

基于空间特征聚合的车道线检测算法^①

叶 伟, 朱 明

(中国科学技术大学 信息科学技术学院, 合肥 230026)

通讯作者: 叶 伟, E-mail: ywjxnf@mail.ustc.edu.cn



摘 要: 车道线检测是无人驾驶任务中最重要的模块之一. 由于车道线具有独特的结构, 且容易受到各种各样复杂环境 (比如光线、遮挡、模糊等) 的影响, 因此车道线检测也是一项很具有挑战性的任务. 传统的卷积神经网络 (CNN) 难以直接学习到精细的车道线空间特征, 本文使用空间特征聚合模块对 CNN 提取的特征在空间维度进行融合增强, 为级联的车道线预测器提供了丰富的空间特征信息. 实验证明, 空间特征聚合模块通过聚合水平和垂直方向的特征图获取精细的全局信息, 在多种复杂环境下都能提升车道线检测算法的性能, 且不会影响检测的速度.

关键词: 车道线检测; 空间特征聚合; 卷积神经网络

引用格式: 叶伟, 朱明. 基于空间特征聚合的车道线检测算法. 计算机系统应用, 2021, 30(12): 235-242. <http://www.c-s-a.org.cn/1003-3254/8194.html>

Lane Detection Algorithm Based on Spatial Feature Aggregation

YE Wei, ZHU Ming

(School of Information Science and Technology, University of Science and Technology of China, Hefei 230026, China)

Abstract: Lane detection is one of the most important modules in self-driving tasks. Lane detection is a challenging task as the structure of the lane line is special, and the detection is easily affected by various environments (such as lighting transformation, obstruction, and the blur of the lane line). Considering the traditional Convolutional Neural Network (CNN) is unable to learn fine spatial features of the lane line directly, in this study, the spatial feature aggregation module is employed to enhance the features extracted by CNN in spatial dimensions, providing rich spatial features for the cascade lane predictor. The experiments show that the module learns fine global information by aggregating feature maps in horizontal and vertical directions and thus improves the performance of the lane detection algorithm in different environments without reducing the detection speed.

Key words: lane detection; spatial feature aggregation; Convolutional Neural Network (CNN)

近年来, 随着人工智能技术的蓬勃发展, 自动驾驶成为最受欢迎的研究方向之一, 很多研究学者在此方向做出了突出的贡献. 车道线检测作为无人驾驶中是最重要的模块之一, 它可以为无人驾驶系统提供准确的位置信息, 使得车辆在行驶过程中不会偏移规定的车道标记线, 保证人车的安全. 目前, 在简单场景中车道线检测算法的性能是能够接受的. 但是, 在自然环境

下存在许多不利条件, 例如其他车辆对车道线的遮挡、恶劣的天气条件、长时间被碾压导致的车道线模糊等等, 使得车道线检测成为具有挑战性的工作.

目前的车道线检测算法主要划分为两种: 基于传统特征提取的方法^[1,2] 和基于深度学习的方法^[3]. 传统的检测算法利用车道线本身的结构信息, 借助人设计的低级特征 (颜色特征^[4]、线分割特征^[5]、霍夫变换

① 基金项目: 安徽省 2019 年重点研究与开发计划 (201904a05020035)

Foundation item: Year 2019, Key Research and Development Program of Anhui Province (201904a05020035)

收稿时间: 2021-02-18; 修改时间: 2021-03-18; 采用时间: 2021-03-26

特征^[6]等)识别车道线.这种方法可以提取出简单环境下的车道线,但是面对复杂的场景鲁棒性较差.而深度学习方法通过多层卷积学习图像的高级特征,然后对特征进行分类完成检测任务.这种方法由于需要大量的数据集进行学习,对各种道路环境兼容性较好.

通常情况下,基于CNN的方法都是把车道线检测作为语义分割^[7]的任务进行逐像素的分类.但是由于车道线的长且窄的特性,车道线的像素数量远小于背景的像素数量,这种类别不平衡会导致分类的准确率大打折扣.同时,普通的语义分割方法并没有考虑到车道线的空间相关性,在提取特征阶段无法学习车道线的精细特征,导致检测的准确性进一步降低.此外,车道线容易受到各种环境(如光线变化,遮挡,路面损坏等)的影响导致图像中车道线的缺失,由于普通的卷积神经网络只能提取到局部信息,无法处理这种中断的情况,影响实际的检测效果.

为了解决以上问题,大量的车道线检测算法被提出,其中一个比较经典的网络结构SCNN^[8],它将传统的卷积层接层的连接形式的转为特征图中片连片(按行或者列切片)卷积的形式,使得图中像素相邻行和列之间能够传递信息.这特别适用于检测长距离连续形状的目标或大型目标,有着极强的空间关系但是外观线索较差的目标.但是其信息顺序传递的方式十分耗时,并且行列之间长距离的信息传递可能会丢失特征图中的原有信息.

基于SCNN网络,本文提出了特征聚合的模块来学习特征图中的空间相关性并且能够更直接有效地传递像素之间的信息.该方法依然是将特征图按照行或者列切片,被切片的特征图经过一维行卷积(或者列卷积)与非线性激活,之后特征图向某个方向(行切片向上或者向下,列切片向左或者向右)平移固定步长得到一个新的特征图,最后两个特征图做基于下标方式的相加.每一次迭代的时候都是将特征图从4个方向(向上、向下、向左、向右)偏移聚合,经过多次迭代实现水平以及垂直方向上的特征聚合,使得特征图中每个像素都能获取整个特征图的空间信息.

1 相关工作

1.1 车道线检测

深度学习的方法有较强的兼容性,通过端到端的

方法能够有效学习车道线的特征. Huval 等人^[9]首次将CNN方法应用于车道线检测. Lee 等人^[10]提出了一种利用车道消失点同时检测车道和路标的多任务CNN. Pizzati 等人^[11]级联实例分割网络和分类网络,以检测带有类型的车道. Philion^[12]采用ResNet50^[13]自动回归车道线的曲线表示,而不是对车道进行分割,从而可以检测道路上任意数量的车道. De Brabandere 等人^[14]提出了一种通用的实例分割网络,通过聚类快速后处理的特征达到检测车道线的效果. 该网络通过判别损失函数学习到一个从图像空间到特征空间的映射,该映射满足同一实例的像素在特征空间中距离较近,否则距离较远的条件. Neven 等人^[15]提出了一种复杂的网络,包括车道分割子网络、像文献^[14]那样的像素嵌入子网络和透视变换网络. 采用迭代聚类所预测出的车道掩码和特征得到车道实例. 最后,对所学的透视变换中的每个车道实例拟合一个三阶多项式,但是没有充分利用特征图的空间信息,对于一些复杂的车道线检测效果欠佳.

Pan 等人^[8]设计了一种称为空间CNN(SCNN)的特殊层来分割道路车道线. SCNN是一种特殊的3D操作,有助于信息沿着行/列传递,并将感受野扩大到整个图像. 这对于车道识别是有用的,因为有些车道可能会充斥在整个图像中,但是SCNN是在同一层相邻的片之间进行信息传递,这种顺序的信息传递方法使其计算花费较大,并且可能随着传播距离的延长,导致信息的丢失.

Hou 等人^[16]提出了一种新的模块自注意力蒸馏(SAD),让ENet编码器^[17]学习两个相邻ENet编码器之间的自我注意,并分割出固定的车道线数量. 由于只在训练的时候使用SAD,因此推理时间较少,但是对于特征图的空间信息利用不足.

为了降低计算量,加快检测速度, Yoo 等人^[18]将车道线检测问题转换为逐行的分类任务,并以一种端到端的方法执行预测. Qin 等人^[19]提出的基于先验结构的深度车道线检测方法,速度非常快,其将车道线的检测问题视为逐行的分类问题,并提出了结构损失去利用车道线的结构模型. 但是由于没有有效利用高级特征的全局信息,对于特殊环境(遮挡、低光照等)下的车道线检测效果并不是很理想.

由于车道线细长的结构特性,关于图像的上下文空间信息的使用是很必要的,并且许多网络也尝试利用空间信息来提升网络性能.在文献[20]中,长短期记忆网络被应用于处理车道线结构的长线性检测. Bell等人^[21]探索循环神经网络(RNN)的运用.循环神经网络能够有效收集上下文信息.

1.2 本文工作

首先,本文基于SCNN模块,提出了空间特征聚合模块. SCNN模块将特征图按行或者列切成特征片,每个特征片依次进行信息的传递,因而比较耗时.本文的空间特征聚合模块以并行的方式传递信息,首先将特征图做卷积操作,然后向指定方向平移指定距离形成新的特征图,再将新的特征图与原始特征图相加得到最终结果,实现了信息传递的并行性,因此相对于SCNN计算代价更低. SCNN模块只沿4个方向迭代各一次,

信息传递的步长固定,且其在某一个方向长距离信息传递的过程中,原有特征图的信息容易丢失.本文的空间特征聚合模块在4个方向各迭代 K 次,每一次信息传递的步长都可变,每个方向的运行过程中都会保留原有特征图的信息.其次,在级联的预测器部分采用文献[19]的思路,减少逐像素分割的计算量.能够在提升检测效果的同时满足检测的实时性.

2 算法设计与实现

本章介绍完整的网络结构图以及空间特征聚合模块.

2.1 网络架构

整体的网络架构如图1所示.网络架构主要由3个模块组成,分别是:特征提取器、空间特征聚合模块、预测器.

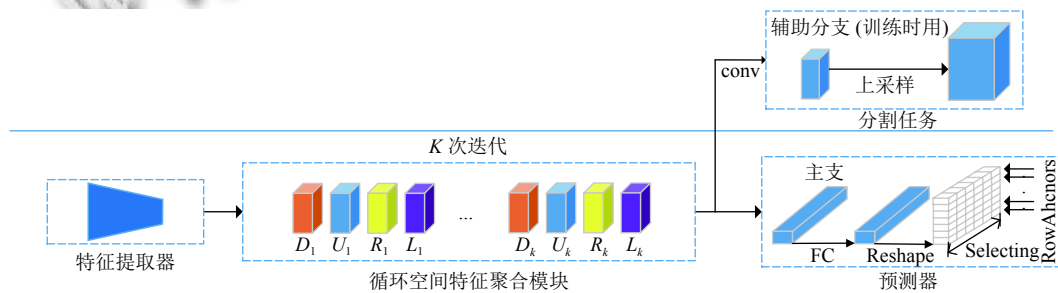


图1 网络架构图

(1) 特征提取器

本文使用VGG^[22]或者ResNet^[13]的全连接层之前的网络作为特征提取器,原始的VGG和ResNet经过多层卷积后得到的特征图的尺寸为原图的1/32,由于车道线占图像素较少,为避免下采样造成语义信息的丢失,本文参考了语义分割方法的操作,将最后的全连接层去掉,并将最后两层的池化层修改为空洞卷积(保证不改变感受野的同时不改变特征图的尺寸),因此,经过特征提取器后的特征图尺寸为原来的1/8.最后将特征图的通道数变为128.

(2) 空间特征聚合模块

经过骨架网络可以提取到图像的语义信息,但是由于CNN本身的局限性,所提取到的特征仅为局部特征.对于特征图上的每个点来说,既没有考虑到感受野之外的像素信息,也没有考虑到感受野内的点与点之

间的相关性.这种简单的特征为后续的预测器分类带来了极大的挑战.如图2所示,为了解决局部的问题,SCNN网络先将特征图进行切片,然后再按照下、上、右、左的顺序进行卷积以获取各行各列像素间的空间信息.以SCNN_D为例,首先将特征图 X 在 H 维度进行切片,得到 H 个 $N \times C \times 1 \times W$ 的特征向量,表示 H 行的特征.每一行的特征计算方式如式(1):

$$X_i = \begin{cases} X_i, & \text{if } i = 0 \\ \text{relu}(f(X'_{i-1})) + X_i, & \text{otherwise} \end{cases} \quad (1)$$

其中, X_i 表示第 i 行特征向量原有值, X'_i 表示第 i 行特征向量更新之后的结果, f 是表示卷积运算,卷积核为 $1 \times w$, relu 为激活函数.第0行的特征向量输入值就是输出值,之后只有第 i 行特征向量收到来自第 $i-1$ 行特征向量的信息时,其才会将信息传递给第 $i+1$ 行.传递

方式为先对第*i*行的特征向量使用一个 $1 \times w$ 的卷积核做卷积运算,然后将卷积后的结果经过非线性激活层 $relu$ 进行信息过滤,再与第*i*+1行的原有特征向量相加,作为第*i*+1行特征向量的结果,该结果同时也作为第*i*+2行特征向量计算的输入,依次计算,得到整个特征图的输出.而 SCNN_U, SCNN_L, SCNN_R 的过程也和 SCNN_D 类似,其中 SCNN_U 和 SCNN_D 恰好方向相反,SCNN_L, SCNN_R 则在维度 *W* 进行切片.

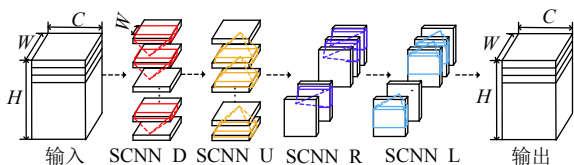


图2 SCNN 模块

但是 SCNN 在长距离的传播过程中会丢失信息,且由于 SCNN 网络速度较慢,本文使用空间特征聚合模块增强特征,这种特征聚合的方式对特征图进行多次迭代,更加充分的利用了特征之间的空间信息.在每一次迭代的时候也是按照从上到下、从下到上、从右到左、从左到右 4 个方向进行信息的搜集,经过 *K* 次迭代之后,特征图中的每一个点都能获得其他点的信息.如图 1 所示,我们分别用 D_k, U_k, R_k, L_k 代表上述的 4 个过程,其中 *k* 表示迭代次数,则第 *k* 次特征聚合模块的迭代过程可用式 (2) 表示.

$$X^{k+1} = L_k(R_k(U_k(D_k(X^k)))) \quad k = 1, 2, \dots, K \quad (2)$$

其中, $K = \lfloor \log_2 L \rfloor$ 表示迭代总次数, *L* 为输入特征图的高或宽.接下来,我们将以 U_k 为例详细的介绍自下而上的过程.

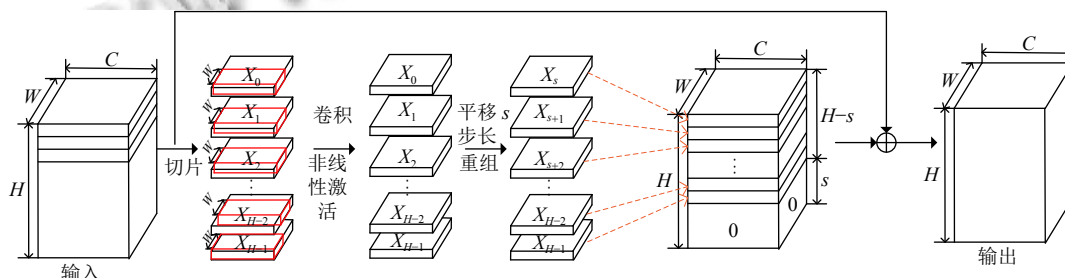


图3 U_k 工作流程 (*s* 表示第 *k* 次迭代的步长)

(3) 预测器

由于进行多次特征融合会消耗减缓网络的推断速

如图 3 所示,首先将特征图 *X* 在 *H* 维度进行切片,得到 *H* 个 $N \times C \times 1 \times W$ 的特征向量,表示 *H* 行的特征.对于每一个特征向量使用一个 $1 \times w$ 的卷积核 *G* 提取与其相隔 s_k 行的特征信息,然后将卷积后的结果经过非线性激活层 $relu$ 进行信息过滤,再与该特征向量相加,作为与输入特征向量相同行的结果.其中 $s_k = L / (2^{K-k})$ 表示步长.此过程可用式 (3) 和式 (4) 表示,其中 *G* 是一个一维的卷积核组, *U* 是输入特征图, *m* 的最大值是输入特征图的通道数 (其值等于特征图的输出通道数,都为 *C*), *i* 和 *j* 分别表示特征图的行和列值, *T* 是中间结果, ρ 是一个非线性函数 (使用的是 $relu$), 式 (4) 中的 $U_{c,i,j}^{k+1}$ 表示经过第 *k* 次 U_k 迭代更新之后的结果.通过第 *k* 次自下而上的信息提取,每一行的特征向量都可以收集到相隔 s_k 行的特征信息.

$$T_{c,i,j}^k = \begin{cases} \sum_{m,n} G_{m,c,n} \cdot U_{m,i+s_k,j+n-1}^k, & 0 \leq i + s_k < H \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

$$U_{c,i,j}^{k+1} = U_{c,i,j}^k + \rho(T_{c,i,j}^k) \quad (4)$$

同样的, D_k, L_k, R_k 的过程也和 U_k 类似,其中 D_k 和 U_k 恰好方向相反, L_k, R_k 则在维度 *W* 进行切片,再聚合其他列的特征.

根据以上的分析,当 $k = 1$ 时 $s_1 = 1$, 此时 $X_{:,h,:}^1$ 接收来自 $X_{:,h+1,:}^0$ 的信息,当 $k = 2$ 时 $s_2 = 2$, 此时 $X_{:,h,:}^2$ (有 $X_{:,0,:}^0$ 和 $X_{:,1,:}^0$) 接收来自 $X_{:,2,:}^1$ (有 $X_{:,2,:}^0$ 和 $X_{:,3,:}^0$) 的信息,此时 $X_{:,0,:}^2$ 就有 $X_{:,0,:}^0, X_{:,1,:}^0, X_{:,2,:}^0$ 和 $X_{:,3,:}^0$ 的信息了,所以经过 $\lfloor \log_2 H \rfloor$ 次迭代之后,每一行都将接收来自相同列其他所有行的信息.自左向右传播类似,所以经过这样 *K* 次迭代之后,每个位置都将获得特征图的全局信息.

度,为了满足车道线检测的实时性条件.因而在预测部分采用了不同于 SCNN 中逐像素分类的方法,而是使

用文献[19]中所述的分类部分来针对经由空间特征聚合处理之后的特征图作预测。该预测器也由两个部分组成:分类分支,分割分支。

分类分支作为网络的主分支负责对每个预定义行中车道线的归属做判断。首先,根据数据集的先验知识(车道线大致位置),挑选出 h 行。然后将每一行等分为 $w+1$ 个单元格,以单元格为单位进行预测。输入图像经过空间特征聚合模块处理之后展开成一维向量,该向量经过全连接层后即得到 $C \times h \times (w+1)$ 的特征图,表示每行网格存在某条车道线的概率。最后使用该特征图在每个预定义行上选择正确的车道线位置。

分割任务作为辅助分支,先将经过空间特征聚合模块处理之后的特征图通道数变为车道线的数量,之后直接上采样与原图尺寸一致的特征图作为分割结果;同时将通道数变换之后的特征图展开成一维向量,经过全连接层处理得到长度为4的向量作为车道线存在与否的结果。该分支只在训练的时候辅助训练,测试的时候不参与信息传播。所以即使使用了辅助任务,在推理的时候也不影响网络的检测速度。

2.2 损失函数

分类任务中的损失函数与[19]中的一致,先使用分类器选出每条车道线在每行中的位置,然后使用交叉熵损失作为分类损失函数,如下所示:

$$P_{i,j,:} = f_{ij}(X), \text{ s.t. } i \in [1, C], j \in [1, h] \quad (5)$$

$$L_{cls} = \sum_{i=1}^C \sum_{j=1}^h L_{CE}(P_{i,j,:}, T_{i,j,:}) \quad (6)$$

其中, X 是输入特征图,其维度为 $C \times h \times w$ 。 f_{ij} 是第 i 条车道线关于第 j 行位置的分类器, $P_{i,j,:}$ 是一个 $w+1$ 维的向量,前 w 个元素表示第 i 条车道线在第 j 行中 w 个网格中的概率,最后一个元素表示在第 j 行是否有第 i 条车道线, $T_{i,j,:}$ 表示正确位置的 one-hot 标签。

分割损失函数与 SCNN 中的一致采用的是分割的二元交叉熵损失 L_{seg} 和存在分类交叉熵损失 L_{exist} ,并进行类别平衡,车道线的分割损失乘以 0.4。

总的损失函数为:

$$L_{total} = L_{cls} + \alpha L_{seg} + \beta L_{exist} \quad (7)$$

实验中超参数 α 和 β 分别设置为 1 和 0.1。

3 实验分析

实验中使用的是两个车道线的基准数据集:

CULane 数据集[8]和 Tusimple 数据集[23]。在接下来的实验设置中,优化器选择 SGD[24],动量设置为 0.9,权重衰减率设置为 $1e-4$,Tusimple 与 CULane 学习率分别为 $2.5e-2$ 和 $1.6e-2$,训练周期分别为 50 和 10,批处理大小分别为 4 和 2。

3.1 Tusimple 数据集测试

Tusimple 数据集的数据采集场景较为单一,数据都是在稳定光照的高速公路上采集的,分辨率为 1280×720 ;数据集总共 6408 张图片,其中训练集 3268 张,验证集 358 张,测试集 2782 张。本文中输入图片大小设置为 368×640 。

Tusimple 数据集使用的评价指标是车道线的精度,定义如下:

$$accuracy = \frac{\sum_{clip} C_{clip}}{\sum_{clip} S_{clip}} \quad (8)$$

其中, S_{clip} 是每个片段中标签点的数量, C_{clip} 是预测正确的点的数量。

对于 Tusimple 数据集,对比了 5 种方法,这 5 种方法主要包括: ResNet18、ResNet34、Res18-SAD[16]、LaneNet[15]、SCNN。本文使用 ResNet18 作为骨干网络,结果显示如表 1。

表 1 Tusimple 数据集上各种方法结果对比

| 模型 | ResNet18 | ResNet34 | Res18-SAD | LaneNet | SCNN | 本文 |
|---------|----------|----------|-----------|---------|--------------|-------------|
| 准确度 (%) | 92.69 | 92.84 | 96.02 | 96.38 | 96.53 | 96.46 |
| 耗时 (ms) | 25.3 | 50.5 | 25.3 | 19.0 | 133.5 | 15.8 |

实验结果表明,相较于 SCNN 本文的方法检测精度降低略微降低,但是速度有较大的提升。速度提升的原因主要有两点,其一是 SCNN 是在原始特征图上进行逐行逐列的顺序操作,每一次只传递某一行或者某一列的信息,其他行和列必须等待接收到传递过来的信息之后才能将它的信息传递出去,这种方法比较耗时;而本文提出的方法是首先对原始特征图使用 $1 \times w$ 或 $w \times 1$ 的共享卷积核执行卷积操作,然后将卷积后的结果沿某个方向平移指定步长形成新的特征图,最后将新的特征图与原始特征图做下标方式的求和运算,这种信息的传递是并行的,不需要逐行逐列的迭代运算,因此花费时间少,速度快,同时在某个方向做卷积操作的时候 SCNN 不共享卷积核,而文中所使用的空间特征聚合模块共享卷积核参数量降低。其二采用了

预定义行的分类方法有效节省了对逐元素分割方法的时间,逐元素分割需要对整个特征图的每一个元素进行分类,但是预定义行只需要针对某一些指定行做分类,因而节省时间。

3.2 CULane 数据集测试

CULane 数据集较为复杂,其数据采集自 9 个不同的场景:正常、拥挤、曲线、强光、夜晚、没有线、阴暗、箭头、交叉路口,都是在城市和高速公路上采集的图片,分辨率为 1640×590,总共 133 235 张图片,其中训练集 88 880 张,验证集 9675 张,测试集 34 680 张。本文中输入图片大小设置为 288×800。

CULane 数据集,其中每一条车道线视为宽 30 像素的线,然后如果预测值和真实值的交并比大于 0.5 则认为其是真阳性,采用 *F1-measure* 作为评价指标,定义如下:

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (9)$$

其中, $Precision = \frac{TP}{TP+FP}$ 并且 $Recall = \frac{TP}{TP+FN}$, *FP* 是假阳性, *FN* 是假阴性, *TP* 是真阳性。

对于 CULane 数据集,几个受欢迎的方法包括 ResNet50、ResNet101、SCNN、RES18-SAD^[16]、Res18-Ultra^[19];本文所使用的方法采用 ResNet18 作为特征提取的骨干网络,结果展示在表 2。相对于其它方法,本文提出的方法在各个场景下的检测性能均有提升,证明了空间特征聚合模块的有效性。

表 2 CULane 数据集几种方法结果对比
(对于交叉路口,只显示 *FP*)

| 模型 | ResNet50 | ResNet101 | SCNN | Res18-SAD | Res18-Ultra | 本文 |
|------|----------|-----------|------|-----------|-------------|-------------|
| 正常 | 87.4 | 90.2 | 90.6 | 89.8 | 87.7 | 91.2 |
| 拥挤 | 64.1 | 68.2 | 69.7 | 68.1 | 66.0 | 71.9 |
| 夜晚 | 60.6 | 65.9 | 66.1 | 64.2 | 62.1 | 67.8 |
| 没有线 | 38.1 | 41.7 | 43.4 | 42.5 | 40.2 | 44.6 |
| 阴暗 | 60.7 | 64.6 | 66.9 | 67.5 | 62.8 | 69.6 |
| 箭头 | 79.0 | 84.0 | 84.1 | 83.9 | 81.0 | 86.9 |
| 强光 | 54.1 | 59.8 | 58.5 | 59.8 | 58.4 | 64.3 |
| 曲线 | 59.8 | 65.5 | 64.4 | 65.5 | 57.9 | 67.6 |
| 交叉路口 | 2505 | 2183 | 1990 | 1995 | 1743 | 1481 |
| 合计 | 66.7 | 70.8 | 71.6 | 70.5 | 68.4 | 73.8 |

3.3 消融实验

为了探索超参数对检测性能的影响,本小结针对 CULane 数据集进行了一系列的消融实验。

考虑到速度的影响,本文选择了两个轻量级的骨

架网络进行对比。结果如表 3 所示,可以看出,使用 ResNet 的特征提取器要明显优于 VGG16。

在空间特征聚合模块中,对于每个特征片进行聚合使用了 1 维的卷积操作,因此卷积核的尺寸可调,改变卷积核的尺寸实验,结果在表 4 中。因此,在所有的实验中,我们设置固定的核尺寸为 9。

表 3 特征提取器替换的影响

| 骨干网络 | VGG16 | ResNet18 |
|-------------------|-------|----------|
| <i>F1-measure</i> | 71.9 | 73.8 |

表 4 核尺寸对模块的影响

| 核尺寸 | 1 | 3 | 5 | 7 | 9 | 11 |
|-------------------|------|------|------|------|------|------|
| <i>F1-measure</i> | 72.4 | 72.6 | 73.3 | 73.5 | 73.8 | 73.8 |

在空间特征聚合模块中,针对迭代次数进行了实验探索。理论上,随着迭代次数的增加,每个像素位置可以聚合更多来自其他位置的信息,从而获得更好的性能。因此,我们对不同的迭代次数进行了比较,如表 5 所示,随着迭代次数的增加,性能会越来越越好。然而,更多的迭代会导致更多的计算时间成本。为了平衡计算时间和性能,我们在其他实验中选择迭代次数为 4。

表 5 迭代次数的影响

| 迭代次数 | 1 | 2 | 3 | 4 | 5 |
|-------------------|------|------|------|------|------|
| <i>F1-measure</i> | 72.2 | 72.5 | 72.8 | 73.8 | 73.9 |

为了验证我们的模块的有效性,统计了与 SCNN 在不同核尺寸下空间信息聚合模块运行的时间,如表 6 所示,可以看到在相同的核尺寸下我们模块的运行时间较 SCNN 几乎快 6 倍。主要原因是 SCNN 传递信息是一种连续的方式,只有当某一特征片接收到来自其他特征片的信息它才会将信息传递给下一个特征片,但是我们的空间特征聚合模块信息的传递是并行的,信息的聚合也是整个特征图基于下标求和的方式完成。

表 6 与 SCNN 不同核尺寸特征聚合模块耗时比较

| 方法 | 核尺寸 | 模块耗时 (ms) |
|------|-----|-----------|
| SCNN | 7 | 23.6 |
| | 9 | 29.0 |
| 本文 | 7 | 4.4 |
| | 9 | 5.0 |

图 4 是在 CULane 数据集上实验之后的可视化的检测结果。

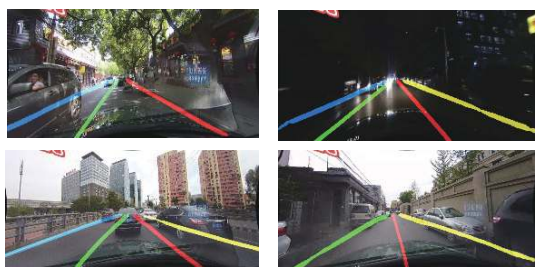


图4 CULane 数据上的检测结果

4 结论与展望

本文基于 SCNN, 设计了一个空间特征聚合模块. 本文利用空间特征聚合模块, 以较小的计算时间融合特征图每个点的全局信息, 在预测器部分, 加入了辅助分支提升系统性能. 实验证明了该网络结构对信息聚合的有效性与时检测的实时性.

参考文献

- Yenikaya S, Yenikaya G, Düven E. Keeping the vehicle on the road: A survey on on-road lane detection systems. *ACM Computing Surveys*, 2013, 46(1): 2. [doi: [10.1145/2522968.2522970](https://doi.org/10.1145/2522968.2522970)]
- Hillel AB, Lerner R, Levi D, *et al.* Recent progress in road and lane detection: A survey. *Machine Vision and Applications*, 2014, 25(3): 727–745. [doi: [10.1007/s00138-011-0404-2](https://doi.org/10.1007/s00138-011-0404-2)]
- Liang D, Guo YC, Zhang SK, *et al.* Lane detection: A survey with new results. *Journal of Computer Science and Technology*, 2020, 35(3): 493–505. [doi: [10.1007/s11390-020-0476-4](https://doi.org/10.1007/s11390-020-0476-4)]
- Cheng HY, Jeng BS, Tseng PT, *et al.* Lane detection with moving vehicles in the traffic scenes. *IEEE Transactions on Intelligent Transportation Systems*, 2006, 7(4): 571–582. [doi: [10.1109/TITS.2006.883940](https://doi.org/10.1109/TITS.2006.883940)]
- Aly M. Real time detection of lane markers in urban streets. 2008 IEEE Intelligent Vehicles Symposium. Eindhoven: IEEE, 2008. 7–12. [doi: [10.1109/IVS.2008.4621152](https://doi.org/10.1109/IVS.2008.4621152)]
- Jung H, Min J, Kim J. An efficient lane detection algorithm for lane departure detection. 2013 IEEE Intelligent Vehicles Symposium (IV). Gold Coast: IEEE, 2013. 976–981. [doi: [10.1109/IVS.2013.6629593](https://doi.org/10.1109/IVS.2013.6629593)]
- Chen Z, Chen ZJ. RBNet: A deep neural network for unified road and road boundary detection. *Proceeding of the 24th International Conference on Neural Information Processing*. Guangzhou: Springer, 2017. 677–687. [doi: [10.1007/978-3-319-70087-8_70](https://doi.org/10.1007/978-3-319-70087-8_70)]
- Pan XG, Shi JP, Luo P, *et al.* Spatial as deep: Spatial CNN for traffic scene understanding. 32nd AAAI Conference on Artificial Intelligence. New Orleans: AAAI, 2018. 7276–7283.
- Huval B, Wang T, Tandon S, *et al.* An empirical evaluation of deep learning on highway driving. *arXiv*: 1504.01716, 2015.
- Lee S, Kim J, Yoon JS, *et al.* VPGNet: Vanishing point guided network for lane and road marking detection and recognition. 2017 IEEE International Conference on Computer Vision (ICCV). Venice: IEEE, 2017. 1965–1973. [doi: [10.1109/ICCV.2017.215](https://doi.org/10.1109/ICCV.2017.215)]
- Pizzati F, Allodi M, Barrera A, *et al.* Lane detection and classification using cascaded CNNs. In: Moreno-Díaz R, Pichler F, Quesada-Arencibia A, eds. *Computer Aided Systems Theory-EUROCAST 2019*. Cham: Springer, 2020. 95–103. [doi: [10.1007/978-3-030-45096-0_12](https://doi.org/10.1007/978-3-030-45096-0_12)]
- Phillion J. FastDraw: Addressing the long tail of lane detection by adapting a sequential prediction network. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach: IEEE, 2019. 11574–11583. [doi: [10.1109/CVPR.2019.01185](https://doi.org/10.1109/CVPR.2019.01185)]
- He KM, Zhang XY, Ren SQ, *et al.* Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas: IEEE, 2016. 770–778.
- De Brabandere B, Neven D, Van Gool L. Semantic instance segmentation with a discriminative loss function. *arXiv*: 1708.02551, 2017.
- Neven D, De Brabandere B, Georgoulis S, *et al.* Towards end-to-end lane detection: An instance segmentation approach. 2018 IEEE Intelligent Vehicles Symposium (IV). Changshu: IEEE, 2018. 286–291. [doi: [10.1109/IVS.2018.8500547](https://doi.org/10.1109/IVS.2018.8500547)]
- Hou YN, Ma Z, Liu CX, *et al.* Learning lightweight lane detection CNNs by Self Attention distillation. 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 1013–1021.
- Paszke A, Chaurasia A, Kim S, *et al.* ENet: A deep neural network architecture for real-time semantic segmentation. *arXiv*: 1606.02147, 2016.
- Yoo S, Lee HS, Myeong H, *et al.* End-to-end lane marker detection via row-wise classification. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition

- Workshops (CVPRW). Seattle: IEEE, 2020. 4335–4343. [doi: [10.1109/CVPRW50498.2020.00511](https://doi.org/10.1109/CVPRW50498.2020.00511)]
- 19 Qin ZQ, Wang HY, Li X. Ultra Fast structure-aware deep lane detection. In: Vedaldi A, Bischof H, Brox T, *et al*, eds. Computer Vision-ECCV 2020. Cham: Springer, 2020. 276–291. [doi: [10.1007/978-3-030-58586-0_17](https://doi.org/10.1007/978-3-030-58586-0_17)]
- 20 Li J, Mei X, Prokhorov D, *et al*. Deep neural network for structural prediction and lane detection in traffic scene. IEEE Transactions on Neural Networks and Learning Systems, 2017, 28(3): 690–703. [doi: [10.1109/TNNLS.2016.2522428](https://doi.org/10.1109/TNNLS.2016.2522428)]
- 21 Bell S, Zitnick CL, Bala K, *et al*. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas: IEEE, 2016. 2874–2883.
- 22 Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv: 1409.1556, 2014.
- 23 TuSimple. TuSimple lane detection. <https://github.com/TuSimple/tusimple-benchmark/>. 2017.
- 24 Bottou L. Large-scale machine learning with stochastic gradient descent. 19th International Conference on Computational Statistics. Paris: Physica-Verlag HD, 2010. 177–186. [doi: [10.1007/978-3-7908-2604-3_16](https://doi.org/10.1007/978-3-7908-2604-3_16)]