第 42 卷 第 3 期
2021 年 3 月

仪 器 仪 表 学 报
Chinese Journal of Scientific Instrument

Vol. 42 No. 3
Mar. 2021

DOI: 10.19650/j.cnki.cjsi.J2107395

面向 AR-HUD 的多任务卷积神经网络研究*

冯明驰,卜川夏,萧 红

(重庆邮电大学先进制造工程学院 重庆 400065)

摘 要:汽车上 AR-HUD 已经得到了广泛应用,其环境感知模块需完成目标检测、车道分割等多个任务,但是多个深度神经网络同时运行会消耗过多的计算资源。针对这一问题,本文提出一种应用于 AR-HUD 环境感知的轻量级多任务卷积神经网络 DYPNet,其以 YOLOv3-tiny 框架为基础,融合金字塔池化模型、DenseNet 的密集连接结构、CSPNet 网络模型的思想,在精度未下降的情况下大幅减少了计算资源消耗。针对该神经网络难以训练的问题,提出了一种基于动态损失权重的线性加权求和损失函数,使子网络损失值趋于同步下降,且同步收敛。经过在公开数据集 BDD100K 上训练及测试,结果表明该神经网络的检测 mAP 和分割 mIOU 分别为 30%,77.14%,使用 TensorRt 加速后,在 Jetson TX2 上已经可以达到 15 frame·s^{-1}左右,已达到 AR-HUD 的应用要求,并成功应用于车载 AR-HUD。

关键词: 增强现实抬头显示器;多任务卷积神经网络;目标检测;语义分割

中图分类号: TH85 TP391 **文献标识码**: A **国家标准学科分类代码**: 460.40

Research on multi-task convolutional neural network facing to AR-HUD

Feng Mingchi, Bu Chuanxia, Xiao Hong

(*College of Advanced Manufacturing Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China*)

Abstract: AR-HUD has been widely used in automobile. Its environment perception module needs to complete target detection, lane segmentation and other tasks, but multiple deep neural networks running at the same time will consume too much computing resources. In order to solve this problem, a lightweight multi-task convolutional neural network (DYPNet) applied in AR-HUD environment perception is proposed in this paper. DYPNet is based on YOLOv3-tiny framework, and fuses the pyramid pooling model, DenseNet dense connection structure and CSPNet network model, which greatly reduces the computing resources consumption without reducing the accuracy. Aiming at the problem that the neural network is difficult to train, a linear weighted sum loss function based on dynamic loss weight is proposed, which makes the loss of the sub-networks tend to decline and converge synchronously. After training and testing on the open data set BDD100K, the results show that the detection mAP and segmentation mIOU of the neural network are 30% and 77.14%, respectively, and after accelerating with TensorRt, it can reach about 15 FPS on Jetson TX2, which has met the application requirements of AR-HUD. It has been successfully applied to the vehicle AR-HUD.

Keywords: augmented reality-head up display (AR-HUD); multi-task convolutional neural network; target detection; semantic segmentation

0 引 言

抬头显示(head-up display,HUD)概念在 1946 年就已经出现,最早主要应用在空军领域中,例如,Hawker-Siddeley Buccanneer 是第一台拥有 HUD 的战斗机[1]。21 世纪之后,越来越多的汽车逐渐引入了 HUD,同时,增强现实技术(augmented reality,AR)也有了突破的进展并逐渐成熟。于是,越来越多的研究人员将 AR 和车载 HUD 技术结合起来,并应用在汽车领域。

在 2014 年,Park 等[2]根据 AR 技术提出了车辆安全信息系统,并研发出车载 AR-HUD 系统,该系统可以在

收稿日期:2021-01-16 Received Date: 2021-01-16

*基金项目:国家自然科学基金(51505054)、重庆市科技局(cstc2019jscx-zdztzxX0050)项目资助

汽车行驶过程中提供实时预警信息,有效帮助驾驶者避开危险,提高驾驶安全性。在 AR-HUD 检测模块中一般至少需要包含目标检测和车道检测或车道线检测技术,用于识别车辆前方目标和变道辅助。Park 等[3]在地面障碍物检测模块中使用广告牌扫描立体匹配算法(billboard sweep stereo matching algorithm)计算地面障碍物,根据高度判断障碍物是否为危险的。虽然该方法使用了机器学习模型,但是支持向量机性能较弱,检测精度无法和深度学习模型相比,而且还使用候选区域检测等传统图像处理算法,无法保证实时检测效果。Kim 等[4]为了提高识别率和减少处理时间,优化了图像处理和特征检测算法,以及 Hough 变换、HOG 和 SVM 等分类算法,利用基于 GPU 的并行处理和线程编程,可以减少更多的处理时间。但是在车载嵌入式平台上,依然无法达到实时检测效果,且检测精度不够高、泛化能力不够强。李卓等[5]使用毫米波雷达和 CCD 摄像机相结合的方法,检测车辆前方目标,在其图像处理过程中使用 Adaboost 算法,该算法虽然精度高,但是训练非常耗时,且对异常样本敏感,非常容易影响最终预测准确性。安喆等[6]基于 SSD[7]目标检测模型构建语义分割模型,对整张图片进行分割,得到目标所属类别结果,虽然该方法检测精度高,但是网络前向推理速度过慢。Abdi 等[8]使用 YOLOv1[9]算法检测车辆前方目标,YOLOv1 精度虽然高,但它属于大型目标检测模型,前方推理速度很慢,不适合应用在车载嵌入式平台上。而且上述文献中对车道检测介绍很少,要不就是使用高精度地图来引导前向行驶和变道。但在高精度地图不能投入使用的条件下,仅仅依靠传统手机导航提供的信息远远达不到实现 AR 导航的目的(民用 GPS 定位精度为 10 m),这就需要实时地对车辆所处的道路场景进行道路特征检测,以为 AR 导航提供实景信息。鉴于此,本文提出一种基于多任务学习框架的轻量级多任务卷积神经网络。

多任务学习是深度学习中的一种,它通过在有监督或无监督的任务之间相互传递信息来提高单个任务的通用能力。在神经网络中执行多任务学习一般有两种形式,分别是硬参数共享与软参数共享[10-11],其中硬参数共享是所有任务层共享主干网络参数;软参数共享是每个任务允许单独的参数与一个辅助任务一起约束参数。多任务学习在深度学习中具有非常重要的地位,因此众多学者对多任务学习做了很多的研究,得到了较多的成果。例如,Long 等[12]提出通过深度相关网络(deep relationship networks)来学习任务之间的关系。Misra 等[13]设计了一种十字绣单元(cross-stitch units),可以自动确定不同任务的共同特征和特定特征。Bakker 等[14]提出了一种基于贝叶斯方法的部分参数共享策略。Zhong 等[15]开发了一个灵活的无效知识转移框架。为了学习任务中的相关性,Zhang 等[16]提出了一种基于凸公式(convex formulation)的解决方案。对于时间序列预测和模式分类,Chandra 等[17-19]为此开发了协同进化多任务框架。在 2017 年,He 等提出了 Mask-RCNN[20]网络,该网络是一款典型的多任务卷积神经网络,具有非常高的精度,只是速度比较慢,在性能较弱的车载嵌入式平台中很难达到实时检测效果,但是它为多任务卷积神经网络如何设计奠定了基础。到目前为止,多任务学习已经相对成熟,可以将其应用在 AR-HUD 中。本文提出的多任务卷积神经网络可同时实现目标检测与语义分割功能,即对一张图片进行前向推理时,得到的结果分别是目标种类、坐标以及分割后的车道。以硬参数共享机制将两种功能融合在一个多任务卷积神经网络中,可以使目标检测和语义分割共用一部分特征参数,能够有效减少参数数量。同时也可以有效节省车载芯片资源,将节约的资源用于别的计算任务。本文的后续安排如下:第 1 节描述了 DYPNet 模型;第 2 节对所设计的 DYPNet 模型进行实验验证;第 3 节对本文进行总结。

1 DYPNet 模型

本文针对 AR-HUD 环境感知模块需求,构建了一种基于多任务学习框架的多任务卷积神经网络,并给其命名为 DYPNet(DenseNet-YOLO-PSPNet)。多任务学习框架分为多种形式,其中,Ruder 根据任务特定层共享参数方式的不同,将多任务学习分为硬参数共享机制与软参数共享机制[21]。硬参数共享机制是神经网络中最常用的多任务学习方法,通常通过在所有任务之间共享隐藏层来应用它,同时保留几个特定于任务的输出层,如图 1(a)所示。

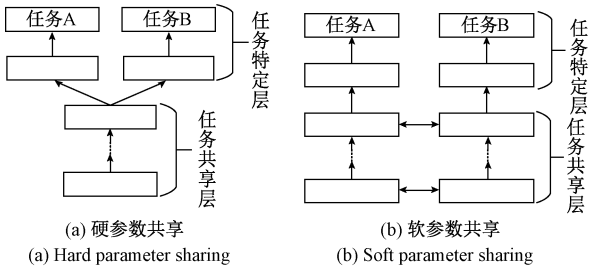


图 1 多任务学习框架

Fig. 1 Multi-task learning framework

硬参数共享机制大大降低了过拟合的风险,因为当有更多的任务需要学习时,网络模型必须找到一个能够捕获所有任务的表示,并且过拟合原始任务的机会较小。在软参数共享机制中,每个任务都有自己的模型和自己的参数,然后对模型参数之间的距离进行正则化,使参数

相似,例如 Duong 等[22] 使用 L2 距离正则化,Yang 和 Hospedales 使用跟踪范数,如图 1(b)所示。

虽然软参数共享机制相比硬参数共享机制检测精度更高,但是软参数共享机制所使用的参数量与计算量更大,检测速度较慢。为了达到实时检测效果,本文选择硬参数共享机制,虽然它结构简单,但是它在检测速度方面具有很大的优势,同时可以节约大量的共享参数。

1.1　DYPNet 模型构建

本文提出的 DYPNet 模型结构如图 2 所示,其中 DCB 模块是 DenseNet[23] 的密集连接模块,如图 3 所示,PSP 模块是 PSPNet[24] 的金字塔池化模型。DYPNet 具有两个任务特定层,分别是目标检测特定层和语义分割特定层,同时这两个任务特定层共享一个主干网络,在降低网络参数量的同时也能够增加网络的泛化能力。

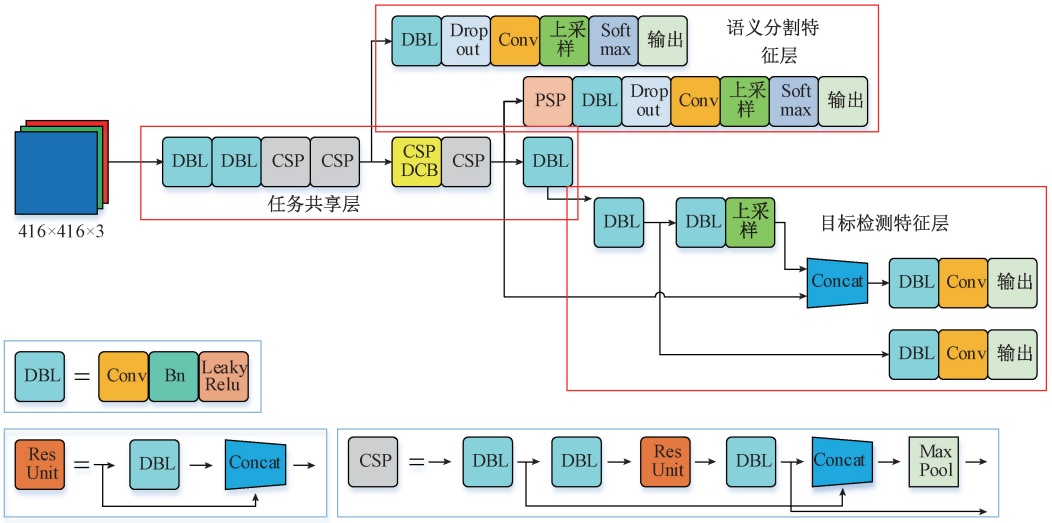


图 2　DYPNet 网络结构

Fig. 2　Structure of DYPNet network

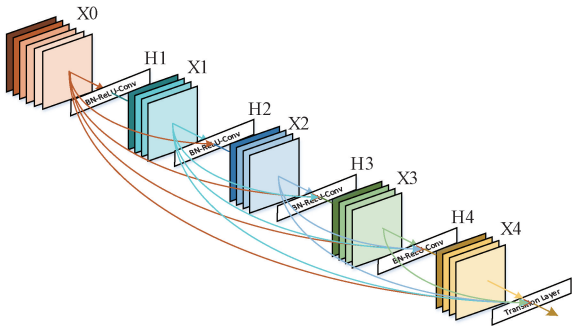


图 3　密集连接结构

Fig. 3　Structure of dense connection

目标检测特定层有两个输出分支,采用特征金字塔网络(feature pyramid network, FPN)[25] 的思想进行构建,其中小尺度输出分支分为两路分支,一路分支进行上采样后和大尺度输出分支进行通道拼接,然后作为大尺度分支输出预测结果;另一路分支则通过一系列的卷积操作输出预测结果。另外,本文的目标检测特定层是基于 anchors 进行训练与预测的,参考 YOLOv3[26] 算法,在每个输出尺度的特征图方格中预设 3 个默认框,本文的目标检测特定层有两个输出尺度,所以共有 6 个默认框。6 个默认框的尺寸各不相同,通过 K-means 聚类算法,在训练数据集中聚类出 6 个方格尺寸。因为本文的目标检测特定层是基于 anchors 进行训练和预测的,所以属于端到端神经网络,从输入端输入图片,从输出端就可以得到结果。端到端神经网络具有前向推理速度快,参数量少等优点,而且检测精度也不逊色于二阶段网络,如 Fast-RCNN[27],Faster-RCNN[28] 等。虽然 Faster-RCNN 网络检测精度高,但是其前向推理速度比较慢,如陈朋等[29] 基于 Faster-RCNN 网络识别车辆假牌、套牌精度可以达到 99% 左右,但是速度只有 0.723 s/帧,而且是在 TITAN X 平台上。

语义分割特定层也是有两个输出分支,语义分割特定层中的两个输出分支与目标检测特定层中的两个输出分支略有不同。在目标检测特定层中,这两个输出分支都负责训练与预测;而在语义分割特定层中,这两个输出分支一个是只负责训练,称为辅助分支,一个是既负责训练又负责预测,称为主分支。

主分支基于金字塔池化模型(pyramid pooling model)进行构建,金字塔池化模型如图 4 所示。金字塔池化尺度融合结构不但实现了将图像全局的上下文信息与局部区域的上下文信息相结合,增加特征信息的丰富度,而且还将图像细节纹理信息与整体的轮廓信息相结合,使网

络可以更加精准分割物体边缘。最后使用双线性插值法对特征图尺寸上采样,使上采样后的尺寸与任务共享层输入尺寸相同。

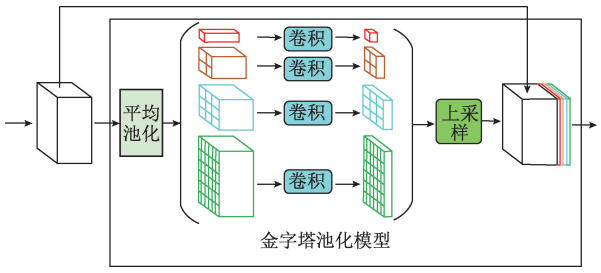


图 4 金字塔池化尺度融合结构

Fig. Structure of pyramid pooling scale fusion

辅助分支虽然在前向推理过程中不使用,但在训练过程可以帮助主分支更快收敛,而且可以有效降低过拟合风险。

1.2 损失函数定义

本文将损失函数分为 3 个部分,第 1 部分是目标检测特定层损失函数;第 2 部分是语义分割特定层损失函数;第 3 部分是整个多任务卷积神经网络的损失函数。

1)目标检测特定层损失函数

目标检测特定层损失函数分为 3 个部分,分别是:

(1)默认框回归损失函数

默认框回归损失函数使用 CIOU 损失函数,如式(1)所示。

$$CIOU\_Loss = 1 - CIOU = 1 - \left(IOU - \frac{\rho^2}{C^2} - \frac{v^2}{(1-IOU)+v} \right) \tag{1}$$

式中:IOU 为真实框与预测框的交并比;ρ 为真实框与预测框的中心点欧式距离;C 为将真实框与预测框包含的最小外接矩形对角线长;v 为衡量长宽比的相似性。v 的具体含义如式(2)所示。

$$v = \frac{4}{\pi^2}\left(\arctan\frac{w^{gt}}{h^{gt}} - \arctan\frac{w}{h} \right)^2 \tag{2}$$

式中:w^{gt}、h^{gt} 分别为真实框的宽高;w、h 分别为预测框的宽高。

(2)置信度损失函数

置信度损失使用二分交叉熵损失函数,如式(3)所示。

$$Loss_{conf} = -\sum_{i=0}^{k\times k}\sum_{j=0}^{m} I_{ij}^{obj}\left[c_i^p \log(c_i^t) + (1-c_i^p)\log(1-c_i^t) \right] - \lambda_{noobj}\sum_{i=0}^{k\times k}\sum_{j=0}^{m} I_{ij}^{noobj}\left[c_i^p \log(c_i^t) + (1-c_i^p)\log(1-c_i^t) \right] \tag{3}$$

式中:k、m 分别为输出特征图尺寸;每个输出特征图方格中默认框数量;I_{ij}^{obj}、I_{ij}^{noobj} 表示输出特征图第 i 个方格第 j 个默认框中含有目标及不含目标;c_i^p、c_i^t 为输出特征图第 i 个方格中预测值、真实值;λ_{noobj} 为不含目标的默认框损失函数权重系数。

在所有的默认框中,不含目标的默认框数量远大于含目标的默认框数量,所以,使用 λ_{noobj} 系数来抑制不含目标的默认框损失值,提高含目标的默认框损失权重,使模式更加“重视”含有目标的默认框造成的损失。

(3)分类损失函数

分类损失同样使用二分交叉熵损失函数,只不过分类损失只计算包含目标的默认框的类别损失值,而不需要计算不含目标的默认框的类别损失,如式(4)所示。

$$Loss_{class} = -\sum_{i=0}^{k\times k} I_{ij}^{obj}\sum_{c\in class}\left[p_i(c)\log(q_i(c)) + (1-p_i(c))\log(1-q_i(c)) \right] \tag{4}$$

式中:$p_i(c)$、$q_i(c)$ 分别为输出特征图第 i 个方格第 c 个类别的预测值和真实值。

最后将这三部分的损失值线性加权就是整个目标检测特定层的总损失值,如式(5)所示。

$$F_1 = CIOU\_Loss + Loss_{conf} + Loss_{class} \tag{5}$$

2)语义分割特定层损失函数

语义分割特定层使用两种损失函数来优化权重,分别是交叉熵损失函数和 Dice Loss 损失函数。其中交叉熵损失函数为主函数,如式(6)所示。

$$Loss = -\sum_x p(x)\log_2 q(x) \tag{6}$$

式中:$p(x)$ 为真实概率分布;$q(x)$ 为预测概率分布。

Dice Loss 损失函数是辅助损失函数,用来辅助交叉熵损失函数做进一步判断。Dice Loss 损失函数如式(7)所示。

$$Dice\_Loss = 1 - \frac{2|y \cap y^{gt}|}{|y| + |y^{gt}|} \tag{7}$$

式中:$|y \cap y^{gt}|$为预测分割图张量与真实分割图张量点乘并求和,$|y|$为预测分割图张量累加和,$|y^{gt}|$为真实分割图张量累加和。

最后将交叉熵损失值和 Dice Loss 损失值线性加权就是整个语义分割特定层的总损失值,如式(8)所示。

$$F_2 = Loss + Dice\_Loss \tag{8}$$

3)多任务卷积神经网络损失函数

本文将多任务卷积神经网络的损失函数定义为两种子网络损失函数的线性加权和,具体如式(9)所示。

$$F_M = W_1 \cdot F_1 + W_2 \cdot F_2 \tag{9}$$

式中:F_M 为多任务卷积神经网络的损失值;F_1 为目标检测特定层网络的损失值;W_1 为目标检测特定层网络的权重系数;F_2 为语义分割特定层网络的损失值;W_2 为语义分割特定层网络的权重系数。

W_1,W_2 根据两种网络的损失值进行动态计算,计算式如式(10),(11)所示。

$$W_1 = \frac{|\Delta F_1|}{|\Delta F_1| + |\Delta F_2|} \tag{10}$$

$$W_2 = \frac{|\Delta F_2|}{|\Delta F_1| + |\Delta F_2|} \tag{11}$$

式中:$|\Delta F_1|$为目标检测特定层网络的当前次 epoch 的损失值与上一次 epoch 的损失值差的绝对值,$|\Delta F_2|$为语义分割特定层网络的当前次 epoch 的损失值与上一次 epoch 的损失值差的绝对值。

将两个分支网络的损失值使用动态损失权重进行加权求和,可保证两个分支网络在训练过程中网络权重同步被优化,避免发生一个分支网络已收敛,另一个分支网络还没有收敛这种情况。

1.3　在 AR-HUD 上的应用

AR-HUD 系统包括 HUD、前视相机、驾驶员瞳孔检测相机、信息处理与控制系统等几部分组成,AR-HUD 系统工作原理如图 5 所示。

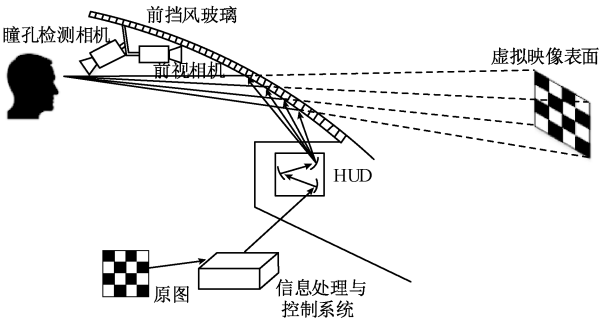


图 5　AR-HUD 系统工作原理图

Fig. 5　Working principle diagram of AR-HUD system

AR-HUD 系统首先通过瞳孔检测相机采集驾驶员驾驶过程中行为信息,其次使用瞳孔检测技术对驾驶员的瞳孔位置进行定位,然后通过汽车挡风玻璃处的前视相机采集汽车前方的环境信息,使用环境感知算法对图片中的场景进行解析。信息处理系统根据车辆、行人位置与驾驶员瞳孔位置获取需要经 HUD 投射的图像,并将图像进行渲染与去畸变处理。最后将处理后的图像发送到 HUD 进行投影,驾驶员观察到的投影图像将会与真实场景相融合,实现场景增强效果。

由于前视相机坐标系、瞳孔检测相机坐标系并不相同,所以为了使目标虚像可以和真实场景相融合,需要将这两个坐标系都转换到车辆坐标系(世界坐标系)下。设一点 P 在前视相机坐标系下的坐标为(x_F, y_F, z_F),前视相机坐标系到车辆坐标系的旋转矩阵为 $\boldsymbol{R}_F$,平移矩阵为 $\boldsymbol{T}_F$,则点 P 在车辆坐标系下的坐标表示如式(12)所示。

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = \boldsymbol{R}_F \begin{bmatrix} x_F \\ y_F \\ z_F \end{bmatrix} + \boldsymbol{T}_F \tag{12}$$

设一点 E 在瞳孔检测相机坐标系下的坐标为(x_E, y_E, z_E),瞳孔检测相机坐标系到车辆坐标系的旋转矩阵为 $\boldsymbol{R}_E$,平移矩阵为 $\boldsymbol{T}_E$,则点 E 在车辆坐标系下的坐标表示如式(13)所示。

$$\begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} = \boldsymbol{R}_E \begin{bmatrix} x_E \\ y_E \\ z_E \end{bmatrix} + \boldsymbol{T}_E \tag{13}$$

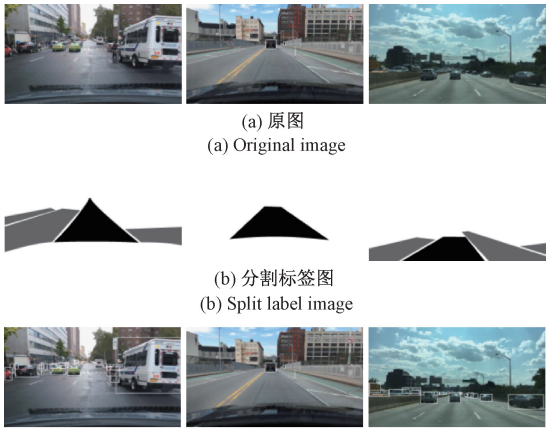
在车辆坐标系下的任意目标点都可以使用三维坐标表示,使用映射函数就可以将所有的三维坐标点映射到二维平面中,通过逆映射函数也可以将二维平面中的坐标映射到三维坐标中。所以,虚拟屏幕上的每个映射点都可以通过逆映射函数映射到车辆坐标系下,实现虚拟投影图像和真实场景相融合的效果。

在环境感知模块中,本文提出的多任务卷积神经网络 DYPNet 可代替单独的目标检测算法,车道分割算法以及车道线识别算法。本文提出的 DYPNet 通过汽车前视相机采集的图片可识别汽车前方的目标并分割车道。将识别的目标坐标、车道信息发送至信息处理系统进行坐标系变换,最后将这些信息通过 HUD 进行投影。

2　实验与分析

实验在个人电脑中进行,使用的语言为 Python-3.6,深度学习框架为 TensorFlow-2.2.0,电脑系统为 Windows 10.0,GPU 为 NVIDIA GeForce RTX 2070 SUPER,显存为 8 G。

本文实验在伯克利数据集(BDD100 K)中进行,BDD100 K 是由伯克利大学 AI 实验室发布的数据集,是目前为止规模最大,场景变化最多的数据集之一。另外 BDD100 K 数据集不但具有足够多的场景和天气变化,而且标注特别精细,其检测难度相比于 COCO 数据集更高,如图 6 所示。



(a) 原图
(a) Original image

(b) 分割标签图
(b) Split label image

(c) 目标检测标签图
(c) Target detection label image

图 6　BDD100 K 部分数据展示图

Fig. 6　Display diagram of BDD100 K partial data

2.1 目标检测实验

本文从 BDD100 K 测试集中随机抽样 500 张图片用作测试,在随机抽样的测试集中,将本文提出的方法与当前主流轻量级网络进行比较,表 1 显示了本文提出的网络在 BDD100 K 数据集上的表现。从表 1 中可以看出,本文提出的 DYPNet 虽然检测速度最慢,但是 mAP 值是最高的,而且速度也达到了 31.57 frame·s^{-1},可以实现实时检测。DYPNet 属于多任务卷积神经网络,不只要考虑目标检测分支性能,还要考虑语义分割分支性能,所以为了使语义分割分支分割效果更好,在主干网络部分添加了 DenseNet 密集连接结构,增加网络特征提取能力。

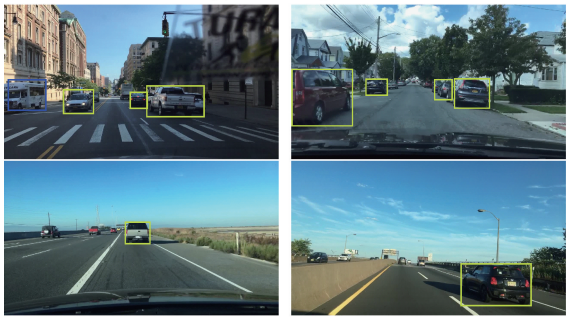
表 1 目标检测分支在 BDD100 K 数据集上的表现

Table 1 Performance of target detection branch on BDD100 K dataset

方法	输入尺寸	mAP/%	模型大小/MB	FPS
YOLOv3-tiny	416×416	23	34.9	34.36
YOLOv4-tiny	416×416	28	22.5	36.49
MobileNet SSD	300×300	25	23.3	35.12
DYPNet	416×416	30	26.0	31.57

图 7 显示了 DYPNet 和 YOLOv4-tiny 在 BDD100 K 数据集上的检测效果对比。

(a) DYPNet算法
(a) DYPNet algorithm



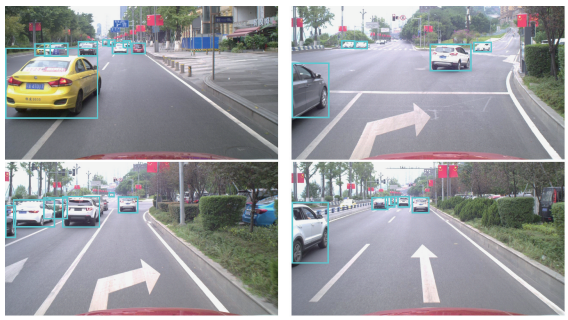
(b) YOLOv4-tiny算法
(b) YOLOv4-tiny algorithm

图 7 在 BDD100 K 数据集上的检测对比

Fig. 7 Detection comparison on BDD100 K dataset

从图 7 可以发现,DYPNet 和 YOLOv4-tiny 对近距离目标检测效果都比较好,但是对远距离和被遮挡目标,YOLOv4-tiny 的检测精度都略低于 DYPNet 网络,有较多的漏检。

图 8 显示了 DYPNet 和 YOLOv4-tiny 在自采数据集的效果对比,从图 8(a)和图 8(b)的对比中可以看出,DYPNet 对近距离目标基本没有出现漏检和误检现象,而且对遮挡目标也有比较高的检测精度。在主干网络中增加 DenseNet 密集连接结构,虽然会略微降低网络速度,但可以有效减少误检和漏检,提高网络对遮挡目标的检测能力。



(a) DYPNet算法
(a) DYPNet algorithm

(b) YOLOv4-tiny算法
(b) YOLOv4-tiny algorithm

图 8 在自采数据集上的检测对比

Fig. 8 Detection comparison on self-collected dataset

2.2 语义分割实验

测试数据集为目标检测使用的相同测试数据集,共包含 500 张图片。在实验机中测试了语义分割分支的速度并计算了模型大小,语义分割分支的模型大小与速度如表 2 所示。

表 2 语义分割分支模型大小与速度

Table 2 Model size and speed of semantic segmentation branch

方法	输入尺寸	FPS	模型大小/MB
DYPNet	416×416	23.5	10.5

从表 1 中可以看出,DYPNet 语义分割分支在实验机

上的检测速度为 23.5 frame·s^{-1},参数量为 0.5 MB,不但达到了实时检测速度,而且对计算机资源消耗也很少。

本文将语义分割类别设为 3 类,分别是 background(不可行驶区域),current lane,navigable area,在测试集中测试,各类别的 IOU 和 mIOU 如表 3 所示。

表 3　在 BDD100 K 数据集上的测试结果

Table 3　Test results on the BDD100 K dataset

类别	IOU/%
background	95.73
current lane	76.08
navigable area	59.61
mIOU/%	77.14

由表 3 可以发现,对不可行驶区域预测精度已经达到了 95%以上,而且这还是在非常复杂的 BDD100 K 数据集中,对 current lane 的预测精度也非常高。

在 PASCAL VOC 2007 和 2012 训练集中训练 DYPNet 网络,并在 PASCAL VOC 2012 测试集上测试 DYPNet 性能。表 4 为 DYPNet 网络和当前几种经典分割网络的对比结果,从表 4 可以看出,本文提出的 DYPNet 网络在 VOC 2012 数据集中的 mIOU 已经超过了当前几种经典的语义分割网络,说明了本文提出的 DYPNet 网络在语义分割方面同样具有优越的性能。

表 4　在 VOC 2012 数据集上的测试结果

Table 4　Test results on the VOC 2012 dataset

方法	mIOU/%
SegNet	0.591
FCN	0.622
DeeplabV2	0.716
DeconvNet	0.725
DYPNet	0.763

图 9 为车道分割效果图,语义分割分支网络根据车道线对车道进行分割,其中较亮的分割区域是当前车道,较暗的分割区域是可行驶区域。由图可以看出,当车道线越明显时,分割边缘效果就越好,当车道虚线之间的间隔较大时,分割边缘会产生一些误差,但总体上效果还是比较可观,左上角图中虽然没有车道线,但是在车的正前方以及右侧方都是可行驶区域,网络依然能够自动将其识别为当前车道,为汽车规划行驶路线。

2.3　多分支融合实验

本文提出的多任务卷积神经网络 DYPNet 包含两个分支网络,分别是目标检测和语义分割,在网络进行前向推理的过程中,两个分支网络分别输出相应的预测结果,在后处理程序中,将两个分支网络的预测结果进行融合,根据融合结果进行可行驶区域规划和汽车防碰撞预警。

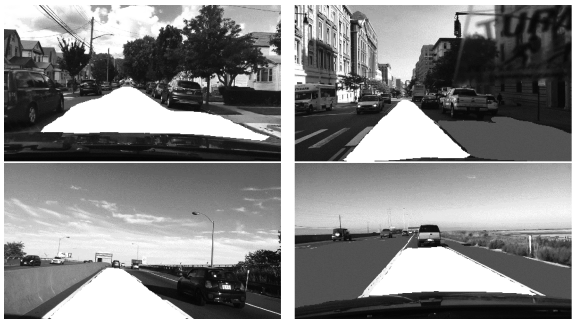


图 9　车道分割效果图

Fig. 9　Renderings of lane segmentation

多任务卷积神经网络在 BDD100 K 数据集上的测试结果如表 5 所示。

表 5　多任务卷积神经网络在 BDD100 K 数据集上的测试

Table 5　Testing of multi-task convolutional neural networks on BDD100 K dataset

方法	输入尺寸	FPS	模型大小/MB	mAP/%	mIOU/%
DYPNet	416×416	20.2	27.9	30	77.14

DYPNet 虽然具有目标检测和语义分割两个子网络,但其模型参数并没有增加多少,因为目标检测和语义分割两个子网络共享一个主干网络,这样会节约大量的可训练参数,网络的推理速度也达到了 20 frame·s^{-1},可以实现实时检测。

DYPNet 效果图如图 10 所示,在每张图片上都有目标检测结果和语义分割结果,且这两个网络分支输出结果都在同一个图像坐标系下。相比较于两种不同网络串行运行,本文提出的多任务卷积神经网络并行运行两种任务,可有效节约计算资源,提高系统整体检测速度。

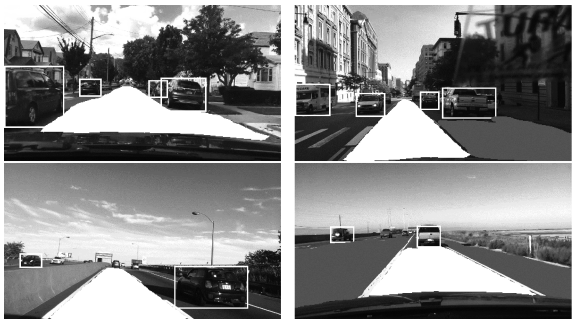


图 10　DYPNet 融合效果图

Fig. 10　Renderings of DYPNet fusion

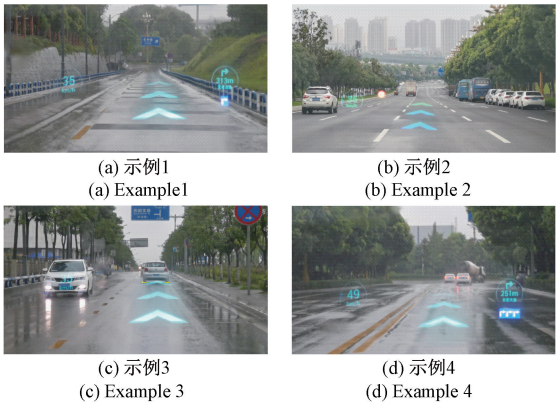
2.4 在 AR-HUD 上的应用

首先在离线情况下,对相机和 AR-HUD 完成标定工作获得一些必要参数,如旋转矩阵、平移向量、畸变系数;然后使用车载嵌入式平台中的瞳孔检测单元测量人眼位置,确定目标在 HUD 上的显示位置;最后将网络模型部署至车载嵌入式 TX2 平台中,并使用 TensorRt 加速。转换后的模型在 TX2 中的速度平均可达到 15FPS,精度稍微下降 1% 左右,仍在接受范围内。AR-HUD 实车图如图 11 所示。

图 11 AR-HUD 实车图

Fig. 11 The real picture of AR-HUD on the car

网络模型在车载嵌入式平台部署完成后,可通过安装在前挡风玻璃内部的前视相机检测前方目标并分割车道。使用标定得到的旋转矩阵和平移矩阵,将网络检测结果从图像坐标系转换为车辆坐标系。通过相机与 AR-HUD 标定结果、瞳孔检测位置以及在车辆坐标系下的网络检测结果,确定目标与车道线在 AR-HUD 显示区域中的位置。图 12 为汽车正常行驶效果图,汽车在正常行驶过程中,HUD 界面会显示车速、导航信息以及导航箭头等信息,驾驶员可以根据需求选择性显示部分信息。对于非当前车道车辆、行人等目标,DYPNet 网络会持续进行跟进检测,但是在 HUD 界面上不会进行显示,以防显示信息过多,对驾驶员驾驶造成干扰。



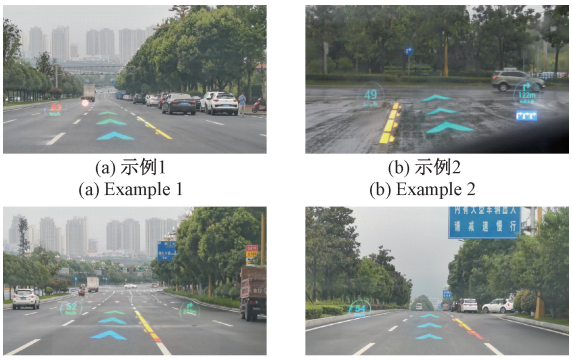
(a) 示例1 (a) Example1　(b) 示例2 (b) Example 2

(c) 示例3 (c) Example 3　(d) 示例4 (d) Example 4

图 12 汽车正常行驶效果图

Fig. 12 Renderings of normal car driving

图 13 为汽车压线行驶效果图,在导航箭头没有从直线行驶箭头变为变道箭头的情况下,汽车偏离当前车道并压到车道线时,HUD 界面会显示一个醒目标识,该标识会被绘制在被压的车道线上,同时,在标识旁边会有一些指向当前车道的标志,提醒驾驶员汽车已压线,回归当前车道正常行驶。



(a) 示例1 (a) Example 1　(b) 示例2 (b) Example 2

(c) 示例3 (c) Example 3　(d) 示例4 (d) Example 4

图 13 汽车压线行驶效果图

Fig. 13 Renderings of car driving on the lane line

图 14 为近车预警效果图,当车辆距离前方目标过近时,会在 HUD 界面上显示一个警告标志,提示驾驶员注意安全,保持安全车距。

(a) 示例1 (a) Example 1　(b) 示例2 (b) Example 2

(c) 示例3 (c) Example 3　(d) 示例4 (d) Example 4

图 14 近车预警效果图

Fig. 14 Renderings of approach car warning

3 结　论

本文综合利用多种经典网络中的优秀结构,设计了一种轻量级的多任务卷积神经网络 DYPNet。DYPNet 对 AR-HUD 范围内的目标车辆进行检测,精度基本可以达到 90% 以上;同时对车道线明显的车道进行分割,精度也可以达到 80% 以上,应用在 AR-HUD 环境感知中,能够以较少的计算资源准确分割出车道并检测汽车前方目

标。为了使多任务卷积神经网络在训练过程中更容易收敛,本文提出了一种基于动态损失权重的线性加权求和损失函数,该损失函数可以使网络更容易收敛,而且能够使两种子网络同步接近收敛状态。

本文提出的方法在近距离检测目标精度及计算速度都已经达到 AR-HUD 实际应用要求,但检测远距离小目标时仍有一定的局限性。今后可以考虑在不降低计算速度的情况下,改变网络结构,增加大尺度输出分支,进而提高网络对远距离小目标的检测精度。

参考文献

[1] MCGREGOR D. A flight investigation of various stability augmentation systems for a jet-lift V/Stol aircraft (Hawker-Siddeley P1127) using an airborne simulator[J]. National Research Council Aeronautical Report LR 500, 1968: 1-41.

[2] PARK H S, KIM K H. AR-based vehicular safety information system for forward collision warning[C]. International Conference on Virtual, Augmented and Mixed Reality, 2014: 435-442.

[3] PARK H S, PARK M W, WON K H, et al. In-vehicle AR-HUD system to provide driving-safety information[J]. Etri Journal, 2013, 35(6): 1038-1047.

[4] KIM K, HWANG Y. The usefulness of augmenting reality on vehicle head-up display[J]. Advances in Human Aspects of Transportation, 2017, 484: 655-662.

[5] 李卓, 周晓, 郑杨硕. 基于 AR-HUD 的汽车驾驶辅助系统设计研究[J]. 武汉理工大学学报(交通科学与工程版), 2017, 41(6): 924-928.
LI ZH, ZHOU X, ZHENG Y SH. Research on the design of automobile driving assistant aystem based on AR-HUD[J]. Journal of Wuhan University of Technology (Transportation Science & Engineering), 2017, 41(6): 924-928.

[6] 安喆, 徐熙平, 杨进华, 等. 结合图像语义分割的增强现实型平视显示系统设计与研究[J]. 光学学报, 2018, 38(7): 85-91.
AN ZH, XU X P, YANG J H, et al. Design and research of augmented reality head-up display system combined with image semantic segmentation[J]. Acta Optica Sinica, 2018, 38(7): 85-91.

[7] LIU W, ANGUELOV D, ERHAN D, et al. SSD: Single shot multibox detector[C]. European Conference on Computer Vision, 2016: 21-37.

[8] ABDI L, MEDDEB A. Driver information system: A combination of augmented reality, deep learning and vehicular Ad-hoc networks[J]. Multimedia Tools and Applications, 2018, 77(12): 14673-14703.

[9] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 779-788.

[10] RUDER S, BINGEL J, AUGENSTEIN I, et al. Latent multi-task architecture learning[C]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019: 4822-4829.

[11] YANG Y, HOSPEDALES T M. Trace norm regularised deep multi-task learning[J]. ArXiv Preprint, 2016, ArXiv:1606.04038.

[12] LONG M, CAO Z, WANG J, et al. Learning multiple tasks with multilinear relationship networks[J]. ArXiv Preprint, 2015, ArXiv:1506.02117.

[13] MISRA I, SHRIVASTAVA A, GUPTA A, et al. Cross-stitch networks for multi-task learning[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 3994-4003.

[14] BAKKER B, HESKES T. Task clustering and gating for Bayesian multitask learning[J]. Journal of Machine Learning Research, 2004, 4(1): 83-99.

[15] ZHONG S, PU J, JIANG Y G, et al. Flexible multi-task learning with latent task grouping[J]. Neurocomputing, 2016, 189: 179-188.

[16] ZHANG Y, YEUNG D Y. Transfer metric learning by learning task relationships[C]. Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010: 1199-1208.

[17] CHANDRA R, ONG Y S, GOH C K. Co-evolutionary multi-task learning for dynamic time series prediction[J]. Applied Soft Computing, 2018, 70: 576-589.

[18] CHANDRA R, CRIPPS S. Coevolutionary multi-task learning for feature-based modular pattern classification[J]. Neurocomputing, 2018, 319: 164-175.

[19] CHANDRA R, ONG Y S, GOH C K. Co-evolutionary multi-task learning with predictive recurrence for multi-step chaotic time series prediction[J]. Neurocomputing, 2017, 243: 21-34.

[20] HE K M, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN[C]. Proceedings of the IEEE International Conference on Computer Vision, 2017: 2961-2969.

[21] RUDER S. An overview of multi-task learning in deep neural networks[J]. ArXiv Preprint, 2017, ArXiv: 1706.05098.

[22] DUONG L, COHN T, BIRD S, et al. Low resource

dependency parsing: Cross-lingual parameter sharing in a neural network parser [C]. Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (ACL) and the 7th International Joint Conference on Natural Language Processing (IJCNLP), 2015: 845-850.

[23] HUANG G, LIU Z, VAN DER MAATEN L, et al. Densely connected convolutional networks [C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 4700-4708.

[24] ZHAO H, SHI J, QI X, et al. Pyramid scene parsing network[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 2881-2890.

[25] LIN T Y, DOLLÁR P, GIRSHICK R, et al. Feature pyramid networks for object detection[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 2117-2125.

[26] REDMON J, FARHADI A. Yolov3: An incremental improvement [J]. ArXiv Preprint, 2018, ArXiv: 1804.02767.

[27] GIRSHICK R. Fast R-CNN [C]. Proceedings of the IEEE International Conference on Computer Vision, 2015: 1440-1448.

[28] REN S Q, HE K M, GIRSHICK R, et al. Faster R-CNN: Towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattem Analysis & Machine Intelligence, 2017, 39(6): 1137-1149.

[29] 陈朋, 汤一平, 何霞,等. 基于多任务 Faster R-CNN 车辆假牌套牌的检测方法[J]. 仪器仪表学报, 2017, 38(12): 3079-3089.
CHEN P, TANG Y P, HE X, et al. Detection method of vehicle fake license plate based on multi-task Faster R-CNN[J]. Chinese Journal of Scientific Instrument, 2017, 38(12): 3079-3089.

作者简介

冯明驰(通信作者),分别 2008 年和 2014 年于中国科学技术大学获得学士学位和博士学位,现为重庆邮电大学副教授,主要研究方向为视觉测量、智能汽车环境感知。
E-mail:fengmc@cqupt.edu.cn

Feng Mingchi (Corresponding author) received his B. Sc. and Ph. D. from University of Science and Technology of China in 2008 and 2014, respectively. Now he is an associate professor at Chongqing University of Posts and Telecommunications. His research interests include vision measurement and environmental perception of intelligent vehicles.

卜川夏,2018 年于重庆邮电大学获得学士学位,现为重庆邮电大学硕士研究生,主要研究方向为计算机视觉。
E-mail:buchuanxia@163.com

Bu Chuanxia received his B. Sc. degree in 2018 from Chongqing University of Posts and Telecommunications. He is currently working toward the M. Sc. Degree in Chongqing University of Posts and Telecommunications. His research interest is Computer Vision.

萧红,2010 年于重庆大学获得博士学位,现为重庆邮电大学副教授,主要研究方向为先进制造技术和智能检测。
E-mail:xiaohong@cqupt.edu.cn

Xiao Hong received her Ph. D. from Chongqing University in 2010. Now she is an associate professor at Chongqing University of Posts and Telecmmunications. Her research interests include advanced manufacturing technology and intelligent detection.