复现主题：实时语义分割模型的复现与优化

1. 背景介绍

1.1 背景

实时语义分割的研究背景是计算机视觉领域中对图像中每个像素进行分类以理解图像内容的关键任务，它在自动驾驶、医学图像分析和环境监测等多个应用领域中发挥着重要作用。随着深度学习技术的发展，尤其是卷积神经网络的进步，语义分割的性能得到了显著提升。然而，在自动驾驶等应用中，模型需要能够快速且准确地分析图像以实现实时决策，这就要求语义分割模型不仅要有高准确度，还要有快速的推理速度。

1.2 发展现状

实时语义分割模型致力于提升分割的速度与准确性。但单纯通过调整模型的深度、宽度和结构来追求速度提升，可能会损害性能表现。因此，必须采取多种策略来实现速度与性能之间的平衡。依据模型的结构设计，主要可分为两类：单分支架构和多分支架构。

单分支架构：这些模型通常通过优化骨干网络模块或设计轻量级的分割头来兼顾性能与速度。尽管多分支模型以其出色的性能和速度带来了挑战，但单分支模型仍在不断发展。例如，STDC[2]通过引入细节聚合模块来解决 BiSeNet 中分支添加导致的延迟问题，该模块采用单分支方式来保持空间信息。SCTNet[7]则利用知识蒸馏技术，以 SegFormer[9]作为教师模型，通过 Conv-Former 块来连接 CNN 和变换器特征之间的语义差异，更有效地学习丰富的语义信息。

多分支架构：单分支架构模型通过跳跃连接在骨干网络和分割头之间传递特征图，以保持空间细节。而多分支模型则通过增加额外的分支来同时学习空间细节和边界信息。BiSeNetV1[3]和 V2[4]提出了一个二分支架构，一个分支专注于深层语义信息，另一个分支关注空间细节，且两个分支之间不共享权重。与此相对，Fast-SCNN[8]和 DDRNet[5]等模型则共享一些低层的骨干网络权重。Fast-SCNN 提取浅层特征并将其分配到两个分支中，一个分支负责保持特征，另一个分支负责提取全局特征，而 DDRNet 则通过多次双边融合来高效整合信息。PIDNet[6]则进一步引入了三分支架构，以捕捉空间细节、深层语义信息和边界信息。

1.3 本文复现模型

本文选取了不同架构中的典型分割模型进行复现。具体模型如下：

- 单分支架构模型：CGNet[1]、STDC[2]
- 二分支架构模型：BiSeNetV1[3]、BiSeNetV2[4]、DDRNet[5]
- 三分支架构模型：PIDNet[6]

2. 方法框架

2.1 输入输出

输入：输入通常是一张图像，这张图像可以是来自各种场景，如城市街道、室内环境、自然环境等。图像的格式通常是 RGB 图像，即包含红色、绿色和蓝色三个颜色通道。
输出：输出是一个与输入图像尺寸相同的图像，称为分割图。在分割图中，每个像素点的值代表该像素所属的类别标签。

2.2 单分支架构模型-STDC

STDC（Short-Term Dense Concatenate network）是一种单分支的网络结构，专为实时语义分割任务设计，通过去除结构冗余来提升效率和性能。它的核心是 STDC 模块，该模块通过逐步降低特征图的维度并聚合这些特征图来形成图像表示，从而获取具有可伸缩接受场和多尺度信息的深度特征。STDC 网络采用 U 型结构，编码器部分由多个 STDC 模块组成，逐步降低空间分辨率，而解码器部分则通过上采样和特征融合恢复高分辨率的分割结果。在解码阶段，STDC 网络提出了细节聚合模块，将空间信息的学习整合到低层的单流方式中，使得网络能够在推理时更精确地保留空间细节。STDC 网络将低层特征和深层特征融合，以预测最终的语义分割结果，这种融合有助于结合低层的细节信息和高层的语义信息，提高分割的准确性。STDC 网络在设计时注重计算成本和推理速度，通过优化模块结构和减少参数数量来实现高效的语义分割。

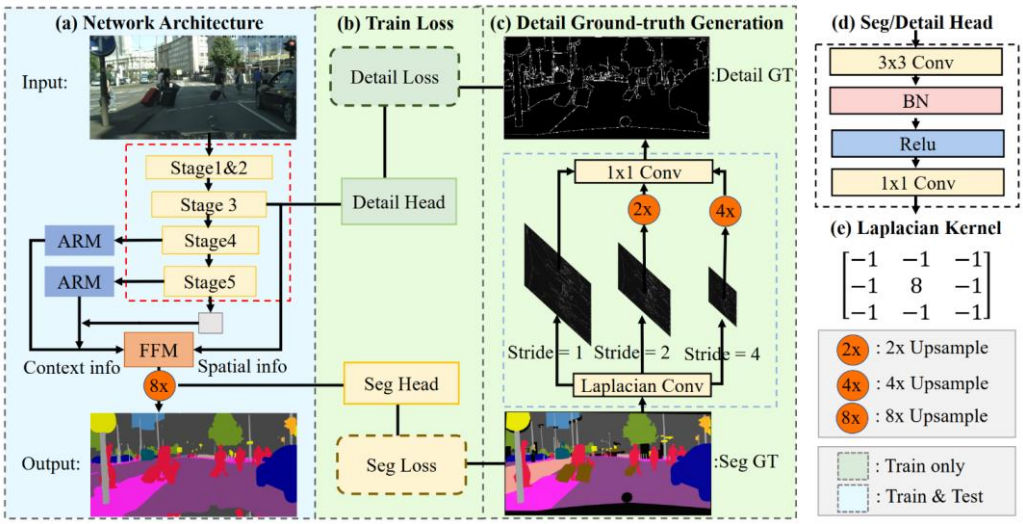

图 1　STDC 架构图

2.3 二分支架构模型-DDRNet

DDRNet（Deep Dual-resolution Networks）是一种为实时语义分割任务设计的深度学习网络架构。它由两个深度分支组成，这两个分支之间执行多次双边融合，以提高特征信息的整

合效率。DDRNet 的一个关键特点是其高分辨率分支生成相对较高分辨率的特征图，而另一个分支则通过多次下采样操作提取丰富的语义信息。这种设计允许网络在保持高分辨率表示的同时，也能捕获高级的上下文信息。此外，DDRNet 还引入了一个名为 Deep Aggregation Pyramid Pooling Module（DAPPM）的新模块，该模块输入低分辨率的特征图，提取多尺度上下文信息，并以级联的方式合并它们，从而在几乎不影响推理时间的情况下扩大有效的感受野并融合多尺度上下文。

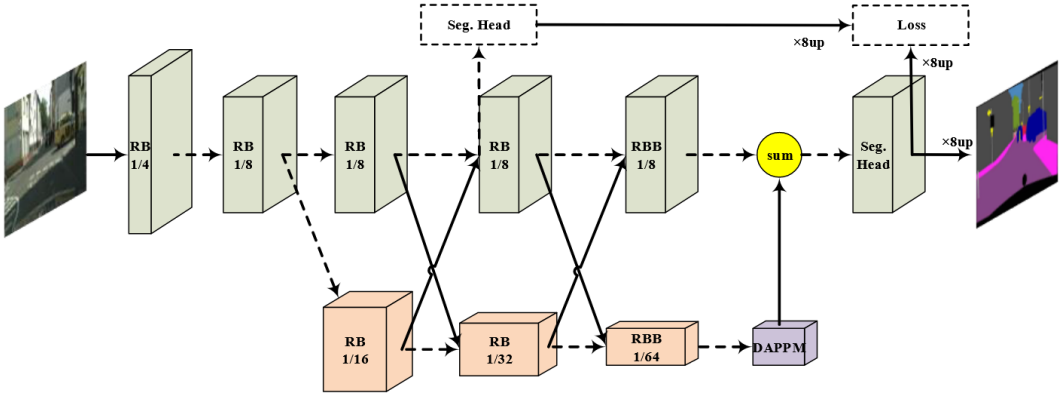

图 2　DDRNet 架构图

2.4 三分支架构模型-PIDNet

PIDNet 是一种三分支的实时语义分割网络架构，灵感来源于比例-积分-微分（PID）控制器。它通过将传统的二分支网络（处理细节和上下文信息）扩展为三个分支来解决直接融合高分辨率细节和低频上下文信息时细节特征被上下文信息淹没的问题，即所谓的过冲现象。PIDNet 的三个分支分别负责解析细节、上下文和边界信息，并通过边界注意力引导细节和上下文分支的融合。这种设计使得 PIDNet 在保持实时处理速度的同时，能够超越其他具有相似推理速度的现有模型，实现更高的分割精度。PIDNet 通过引入像素注意力引导融合模块（Pag）和边界注意力引导融合模块（Bag），以及针对边界检测的额外损失函数，增强了模型的性能。此外，PIDNet 还采用了并行聚合金字塔池化模块（PAPPM）来提高上下文信息的聚合效率。PIDNet 的设计不仅提高了分割的准确性，还保持了模型的推理速度，使其成为实时语义分割领域中的一个强有力的竞争者。

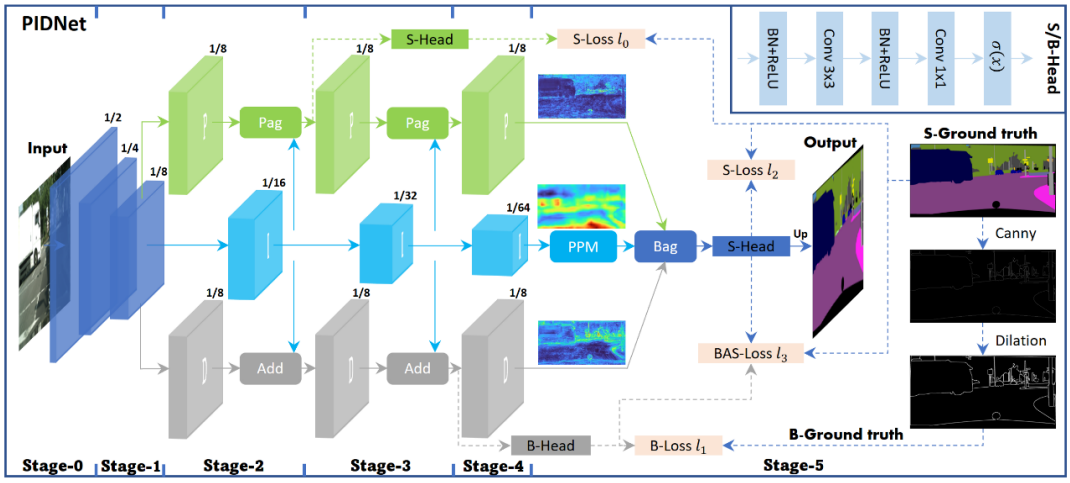

图 3　PIDNet 架构图

3. 实验结果

3.1 实验环境

我们的实验基于 MMSegmentation 框架，对多个模型进行了复现。实验环境如下：

- Python 版本：3.8.10
- PyTorch 版本：1.12.1
- Torchvision 版本：0.13.1
- MMEngine 版本：0.7.3
- MMCV 版本：2.0.0
- MMSegmentation 版本：1.0.0

3.2 实验过程

(1) 安装 MMSegmentation

```
第 1 步：使用 MIM 安装 MMCV
pip install -U openmim
mim install mmengine
mim install "mmcv>=2.0.0"

第 2 步：安装 MMSegmentation
git clone -b main https://github.com/open-mmlab/mmsegmentation.git
cd mmsegmentation
pip install -v -e .
```

(2) 验证安装是否成功

```
第 1 步：下载配置和权重文件
mim download mmsegmentation --config ddrnet_23-slim_in1k-pre_2xb6-120k_cityscapes-
```

1024x1024 --dest .

下载完成后，得到两个文件：

ddrnet_23-slim_in1k-pre_2xb6-120k_cityscapes-1024x1024.py

ddrnet_23-slim_in1k-pre_2xb6-120k_cityscapes-1024x1024_20230426_145312-6a5e5174.pth

第 2 步：验证推理演示

```
python ./mmsegmentation/demo/image_demo.py \
./mmsegmentation/demo/demo.png \
ddrnet_23-slim_in1k-pre_2xb6-120k_cityscapes-1024x1024.py \
ddrnet_23-slim_in1k-pre_2xb6-120k_cityscapes-1024x1024_20230426_145312-6a5e5174.pth \
--device cuda:0 \
--out-file result.jpg
```

(3) 复现模型

第 1 步： 准备数据集

访问 Cityscapes 数据集的官方网站（https://www.cityscapes-dataset.com/）下载所需的数据集，并将其解压至项目目录中的 dataset 文件夹下。

第 2 步：下载配置文件

```
mim download mmsegmentation --config ddrnet_23-slim_in1k-pre_2xb6-120k_cityscapes-1024x1024 --dest .
```

下载完成后，得到配置文件：

ddrnet_23-slim_in1k-pre_2xb6-120k_cityscapes-1024x1024.py

第 3 步：开始训练

```
CUDA_VISIBLE_DEVICES=0,1 bash \
./mmsegmentation/tools/dist_train.sh \
ddrnet_23-slim_in1k-pre_2xb6-120k_cityscapes-1024x1024.py \
2 \
--work-dir ./weight/seg
```

3.3 论文结果

表 1 论文结果

模型	GPU	ImageNet	速度（FPS）	性能（mIoU）
单分支架构模型				
CGNet	V100	No	31.14	68.3
STDC1	V100	Yes	23.06	74.9
STDC2	V100	Yes	23.71	76.7
二分支架构模型				
BiSeNetV1	V100	Yes	31.8	74.4

BiSeNetV2	V100	No	31.8	73.6
DDRNet-23-Slim	A100	Yes	85.9	77.84
DDRNet-23	A100	Yes	33.4	80.0
三分支架构模型				
PIDNet-S	A100	Yes	80.8	78.7
PIDNet-M	A100	Yes	72.0	80.2
PIDNet-L	A100	Yes	60.1	80.9

表 1 展示了基于 Cityscapes 数据集的实验结果，揭示了单分支模型在性能上的不足，且其速度优势并不显著。考虑到不同模型在不同 GPU 上的速度测试会产生不公平的结果，本文在复现实验时统一采用了相同的 GPU 进行速度测试，以确保结果的公正性和可比性。

3.4 复现结果

表 2　复现结果

模型	**GPU**	**ImageNet**	**速度（FPS）**	**性能（mIoU）**
单分支架构模型				
CGNet	RTX3090	No	54.9	68.1
STDC1	RTX3090	No	98.4	71.8
STDC2	RTX3090	No	74.7	74.9
二分支架构模型				
BiSeNetV1	RTX3090	No	65.9	74.4
BiSeNetV2	RTX3090	No	74.4	73.6
DDRNet-23-Slim	RTX3090	No	131.7	76.3
DDRNet-23	RTX3090	No	54.6	78.0
三分支架构模型				
PIDNet-S	RTX3090	No	102.6	76.4
PIDNet-M	RTX3090	No	42.0	78.2
PIDNet-L	RTX3090	No	31.8	78.8

表 2 呈现了复现结果。在这一过程中，本文刻意没有使用 ImageNet 预训练权重，目的是为了评估在缺乏预训练权重支持时，模型性能会受到多大程度的影响。实验结果显示，大多数模型的性能都有所下降，而 BiSeNetV1 却是个例外，其性能并未降低。此外，本文还利用 RTX3090 显卡对各模型进行了测试，进一步证实了单分支模型在性能上的不足，并且它们的速度优势并不如预期中那么明显。从整个结果来看，DDRNet 在性能和速度之间取得了更好的平衡。此外，本文还对 DDRNet 和 PIDNet 的预测结果进行了可视化分析，如图 4 所示。观察结果显示，两个模型在分割效果上的差异并不显著。

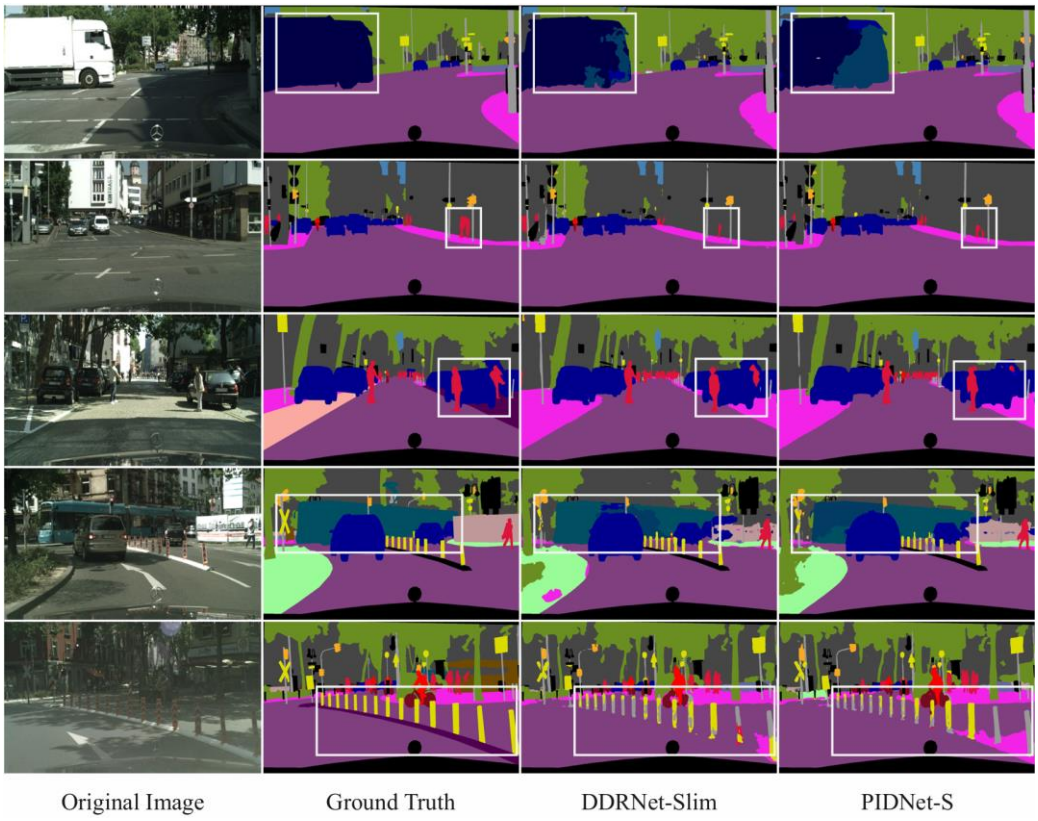

图 4　模型分割结果可视化

4. 讨论与改进思路

4.1 先前方法的不足

尽管现有的实时语义分割模型在准确性和速度之间取得了不错的平衡，但它们的多路径 Block 仍然影响整体速度。多路径块在训练中表现出色，但在推理阶段并不适合，因为它们增加了计算成本和内存使用，最终影响了推理速度。

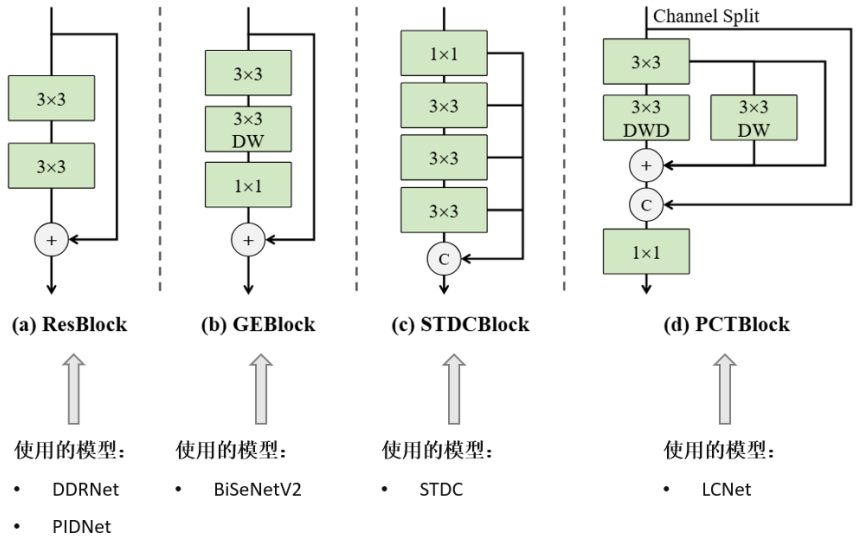

图 5　不同模型使用的 Block

4.2 改进思路

可重参数 Block（RB）：在训练时使用多路径来增强模型的性能，在推理时重参数化为单路经来增加模型的速度，这种转化不会损失性能，如图 6 所示。

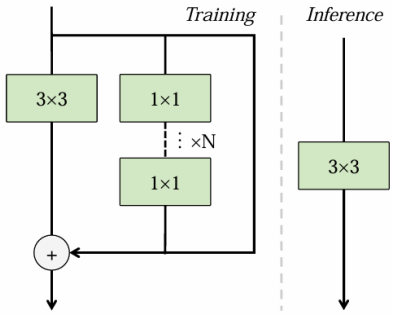

图 6　可重参数化 Block

4.3 改进结果

基于二分支架构，本文用提出的 RB 构建了网络，并且将该网络命名为 Reparameterizable Dual-Resolution Network (RDRNet)，如图 7 所示。

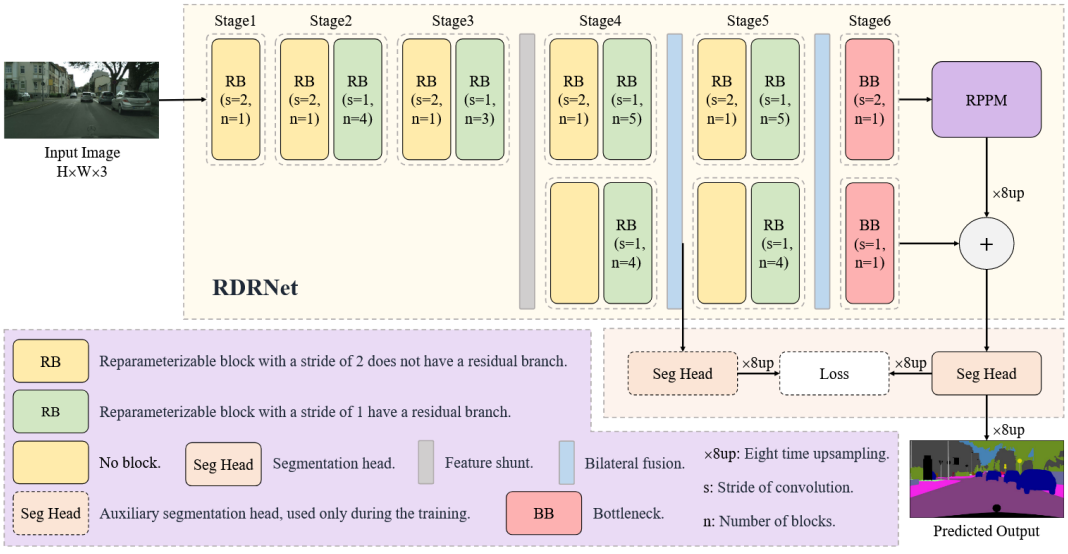

图 7　RDRNet 架构图

4.4 实验结果

表 3　RDRNet 实验对比

模型	GPU	ImageNet	速度（FPS）	性能（mIoU）
单分支架构模型				
CGNet	RTX3090	No	54.9	68.1
STDC1	RTX3090	No	98.4	71.8
STDC2	RTX3090	No	74.7	74.9

二分支架构模型				
BiSeNetV1	RTX3090	No	65.9	74.4
BiSeNetV2	RTX3090	No	74.4	73.6
DDRNet-23-Slim	RTX3090	No	131.7	76.3
DDRNet-23	RTX3090	No	54.6	78.0
三分支架构模型				
PIDNet-S	RTX3090	No	102.6	76.4
PIDNet-M	RTX3090	No	42.0	78.2
PIDNet-L	RTX3090	No	31.8	78.8
RDRNet 基于二分支架构				
RDRNet-S	RTX3090	No	129.7	76.8
RDRNet-M	RTX3090	No	52.5	78.9
RDRNet-L	RTX3090	No	39.0	79.3

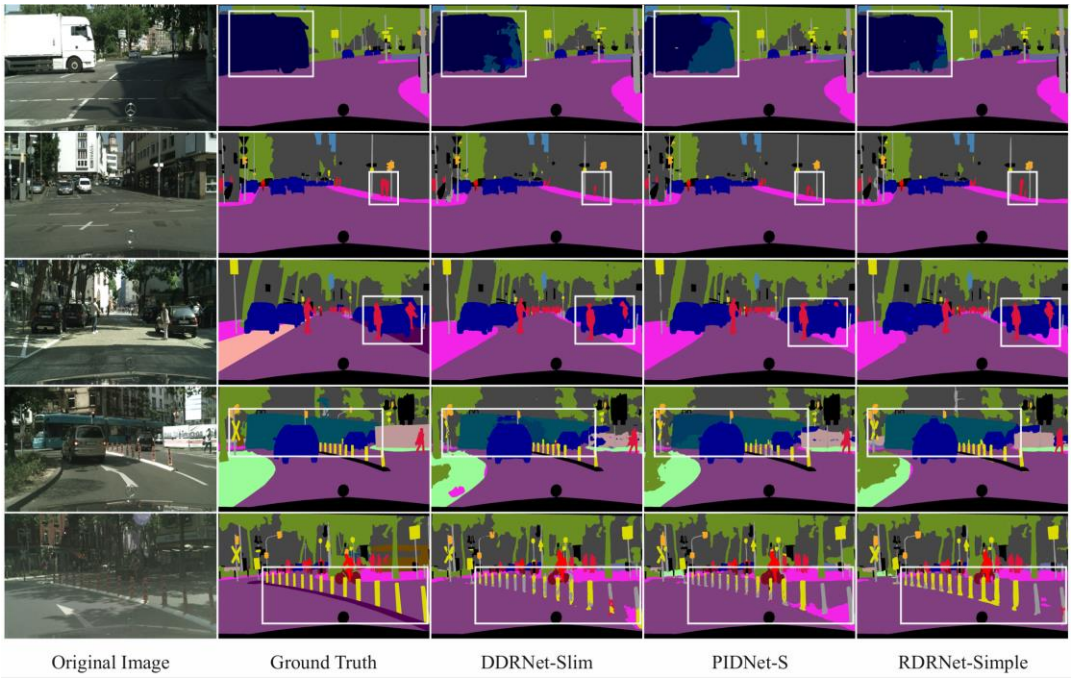

图 8　RDRNet 可视化

从表 3 中可以明显看出，RDRNet 在性能与速度之间实现了更佳的平衡，并且在分割效果上超越了 DDRNet 和 PIDNet。

4.5 论文发表

目前本文所讨论的改进模型（RDRNet）已经被整合成论文发布在了 arxiv 上，并且代码也被发布在了 github 上。

论文地址：https://arxiv.org/abs/2406.12496

代码地址：https://github.com/gyyang23/RDRNet 15stars

参考文献：

[1] Wu, Tianyi, et al. "Cgnet: A light-weight context guided network for semantic segmentation." IEEE Transactions on Image Processing 30 (2020): 1169-1179.

[2] Fan, Mingyuan, et al. "Rethinking bisenet for real-time semantic segmentation." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021.

[3] Yu, Changqian, et al. "Bisenet: Bilateral segmentation network for real-time semantic segmentation." Proceedings of the European conference on computer vision (ECCV). 2018.

[4] Yu, Changqian, et al. "Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation." International journal of computer vision 129 (2021): 3051-3068.

[5] Pan, Huihui, et al. "Deep dual-resolution networks for real-time and accurate semantic segmentation of traffic scenes." IEEE Transactions on Intelligent Transportation Systems 24.3 (2022): 3448-3460.

[6] Xu, Jiacong, Zixiang Xiong, and Shankar P. Bhattacharyya. "PIDNet: A real-time semantic segmentation network inspired by PID controllers." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023.

[7] Xu, Zhengze, et al. "SCTNet: Single-Branch CNN with Transformer Semantic Information for Real-Time Segmentation." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 38. No. 6. 2024.

[8] Poudel, Rudra PK, Stephan Liwicki, and Roberto Cipolla. "Fast-scnn: Fast semantic segmentation network." arxiv preprint arxiv:1902.04502 (2019).

[9] Xie, Enze, et al. "SegFormer: Simple and efficient design for semantic segmentation with transformers." Advances in neural information processing systems 34 (2021): 12077-12090.