriptions. (3) type 2: specified descriptions of salient objects. (4) type 3: specified descriptions of non-salient objects. (5) type 4: common descriptions contain both salient and non-salient objects. Then, according to the different types, we train different models to analyze the effects of different text descriptions on visual attention, and we find that image saliency is significantly influenced by a text description, and different text descriptions for the same image may have different impacts on the corresponding visual attention.

The training data is shown in Fig7. By comparing the KL value and CC value, we observe that GSGNet demonstrates robust performance across all tasks. The model shows diverging trend in these cases. In type 0 (pure picture) and type 1 (with general descriptions), our model obtains high CC values (above 0.8) and low KL values (below 0.4) near the 30th training epoch, which proves the superiority of the model during training. In type 2 (salient descriptions), the CC values are still high while the KL values increase. That may because the model can't extract all the features of pictures. However, its performance on non-salient tasks is not as strong. Taken together, our model achieves competitive performance with other saliency prediction models.

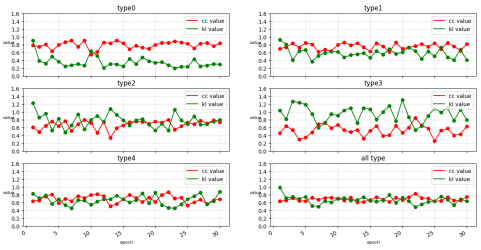


Fig. 7: training data

During training, we validate the proposed model on the test dataset every epoch. We find that the results of the validation are slightly worse than those of the training. This may be caused by the overfitting of the proposed model. However, the overall trends of metrics changes by epoch are the same with those in training. The validation data is shown in Fig8.

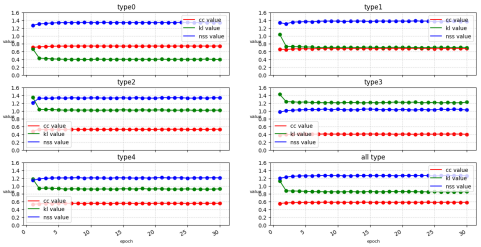


Fig. 8: validation data

C. Test and Analysis

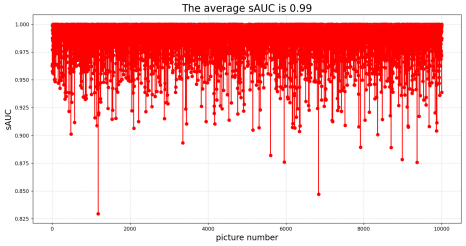
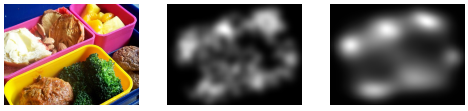


Fig. 9: test results

We could not train a very complete large model due to the defective experimental equipment and insufficient memory of the graphics card (NVIDIA GEFORCE RTX 3050), so we fine tune a pretrained model of GSGNet. Using this model, We perform the test task of sAUC on the SALICON dataset. This is because that this pretrained model is trained on SALICON dataset and SALICON dataset is a classic dataset

with 10000 training pictures, 5000 validation pictures and 5000 test pictures. The results is shown in Fig9, from which we can see this method really has an impressive performance for saliency prediction.

Meanwhile, we obtain the prediction saliency map using our method while tesing. Fig10 is a plot of the model predictions versus the ground truth. From this we can see the accuracy of the prediction. (This prediction map is obtained under the pure situation).



(a) original (b) fixation (c) map

Fig. 10: test examples

D. Ablation study

By adjusting the value of α in the loss function, we can compare the differences in model performance and choose the best loss function. The next table shows the results. (BCE-1 means that BCE loss is only utilized for epoch one.)

TABLE I: ablation study of loss function

Type	kl	CC	NSS
base	0.1839	0.9122	1.9405
BCE-a	0.1841	0.9111	1.9356
BCE-1	0.1822	0.9129	1.9419
BCE-1 + 0.5*NSS	0.1887	0.9030	2.0088

V. Conclusion

In this paper we propose a image saliency prediction model under text-guidance. We use a semantic-guided network for predicting saliency maps. Features in the encoder's deepest layer contain rich semantic and contextual information, which are further refined by the channel-squeeze spatial attention (CSSA)-based blocks, which capture global representations from the processed spatial attention maps. Additionally, multi-level features are integrated by the local-global fusion block (LGFB) combining the merits of CNNs and transformers, fusing local and long-range spatial information at multiple perceptual levels. We have conducted training, validation and test experiments to verify the effectiveness and robustness of our proposed model.

There are still some thoughts to be completed, e.g. how to combine text features with image features more effectively. Previously, we use BERT encoder to extract text feature embeddings and adopt the features concatenation methods. We perform a simple linear transformation on the text modality to convert it into the feature vector size required by the image modality, and then concatenate the text feature vector with the image feature vector.

However, this method may cause huge consumption of computing resources, too many parameters and redundant information. Addressing this challenge, BLIP emerges as a well-suited solution. BLIP facilitates this alignment by implementing cross-attention mechanisms. Specifically, it combines text embeddings obtained from BertTokenizer with image embeddings derived from ViT within the BLIP architecture. This approach allows for a more nuanced and coordinated interplay between textual and visual information, essential for tasks requiring a deep understanding of both modalities. Subsequent work will be devoted to this point.

References

[1] Yinan Sun, Xiongkuo Min*, Huiyu Duan, Guangtao Zhai*, *The Influence of Text-guidance on Visual Attention*. Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University Shanghai, China

[2] Yinan Sun, Xiongkuo Min*, Member, IEEE, Huiyu Duan, and Guangtao Zhai*, Senior Member, IEEE, *How is Visual Attention Influenced by Text Guidance? Database and Model*.

[3] Jiawei Xie, Zhi Liu, Gongyang Li, Xiaofeng Lu, Tao Chen, *Global semantic-guided network for saliency prediction*.

[4] Junting Pana, Cristian Canton-Ferrerb, Kevin McGuinnessc, Noel E. O'Connorc, Jordi Torresd, Elisa Sayrola, Xavier, Giro-i-Nieto, *SalGAN: visual saliency prediction with adversarial networks*

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, https://github.com/4AI/bert-encoder

[6] Junnan Li, Dongxu Li, Caiming Xiong, Steven Hoi, *BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation*, https://github.com/salesforce/BLIP