Software Engineering and Applications 软件工程与应用, 2023, 12(3), 381-392
Published Online June 2023 in Hans. https://www.hanspub.org/journal/sea
https://doi.org/10.12677/sea.2023.123038

Hans汉斯

基于ResNet-ViT和注意力机制的车道线检测方法

何　飞，唐春晖

上海理工大学，光电信息与计算机工程学院，上海

收稿日期：2023年4月3日；录用日期：2023年5月23日；发布日期：2023年5月31日

摘　要

车道线检测是自动驾驶领域中的重要感知任务。针对当前基于卷积神经网络(CNN)的车道线检测方法存在网络推理速度慢和对细长车道线结构建模能力不佳的问题，提出了一种基于ResNet-ViT和注意力机制的车道线检测方法。具体地，该方法首先搭建主干网络ResNet用于特征提取，并在主干网络中引入Vision Transformer (ViT)的编码结构，以提高网络对车道线细长结构的建模能力。其次，设计辅助分割网络，在其中嵌入通道注意力机制模块，以增强网络对重要通道的学习能力；辅助分割网络与主干网络通过共享部分参数来实现权重共享，从而提高模型的效率和泛化能力。最后，特征解码部分引入行锚分类的思想，在特征图行方向上预测车道线的位置坐标，输出带有车道线标记点的图像。经过实验验证，本文所提出的方法在TuSimple数据集上的准确率达到96.04%，推理速度达到98帧/秒，验证了其有效性。

关键词

车道线检测，ResNet-ViT，注意力机制，行锚分类

Lane Detection Method Based on ResNet-ViT and Attention Mechanism

Fei He, Chunhui Tang

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Apr. 3rd, 2023; accepted: May 23rd, 2023; published: May 31st, 2023

Abstract

Lane detection is a crucial task in the field of autonomous driving. However, the current lane de-

文章引用: 何飞, 唐春晖. 基于 ResNet-ViT 和注意力机制的车道线检测方法[J]. 软件工程与应用, 2023, 12(3): 381-392. DOI: 10.12677/sea.2023.123038

tection methods based on Convolutional Neural Networks (CNN) suffer from slow network inference speed and poor ability to model the slender lane structure. To overcome these limitations, this paper proposes a lane detection method based on ResNet-ViT and attention mechanism. Specifically, the proposed method first constructs a backbone network ResNet for feature extraction, and incorporates the Vision Transformer (ViT) coding structure into the backbone network to enhance the network's ability to model the slender structure of lane lines. Additionally, an auxiliary segmentation network is designed, in which a channel attention mechanism module is incorporated to enhance the network's learning ability for important channels. The auxiliary segmentation network and the backbone network share some parameters to achieve weight sharing, thereby improving the efficiency and generalization ability of the model. Finally, the line anchor classification concept is introduced in the feature decoding part to predict the position coordinates of the lane lines in the line direction of the feature map and generate the image with lane mark points. Experimental results on the TuSimple dataset demonstrate that the proposed method achieves an accuracy of 96.04% and an inference speed of 98 frames per second, verifying its effectiveness.

Keywords

Lane Detection, ResNet-ViT, Attention Mechanism, Row Anchor Classification

1. 引言

随着自动驾驶技术的蓬勃发展，车道线检测已经被广泛应用于车辆辅助驾驶及自动驾驶环境感知模块中。快速且准确的进行车道线检测是保障车辆安全行驶的前提。在当前的研究中，视觉传感器被广泛采用来检测车道线，主要是因为在道路图像中，车道线的视觉特征较为明显，并且相较于 GPS 和激光雷达，视觉传感器具有价格低、鲁棒性好等优势。然而，车道线在实际路况中的形状、颜色呈现多样性，并且车辆行人的遮挡、车道线使用时间过长出现的磨损、天气因素等等都会影响车道线的连续性。因此，如何提高车道线检测的准确性和实时性是自动驾驶领域亟待解决的问题。为了解决这些问题，我们需要研究如何处理车道线的多样性，并且探索如何在保障实时性的前提下提高车道线检测的精确度。

目前，车道线检测可分为基于传统图像处理的方法和基于深度学习的方法。传统图像处理方法大多是基于人工特征提取，并采用直曲线模型来拟合车道线。Chao Ma 等人[1]提出了一种基于 CIELab 颜色特征聚类的车道线检测算法，该算法通过颜色聚类识别车道线，依据道路的几何特征，采用二次曲线来匹配车道。段建民等人[2]提出了一种改进的顺序随机抽样一致性(RANSAC)的车道线检测算法，采用改进简单图像统计(SIS)阈值算法对图像进行二值化处理，构建车道线模型后利用改进的顺序 RANSAC 算法拟合车道线，最后进行模型配对确定车道线。吴彦文等人[3]提出了一种基于视觉传感器与高精度地图相融合的车道线检测与跟踪方法，首先采用改进的霍夫变换提取边缘线段，其次根据滤波预测更新车道线模型状态参数，最后结合高精度地图中车道线先验模型参数跟踪车道线轨迹。虽然这些方法取得了一定的成果，但存在过度依赖人工提取的特征、检测精度低、环境适应性差等问题。为了克服这些问题，具有强大建模能力和特征学习能力的深度学习方法成为国内外学者研究的重点内容。在深度学习方法中，Kim 等人[4]提出了一种基于 CNN 的车道线检测算法，利用 CNN 提取图像中的特征信息，采用聚类等后

处理的方式获得车道线检测的输出结果。Li 等人[5]将 CNN 和循环神经网络(RNN)相结合，利用 CNN 提取每帧图像中的特征信息，并将其输入到 RNN 中进行车道线的预测。为了提高车道线的检测速度，Qin 等人[6]提出了一种超快的车道线检测(UFLD)方法，将车道线检测过程视为基于行的选择分类问题，大大提高了网络模型推理速度。然而，由于不同于以往逐像素点的分类，基于行的分类方法对原始图像进行特征提取时会丢失细节信息，导致检测精度欠佳。为了克服检测精度欠佳的问题，Neven [7]等人提出了一种端到端的 LaneNet 模型，以逐像素的方式检测是否属于车道线。Sun 等人[8]提出了多孔卷积和空间金字塔相结合的方法来提高车道线检测精度。Liu 等人[9]提出使用 Transformer 捕获车道线中细长车道线特征和全局特征，并采用多项式参数模型来描述车道线，实现端到端训练的同时降低了计算量。

针对基于 CNN 的车道线检测方法存在网络推理速度慢和对细长车道线结构建模能力不佳的问题，提出一种基于 ResNet-ViT 和注意力机制的车道线检测方法，该方法首先结合 ResNet [10]和 ViT [11]的编码模块来进行特征提取，学习图像全局特征并细化局部特征，从而减少细节信息的丢失，获得更丰富的特征表示。其次，构建一个通道注意力模块，可以识别不同通道之间的特征差异，并增强模型对车道线细节信息的关注度。最后，采用行锚分类的思想，以网格化行方向位置选择替代了以往的像素级分类，从而提高模型的检测速度，满足实时性的要求。

2. 车道线检测算法设计

本文旨在提出一种同时满足车道线检测实时性和检测精度要求的方法。UFLD 方法采用 ResNet 进行特征提取，并提出行分类的车道线检测思想，提高了车道线检测任务的实时性；而 ViT 网络模型能够对不同尺寸和分辨率的输入图像进行处理。根据文献 DETR[12]的研究结果得出两者的结合可以较好地适应图像分类任务。因此，本文提出一种基于 ResNet-ViT 和注意力机制的车道线检测方法。该方法的网络模型整体框架由特征提取网络、辅助分割网络和分类网络三部分组成，如图 1 所示：

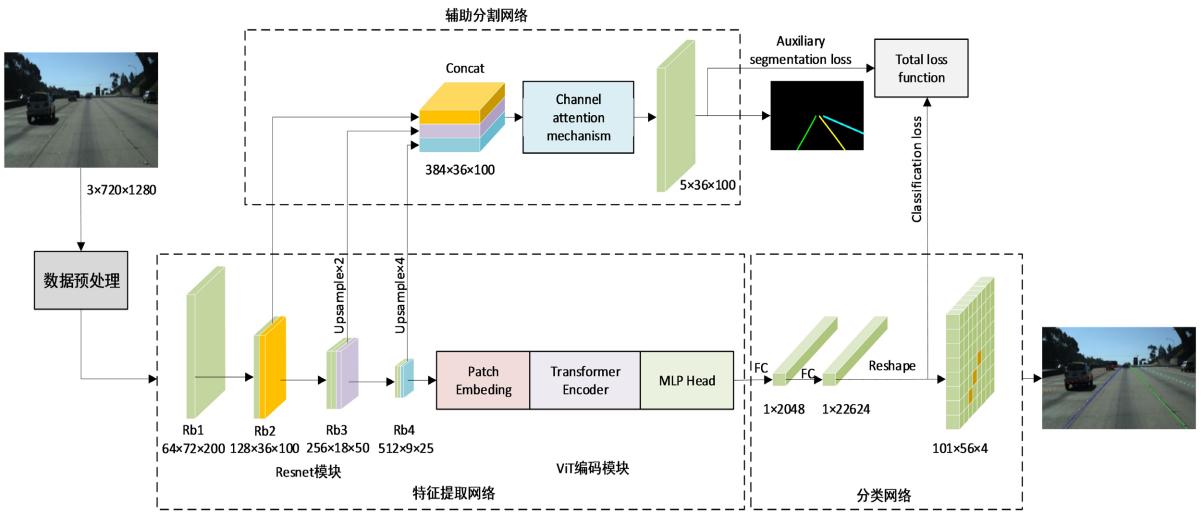

Figure 1. Overall framework of network model
图 1. 网络模型整体框架

2.1. ResNet-ViT 特征提取网络

在进行特征提取前会对图像数据进行预处理，原始图像尺寸为[3, 720, 1280]，表示图像的通道数为 3、高为 720、宽为 1280。考虑到模型下采样率和降低计算量的要求，将原始图像的尺寸统一缩小为[3, 288, 800]

来做标签图像，以图像左上角为原点，沿着图像垂直方向从上往下每隔 4 个像素点采样一个点，共 56 个采样点，确定 56 个行锚(Row Anchor)。每一个 row anchor 上划分 100 个网格，将图像的宽映射到 100 个网格中，得到 100 个网格中真实标签所处的位置。另外采用旋转、垂直和水平位移来进行数据增强，旨在提高模型的泛化能力。

本文提出的 ResNet-ViT 特征提取网络主要表现为将 ViT 网络模型的编码部分嵌入到 ResNet 网络的输出之后，用以提取更丰富的车道线细节特征，生成特征图。输入至特征提取网络的图像尺寸为[3, 288, 800]，ResNet 网络的第一层为卷积层，使用 64 个大小为 7 × 7 的卷积核进行卷积操作，输出尺寸为[64, 144, 400]；接下来，使用最大池化层将特征图的维度缩减至[64, 72, 200]，以减少计算量；然后，特征图被送入 4 个堆叠的残差块(Residual Block, Rb)，在网络模型整体框架中分别为 Rb1、Rb2、Rb3、Rb4，每个 Rb 包括两个卷积层和一个跳跃连接，跳跃连接将输入直接加到残差块的输入上。这些残差块不断对特征图像进行下采样，缩减其尺寸并逐渐提高通道数，最终特征图的输出尺寸为[512, 9, 25]，这些特征图旨在对图像中车道线进行初步特征提取，并作为后续 ViT 编码模块的输入。

ViT 是将 Transformer [13]在 NLP 领域的思想成功运用至图像分类任务上，证明了 Transformer 可以应用于机器视觉领域。ViT 论文中，作者通过实验证明 ResNet + ViT 的融合模型在所有数据单次迭代次数较少的情况下，图像分类的实验效果比纯 ResNet 或纯 ViT 模型的效果好。本文的车道线是一种基于行方向的位置分类，本质也是分类问题。所以，在特征提取网络 ResNet 后引入 ViT 的编码模块，旨在捕获图像全局信息，提取更加丰富的车道线特征，进而提高车道线位置分类的精度。ViT 的编码模块对特征图的处理如图 2 所示：

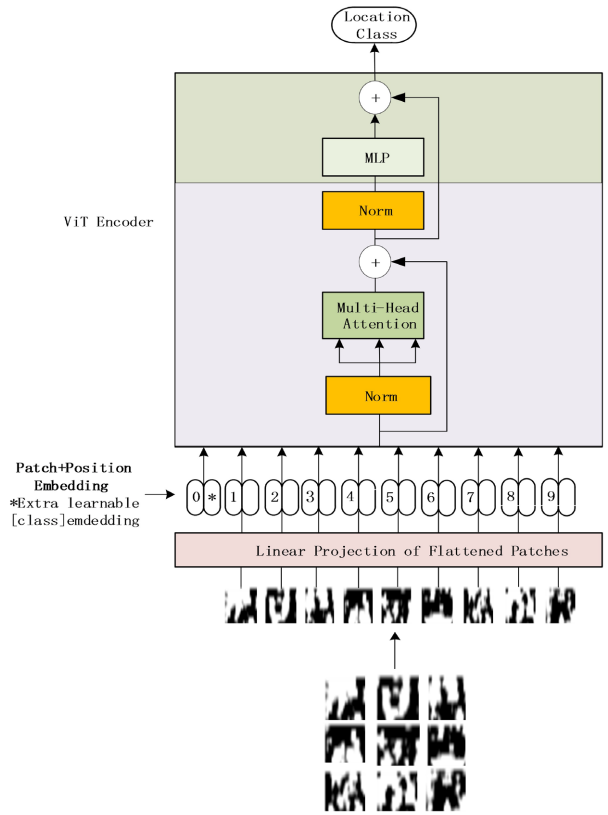

Figure 2. Processing of feature maps by ViT encoding module
图 2. ViT 编码模块对特征图的处理

ResNet 网络输出的特征图为[512, 9, 25]的三维矩阵，而 ViT 中编码模块要求输入的是向量序列，即二维矩阵[num_token, token_dim]，token 是特征图被划分出来的块的向量表示，token_dim 表示每个 token 的向量维度。所以需要先通过 Embedding 层对图像数据进行变换。首先，对尺寸为[512, 9, 25]的特征图进行分块处理，使用 15 个大小为 3 × 5 的卷积核进行卷积操作，将特征图划分成 15 个图像块，每个图像块的大小为 3 × 5。其次，将每个图像块视为一张小图像，并展平为一个向量，这个向量即为 ViT 中编码部分的一个 token，共 15 个 token，每个 token 向量的长度为 15；将这些向量拼接成一个序列，作为 ViT 中编码模块的输入。设定超参数 token_dim 为 100，表示进行 101 分类，则输入序列为[15, 100]。在输入序列的前面需要添加一个特殊的 token，表示整个序列的起始位置。参考 Devlin 等人[14]的标记方式，拼接一个长度为 100 的可训练参数 class token，其大小为[1, 100]，拼接后得到的矩阵大小为[16, 100]，即得到 16 个 token，每个 token 向量维度为 100。在序列中的每一个 token 向量前，需要添加一个位置编码向量，旨在让模型知道每个 token 在原始图像中的位置。使用标准可训练的 1D 位置编码，因为是直接叠加在 token 上的，所以其大小也为[16, 100]，将生成的位置编码矩阵与输入张量相加得到新的嵌入向量，输入至 ViT 的编码模块。

图 2 中 ViT 的编码模块主要由归一化(Norm)、多头注意力机制(Multi-Head Attention)和分类层(MLP)组成。Norm 对输入数据进行归一化，以提高训练稳定性和可靠性。Multi-Head Attention 是基于自注意力机制的结构，可以在不同的图像区域中学习并捕捉关键的特征信息。MLP 作为分类层，预测最终车道线的位置坐标。ViT 编码模块中自注意力机制可以用来学习输入特征图中不同像素之间的关联性，以便更好地检测车道线。自注意力机制的公式如下式(1)，式(2)所示：

$$\begin{cases} \boldsymbol{Q} = \boldsymbol{X}\boldsymbol{W}^{q} \\ \boldsymbol{K} = \boldsymbol{X}\boldsymbol{W}^{k} \\ \boldsymbol{V} = \boldsymbol{X}\boldsymbol{W}^{v} \end{cases} \tag{1}$$

$$\text{Attention}\left(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\right) = \text{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\boldsymbol{V} \tag{2}$$

式中：$\boldsymbol{X}$ 为输入特征矩阵，$\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$ 分别为查询矩阵、键矩阵、值矩阵，$\boldsymbol{W}^{q}$、$\boldsymbol{W}^{k}$、$\boldsymbol{W}^{v}$ 为可学习的超参数矩阵。

在车道线检测任务中，首先需要将输入特征矩阵转换成查询矩阵、键矩阵和值矩阵。可以通过使用三个不同的可学习矩阵分别与输入特征矩阵相乘来实现。接下来，需要计算图像中每个位置与其他所有位置的相似度，并使用这些相似度来计算注意力权重。这里使用的相似度度量方式是点积注意力(Dot-Product Attention)，计算每个查询向量与所有键向量之间的注意力分数，然后对每个查询向量的注意力分数进行归一化，得到注意力权重向量。最后，将注意力权重向量与值矩阵相乘，得到一个新的值矩阵，表示对每个查询向量的注意力池化结果。通过自注意力机制的计算，ViT 编码模块可以捕捉到输入特征图中的局部和全局的关联性，从而提高车道线检测的性能。

Multi-Head Attention 的作用是将输入特征图分别投影到不同的子空间中，从而增强特征表示的多样性。具体来说，将查询矩阵、键矩阵和值矩阵分别经过 h 个不同的线性变换，得到 h 组新的查询矩阵、键矩阵和值矩阵，然后对每组查询矩阵、键矩阵和值矩阵分别进行自注意力机制的计算。最后，将 h 个注意力头的输出结果拼接在一起，并经过一个线性变换得到最终的输出矩阵。Multi-Head Attention 可以学习到不同类型的特征之间的关系，例如车道线的形状、纹理等，帮助模型更好地识别车道线。其计算公式如下式(3)、式(4)所示：

$$\text{head}_i = \text{Attention}\left(\boldsymbol{Q}\boldsymbol{W}_i^Q, \boldsymbol{K}\boldsymbol{W}_i^K, \boldsymbol{V}\boldsymbol{W}_i^V\right) \tag{3}$$

$$\text{MultiHead}\left(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}\right) = \text{Concat}\left(\text{head}_1,\cdots,\text{head}_h\right)\boldsymbol{W}^O \tag{4}$$

式中，$\boldsymbol{W}_i^Q,\boldsymbol{W}_i^K,\boldsymbol{W}_i^V$ 分别代表对第 i 个自注意机制的 $\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}$ 进行线性变换的矩阵，h 表示自注意力的个数，$\boldsymbol{W}^O$ 表示多头注意力的权重矩阵。

2.2. 基于注意力机制的辅助分割网络

在 Vgg [14]等分类网络中加入注意力机制，分类效果会有明显的提升。在 Faster-RCNN [15]等目标检测网络中加入注意力机制时，其目标检测效果也会提升。因此本文尝试在车道线分割网络中加入注意力模块，以增强网络的车道线特征提取能力。本文引入辅助分割网络。在该网络中采用通道注意力机制[16]，旨在提高网络对各通道特征图上车道线细节的关注度，从而训练得到更加优秀的网络模型。通道注意力机制可以赋予每个通道不同的权重，建立起特征图通道间的相互关系，从而增强网络判断重要特征通道的能力。下面给出通道注意力机制的作用原理，其结构图如图 3 所示：

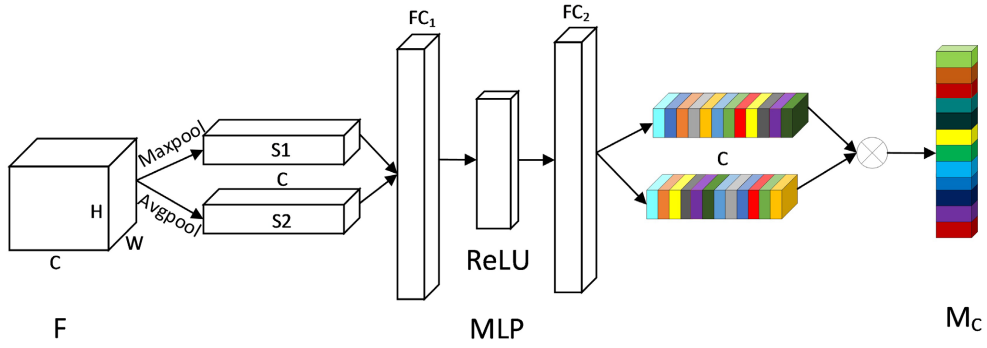

Figure 3. Structure of channel attention mechanism
图 3. 通道注意力机制结构图

如图 3 所示，输入通道注意力模块的特征图为 $F \in R^{H*W*C}$，其中 H 和 W 分别表示输入特征图的高和宽，C 表示输入特征图的通道数。我们首先使用平均池化和最大池化操作将 F 转换为平均池化特征 S1 和最大池化特征 S2，S1 和 S2 的尺寸为 1*1*C。然后，将这两个输出特征送到一个多层感知器(MLP)中，MLP 由全连接层 FC1、FC2 和 ReLU 非线性激活函数组成，以融合所有通道特征，得到两个一维特征向量。最后，我们使用逐元素求和将这两个特征向量进行合并，并使用 Sigmoid 函数对它们进行归一化处理，从而得到各通道的权重值 M_C。通道注意力机制的计算方式如下式(5)所示：

$$\begin{aligned} Mc(F) &= \sigma\left\{MLP\left[AvgPool(F)\right] + MLP\left[MaxPool(F)\right]\right\} \\ &= \sigma\left\{MLP\left[\frac{1}{H\times W}\sum_{i_0=1}^{H}\sum_{j_0=1}^{W} f_{x0}(i_0,j_0)\right] + MLP\left[\max_{i\in H, j\in W} f_{x0}(i_0,j_0)\right]\right\} \end{aligned} \tag{5}$$

式中：σ 表示 Sigmoid 函数，$f_{x0}(i_0,j_0)$ 表示输入特征图 F 的第 x_0 通道中坐标为 (i_0,j_0) 点的像素值。

2.3. 分类网络

本文采用 UFLD 的行锚分类思想，将车道线视为逐行分类问题。如图 4 所示，在图上划分若干个网格单元，车道线检测问题变成在每行上寻找特定网格的问题。

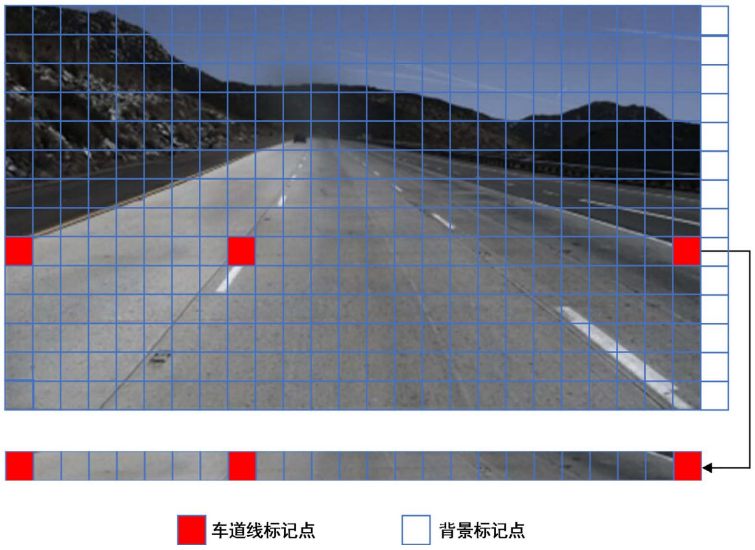

Figure 4. Lane position representation
图 4. 车道位置表示

假定在图像上预定义了 h 个行，在每一行上划分了 w 个网格，待检测的车道线数量为 C 条，X 表示图像的全局特征，f_{ij} 表示第 i 条车道线在第 j 行的位置分类器。那么，车道线的预测公式可表示为式(6)所示：

$$P_{i,j,:} = f^{ij}(X), i \in [1,C], j \in [1,h] \tag{6}$$

经过特征提取网络处理后，得到了饱含车道线细节的特征图。这些特征图输入到全连接层，输出分类张量。然后，将分类张量重构为 C 通道的特征图，每个通道的特征图分成 h 行，并在每一行上进行切分，形成 w 个网格。在每一行的最后增加一个网格单元，代表对应的行中不存在车道线。最后，使用 SoftMax 分类算法，得到一行中每个网格为车道线的概率。接下来，对每行的网格单元概率求期望，计算出车道点的横坐标。这个过程的概率及期望公式如下式(7)、式(8)所示：

$$\text{Prob}_{i,j,l} = \text{softmax}\left(P_{i,j,l}\right) \tag{7}$$

$$\text{Loc}_{i,j} = \sum_{k=1}^{w} k \times \text{Prob}_{i,j,k} \tag{8}$$

式中：$\text{Prob}_{i,j,l}$ 表示第 i 条车道线出现在第 j 行、第 l 列网格单元上的概率；$\text{Loc}_{i,j}$ 表示对行锚中各单元概率求期望后得到的车道点横坐标。

2.4. 损失函数

本文将每条车道线都看作是在不同行上进行分类的问题，因此分类损失函数求的是每条车道线在预定义的行中的预测位置与真实位置的独热编码的交叉熵之和。假设需要检测的车道线数量为 C，预定义的行数量为 h。使用损失函数 L_{cls} 来表示第 h 个行第 C 条车道线交叉熵的损失。假设第 i 条车道线在第 j 行的标签为 T_{ij}，如果判断图中有车道线则 T_{ij} 为 1，如果没有则为 0，假设模型的预测值为 P_{ij}，那么车道线的分类损失可表示为式(9)中的交叉熵损失函数。通过最小化这个损失函数，可以训练出能够准确检测车道线的行锚分类网络模型。

$$L_{cls} = -\sum_{j}^{h}\sum_{i}^{c} T_{ij}\log\left(P_{ij}\right) + \left(1-T_{ij}\right)\log\left(1-P_{ij}\right) \tag{9}$$

另外，辅助分支使用交叉熵作为辅助分割损失，可以有效地提高车道线像素识别的精度。最终的损失函数是分类损失和分割损失的加权平均，其中分类损失是每条车道线在预定义的行中的预测位置与真实位置的独热编码的交叉熵之和，分割损失则是使用注意力机制得到的辅助分支的交叉熵损失 L_{seg}，α、β 为损失系数，总损失计算公式如式(10)所示：

$$L_{\text{total}} = \alpha L_{cls} + \beta L_{seg} \tag{10}$$

3. 实验与讨论

3.1. 实验数据集

本文采用 TuSimple 数据集[17]来验证所提出的车道线检测方法的性能。TuSimple 数据集是当下最广泛应用于车道线检测的数据集之一，它包含 3268 张训练图片、358 张验证图片和 2782 张测试图片。这些图片在不同的天气和交通状况下拍摄，以尽可能模拟真实的驾驶环境。

3.2. 评价指标

为便于同其他车道线检测方法进行对比分析，本文采用官方提供的评价标准：准确率(Acc)、假阳性率(FP)和假阴性率(FN)。计算方式如下式(11)、(12)、(13)所示：

$$\text{准确率(Acc)：}\quad \text{Acc} = \sum \frac{C_{pred}}{C_{gt}} \tag{11}$$

$$\text{假阳性率(FP)：}\quad \text{FP} = \frac{F_{pred}}{N_{pred}} \tag{12}$$

$$\text{假阴性率(FN)：}\quad \text{FN} = \frac{M_{pred}}{N_{gt}} \tag{13}$$

式中：C_{pred} 是预测正确的车道线点数，C_{gt} 是车道线真实的点数；F_{pred} 是预测错误的车道点数量，N_{pred} 是预测正确的车道点数量；M_{pred} 是未被预测到的真实车道点数量，N_{gt} 是真实的车道点数量。

此外，由于本文设计了在训练阶段使用的辅助分割网络，那么交并比(IoU)也是一个重要的评估指标。IoU 值越高，表示预测结果与真实标注越接近，训练出的模型性能越好。

3.3. 实验环境与设置

本文的实验环境包括Ubuntu操作系统，NVIDIA TITAN Xp显卡。我们使用Python3.7语言以及pytorch深度学习框架来构建网络模型。为了增加训练数据的多样性，我们对原始图像进行了裁剪、旋转和平移等数据增强操作。在模型训练过程中，我们使用了 Adam 优化器，学习率为 4e−4，学习率衰减方式采用 cosine 方式，权重衰减系数为 1e−4，动量因子为 0.9。

3.4. 实验结果分析

3.4.1. 定量评估

为了充分评估本文提出的方法对车道线检测的有效性，并将其与近年来提出的 SCNN [18]、VGG-LaneNet [19]、LaneNet [20]、PolyLaneNet [21]、PointLaneNet [22]和 UFLD 方法进行比较，我们进行了一系列的实验并在 Tusimple 数据集上进行了定量分析，评估结果如下表 1 所示。

该表格展示了几种不同车道线检测方法在 TuSimple 数据集上的性能指标，包括准确率(Acc)、误检率(FP)、漏检率(FN)、每秒处理帧数(FPS)和平均交并比(mIoU)。可以看出，本文提出的方法在准确率和

Table 1. Comparison results with other methods on Tusimple dataset
表 1. 与其他方法在 Tusimple 数据集上的比较结果

方法	Acc/(%)	FP/(%)	FN/(%)	FPS/(帧/秒)	mIoU/(%)
SCNN	96.53	6.17	1.80	7.5	57.37
VGG-LaneNet	94.03	10.2	11.0	1.7	41.34
LaneNet	94.42	9.0	9.0	62.5	56.59
PolyLaneNet	93.36	9.42	9.33	115.0	N/A
PointLaneNet	96.34	4.67	5.18	71.0	N/A
UFLD	95.87	19.2	4.0	144	N/A
Ours	96.04	13.8	3.87	98	N/A

注意：表中的 N/A 表示相关论文未提及该指标。

漏检率方面表现良好，达到了 96.04%和 3.87%的结果，但误检率较高，达到了 13.8%。相较之下，SCNN 方法因其使用了复杂的后处理，所以在误检率和漏检率方面表现出色，达到了 6.17%和 1.80%的结果，但推理速度较慢，只有 7.5 帧/秒。PointLaneNet 方法可以直接得到车道线点的坐标，更加契合 Tusimple 数据集的评估方式，因此本文方法在准确率上略差于 PointLaneNet。其他方法的表现情况各有所长，例如 UFLD 方法的误检率较高，但推理速度很快，达到了 144 帧/秒。总的来说，本文的方法在检测精度和速度方面展现了更综合的优势。

3.4.2. 消融实验

为验证本文所提方法中不同模块的有效性，本小节对各模块进行拆解，以 UFLD 为基础模型，实验设置相同的情况下在 Tusimple 数据集上进行对比实验，其结果如下表 2 所示。

Table 2. Quantitative results of the proposed module
表 2. 所提模块的定量结果

基础模型	通道注意力机制	ViT 编码模块	Acc/(%)	IoU/(%)	FPS/(帧/秒)
√			95.87	89.00	144
√	√		95.90	94.00	144
√		√	96.02	87.50	99
√	√	√	96.04	93.80	98

注意：IoU 指标是辅助分割网络参与训练时的交并比。

通过消融实验结果的分析，可以发现在基础模型上添加注意力机制和 ViT 编码模块可以明显提高模型的准确率和交并比，相比其他组合方式更具优势。具体来说，将注意力机制添加到基础模型中，可以显著提高模型的 IoU 指标，相比其他组合方式提高了五个百分点。这是由于注意力机制可以让模型更加关注重要的车道线部分，进而提高模型的准确率。而 ViT 编码模块可以更好地捕捉车道线之间的关系，将其添加到模型中可以进一步提高检测准确率，最终达到了 96.02%的准确率。在基础模型中同时添加注意力机制和 ViT 编码模块，可以使模型达到最佳的性能表现，准确率和 IoU 分别达到 96.04%和 93.80%。总体来看，随着模型中添加的复杂结构越来越多，准确率和交并比逐步提高，但帧率也会随之下降。

3.4.3. 可视化评估

本文方法是基于 UFLD 方法改进的，所以本文主要与 UFLD 方法进行可视化结果对比。数据均采用 Tusimple 数据集中的图片，对比结果如图 5 所示。

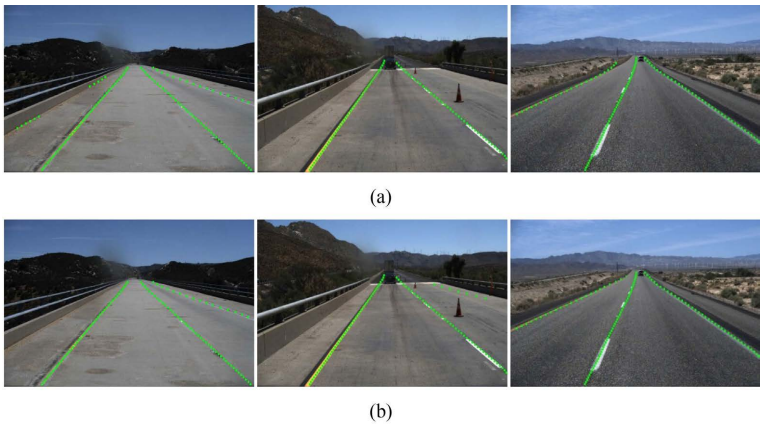

Figure 5. Comparison of visualization results between the proposed method and the UFLD method. (a) UFLD results; (b) Results of our method
图 5. 本文方法和 UFLD 方法的可视化结果比较。(a) UFLD 结果；(b) 本文结果

图 5 展示了本文方法与 UFLD 方法的检测效果对比。结果显示，UFLD 方法对于车道线细节部分的处理效果不佳。在图 5(a)第一幅图像中，UFLD 方法出现了误检情况，将路面边缘错误地标记为车道线；第二幅图像由于最右侧车道线的不明显，UFLD 方法出现了漏检情况；图 5(a)中第三幅图像显示车道线检测点在远端出现了偏移。相比之下，本文提出的方法在图 5(b)中的检测结果未出现误检或漏检情况，并且在远端车道线检测方面表现良好。

图 6 显示了本文方法在 Tusimple 数据集上的一些经典场景下生成的视觉结果。这些场景包括遮挡、多车道、曲线等情况。可以看出，本文方法在各种情况下均表现良好。

Figure 6. This paper method produces visual results on the Tusimple dataset
图 6. 本文方法在 Tusimple 数据集上生成的视觉结果

4. 结论

本文提出一种基于 ResNet-ViT 和注意力机制的车道线检测方法，旨在提高车道线检测的准确性和稳定性。该方法在特征提取阶段和辅助分割网络中分别引入 ViT 编码模块和通道注意力机制，以优化车道线检测的性能。实验结果表明，在 ResNet 的基础上引入 ViT 编码模块，能够显著提高网络的特征提取能力，进而提高车道线检测的精度。此外，本文在辅助分割网络中加入通道注意力机制，可以明显提高 IoU 值，增强车道线像素分割能力。在 Tusimple 数据集上的定量和可视化评估中，本文所提出的方法具有一定的优势。

基金项目

国家自然科学基金 61374197。

参考文献

[1] Chao, M. and Mei, X. (2010) A Method for Lane Detection Based on Color Clustering. *Third International Conference on Knowledge Discovery & Data Mining*, Phuket, 9-10 January 2010, 200-203. https://doi.org/10.1109/WKDD.2010.118

[2] 段建民, 李岳, 庄博阳. 基于改进 SIS 算法和顺序 RANSAC 的车道线检测方法研究[J]. 计算机测量与控制, 2018, 26(8): 280-284, 289.

[3] 吴彦文, 张楠, 周涛, 严巍. 基于多传感融合的车道线检测与跟踪方法的研究[J]. 计算机应用研究, 2018, 35(2): 600-603, 607.

[4] Kim, J. and Lee, M. (2014) Robust Lane Detection Based on Convolutional Neural Network and Random Sample Consensus. In: Loo, C.K., Yap, K.S., Wong, K.W., Teoh, A. and Huang, K., Eds., *Neural Information Processing*, Springer, Cham, 454-461. https://doi.org/10.1007/978-3-319-12637-1_57

[5] Li, J., Mei, X. and Prokhorov, D. (2016) Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scene. *IEEE Transactions on Neural Networks & Learning Systems*, **28**, 690-703.

[6] Qin, Z.Q., Wang, H. and Li, X. (2018) Ultra Fast Structure-Aware Deep Lane Detection. In: Vedaldi, A., Bischof, H., Brox, T. and Frahm, J.M., Eds., *Computer Vision—ECCV* 2020, Springer, Cham, 276-291. https://doi.org/10.1007/978-3-030-58586-0_17

[7] Neven, D., De Brabandere, B., Georgoulis, S., Proesmans, M. and Van Gool, L. (2018) Towards End-to-End Lane Detection: An Instance Segmentation Approach. 2018 *IEEE Intelligent Vehicles Symposium*, Changshu, 26-30 June 2018, 286-291. https://doi.org/10.1109/IVS.2018.8500547

[8] Sun, Y., Wang, L., Chen, Y.Q. and Liu, M. (2019) Accurate Lane Detection with Atrous Convolution and Spatial Pyramid Pooling for Autonomous Driving. 2019 *IEEE International Conference on Robotics and Biomimetics*, Dali, 6-8 December 2019, 642-647. https://doi.org/10.1109/ROBIO49542.2019.8961705

[9] Liu, R., Yuan, Z., Liu, T. and Xiong, Z.L. (2021) End-to-End Lane Shape Prediction with Transformers. 2021 *IEEE Winter Conference on Applications of Computer Vision*, Waikoloa, 3-8 January 2021, 3693-3701. https://doi.org/10.1109/WACV48630.2021.00374

[10] He, K.M., Zhang, X.Y., Ren, S.Q. and Sun, J. (2016) Deep Residual Learning for Image Recognition. 2016 *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, 27-30 June 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90

[11] Dosovitskiy, A., Beyer, L., Kolesnikov, A., *et al*. (2021) An Image is Worth 16 × 16 Words: Transformers for Image Recognition at Scale. https://arxiv.org/abs/2010.11929

[12] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A. and Zagoruyko, S. (2020) End-to-End Object Detection with Transformers. In: Vedaldi, A., Bischof, H., Brox, T. and Frahm, J.M., Eds., *Computer Vision—ECCV* 2020, Springer, Cham, 213-229. https://doi.org/10.1007/978-3-030-58452-8_13

[13] Vaswani, A., Shazeer, N., Parmar, N., *et al*. (2017) Attention Is All You Need. 31*st Annual Conference on Neural Information Processing Systems (NIPS)*, Long Beach, 4-9 December 2017, 6000-6010.

[14] Simonyan, K. and Zisserman, A. (2015) Very deep Convolutional Networks for Large-Scale Image Recognition. https://arxiv.org/abs/1409.1556

[15] Ren, S.Q., He, K.M., Girshick, R. and Sun, J. (2017) Faster R-CNN: Towards Real-Time Object Detection with Re-

gion Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **39**, 1137-1149. https://doi.org/10.1109/TPAMI.2016.2577031

[16] Woo, S., Park, J., Lee, J.Y. and Kweon, I.S. (2018) CBAM: Convolutional Block Attention Module. In: Ferrari, V., Hebert, M., Sminchisescu, C. and Weiss, Y., Eds., *Computer Vision—ECCV* 2018, Springer, Cham, 3-19. https://doi.org/10.1007/978-3-030-01234-2_1

[17] Bigelow, P. (2022) TuSimple Embracing Self-Driving Challenges. *Automotive News*, **96**.

[18] Pan, X.G., Shi, J.P., Luo, P., Wang, X.G. and Tang, X.O. (2018) Spatial as Deep: Spatial CNN for Traffic Scene Understanding. *Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, 2-7 February 2018, 7276-7283. https://doi.org/10.1609/aaai.v32i1.12301

[19] Guo, Z., Huang, Y., Wei, H., *et al*. (2021) DALaneNet: A Dual Attention Instance Segmentation Network for Real-Time Lane Detection. *IEEE Sensors Journal*, **21**, 21730-21739. https://doi.org/10.1109/JSEN.2021.3100489

[20] Neven, D., Brabandere, B.D., Georgoulis, S., Proesmans, M. and Van Gool, L. (2018) Towards End-to-End Lane Detection: An Instance Segmentation Approach. 2018 *IEEE Intelligent Vehicles Symposium*, Changshu, 26-30 June 2018, 286-291. https://doi.org/10.1109/IVS.2018.8500547

[21] Tabelini, L., Berriel, R., Paixo, T.M., Badue, C. and Oliveira-Santos, T. (2020) PolyLaneNet: Lane Estimation via Deep Polynomial Regression. arXiv: 2004.10924v2. https://arxiv.org/abs/2004.10924v2

[22] Chen, Z., Liu, Q. and Lian, C. (2019) PointLaneNet: Efficient End-to-End CNNs for Accurate Real-Time Lane Detection. 2019 *IEEE Intelligent Vehicles Symposium*, Paris, 9-12 June 2019, 2563-2568. https://doi.org/10.1109/IVS.2019.8813778