Saliency Prediction with Relation-aware Global Attention Module

Ge Cao and Kang-Hyun Jo*

School of Electrical Engineering, University of Ulsan, Ulsan, Republic of Korea
caoge@islab.ulsan.ac.kr; acejo@ulsan.ac.kr

Abstract. The deep learning method has achieved great success in saliency prediction task. Like depth and depth, the attention mechanism has been proved to be effective in enhancing the performance of Convolutional Neural Network (CNNs) in many studies. In this paper, we propose a new architecture that combines encoder-decoder architecture, multi-level integration, relation-aware global attention module. The encoder-decoder architecture is the main structure to extract deeper features. The multi-level integration constructs an asymmetric path that avoid information loss. The Relation-aware Global Attention module is used to enhance the network both channel-wise and spatial-wise. The architecture is trained and tested on SALICON 2017 benchmark and obtain competitive results compared with related research.

Keywords: Saliency prediction · attention mechanisms · Relation-aware global attention

1 Introduction

Saliency prediction is a very basic research field in computer vision but can be used in many other tasks like object recognition [1], tracking regions of interest [2], image retargeting [3] and so on. For human visual attention, we would pick the most interesting region in your mind the first time when we see a scene. Then our attention will be distracted and begin to notice other things that are more subtle and detailed. These operations effectively help humans focus limited attention on key information, save visual resources, and obtain the most significant information quickly.

For the saliency prediction task, it describes the spatial location of an image that attracts the observer most. For traditional saliency prediction, some low-level features such as color, texture, and semantic concepts. Obviously, these kinds of methods cannot achieve satisfactory performance. With the development of CNNs, saliency prediction task can also use deep learning method to train complicated models thanks to generous data-driven methods and large scale annotated datasets [4]. And many works [5] achieved great results in saliency prediction tasks.

* Corresponding author

To extract deep feature and high-level semantic information, this paper constructs an encoder-decoder architecture to do feature extraction. For enhancing the robustness of the network and avoid information loss by pooling operations in CNNs, and U-Net [6] like architecture, we called multi-level integration is used. Recently, many works resort to attention mechanism and surely obtain better results. In this paper, we propose a new architecture that combines encoder-decoder architecture, multi-level integration, and relation-aware global attention (RGA) module which proposed by [7]. In the paper [7], they proved that both channel-wise relation-aware global attention (RGA-C) module and spatial-wise relation-aware global attention (RGA-S) module improve the performance of the baseline network. As shown in their visualization results we can surely find that the attention mechanism makes the network pay more attention to the discriminative features part. In this paper, we combine the RGA-SC (sequential spatial-channel) module in encoder-decoder architecture and successfully improve the performance of the baseline.

To prove the effectiveness of the architecture proposed by this paper, we compared with [8] which proposed similar architecture with this paper. In [8], the self-attention module is used to enhance the global relation of the final convolutional layer of the encoder. Compared with [8], the experimental results show that using the RGA-SC module could effectively predict the saliency map.

The remaining content is organized as follows. Section 2 summarized the related work. The details of each component in the whole architecture and the loss functions used are introduced in Sect. 3. Section 4 provides the details and results of the experiments. Finally, Sect. 5 concludes the paper.

2 Related Work

The traditional methods of saliency prediction task focus on low-level features. Far-reaching work by Itti [9] relied on color, intensity, and orientation maps, and then using Gabor filter to integrate the features to get global saliency feature map. Bruce [10] use low-level local features in combination with information-theoretic ideas. Borji [11] combined low-level features of previous best bottom-up models with top-down cognitive visual features and learn a direct mapping from those features to human eye fixations.

With the continuous development of deep learning techniques and large-scale datasets, the performance of saliency prediction is steadily improving. Most of this success can be attributed to Convolutional architectures. Some work researched to combine CNN with recurrent neural networks [12]. The LSTM architecture makes a great improvement in saliency prediction. And [12] also proposed to use combined evaluation metrics loss to train the model. In attention mechanism fields, Woo [13] use large filter size of 7×7 over the spatial features in their Convolutional Block Attention Module (CBAM) to produce a spatial attention map. Followed by [13], [7] proposed Relation-aware Global Attention (RGA) module with spatial-wise and channel-wise sequentially (RGA-SC). Their extensive ablation studies demonstrate that our RGA can significantly enhance the feature representation power.

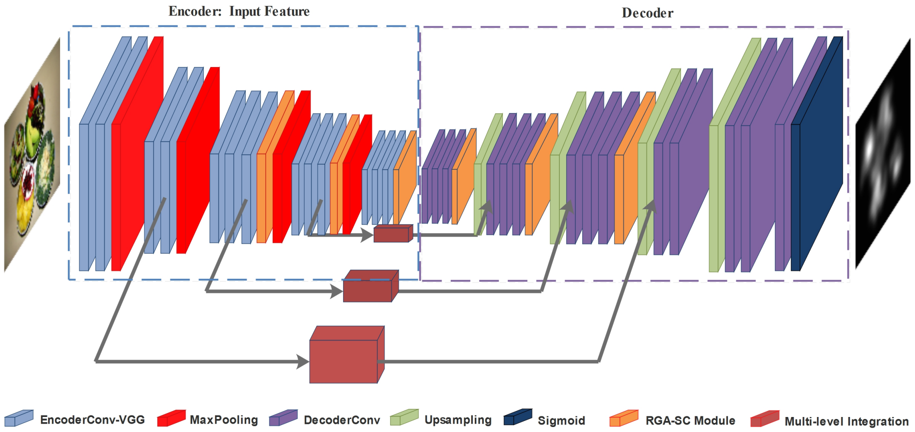

Fig. 1. The overview of the proposed saliency prediction model. The left image is the input image, we use the proposed CNNs architecture to extract feature and process the saliency prediction. The module shown in different color would be introduced in detail in Sec. 3.

3 Proposed Architecture

In this section, we introduce the proposed architecture.

3.1 Overview Structure

The whole architecture of the proposed network is introduced in this subsection. For saliency prediction, the fully convolutional network (FCN) has achieved great performance. To extract the deeper features and high-level semantic information, we need an encoder to decrease the size of the feature map and save computation resources. As shown in Fig. 1, the encoder of the proposed architecture has convolutional layers, max-pooling operations which decrease the size and increase the receptive field, and the RGA-SC module which would introduce in subsection 2. After the process of the encoder, the proposed network obtain a multi-channel tensor which has a smaller size feature map and high-level information. To reduce the size, we use upsampling operations in the decoder and finally get the saliency probability map through the final sigmoid layer.

The size of input images is generally changed to 256 × 256 with initial 3 channels. In the encoder part, except the RGA-SC module, the architecture of the network is identical in architecture to VGG16 [14] except the final max-pooling layer and three fully connected layers. Through the encoder part, the size of the input feature map is changed to 16 × 16, where the RGA-SC module would not change the size of the feature map. For the decoder part, the order of its layer is revered with the encoder with max-pooling operations replaced by upsampling operations to restore the size of the feature map. At the final of the network is a 1 × 1 convolutional layer with sigmoid non-linearity which ultimately produces the predicted saliency map. There also have 3 U-Net like architecture that concatenates the feature maps in the same scale in encoder and decoder, we call multi-level integration (MI) in this paper. Each MI module has 3 convolutional layers followed by ReLU.

3.2 Relation-aware Global Attention Module

Though the RGA module [7] is proposed to solve problems for other field, we could also draw lessons from it in saliency prediction. In RGA module, spatial-wise Relation-aware Global Attention (RGA-S) and channel-wise Relation-aware Global Attention (RGA-C) are introduced respectively.

Spatial Relation-aware Global Attention (RGA-S)
Given input tensor $X \in \mathbb{R}^{C \times H \times W}$, where C denotes channel, H denotes height, W denotes width. For learning the spatial attention map of size $H \times W$, RGA-S block is proposed. Taken C-dimensional feature vector at each position as a feature node, so there are $N = H \times W$ nodes. Each feature node of RGA-S can be represented as $\mathbf{x}_i \in \mathbb{R}^C$, where $i = 1, 2, ..., N$.

Then the pairwise relation between i-th node and j-th node can be defined as:

$$r_{i,j} = f_s(\mathbf{x}_i, \mathbf{x}_j) = \theta_s(\mathbf{x}_i)^T \phi_s(\mathbf{x}_j), \tag{1}$$

where θ_s and ϕ_s are both 1 × 1 convolutional layer followed by bath normalization (BN) and ReLU activation, and BNs are omitted to simplify the equation. Then the pairwise relations for all the nodes can be represented by Affinity Matrix $R_s \in \mathbb{R}^{N \times N}$.

For the i-th feature node, the relation vector can be denoted as:

$$\mathbf{r}_i = [R_s(i, :), R_s(:, i)] \in \mathbb{R}^{2N} \tag{2}$$

To learn the attention of the i-th feature node, they combine the pairwise relations $\mathbf{r}_i$ and the feature node itself to get the spatial relation-aware feature $\tilde{y_i}$:

$$\tilde{y_i} = [pool_c(\psi_s(\mathbf{x}_i)), \varphi_s(\mathbf{r}_i)] \in \mathbb{R}^{1+N/s_1}, \tag{3}$$

where ψ_s and φ_s are both 1 × 1 convolutional layer followed by BN and ReLU activation, $pool_c$ denotes global average pooling along the channel dimension to reduce the channel of the input tensor to 1.

For mining valuable information from the spatial relation-aware feature $\tilde{y_i}$, two 1 × 1 convolutional layer W_1 and W_2 followed by BN are implemented to get spatial attention value a_i:

$$a_i = Sigmoid(W_2 ReLU(W_1 \tilde{y_i})), \tag{4}$$

where W_2 decreases the channel with a ratio and W_2 shrinks the channel to 1 to save computation resources.

Channel Relation-aware Global Attention (RGA-C)
Similar to RGA-S, when RGA-C get the input tensor $X \in \mathbb{R}^{C \times H \times W}$, they take the $d = H \times W$-dimensional feature map at each channel as a feature node. So each feature node of RGA-C can be represented as $\mathbf{x}_i \in \mathbb{R}^d$, where $i = 1, 2, ..., C$.

Then the pairwise relation between i-th node and j-th node can be defined as:

$$r_{i,j} = f_s(\mathbf{x}_i, \mathbf{x}_j) = \theta_s(\mathbf{x}_i)^T \phi_s(\mathbf{x}_j), \tag{5}$$

where θ_s and ϕ_s are same to RGA-S. Then the pairwise relations for all the nodes can be represented by Affinity Matrix $R_c \in \mathbb{R}^{C \times C}$.

For the i-th feature node, the relation vector can be denoted as:

$$\mathbf{r}_i = [R_c(i,:), R_c(:,i)] \in \mathbb{R}^{2C} \tag{6}$$

Then follow the Eq. (3) and (4), we can get the channel relation-aware feature $\tilde{y_i}$ and channel attention value a_i.

3.3 Multi-level Integration and Loss Function

The proposed architecture employs a U-Net like architecture that symmetrically expands the feature maps after upsampling operations in the decoder. Information vanishing is inevitable Due to the max-pooling operation in the encoder. As shown in Fig. 1, three MI module are combined in the proposed architecture. The input feature map of MI is the feature map before the last three max-pooling operations. Every step of expansion is composed of an upsampling of the feature map and concatenation with the same scale feature map from the encoder. Additionally, three 3×3 convolutional layers with stride 1 padding 1 followed by ReLU are used in each MI.

For the loss function, we follow the paper [8]:

$$L(\hat{\mathbf{I}}, \mathbf{I}) = \alpha KLdiv(\hat{\mathbf{I}}, \mathbf{I}) + \beta CC(\hat{\mathbf{I}}, \mathbf{I}) + \gamma SIM(\hat{\mathbf{I}}, \mathbf{I}), \tag{7}$$

where $\hat{\mathbf{I}}$ and $\mathbf{I}$ are predicted saliency maps and the ground truth, α, β and γ are three coefficients take 10, -1, -1 followed [8].

4 Experiments and Analysis

In this section, we show the details of experiments and comparison results.

4.1 Experimental Setup

We use the largest available dataset SALICON [4] to train and test the proposed model. The dataset consists of 10,000 images for training, 5,000 images for validating and 5,000 images for testing. In this paper, we train the proposed model on SALICON datasets with 10,000 training images and use 5,000 images for validating. All experiments are conducted with the deep learning Pytorch framework. The model trains for 10epochs with the learning rate 1e-4 and reducing after every 3 epochs. The ADAM optimization algorithm is employed to train the whole network. The proposed model is trained with a batch size of 8 on one NVIDIA GeForce GTX 1080Ti GPU with 11 GB memory.

4.2 The contribution of each component

In Table 2, we compare the contribution of each component, where VGGM denotes the original encoder-decoder architecture without multi-level integration and RGA-SC module. VGGM+RGA(1) denotes the original encoder-decoder architecture with only one RGA-SC module which added after the final convolutional layer and symmetrical position. VGG+RGA(1)+MI denotes the previous model plus three multi-level integration. It is obviously that each component effectively enhance the network, we call the model combined with the original encoder-decoder architecture, RGA-SC module and multi-level integration module VGGRGA. Table 2 compare the performance when add different number of RGA-SC module. VGGRGA(n) means there are 2 × n RGA-SC modules in the architecture. VGGRGA(3) is shown as Fig. 1. We don't do the experiments of VGGRGA(4) and (5) is due to the limited computation memory. After all, we can find that even if add more RGA-SC module, the performance would not improve a lot from Table 2.

Table 1. Performance comparison of different version on validation set of Salicon-2017

Model	KLdiv↓	CC↑	SIM↑
VGGM	0.272	0.854	0.745
VGGM+RGA(1)	0.266	0.858	0.748
VGGM+RGA(1)+MI	**0.241**	**0.876**	**0.768**

The results are evaluated by three Evaluation metrics: Kullback-Leibler Divergence (KLdiv), Pearson Cross-Correlation (CC) and Similarity (SIM). Differently from the KLdiv loss which value should be minimized, the CC and the SIM loss is maximized to obtain the higher performance in saliency prediction.

4.3 Comparison with other work

VGGSSM [8] is a similar work which using self-attention as enhancing tool to improve the performance of network for saliency prediction task. They use self-

Table 2. Performance comparison of the proposed model with 1, 2, 3 RGA-SC module

Model	KLdiv↓	CC↑	SIM↑
VGGRGA(1)	0.241	0.876	0.768
VGGRGA(2)	0.239	0.876	0.768
VGGRGA(3)	**0.239**	**0.877**	**0.769**

attention after the final convolutional layer of encoder to enhance global relation among all the pixels in feature map. We compare the results with them in validation set of Salicon-2017 dataset. And the comparison results show the proposed model obtains competitive performance in saliency prediction. Although RGA module achieved better results in the contest between itself and self-attention module. It is likely that RGA module can stack more in the structure with the same computing source, which leads to better effect. But the advantage of RGA shows that it takes less memory in the calculation.

Table 3. Performance comparison with the proposed model and VGGSSM [8]

Model	KLdiv↓	CC↑	SIM↑
VGGSSM	0.251	0.876	0.769
VGGRGA(3)	**0.239**	**0.877**	**0.769**

4.4 Conclusions

In this paper, a saliency prediction model VGGRGA upon encoder-decoder architecture is proposed. The model integrates three important components and the experimental results demonstrate the efficiency of the components. Additionally, this paper also research the results with different numbers of RGA module. Furthermore, we would compare all the attention methods' contribution to saliency prediction task.

Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the government (MSIT).(No.2020R1A2C2008972)

References

1. B. Schauerte, J. Richarz and G. A. Fink, "Saliency-based identification and recognition of pointed-at objects," 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, 2010, pp. 4638-4643, doi: 10.1109/IROS.2010.5649430.

2. S. Frintrop and M. Kessel, “Most salient region tracking,” 2009 IEEE International Conference on Robotics and Automation, Kobe, 2009, pp. 1869-1874, doi: 10.1109/ROBOT.2009.5152298.
3. T. Saeko, R. Ramesh, G. Michael, G. Bruce. (2005). “Automatic image retargeting.” Proceedings of the 4th International Conference on Mobile and Ubiquitous Multimedia, MUM ’05. 154. 59-68. 10.1145/1149488.1149499.
4. M. Jiang, S. Huang, J. Duan and Q. Zhao, “SALICON: Saliency in Context,” 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, 2015, pp. 1072-1080, doi: 10.1109/CVPR.2015.7298710.
5. N. Reddy, S. Jain, P. Yarlagadda, V. Gandhi. (2020). “Tidying Deep Saliency Prediction Architectures.” abs/2003.04942
6. O. Ronneberger, P. Fischer, T. Brox: “U-Net: convolutional networks for biomedical image segmentation.” arXiv e-prints, arXiv:1505.04597 (2015)
7. Z. Zhang, C. Lan, W. Zeng, X. Jin and Z. Chen, “Relation-Aware Global Attention for Person Re-Identification,” 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020, pp. 3183-3192, doi: 10.1109/CVPR42600.2020.00325.
8. G. Cao, Q. Tang, K. Jo, “Aggregated Deep Saliency Prediction by Self-attention Network”, Springer International Publishing, 2020, pp. 87-97.
9. L. Itti, C. Koch and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis,” in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 11, pp. 1254-1259, Nov. 1998, doi: 10.1109/34.730558.
10. Neil D. B. Bruce and John K. Tsotsos, “Saliency, attention, and visual search: AN information theoretic approach,” Journal of vision, vol. 9, no. 3, pp. 5-5, 2009.
11. A. Borji, “Boosting bottom-up and top-down visual features for saliency estimation,” 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, 2012, pp. 438-445, doi: 10.1109/CVPR.2012.6247706.
12. M. Cornia, L. Baraldi, G. Serra and R. Cucchiara, “Predicting Human Eye Fixations via an LSTM-Based Saliency Attentive Model,” in IEEE Transactions on Image Processing, vol. 27, no. 10, pp. 5142-5154, Oct. 2018, doi: 10.1109/TIP.2018.2851672.
13. S. Woo, J. Park, J. Lee, I. Kweon, “CBAM: Convolutional Block Attention Module,” in Proceedings of the European Conference on Computer Vision (ECCV), Sept. 2018.
14. K. Simonyan, A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv e-prints, arXiv:1409.1556 (2014).