# **Global Context-Aware Progressive Aggregation Network for Salient Object Detection**



AAAI 2020

## **主要工作**

这篇文章和F3Net在想法上有很大的相似之处，都认为：the previous works mainly adopted multiple-level feature integration **yet ignored the gap between different features**.

而另一条there also exists a dilution process of high-level features as they passed on the top-down pathway实际上也是借用自之前的PoolNet。

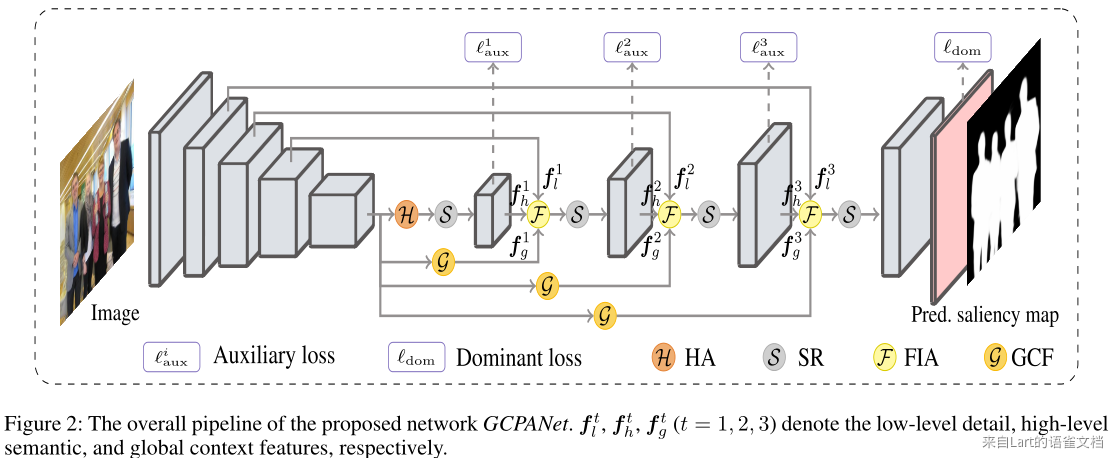
因此，总体而言，本文认为现有的基于FCN的模型存在这样的两个问题：

1. Due to **the gap between different level features**, the simple combination of semantic information and appearance information is insufficient and **lacks consideration of the different contribution of different features** for salient object detection;
2. Most of the previous works **ignored the global context information**, which **benefits for deducing the relationship among multiple salient regions** and producing more complete saliency result.

为了处理这两个问题，这里提出了几个模块：

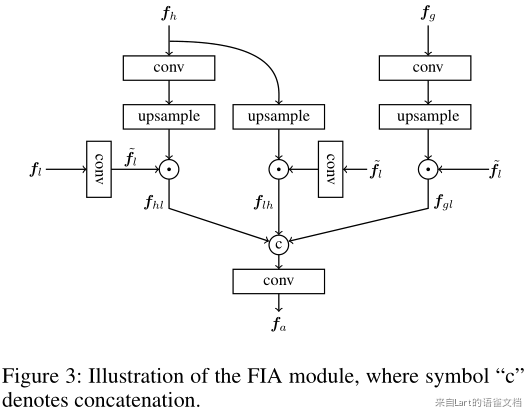
* 对于第一个问题：**Feature Interweaved Aggregation** (FIA) module fully integrates the high-level semantic features, low-level detail features, and global context features, which is expected to **suppress the noises but recover more structural and detail information**.
* 通用提升：
  + **Head Attention** (HA) module is used to **reduce information redundancy and enhance the top layers features** by leveraging the spatial and channel-wise attention
  + **Self Refinement** (SR) module is utilized to **further refine and heighten the input features**
* 对于第二个问题：**Global Context Flow** (GCF) module generates the global context information at different stages, which aims to **learn the relationship among different salient regions** and **alleviate the dilution effect of high-level features**

## **主体结构**



主要包含四个组件，这里分别简单介绍。

### **FIA**



这里多处使用乘法操作。The multiplication operation can strengthen the response of salient objects, meanwhile suppress the background noises. 从图中可以比较直观的了解整体的计算过程。注意，这里中间和右侧分支使用的输入时IMG_259而不是IMG_260，也就是左侧分支的中间特征。

整个模块的输入包含三个部分：

1. the high-level features from the output of the previous layer
2. the low-level features from the corresponding bottom layer
3. the global context feature generated by the GCF module

### **SR**

例如，在预测的显著物体上有一些洞，这是由不同层的矛盾反应引起的。因此，我们开发了一个 SR 模块，在通过 HA 模块和 FIA 模块后，通过使用乘法和加法操作来进一步细化和增强特征图。

SR本身很简单，就是一个三层的卷积结构。没啥好说的。

### **HA**

由于编码器组件的顶层特征对于显著目标检测通常是冗余的,我们设计了一个接在顶层后的 HA 模块，通过利用空间和通道注意机制来学习更具有选择性和代表性的特征。

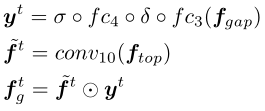
输入特征图IMG_261，先将通道调整成256，得到IMG_262然后使用简单的卷积结构得到第一阶段特征IMG_263。

之后再通过全局平均池化来处理IMG_264变成了通道级特征矢量IMG_265，后接两个全连接层，分别使用ReLU和Sigmoid作为激活函数，从而得到权重矢量IMG_266。

最终的输出使用IMG_267，即用IMG_268对IMG_269进行通道加权。

### **GCF**

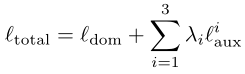
与PoolNet不同，这里考虑了不同阶段的不同贡献。首先使用全局平均池化获取全局上下文信息，然后**为每个阶段的全局上下文特征图的不同通道重新分配不同的权重**。要注意，这里的GCF按照图示，针对不同的阶段会设置不同的GCF。



这里的输出会被送到FIA的右侧分支作为输入。

### **损失函数**

使用的深监督形式下的交叉熵损失。



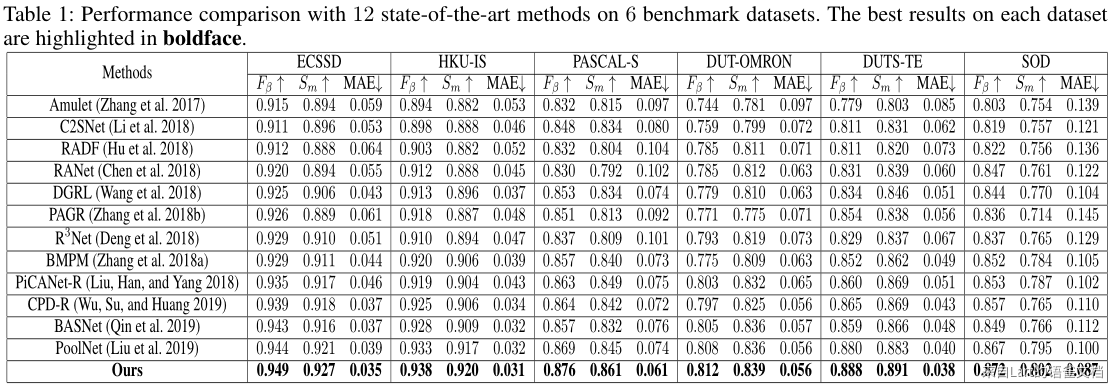
To facilitate the optimization of the proposed network, we add auxiliary loss at three decoder stages. Specifically, a **3×3 convolution operation is applied for each stage to squeeze the channel of the output**

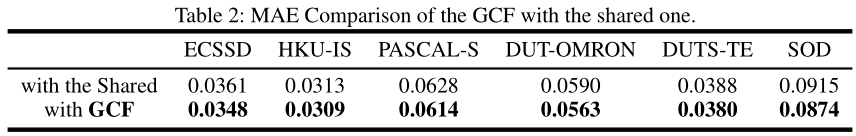
**feature maps to 1**. Then these maps are **up-sampled to the same size as the ground truth via bilinear interpolation** and sigmoid function is used to normalize the predicted values into [0,1].

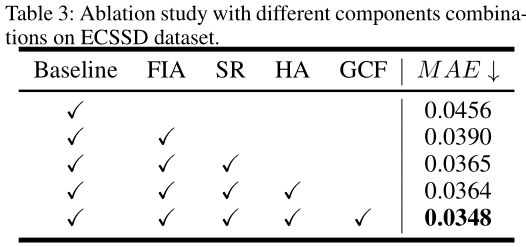
The auxiliary loss branches only exist during the training stage, whereas they are abandoned when inference.

## **实验细节**

* We adopt ResNet-50 (He et al. 2016) pretrained on ImageNet (Deng et al. 2009) as our network backbone.
* In the training stage, we **resize each image to 320×320 with random horizontal flipping, then randomly crop a patch with the size of 288 × 288 for training**.
* During the inference stage, images are simply **resized to 320 × 320** then fed into the network to obtain prediction without any other post-processing (e.g., CRF).
* We use Pytorch (Paszke et al. 2017) to implement our model.
* Mini-batch Stochastic gradient descent (SGD) is used to optimize the whole network with the **batch size of 32**, the momentum of 0.9, and the weight decay of 5e-4.
* We use the **warm-up and linear decay strategies** with the maximum learning rate **5e-3 for the backbone and 0.05 for other parts** to train our model and stop training after 30 epochs.
* The inference of a **320×320 image** takes about 0.02s (over 50 fps) with the acceleration of one NVIDIA Titan-Xp GPU card.







## **相关链接**

* 论文：[https://arxiv.org/pdf/2003.00651.pdf](https://arxiv.org/pdf/2003.00651.pdf" \t "_blank)
* 代码：[https://github.com/JosephChenHub/GCPANet](https://github.com/JosephChenHub/GCPANet.git" \t "_blank)