# 语义分割记录

叶亮

2021 年 6 月 23 日

# 目录



# 1 Basic mathematics

## 1.1 Normalization

Batch Normalization:

$$y = \frac{x - \mathrm{E}[x]}{\sqrt{\mathrm{Var}[x] + \epsilon}} * \gamma + \beta \tag{1}$$

$$\hat{x}_{\text{new}} = (1 - \text{momentum}) \times \hat{x} + \text{momemtum} \times x_t \tag{2}$$

### 1.2 Entropy

信息熵:

$$H(x) = -\sum_{x\in\chi} p(x)\log p(x) = \sum_{x\in\chi} p(x)\log\frac{1}{p(x)} \tag{3}$$

即事件发生概率的倒数的期望，熵越大代表事件发生的不可能性大，里面包含的信息量越大。

信息论解释：按照真实分布 p 来编码样本所需的编码长度的期望，信息熵 $H(p)$.

$$\sum_i p(i) * \log\frac{1}{p(i)} \tag{4}$$

交叉熵：按照不真实分布 q 来编码样本所需的编码长度的期望,$H(p,q)$

$$\sum_i p(i) * \log\frac{1}{q(i)} \tag{5}$$

引申出 KL 散度 $D(p||q) = H(p,q) - H(p)$, 它也叫相对熵，表示两个分部的差异，差异越大，相对熵越大。

$$\sum_i p(i) * \log\frac{p(i)}{q(i)} \tag{6}$$

$$D_K L(p||q) = \sum_{i=1}^{N} p(x_i)\cdot(\log p(x_i) - \log q(x_i)) \tag{7}$$

## 2 SF-segnet

分割任务中影响性能的两个重要因素：分辨率和语义表征能力。常用的方法有:

atrous convolution: 在后几个 stage 使用来保持高分辨率和较强的语义表征. 缺点：较大的计算能力和显存占用。

fpn-like network: 通过双边连接来融合 low-level 和 high-level 的特征，提高语义表征能力。

论文核心：fpn-like 的连接方式 ineffect. 提出学习 Semantic Flow between layers with different resolutions。即 Flow Alignment Module(FAM), 通过取相邻 level 的特征作为输入, 输出 offset field, and then warp the coarse feature to the fine feature with higher resolution according to the offset field. FAM 为即插即用式，可插入到任意 backbone 中，called **SFNet**. 灵感来源于光流。

在场景解析任务中，主要有两个范式 (paradigm) 用于高分辨率语义分割。1. 沿主路径 keep spatial and semantic information. 2. distributes spatial and semantic information on different parts in a network, then merges back via different strategies.

The first is atrous convolution. The second is fuse multi-level feature maps for both spatiality and semantics

### 2.1 Method

文章采用了 Encoder-decoder 的架构。其中，Encoder 部分为四个阶段的 backbone, 对应 stage1, 2, 3, 4. 步长分别为 4, 8, 16, 32. Decoder 部分为 FPN 的改进版。将之前的 top-down

部分的上采样相加模块替换成了 Flow Alignment module(FAM). 通过语义流的方式来计算上采样的插值。在 pytorch 中的实现为：通过 torch.grid_sample 来实现上采样差值运算。

# 3 BiSeNet

旷视提出，“BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation”

## 3.1 Introduction

提出双边分割网络, 空间路径: 用较小的 stride 来保留空间信息，并生成高分辨率特征图; 上下文路径: 快速下采样策略来得到足够的感受野。提出一个特征融合模块来融合两路特征。2048x1024, 68.4% mIoU on cityscape, 105 FPS Titan XP. 常用的加速方法如图 1所示。(a) 和 (b) 均会导致

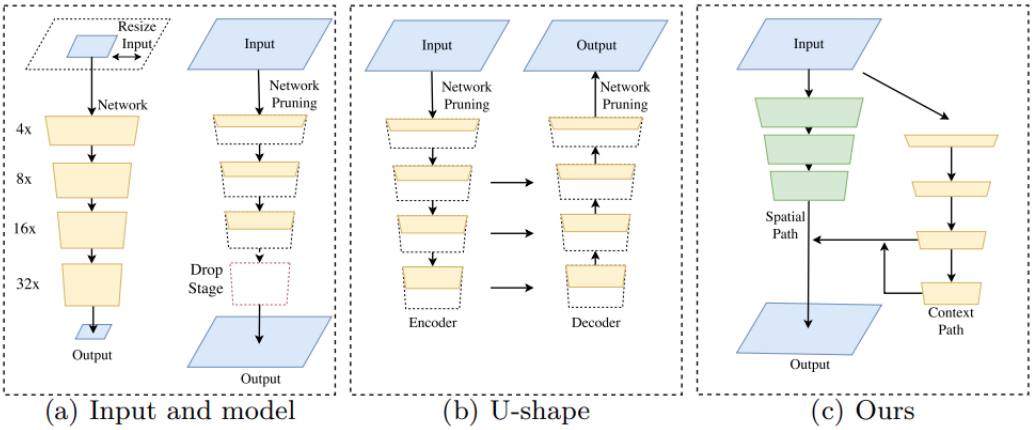


图 1: 不同方法比较.(a) 裁剪和 resize; (b) U-shape 结构;(c) BiSeNet

空间细节的损失。完整的 U-shape 结构其速度不理想。(c) 为提出的模型，对于空间路径，使用三个卷积层得到步长为 8 的特征图；上下文路径，则在 Xception 尾部加入全局平局池化。在此基础上，提出特征融合模块 (FFM) 和注意力改善模块 (ARM). 主要贡献如下:

We propose a novel approach to decouple the function of spatial information preservation and receptive field offering into two paths. Specifically, we propose a Bilateral Segmentation Network (BiSeNet) with a Spatial Path (SP) and a Context Path (CP).

We design two specific modules, Feature Fusion Module (FFM) and Attention Refinement Module (ARM), to further improve the accuracy with acceptable cost.

We achieve impressive results on the benchmarks of Cityscapes, CamVid, and COCO-Stuff. More specifically, we obtain the results of 68.4% on the Cityscapes test dataset with the speed of 105 FPS.

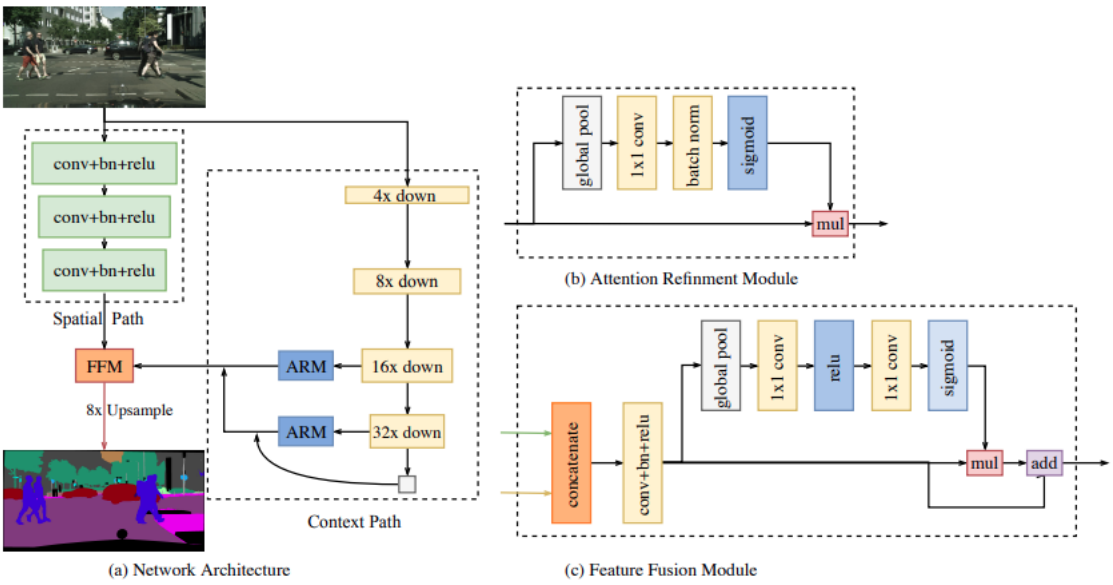


图 2: BiSeNet 网络结构.(a) 模块的长表示空间尺寸，厚度表示通道数; (b) ARM; (c) FFM

## 3.2 Method

### 3.2.1 Spatial path

使用三个卷积，每个卷积 $stride = 2$, 得到 1/8 特征图。

## 3.3 Context path

使用轻量网络，Xception 来快速下采样得到较大感受野的特征图，并在尾部加入全局平均池化。

ARM: 使用全局平均池化来得到上下文信息，并计算注意力向量来引导学习。

## 3.4 Network Architecture

其结构如图 2所示。使用预训练的 Xception 作为 backbone。

Loss: 添加了两个辅助 loss 帮助训练，所有 loss 均采用 softmax. 如公式 (1) 所示; 使用 $\alpha$ 来平衡主 loss 和辅助 loss 的权重，如公式 (2) 所示；在本文中，$\alpha$ 为 1，

$$loss = \frac{1}{N}\sum_{i} L_i = \frac{1}{N}\sum_{i} -\log\left(\frac{e^{p_i}}{e^{\sum_j p_j}}\right) \tag{8}$$

其中，$p$ 为网络预测输出.

$$L(X;W) = l_p(X;W) + \alpha\sum_{i=2}^{K} l_i(X_i;W) \tag{9}$$

其中,$X_i$ 为 Xception 的第 i 个 stage 的输出特征，在论文中 $K = 3$。

### 3.5 Experiment

Xception39 as backbone.

Training details: SGD with batch size 16, momentum 0.9, weight decay $1e^{-4}$, "poly" learning rate, learning rate = initial learning rate x $(1 - \frac{iter}{max_iter}^{power})$, where power=0.9, initial learning rate = $2.5e^{-2}$.

Data augmentation: mean subtraction, random horizontal flip, random scale0.75, 1.0, 1.5, 1.75, 2.0, and random crop into fixed size.

cityscape: Res-18 74.8% on val, 74.7% on test with FPS 65.5. Xception39, 69% on val, 68.4% on test with FPS 105.8

## 4 DDRNet

"Deep Dual-resolution Networks for Real-time and Accurate Semantic Segmentation of Road Scenes" [**?**]

Inspired by HRNet, 作者使用双分支网络，如图所示：

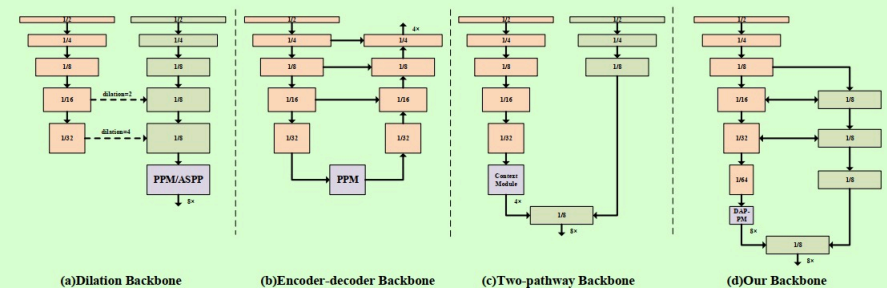


图 3: 不同架构的分割网络

### 4.1 相关工作

**高性能分割**

大多数 sota 网络使用 dilated backbone. Deeplabv3+ 使用更简单的 decoder 部分，通过将 low-level 和 high-level 的特征图融合进行预测，缓解了由 dilated conv 生成的高分辨率特征图的性能需求。HRNet 则是通过更注重高分辨率特征表示。

**实时语义分割**

所有的实时语义分割方法采用两种基本的架构：encoder-decoder 和 two-pathway 架构。

1) Encoder-decoder: 相比 dilated convolution, encoder-decoder 计算资源需求更少。通常 encoder 的输出步长为 32，经过 decoder 逐渐上采样到 1/4 or 1/8, 且通常融合 decoder 和 encoder 的对应步长的特征图来提升表征能力。因此，其对显存的需求更少。

2) Two-pathway: encoder-decoder 中的逐级下采样会损失 partial information, 且无法恢复。因此, two-pathway 架构提出, 通过在正常 encoder 之外, 新建一个 shallow pathway of high resolution

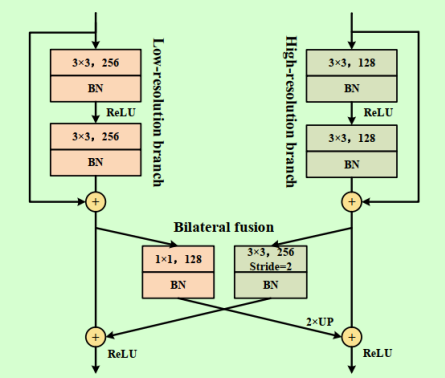

Fig. 3. The details of bilateral fusion in DDRNet. Summation fusion is implemented before ReLU.

图 4: bilateral 融合

来提供较丰富的空间信息。为了 trade-off, two-pathway 可以是较轻的 encoder 和较宽的 shallow branch. 在 BiSeNet 中，两个分支在一开始就进行分离。在 Fast-SCNN 中，两个分支使用相同的下采样。DDRNet 则在早期使用相同的下采样，并在之后的 stage 中交换信息。

## 4.2 Method

**A Rethinking HRNet** 语义分割需要高分辨率特征图来做 dense prediction, 以及较大的感受野来解析场景。而多尺度表征能力，对目标检测任务更有意义，因为网络需要在一张图上尽可能多地对不同尺度的目标进行检测。因此，HRNet 可以分为两个分支：1. 用于维持高分辨率特征图; 2. 用于生成较大感受野。本文通过优化，可以显著减少 HRNet 的显存消耗。

**B Dual-resolution for image classification** 主要介绍 DDR 的模型结构，包括 DDR-23 和 DDR-39。两者分别使用 Res-18 和 Res-34，通过修改开始的 7x7conv 为两个 3x3 conv，使用双边连接模块来做特征融合。如图 Fig .4

**C Deep Aggregation Pyramid Pooling Module** 如图 Fig .5所示:

**D Overall architecture for Semantic Segmentation** segmentaiton head channels set to 64,128,256 for DDR-23-slim, DDR-23, DDR-39. 与其他方法一样，采用 deep supervision 方法，在 stage 3 的阶段加入 auxiliary head，来辅助训练。如图 Fig .6所示：

## 4.3 EMANet

”Expectation-Maximization Attention Networks for Semantic Segmentation”.

self-attention mechanism(自注意力机制)，通过在特征图中所有位置加权求和的方式，来 capture long-range relations. 这篇论文中，将注意力机制表述成 expectation-maximazation(期望最大化) manner 并 iteratively estimate a much more compact set of bases upon which the attention maps are computed. 主要贡献：

- 将自注意机制定义为 EM iteration manner,which can learn a more compact basis set and largely reduce the computational complexity.

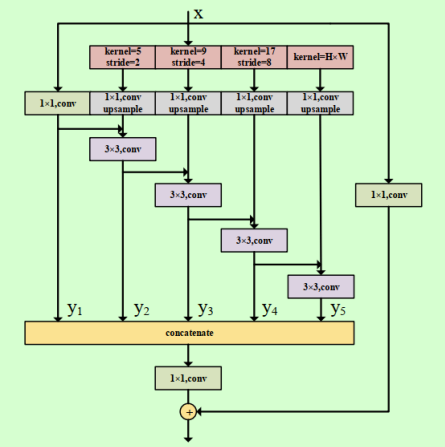


Fig. 5. The detailed architecture of Deep Aggregation Pyramid Pooling Module. The number of multi-scale branches can be adjusted according to input resolution.

图 5: DAPPM

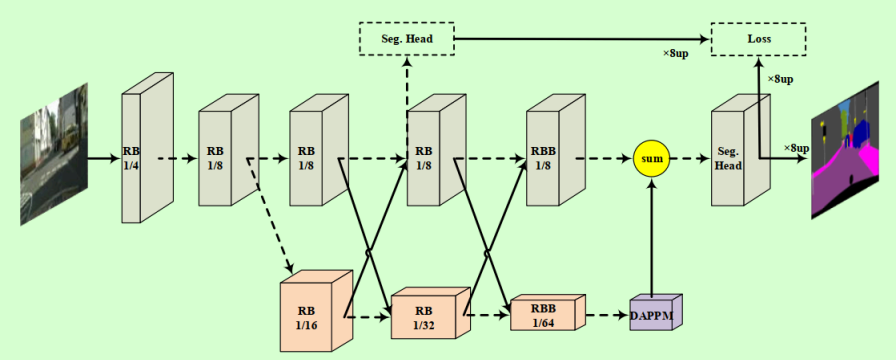


Fig. 4. The overview of DDRNets on semantic segmentation. "RB" denotes sequential residual basic blocks. "RBB" denotes single residual bottleneck block. "DAPPM" denotes the Deep Aggregation Pyramid Pooling Module. "Seg. Head" denotes the segmentation head. Black solid lines denote information paths with data processing (including upsampling and downsampling) and black dashed lines denote information path without data processing. "sum" denotes pointwise concatenation. Dashed boxes denote the components which are disregarded in the inference stage.

图 6: 分割架构

- proposed EMA module and set up specific manners for bases' maintenance and normalization.
- Extensive experiments

## 4.4 相关工作与知识点

### 4.4.1 Expectation-Maximization Algorithm

EM aims to find the maximum likelihood solution for latent variable models. Denote $X = x_1, x_2, ..., x_N$ 作为观测样本, 每个 $x_i$ 有对应的 latent variable$z_i$. 将 $X, Z$ 表示完整数据 (complete data), 似然函数的形式为 $\ln p(X, Z|\theta)$, where $\theta$ 表示模型的所有参数。latent variable $Z$ is given by 后验分布 $p(Z|X, \theta)$. EM 算法通过两步来最大化似然函数 $\ln p(X, Z|\theta)$, i.e., E and M.