分类号：TP391　　单位代码：10363
密　级：公 开　　学　号: 2210330111

题　目 基于视频图像的驾驶员行为识别方法研究

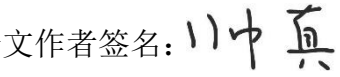
题　目 RESEARCH ON DRIVER BEHAVIOR RECOGNITION METHOD BASED ON VIDEO IMAGES

学　生　姓　名：帅真
校　内　导　师：杨会成（教授）
校　外　导　师：冯海洪（高级工程师）
专　　业：电子信息
研　究　方　向：人工智能与系统集成

论文答辩日期：2024 年 5 月 23 日

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名： 

日期：　　2024 年 5 月 31 日

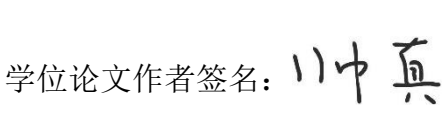
安徽工程大学学位论文版权使用授权书

学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 □，在 ____ 年解密后适用本版权书。

本学位论文属于

不保密 ☑。

学位论文作者签名：  　　指导教师签名： 

日期： 2024 年 5 月 31 日　　日期： 2024 年 5 月 31 日

基于视频图像的驾驶员行为识别方法研究

摘 要

驾驶员的行为是道路交通安全的关键因素之一。对于驾驶员的危险行为进行准确的识别，并及时地进行相应提示，可以极大程度地保证驾驶员人身安全和道路安全。近年来，计算机视觉技术在驾驶行为识别领域的研究日益受到学者们的瞩目，各种算法和模型不断涌现，旨在实现驾驶行为的自动、精确和高效识别，这已成为人工智能领域的一个研究焦点。然而现有的研究工作面临一个重要的瓶颈问题，即如何兼顾识别精度和计算效率。通常来说，高精度模型耗费过多的计算资源，不适用于嵌入式设备；反之，精简的模型难以取得相对较高的识别精度。针对上述问题，本文面向算力有限的嵌入式设备，提出轻量化的驾驶员行为识别模型，实现了实时性和精度的平衡。本文主要研究内容和结论如下：

（1）针对驾驶员姿态估计中常见的非欧式空间数据结构，提出了一种基于图卷积技术的驾驶员行为识别新方法。通过深入剖析驾驶员骨架图在空间和时间维度的分布特征，实现对驾驶员行为的精准识别。鉴于驾驶室环境的特殊性及驾驶行为主要集中于上半身的特点，优化了驾驶员行为姿态估计图，并将其转化为驾驶员行为姿态时空图数据集，以更好地适应实际驾驶场景。针对驾驶员行为实时识别的需求，在此基础上进一步研究了 MobileNet 与空洞卷积的轻量化方法，对模型的复杂度和参数量进行优化。所提方法在 AHPU-SET 数据集上可以取得 90%的识别精度，在 Jetson TX2 上识别速度为 26FPS，与 ST-GCN 相比准确率提高了 5.32%，充分验证了所提方法的有效性。

（2）由于骨架信息只关注于人体的行为特征，无法区分与静态

物体交互的动作类别，因此进一步提出了一种基于压缩视频的驾驶员行为多分支融合识别方法。通过压缩视频获取视频帧、运动矢量和残余误差，设计了多分支时空卷积神经网络模型，使用轻量化特征提取网络结构对外观信息和运动信息编码。同时设计了跨层连接模块与时空特征池化模块用于融合外观信息和运动信息，实现驾驶员行为识别的时空表征。为了弥补由于轻量化网络可能造成的精度下降，采用知识蒸馏学习策略，从泛化能力强的老师模型中学习补充知识来引导学生模型，提升学生模型的性能。所提方法在 AHPU-SET 数据集上的准确率达到了 96.5%，在 Jetson TX2 上识别速度为 25 clips/s，相较于 Coviar 网络准确率提升了 3.5%，运算速度也有提升。

综上，本文提出了可部署于嵌入式设备的驾驶员行为识别方法，在 AHPU-SET 数据集上进行了实验，结果证明了所提方法准确率高、复杂度低、实时性好，为实时准确地识别驾驶员行为提供了有效的技术支撑。

关键词：驾驶员行为识别，图卷积神经网络，压缩视频，嵌入式设备，模型压缩

RESEARCH ON DRIVER BEHAVIOR RECOGNITION METHOD BASED ON VIDEO IMAGES

ABSTRACT

Driver behavior is one of the key factors of road traffic safety. Accurate recognition of dangerous driver behavior and timely corresponding prompts can greatly ensure drivers' personal safety and road safety. In recent years, the research of computer vision technology in the field of driver behavior recognition has received increasing attention from scholars, and a variety of algorithms and models continue to emerge, aiming at realizing automatic, accurate and efficient recognition of driver behavior, which has become a research focus in the field of artificial intelligence. However, existing research efforts face an important bottleneck problem, i.e., how to balance recognition accuracy and computational efficiency. Generally speaking, high-precision models consume too much computational resources and are not suitable for embedded devices; conversely, streamlined models are difficult to achieve relatively high recognition accuracy. Aiming at the above problems, this paper proposes a lightweight driver behavior recognition model for embedded devices with limited arithmetic power, which achieves a balance between real-time and accuracy. The main research content and conclusions of this paper are as follows:

(1) For the common non-Euclidean spatial data structure in driver posture estimation, a new method of driver behavior recognition based on graph convolution technique is proposed. By deeply analyzing the distribution characteristics of the driver's skeleton map in the spatial and

temporal dimensions, accurate recognition of driver behavior is achieved. In view of the special characteristics of the cab environment and the fact that the driver behavior mainly focuses on the upper body, the driver behavior pose estimation map is optimized and transformed into a driver behavior pose spatio-temporal map dataset to better adapt to the actual driving scene. Aiming at the demand of real-time recognition of driver behavior, the lightweight method of MobileNet with dilated convolution is further investigated on this basis to optimize the complexity and number of parameters of the model. The proposed method can achieve 90% recognition accuracy on the AHPU-SET dataset, and the recognition speed is 26FPS on Jetson TX2, which improves the accuracy by 5.32% compared with that of ST-GCN, which fully verifies the effectiveness of the proposed method.

(2) Since the skeleton information only focuses on the behavioral features of the human body and cannot distinguish the action categories interacting with static objects, a multi-branch fusion recognition method for driver behavior based on compressed video is further proposed. Video frames, motion vectors and residual errors are obtained from compressed video, and a multi-branch spatio-temporal convolutional neural network model is designed to encode appearance and motion information using a lightweight feature extraction network structure. Meanwhile, a cross-layer connection module and a spatio-temporal feature pooling module are designed for fusing appearance and motion information to achieve spatio-temporal characterization for driver behavior recognition. To compensate for the possible accuracy degradation due to the lightweight network, a knowledge distillation learning strategy is used to learn supplementary knowledge from the teacher model with strong generalization ability to guide the student model and improve the performance of the student model. The proposed method achieves an accuracy of 96.5% on the AHPU-SET

dataset and a recognition speed of 25 clips/s on Jetson TX2, which is a 3.5% improvement in accuracy and an increase in computing speed compared to the Coviar network.

In summary, this paper proposes a driver behavior recognition method that can be deployed in embedded devices, and experiments are conducted on the AHPU-SET dataset, and the results demonstrate that the proposed method has high accuracy, low complexity, and good real-time performance, which provides effective technical support for real-time and accurate recognition of driver behavior.

KEY WORDS: driver behavior recognition, graph convolutional neural networks, compressed video, embedded devices, model compression

目录

第1章 绪论

1.1 研究背景与意义

随着汽车数量的迅猛增长，道路交通安全问题日益凸显，成为公众关注的焦点。在道路交通系统中，人、车、道路、环境四要素共同构成了一个复杂的交互网络，其中人的因素尤为关键。据权威统计，近九成交通事故源于人为因素，而驾驶员的危险行为更是占据了其中的近七成[1]。这一数据揭示了驾驶员行为对交通安全的重要影响，突显了改善驾驶员行为、降低交通事故率的紧迫性。为此，国务院安委会明确要求“两客一危”车辆安装智能视频监控的报警装置。该装置具备实时检测驾驶员不安全驾驶状态的能力，一旦识别到潜在的安全隐患，将立即触发警报系统，从而有效预防交通事故的发生，进而显著增强道路的安全性能。实现对驾驶员行为的实时自动识别，对于预防和减少由于驾驶员错误决策和危险行为导致的交通事故具有重大意义。因此开发一种能够有效检测驾驶员行为的方法就显得尤为重要。

当前，关于驾驶员驾驶状态与行为的检测研究日益深化，其研究重点主要聚焦于几个关键领域：基于驾驶员生理状态感应参数[2]、基于车辆运行状态信息[3]以及基于计算机视觉技术[4]。在基于生理状态感应方面，这种方法能够提供丰富的驾驶员生理数据，包括心率、脑电波等，能够直观地反映驾驶员的生理状态，从而判断其是否处于可能影响驾驶安全的状态。然而，应用过程中驾驶员需要佩戴生物感应的仪器，这无疑会对驾驶体验造成一定的干扰。此外，生物感应仪器的价格普遍较高，这也使得其在实际应用中的普及面临一定的困难。在基于车辆运行状态方面，通过收集方向盘操作、刹车使用情况等关键信息，为分析驾驶员的驾驶行为提供了重要依据。然而，这种方法在判断驾驶员是否处于分心状态方面存在一定的局限性。因为仅凭车辆行驶数据，往往难以准确捕捉驾驶员的注意力分散情况，这在一定程度上影响了其检测结果的精确性。相比之下，基于计算机视觉，特别是车载摄像头的驾驶行为检测方法，因其实时性强、数据获取便捷且对驾驶员行为特征捕捉精准，成为最具实用前景的研究方向。这种方法不仅适用于私家车驾驶安全的日常监测，也便于交通监管部门进行有效管控。随着深度

学习技术[5]的快速发展，其在视频行为识别领域的应用日益广泛。基于深度学习的驾驶行为识别研究，不仅提升了识别的准确率和效率，也在实际应用中取得了显著成效。未来，随着技术的不断进步和成本的不断降低，这种基于计算机视觉和深度学习的驾驶行为检测方法将在保障驾驶安全方面发挥更加重要的作用。驾驶员行为识别作为车联网（Internet of Vehicles，IoV）系统的重要功能，对于提升行车安全性具有不可忽视的作用。通过车载摄像头实时监测汽车座舱内的驾驶员状态，车联网系统能够迅速捕捉驾驶员的分心驾驶行为，及时发出提醒，或自动切换到自动驾驶模式，确保行车过程中的安全。随着自动驾驶技术的不断发展，驾驶员行为识别将成为实现完全自动驾驶的关键一环，为未来的智慧交通奠定坚实基础。

1.2 国内外研究现状

1.2.1 基于骨架特征的驾驶员行为识别方法

基于骨架的行为识别方法是一种通过分析人体骨架序列的运动模式来进行行为判定的技术。该方法的核心在于利用人体骨架关节点的二维或三维坐标时间序列作为输入，进而构建相应的识别模型。骨架序列占用空间小，且对于光照、颜色、纹理等干扰因素具有较强的鲁棒性。基于骨架的行为识别方法，根据其技术特性和应用方式，主要可划分为以下三类。（1）基于循环/卷积神经网络（2）基于图卷积网络（3）基于 Transformer 网络。

在早期对骨架行为识别的探索中，主流的技术手段主要围绕着循环神经网络（Recurrent Neural Network，RNN）和卷积神经网络（Convolutional Neural Network，CNN）的设计展开。Shahroudy 等人[6]则提出了一种递归神经网络结构，该结构专注于建模每个身体部位特征的长期时间依赖性，通过捕捉这种时间相关性能够更准确地识别出行为模式，提高了识别的准确性。Li 等人[7]尝试将骨骼序列编码为一张时空编码图，保留了骨骼序列的时空信息，然后将这张时空编码图输入到卷积神经网络中，通过网络的逐层提取和抽象，最终得到行为类别的识别结果。Li 等人[8]提出了一种分层聚合端到端共现特征学习框架，充分利用了 CNN 的自动特征学习能力，从骨架序列中逐层提取共现特征，从而有效地捕捉了人体行为

的复杂性和动态性。Ke 等人[9]提出了一种骨架序列处理方法，将 3D 坐标的每个通道转换为片段，每个片段都蕴含了丰富的时间信息和空间关系，在此基础上设计了一种多任务 CNN 来学习这些生成的片段，从而实现了对行为的精准识别。Duan 等人[10]将人体骨架的一系列数据转换为 3D 的热力图形式，再使用 3D CNN 提取运动特征。这类方法存在一些局限性，在处理过程中往往忽视了骨架关节点的空间依赖关系，导致行为识别的精度受到一定的限制。

Yan 等人[11]提出了一种创新的时空图模型，将图神经网络的应用范围扩展至骨架行为识别领域。该模型建立在骨架图的基础上，每个节点对应一个人体关节，通过空间边和时间边分别捕捉关节间的自然连通性和时间上的连续性。多个时空图卷积层的构建使得信息在时空维度上得以有效集成，显著提升了行为识别的性能。基于时空图卷积神经网络（Spatial Temporal Graph Convolutional Networks，ST-GCN），Shi 等人[12]利用骨骼数据中的关节和骨骼运动学依赖关系构建有向无环图，这种网络结构能够有效地提取关节、骨骼及其关系信息，进而实现准确的行为识别。在行为识别领域，使用的骨架图通常是启发式预定义的，主要用于表示人体的物理结构。然而，这种预定义的骨架图对于行为识别的效果并非最佳。以读书和拍手为例，这些行为中两只手的关系对于识别至关重要，但在现有的 ST-GCN 预定义人体图中，两只手之间的距离较远，导致难以准确捕获它们之间的关系。Obinata 等人[13]为增强 GCN 的时序建模能力，设计了一个时间扩展模块，该模块能有效扩展相邻帧之间的时态图，从而更准确地提取人体运动中多个相邻关节的相关特征。为了提升模型的推理效率，Song 等人[14]提出了一种多输入分支的早期融合方法，旨在丰富骨骼特征。该方法在早期融合阶段保持紧凑表示，同时结合瓶颈结构的残差 GCN 和局部注意力模块，有效减轻了计算成本并降低了训练难度，同时保持了模型的精度。然而，基于 GCN 的算法存在通道共享同一套拓扑结构的问题，这限制了模型的能力上限。针对这一问题，Chen 等人[15]提出了细化训练拓扑结构的方法，以突破这一限制，从而提高模型的表现能力。尽管图卷积网络在行为识别中因参数量精简、计算复杂度低而展现出显著优势，但其对于短期运动特征的忽略却成为了在识别特定对称动作时的一大障碍。这种局限性使得图卷积网络在处理形态高度相似的对称动作时，难以捕捉其细微差异，从而影响了识别的准确性。

Plizzari 等人[16]在研究中引入了 Transformer 模型[17]，进而构建了一种创新的时空 Transformer 网络（Spatial-Temporal Transformer network，ST-TR），该网络充分运用了 Transformer 的自注意力机制，以建模关节之间的复杂依赖关系。空间自注意力模块负责捕捉不同身体部位在同一帧内的交互模式，而时间自注意力模块则专注于模拟帧与帧之间的相关性，从而全面理解行为的动态变化。Wang 等人[18]提出了一种基于 Transformer 的新型网络，该网络将身体骨架分为多个部分，分别提取整体和部位的特征，并建立两者间的关系，避免了单个关节间相互作用的局限性，有助于更全面地理解行为。基于 Transformer 的方法在性能上虽然与图卷积网络方法相近，但在实际应用中却面临着显著的挑战。这些方法在训练过程中往往表现出较高的难度，需要长时间的迭代与优化，导致了训练周期的延长。由于 Transformer 模型结构的复杂性，其计算复杂度相对较高，这在一定程度上影响了模型在实际应用中的效率。

基于骨架的驾驶行为识别方法主要依赖 GCN 或 RNN 来抽取姿态特征。其中，Li 等人[19]的研究聚焦于利用时空领域图来估计驾驶员的关节位置，并通过基因加权算法筛选出与驾驶行为高度相关且位置变化显著的关节点。这种方法有效地利用了感官输入与驾驶员行为之间的相关性，从而提高了对驾驶员详细动作的识别精度。Pan 等人[20]则提出了一种实时驾驶活动识别模型，通过车载单目摄像头采集驾驶员上半身的骨架信息，并利用 GCN 进行单帧空间结构特征的推理。为了捕捉时序运动特征，进一步使用长短期记忆网络（Long Short-Term Memory，LSTM）网络对连续帧进行处理，在实时性和准确性方面均取得了良好的表现。Martin 等人[21]的研究进一步扩展了基于骨架的识别方法的应用范围，通过整合额外的输入模式，如内部元素和物体信息，将这些数据与骨架信息相结合，提高了与物体交互相关行为的识别精度。在驾驶行为识别领域，基于骨架的方法在多数场景下均展现出优越的性能。然而，面对身体姿态高度相似但交互物体迥异的行为，例如抽烟与喝水，由于缺乏交互物体的外观信息，这些方法在区分这些细微差异时难以保持高精度，从而影响了整体识别的准确性。

基于 RGB 视频帧和骨架的驾驶行为识别方法，通过双流架构整合了 RGB 视频帧与骨架信息的优势。其中，Weyers 等人[22]的研究聚焦于驾驶员身体关键点的定位，特别是手部的图像区域，并结合三维身体关键点作为递归神经网络的输

入。这种设计有效捕捉了手部动作与驾驶行为之间的关联，提升了识别精度。Behera 等人[23]进一步认识到人类活动是姿势与语义上下文线索的结合，通过建模身体关节结构，将身体与物体的相互作用表示为关系对，提取结构信息，并提出了一种多流深度融合网络，将高级语义信息与 CNN 特征相结合，增强了识别模型的理解能力。这类方法避免了基于光流方法的一些局限性，但不同模态信息的融合过程中，仍然会导致信息损失，这使得难以充分利用不同模态之间的互补信息，从而影响了最终的识别精度。

1.2.2 基于视频图像的驾驶员行为识别方法

视频行为识别技术的发展历程可归纳为三大类别：（1）基于光流的双流模型；（2）基于 3D 卷积模型；（3）轻量级模型。

Karpathy 等人[24]深入研究了卷积神经网络（Convolutional Neural Network，CNN）在视频行为识别任务中的应用，利用单个 2D CNN 有效地提取视频帧的空间特征并探索多种特征融合方式，以实现对视频行为的准确识别。然而在大规模数据集上预训练的模型迁移至小规模数据集时，性能往往不尽如人意，甚至逊于基于手工特征的传统方法。尽管 2D CNN 在提取空间特征方面表现出色，但处理视频的方式主要基于单帧分析，对于时序信息的捕捉和处理相对较弱，在处理涉及连续帧之间关系以及复杂动态变化的视频行为时，难以达到理想的识别效果。于是，Simonyan 等人[25]提出了基于光流的双流网络。该网络的一个分支处理静态帧，负责建模空间信息；另一分支则利用光流[26]提取连续帧间运动的互补信息。两分支的分类结果经融合后，实现了对视频行为的精准识别。在此基础上，双流网络结构得到了进一步的优化。Ng 等人[27]利用权值共享的 CNN 提取时序有序特征，并结合 LSTM 进行时序处理和特征融合。但此方法在短视频处理中由于特征相似性高而提升有限。Feichtenhofer 等人[28]则深入探讨了特征融合的策略，在双流网络的基础上，研究了不同层次的时空特征图融合方法，进一步提升了识别性能。Wang 等人[29]关注于长期时间信息的利用，设计了一种基于分段采样的双流网络模型。该模型通过引入稀疏采样策略，直接从视频片段中挑选出关键帧，有效地建模了帧之间的长时依赖关系，从而实现了视频片段级别的行为识别。

时空模型的核心思想在于通过 3D 卷积核在视频序列中同时沿时间和空间维

度进行卷积操作，以捕获视频中的短时时空信息。此类方法最初由 Ji 等人[30]提出，为视频行为识别领域带来了新的视角。在此基础上，Tran 等人[31]提出了C3D(Convolution 3D)网络结构，将 2D 的 VGG16[32]扩展至 3D 版本，由于当时公开数据集的规模受限，该模型在训练过程中出现了欠拟合现象，导致性能未能达到预期。由于 3D CNN 在训练过程中的高计算成本和对视频数据的庞大需求，Qiu[33]和 Tran 等人[34]通过使用伪 3D(Pseudo-3D, P3D)卷积核的方法来减少参数量，该方法将传统的3×3×3卷积核分解成1×3×3和3×1×1两个较小的卷积核，原先的卷积操作需要 27 个参数减少为仅需要 9 个参数，同时将 Resnet 作为基础模型，这一改进显著提升了模型的性能，相较于最初的 3D 网络结构有了明显的提升。为了缓解 3D CNN 在训练过程中的困难，研究者们开源了一个大规模的视频数据集 Kinetics400，通过将 2D 网络的预训练参数迁移至 3D 网络，显著提升了 3D CNN 的训练效率和准确性，为行为识别任务提供了更为强大的支持。Carreira 等人[35]收集并发布了数据量达到 60 万的 Kinetic 数据集，通过在该数据集上进行预训练，3D 模型展现出了显著的性能提升，首次在性能上超越了双流网络方法。这一突破性的进展，充分验证了在训练过程中 3D 模型此前的确存在的欠拟合问题。同时还提出了膨胀 3D 卷积神经网络（Inflated 3D ConvNet，I3D），该模型通过将 2D 卷积核在时间维度上进行膨胀，接着赋值给对应的 3D 卷积核，实现了对运动信息的有效捕捉。双流 I3D 更进一步从堆叠的光流中学习显式的运动表示，取得了显著的效果。Wang 等人[36]提出了 Non-local 模块，旨在捕获视频中的长时依赖性，这一创新有助于模型更好地理解和识别复杂的时间动态。Feichtenhofer 等人[37]提出了一种 3D 双流网络，该网络包含慢速高分辨率流和快速低分辨率流。慢速高分辨率流专注于学习静态图像的场景信息，而快速低分辨率流则用于描述运动信息，旨在以较少的参数捕捉高频运动信息，从而提高行为识别的效率。

Lin 等人[38]提出了一种轻量级的时序移位模块（Temporal Shift Module，TSM），通过交换相邻帧的部分通道信息来捕捉时序特征，该方法性能接近 3D CNN，但计算复杂度更低，接近于 2D CNN。Shao 等人[39]在此基础上进一步改进，让网络自适应地学习时序移位的方向和大小，从而更精准地捕捉时序动态。Li 等人[40]受（Squeeze-and-Excitation Networks，SENet）[41]的启发，引入了注意力机制，主

要目标在于激活与运动紧密相关的通道特征以扩大模型的感受野，更高效地捕获长期的时序信息。Liu 等人[42]也探索了注意力机制在行为识别领域的应用，不仅增强了与运动相关的特征，还使用通道卷积提取时序的上下文信息，从而提升了行为识别的准确性。这类轻量级方法在计算复杂度上有所优化，推理速度更快，适用于实时应用场景。然而，与基于 3D 卷积的方法相比，这类方法的性能稍逊一筹。

基于 RGB 视频帧的驾驶行为识别方法主要依赖于 2D/3D CNN 或 RNN 技术，其研究核心在于提取有效的动作和时序特征，或利用注意力机制聚焦关键行为区域。Arefin 等人[43]通过修改 AlexNet 网络[44]进行修改并融合方向梯度直方图特征，虽然识别性能略有下降，但是显著减少了模型参数，提升了计算效率。Chuang 等人[45]设计了多视角的驾驶员行为识别系统，该系统能够处理单个摄像头视野受限的问题，并通过 RNN 有效捕捉驾驶员行为的连续时间特性。Lu 等人[46]提出了一种位置敏感的感兴趣区域对齐方法，显著提升了模型对微小目标物体的感知敏锐度，不仅能够有效增强同类特征之间的内在一致性，还能显著区分不同类别特征间的差异性，从而实现了识别精度的提升。Wang 等人[47]针对网络模型识别准确率低的问题，提出了一种基于 CNN 和 LSTM 的危险驾驶行为识别模型，引入无监督关注机制对算法进行优化，使模型能够聚焦于特定的视觉区域。通过整合注意力加权模块和卷积 LSTM，该模型在一定程度上提高了识别精度。当前基于 RGB 视频帧的驾驶行为识别方法多直接沿用通用的行为识别技术，RGB 视频帧虽然能够提供丰富的颜色信息，但在某些情况下，却无法提供有效的场景信息，尤其是在光线不足或环境噪声较大的情况下，视频质量会大打折扣，影响了识别的准确性。

基于 RGB 视频帧和光流的驾驶行为识别方法是从视频中提取出驾驶行为的短期运动信息，从而增强模型的时序建模能力。具体而言，基于 RGB 视频帧和光流的驾驶行为识别方法通常遵循以下流程：首先，从视频帧中抽取光流信息，这是一种描述像素点运动趋势的技术，能够反映出视频中物体的动态变化；然后，利用双流 CNN 模型同步提取 RGB 帧和光流的特征。RGB 帧提供了外观信息，而光流则提供了运动信息。这两个模态的特征在模型中进行融合，确保模型能够同时学习到动作信息和时序信息。Hu 等人[48]提出了一种基于双流网络的驾驶行

为识别方法，该方法中时间流关注光流中的运动信息，而空间流关注视频帧的外观信息。通过优化双流信息的融合策略，这种方法能够更准确地捕捉驾驶行为的特征。在另一项工作中，Hu 等人[49]进一步提出了一个混合时空深度学习框架，结合光流预测技术和编码器-解码器时空卷积神经网络。这种方法不仅能够捕获驾驶员行为的短期时空信息，还能通过特征细化网络对短期特征进行优化，同时引入了 LSTM 来整合长期时空信息。最终，通过全连接神经网络对提取的特征进行分类，实现驾驶员动作的精准识别。尽管这类方法相较于仅使用 RGB 视频帧的方法在精度上有所提升，但也存在一些局限性。首先，光流的计算受摄像头移动的干扰较大，可能导致误差。其次，光流的提取过程耗时较长，增加了计算负担。因此，在追求更高精度的同时，也需要权衡计算成本和实时性要求。

1.3 论文的主要研究内容

针对现有的驾驶员行为识别模型难以平衡速度和精度的问题，本文提出了基于深度学习的驾驶员行为识别方法，可部署于嵌入式平台实现实时识别，主要的研究内容如下：

（1）基于图卷积的多模态融合驾驶员行为识别方法

提出了一种基于图卷积的驾驶员行为识别方法，使用图卷积神经网络对驾驶员行为视频片段进行深度的特征提取，通过捕捉视频中的时空依赖关系，有效地提取出驾驶员行为的内在特征。采用 Openpose 算法[50]对驾驶员的姿态进行精准估计与行为识别，鉴于驾驶过程中的动作主要集中在驾驶员的头部和手部，设计了驾驶员的姿态时空图，并构建了一个专门由驾驶姿态时空图组成的数据集。最后利用全连接层对所提取的特征进行高效的驾驶行为分类，实现了对驾驶员行为的精确识别。由于模型最终部署在 Jetson TX2 嵌入式设备上进行精度和效率分析，需要对模型进行轻量化处理。采用 MobileNet 与空洞卷积对 Openpose 算法进行了针对性的改进，旨在保证识别精度的同时，大幅降低模型的计算复杂度，使其更适应嵌入式平台的运行需求。

（2）基于压缩视频的多分支融合驾驶员行为识别方法

提出了一种基于压缩视频的多分支轻量化驾驶员行为识别框架。首先，直接将压缩视频处理成为三个分支，为了降低计算成本，采用轻量化卷积神经网络进

行时空建模。一个分支通过使用轻量级的二维卷积网络捕获外观信息，另外两个分支采用轻量级三维卷积网络分别从运动向量和残差帧中学习时间信息。其次，设计了跨层连接模块和时空融合模块分别用于融合不同网络分支的中间层特征图和末层特征向量，通过融合多分支结果以得到最终的准确度。最后，引入老师模型引导轻量化模型，从高容量的时空深度学习模型中提炼出互补的知识，将其迁移到所提出的多分支轻量化模型中进一步改善识别精度。

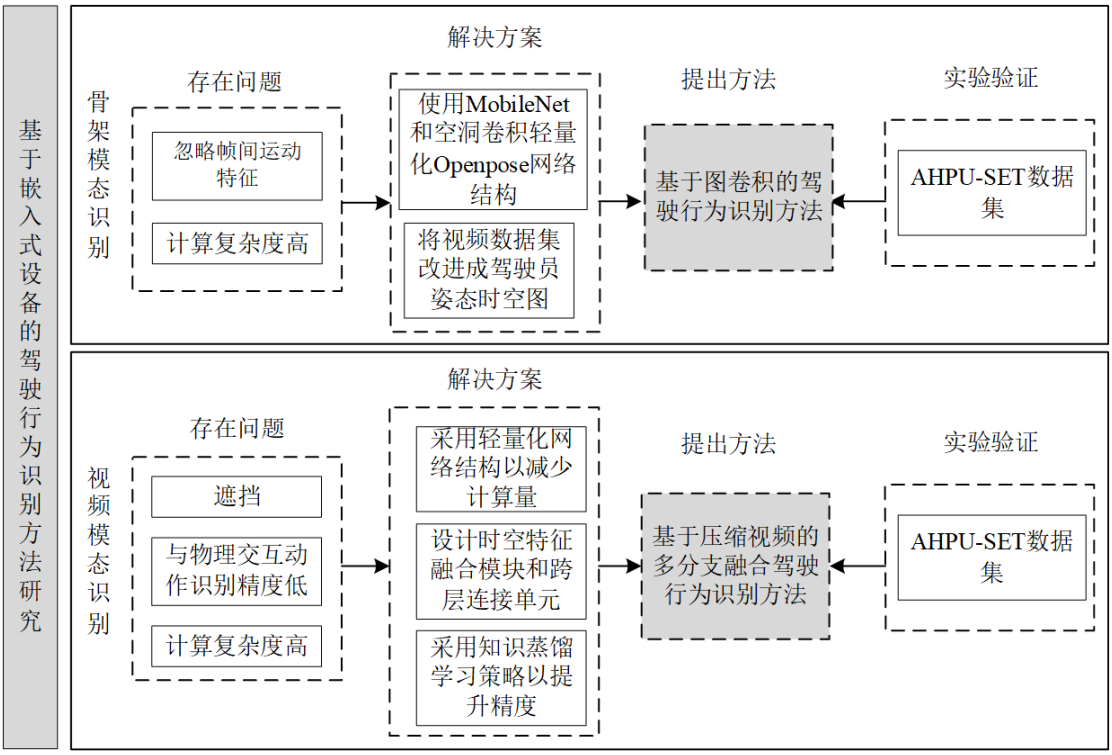


图 1-1 本文技术路线图

Fig 1-1 Technology roadmap for this paper

1.4 论文的结构安排

本文每一章的研究内容如下：

第一章：绪论。介绍了驾驶员行为识别的研究背景与意义，阐述了国内外行为识别以及驾驶员行为识别的研究现状，介绍了论文的研究内容和结构安排。

第二章：相关理论与相关工作。介绍了卷积神经网络中的卷积层、激活层等理论基础、轻量化网络结构、驾驶行为识别任务中常用的主流数据集、评价指标以及嵌入式部署。

第三章：基于图卷积的驾驶员行为识别方法。首先介绍了基于图卷积的驾驶

行为识别方法的算法流程；其次介绍了自建的驾驶员姿态时空图数据集，以及相关的网络结构；然后介绍了实验设计以及相关结果分析；最后进行章节总结。

第四章：基于压缩视频的多分支融合驾驶员行为识别方法。首先介绍了算法目标以及整体的算法流程；其次介绍了算法结构中的各部分组成的设计原理；然后详细描述了算法的实验过程，主要包括超参数设置、各模块有效性验证、以及与现有方法的对比试验；最后进行章节总结。

第五章：总结与展望。简要总结了本文的主要工作，分析了现有的工作存在的问题，展望了未来工作方向。

第 2 章 相关理论及相关工作

本章主要介绍了研究内容中所涉及的相关方法的理论基础，包括卷积神经网络各部分的基本结构以及一些使用的经典网络结构，并对相应网络模型的评价指标和数据集进行了简单的介绍和分析。

2.1 卷积神经网络

卷积神经网络（Convolutional Neural Network，CNN）是一种深度学习模型，其核心思想在于利用卷积操作，有效地从图像等复杂数据中提取出关键特征，进而实现数据的精确分类与识别。CNN 的基本结构如图 2-1 所示，卷积层负责通过一系列卷积核（滤波器）对输入数据进行局部特征提取，捕捉图像中细节信息；池化层则通过下采样操作，减少数据的空间尺寸，同时保留关键特征，增强网络的鲁棒性；而全连接层则负责将前面层提取的特征进行整合，形成最终的分类或识别结果。

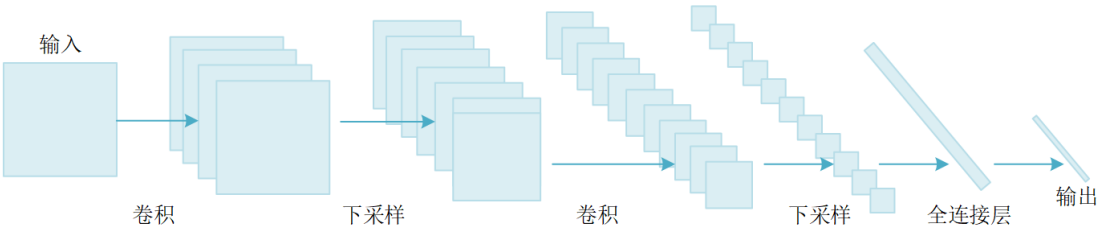


图 2-1 卷积神经网络结构

Fig 2-1 Structure of convolutional neural network

CNN 具有三大显著特性：稀疏连接、参数共享以及平移不变性。首先，稀疏连接特性使得 CNN 在处理图像时能够关注局部特征，使得模型更加高效且易于训练。其次，参数共享特性使同一个卷积核会在输入数据的不同位置进行滑动并共享相同的参数，不仅进一步减少了模型的参数量，还使得模型能够更有效地学习到输入数据的内在规律和结构特征，提取出更具泛化能力的特征表示。最后，平移不变性使得 CNN 对图像的平移变换具有鲁棒性。无论图像中的目标物体如何移动，CNN 都能通过其卷积和池化操作提取出稳定的特征表示，从而实现对目标的准确识别。这些特性在显著降低模型复杂度的同时，也实现了参数量的精简，从而允许图像等类型的数据直接作为模型的输入。此外还突破了传统算法在

复杂特征提取与数据重建方面所面临的难题，为模型处理图像数据提供了更为高效和便捷的方式。

卷积层作为 CNN 的核心组成部分，通过在输入数据上滑动卷积核进行卷积操作，从而有效提取数据的局部特征。卷积操作可以视为一种特殊的加权求和过程，其中卷积核中的权重参数是通过训练学习得到的。如下图 2-2 所示为二维卷积操作，将输入图像视为一个 n 维矩阵，一个 $m \times m$ 维（$m < n$）的卷积核从图像的左上角开始，以一定的步长逐步扫描每个像素，每次扫描时，卷积核与对应位置的区域进行运算，得出一个特征值。当卷积核按照设定的步长遍历整张图像后，就生成了一张特征图。不同的卷积核由于其权重和偏置的不同，能够提取出输入图像中不同维度的特征，这些特征图随后被堆叠起来，共同构成了下一层网络的输入数据。

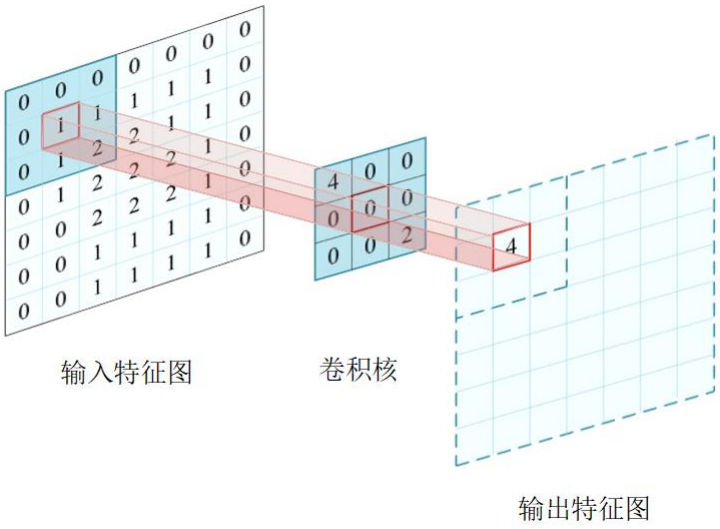


图 2-2 二维卷积计算示例图

Fig 2-2 Example diagram of 2D convolutional computation

卷积层中常伴随着激励层的设置，它的作用在于对卷积核的计算结果执行非线性映射，从而增强卷积神经网络处理复杂数据的能力。下面介绍三种常见的激活函数：

（1）Sigmoid 函数

Sigmoid 函数的输出值域限定在 0 到 1 之间，因此具有归一化神经元输出的效果。当输入为极大的负数时，输出趋于 0；而当输入为极大的正数时，输出则趋近于 1。计算公式如下：

$$\text{Sigmoid}(x)=\frac{1}{1+e^{-x}} \tag{2-1}$$

Sigmoid 函数曲线是一种 S 型曲线，函数图像如下图 2-3。Sigmoid 函数虽在早期深度学习中应用广泛，但其计算量大、易导致梯度消失等缺点限制了其应用。

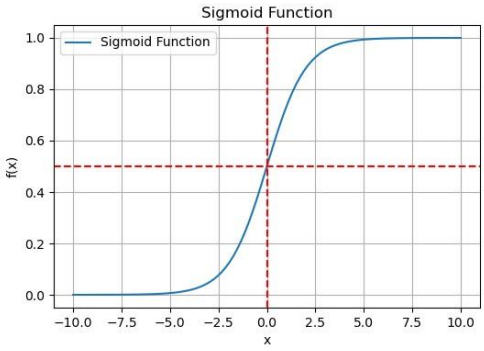


图 2-3 Sigmoid 函数曲线图

Fig 2-3 Plot of sigmoid function

（2）Tanh 函数

Tanh 函数与 Sigmoid 函数在性质上有诸多相似之处，计算公式如下：

$$\text{Tanh}(x)=\frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \tag{2-2}$$

Tanh 函数输出间隔为 1，并以 0 为中心，这使得它在某些方面优于 Sigmoid 函数，函数图像如下图 2-4。然而，当输入值过大或过小时，Tanh 函数的梯度也会变得很小，不利于权重更新。

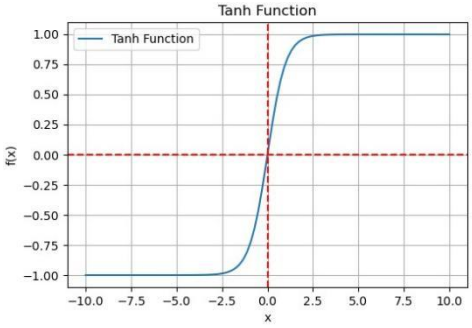


图 2-4 Tanh 函数曲线图

Fig 2-4 Plot of Tanh function

（3）ReLU 函数

ReLU 函数采用取最大值的运算方式，计算公式如下：

$$\mathrm{ReLU}(x)=\begin{cases}x & \text{如果}x>0\\ 0 & \text{如果}x\le 0\end{cases} \tag{2-3}$$

由于 ReLU 函数能使部分神经元输出为 0，从而实现网络的稀疏性，减少参数间的相互依赖，有助于缓解过拟合问题，函数图像如下图 2-5。相较于 Sigmoid 和 Tanh 函数，ReLU 函数的计算速度更快，且能有效解决梯度消失等问题。因此，在深层卷积神经网络中，ReLU 函数或其变体更为常用。

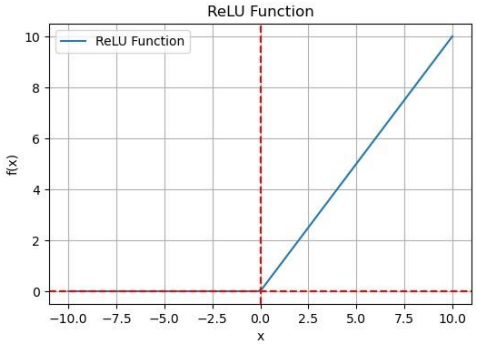
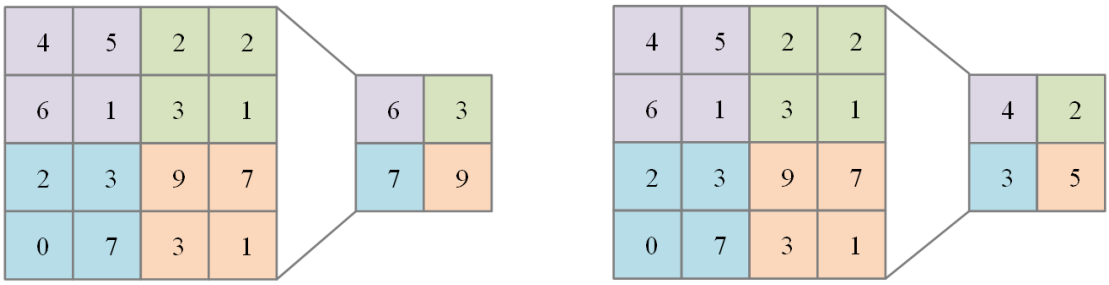


图 2-5 ReLU 函数曲线图

Fig 2-5 Plot of ReLU function

在深度学习的框架中，池化函数同样不可或缺。通过有效地降低输入数据的维度，不仅能够简化模型的复杂度，减少计算量，还能够显著提升模型的鲁棒性和泛化能力。在 CNN 中，池化函数常用于处理卷积层的输出，在保留关键特征的同时，剔除冗余信息，以达到优化模型的目的。常用的池化方式有两种，分别为最大池化和平均池化，这些方式能够有效地提升模型的性能，如下图 2-6 所示。



（1）最大池化　　（2）平均池化

图 2-6 最大池化（左）和平均池化（右）操作示例图

Fig 2-6 Example diagram of maximum pooling (left) and average pooling (right) operations

最大池化通过保留滤波器滑动区域内的最大值，凸显出可能探测到的特定特征，同时忽略其他非关键值，进而减少噪声的干扰，提升了模型的稳健性。此过程仅需设定滤波器尺寸和步进长度这两个超参数，无需其他参数通过模型训练得

出，因此其计算量相对较小。平均池化则是在滤波器算子所覆盖的区域内计算所有值的平均值，同样在特征提取与降维中有重要作用。

全连接层在 CNN 中犹如一个“分类器”，如下图 2-7 所示。卷积层、池化层以及激活函数层将原始数据映射至隐层特征空间，全连接层则通过特定的权重和偏置参数，将池化层输出的特征向量转化为具体的输出类别。这一过程可以视为一种特征的整合与转换，它将隐层特征空间中的分布式特征表示进一步提炼和抽象，最终映射到与任务相关的标记空间。通过这种方式，全连接层不仅增强了模型的表达能力，还使得模型能够更准确地理解和识别输入数据的本质特征，进而实现高精度的分类或识别任务。

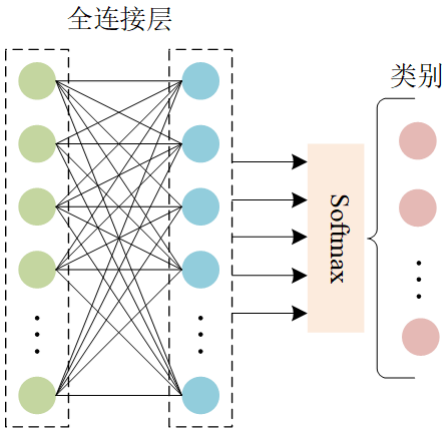


图 2-7 全连接层示意图

Fig 2-7 Schematic diagram of the full connectivity layer

2.2 轻量化神经网络

轻量化神经网络（Lightweight Neural Networks）是通过优化网络结构、减少参数和计算量等手段，使得模型能够在保持一定性能的同时，具备更小的体积和更低的计算复杂度，从而实现在嵌入式设备上的高效部署与推理。近年来，轻量化神经网络在移动端和嵌入式设备的视觉识别、语音识别及自然语言处理等领域得到了广泛应用与研究。

（1）ShuffleNet 系列

ShuffleNet 作为一种轻量级神经网络模型，旨在实现高效的计算性能与准确的预测结果，有效平衡了轻量级模型在准确性与计算效率之间的冲突。其核心创新在于引入通道重排与分组卷积方法，能够显著降低计算复杂度以及参数数量。

自 Megvii 于 2017 年首次提出 ShuffleNet 以来，ShuffleNet 系列不断推出新版本，如 ShuffleNetV2[51]。新版本在保持原有优点的基础上，进一步引入了“多尺度特征图融合”技术，通过将不同尺度的特征图进行融合，增大了模型的感受野，提升了模型的表示能力。

通道重排：首先，将卷积核划分为多个较小组别；随后，在这些小组内分别对输入数据与卷积核进行卷积操作；最后，将各组的卷积结果合并，得到最终输出。在此过程中，通道重排将输入张量按通道数分割为若干子张量，并依据特定规则重新排列这些子张量的通道顺序，再将其拼接。这样，原本分散于不同通道的特征得以有效混合，解决了各组间信息隔离的问题，提升了特征的多样性，进而增强了模型的预测精度。下图 2-9 详细展示了通道重排的操作流程。

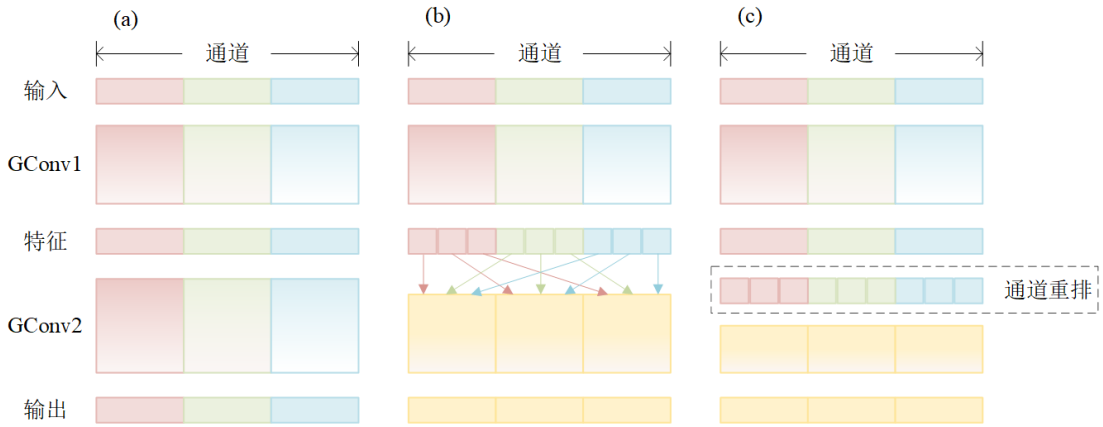


图 2-8 通道重排

Fig 2-8 Channel rearrangement

在上图中，GConv 代表分组卷积操作，其中图 2-8 中(a)表示的是两个叠加的卷积层，具有相同组数，在这种配置下，每个输出通道仅与输入中的特定通道相关，而不同组之间的通道则相互独立。然而，当 GConv2 在 GConv1 之后从各个不同的组中提取数据时，如图 2-8 中(b)，输入和输出通道之间才完全相关。图 2-8 中(c)展示了通道重排如何实现分组卷积中的跨组通道信息获取。具体而言，前一层生成的特征图先被分割成若干子组，随后在下一层中，每个组被赋予不同的子组，从而确保输入和输出通道之间的全面关联。具体而言，在前一层生成的特征图经过处理后，会被划分成数个独立的子组。随后，在后续的卷积层中，这些子组会进行重新的组合分配，以确保输入与输出通道之间建立起全面的连接关系。假设有一个包含了特定组数的卷积层，并且这些组对应的输出通道数是已知的。

为了进行通道重排，首先需要对输出通道的维度进行重新塑形，将其转变为一个特定的形状，接着通过转置操作，进一步调整这些通道的顺序，使其适应下一层的输入要求，最后经过展平处理，得到一个适合作为下一层输入的数据结构，从而完成了整个通道重排的过程。

分组卷积：其中输入数据被均等地划分为若干组，每组数据都与其对应的卷积核进行独立的卷积运算，生成相应的输出。这些输出随后按照一定顺序组合，形成最终的卷积结果。在分组卷积中，分组数是一个关键参数，它决定了输入数据和卷积核的分组方式。每个组内的卷积核大小和数量保持一致，同时组内输入数据的通道数也相同。然而，不同组之间的卷积核和输入数据的特性可以有所差异。下图 2-10 详细展示了分组卷积的具体操作流程。

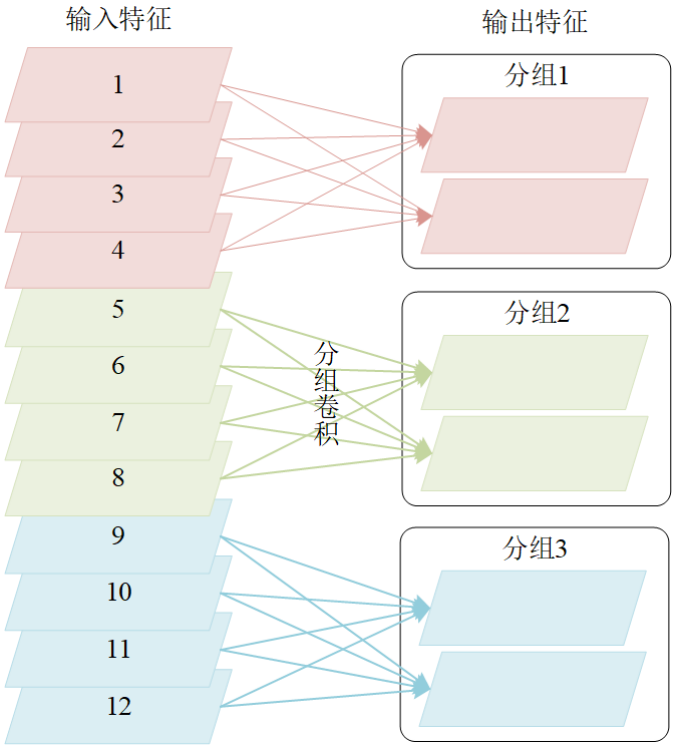


图 2-9 分组卷积

Fig 2-9 Group convolution

假设输入特征图的大小为$W \times H \times C$，分别对应特征图的宽、高和通道数，单个卷积核尺寸为$k \times k \times C$，分别对应卷积核的宽、高和通道数，输出的特征图尺寸为$W^{'} \times H^{'}$，输出的宽和高与卷积的步长相关，输出通道数等于卷积核个数。常规卷积的参数量为k^2C，运算量为$k^2CW^{'}H^{'}$。将卷积的输入特征图分为g个组，每个卷积核也相应的分为g个组，在对应的组内做卷积，每组卷积生成一个特征

图。则输入每组特征图尺寸为$W \times H \times C / g$；单个卷积核每组的尺寸为$k \times k \times C / g$；输出特征图尺寸为$W^{'} \times H^{'} \times g$，共生成$g$张特征图。分组卷积的参数量为$k^2 \times C / g \times g = k^2 C$，运算量为$k^2 \times C / g \times W^{'} \times H^{'} \times g = k^2 C W^{'} H^{'}$。综上可知，分组卷积使用少量的参数量和运算量就能生成g张特征图，这就意味着能够编码更多的信息。然而，分组卷积存在一个明显缺陷：当输入数据被分割成多份时，各组之间的信息流通被阻断，每组数据变得相互独立，缺乏关联性。所以引入通道重排技术，增强了各组之间的信息交互，解决了分组卷积中信息孤立的问题，提高了模型的性能。

（2）MobileNet 系列

MobileNet[52]是一种创新的轻量级卷积神经网络架构，其核心在于深度可分离卷积技术的运用，将复杂的卷积任务分解为更小的单元，从而显著降低了计算量和参数量。这一创新理念源自致力于卷积神经网络研究的 Inception 团队，他们在此领域取得了显著成果。2014 年，该团队率先提出了 Inception 模块，并在此基础上持续探索和优化，其中深度可分离卷积便是其一项重大突破，在轻量级神经网络设计领域具有显著的应用价值。其应用主要体现在两个方面，均有效降低了模型的计算量和参数量，为模型轻量化提供了有力支持。首先，深度可分离卷积通过将卷积核的通道数进行分离，分解为深度卷积和逐点卷积两个步骤，显著减少了所需参数量。深度卷积专注于在每个通道内进行独立的卷积操作，而逐点卷积则负责将不同通道的信息进行融合。这种分离式设计使得每个步骤的参数量得到有效控制，进而实现了模型的整体轻量化。这种轻量化设计使得模型在保持较高性能的同时，能够减少需要占用的存储空间以提高运算速度，从而更适用于资源受限的环境。其次，深度可分离卷积在深度卷积阶段，对每个像素位置上的每个通道进行独立的卷积操作。这种局部化的处理方式使得每个卷积核只需要处理较少的空间点数，进一步降低了计算量和参数量。这种处理方式不仅提高了计算效率，还有助于模型更好地捕捉图像的局部特征。局部特征的准确提取对于许多计算机视觉任务至关重要，例如目标检测、图像分割等。因此，深度可分离卷积的应用有助于提升这些任务的性能。

以一个例子来说明，首先假设输入特征图的大小为$Cin \times H \times W$，滤波器大小为$Cin \times k \times k$，所得输出特征图尺寸假定不变，大小为$Cout \times H \times W$，则常规卷积

操作需要的参数总数量为$Cin \times Cout \times H \times W$。如下图 2-11 所示，常规卷积操作作用于一个大小为64×64像素、三通道彩色图片输入层，通过一个由 4 个滤波器组成的卷积层，生成 4 个与输入层尺寸相同的特征图，可以计算出卷积层的参数总数量是$3 \times 4 \times 3 \times 3 = 108$。

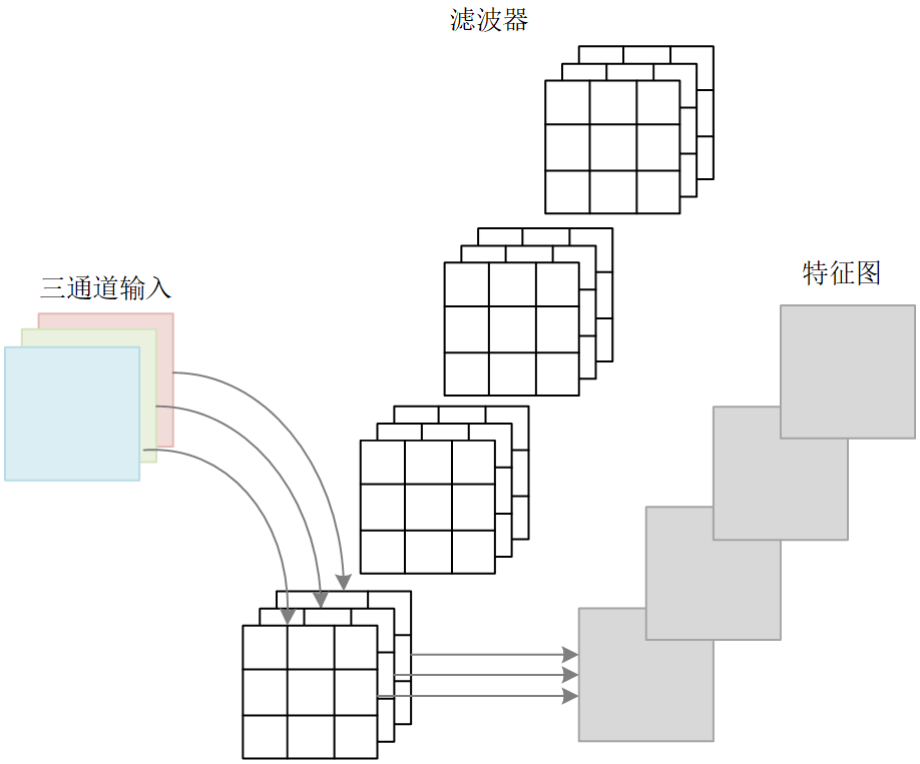


图 2-10 常规卷积操作

Fig 2-10 Operation of conventional convolution

深度可分离卷积包含深度卷积和逐点卷积两个步骤，如下图 2-11 所示。假设深度卷积的滤波器大小为$Cin \times k \times k$，逐点卷积的滤波器大小为$Cin \times 1 \times 1$，则需要的参数总数量为$Cin \times k \times k + Cin \times Cout$。如下图 2-11 所示，深度可分离卷积的操作过程分为两部分：一是逐层卷积，针对每个输入通道进行处理；二是逐点卷积，对输出特征图进行通道间的信息融合。在此例中，滤波器数量与输入层的通道数相同。经过计算，三通道图像经深度卷积可生成 3 个特征图，每个特征图的参数数量为$3 \times 3 \times 3 = 27$，随后通过1×1卷积技术，这些特征图被进一步组合，生成新的输出特征图。此步骤中，卷积核的尺寸为$1 \times 1 \times M$，M 为上一层的通道数。若有 N 个卷积核，则生成 N 个输出特征图，此步骤参数量为$1 \times 1 \times 3 \times 4 = 12$。两部分参数相加，总量为 39，远少于常规卷积的 108 个参数。对比可见，深度可分离卷积能够显著减少参数量，仅为常规卷积的约 1/3。

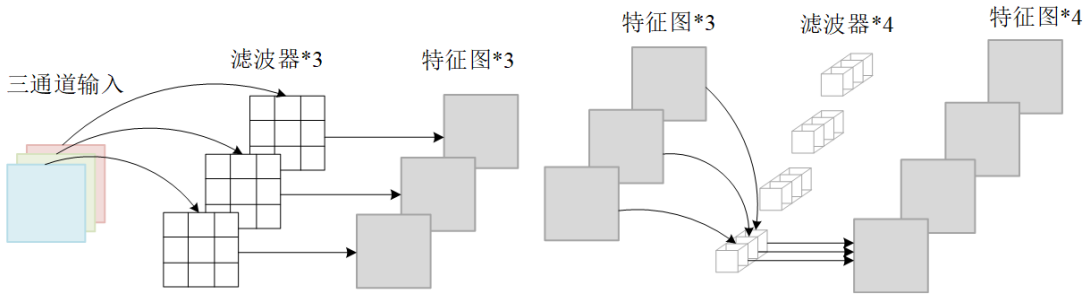


图 2-11 逐深度卷积和逐点卷积

Fig 2-11 Depth-by-depth and point-by-point convolution

MobileNet 系列算法，作为一种轻量化神经网络方法，旨在提供高效的图像分类和目标检测模型。该系列由 Google 开发，适用于移动设备等资源有限的环境，实现高质量的计算机视觉任务。MobileNet 系列通过采用深度可分离卷积和残差连接等技术，有效减少了网络参数和计算量。MobileNetV1 作为系列的开山之作，利用深度可分离卷积降低计算复杂度和参数数量，并将标准卷积核替换为特定尺寸的卷积核。随着 MobileNet 系列的不断发展和优化，先后推出了 MobileNetV2[53]和 MobileNetV3[54]等版本，进一步优化了网络结构和性能。这些版本在保持轻量级特性的同时，通过引入新的优化策略和技术，提高了模型的准确性和效率。MobileNetV2 采用了倒置残差结构和线性瓶颈层等设计，进一步减少了参数量和计算量；MobileNetV3 则结合了自动化搜索技术和硬件感知优化，实现了更高的性能和更低的延迟。MobileNet 系列算法在人脸识别、图像分类、目标检测等领域有着广泛应用。特别是 MobileNetV2 在 Imagenet 图像分类任务中，达到了与 ResNet-50 相当的准确率，但参数量和计算量仅为 ResNet-50 的约 1/14 和 1/34，使其成为资源受限环境中实现高效计算机视觉任务的重要工具。

2.3 评价指标及数据集

2.3.1 评价指标参数

在本文中采用了多种指标来全面评估模型的性能。首先，本文利用准确率（Accuracy）这一指标来衡量模型的预测精度。准确率指的是模型预测结果中，预测概率最高的动作类别与真实动作类别一致的样本数占总样本数的比例。这一指标直观地反映了模型在分类任务上的表现，是评估模型性能的关键指标之一，计算公式表示为：

$$\text{Accuracy} = \frac{TP + TN}{N} \tag{2-4}$$

其中 N 表示样本总数量，TP 表示将正类样本正确预测为正类的数量，TN 表示将负类样本正确预测为负类的数量。

此外，为了更深入地了解模型的计算量和运算复杂度，本文采用了参数量（Parameters）和浮点运算次数（Floating Point Operations, FLOPs）这两个指标。参数量是模型中所有可训练参数的总和，它直接反映了模型的复杂度和存储需求。而浮点运算次数则是指模型在进行一次前向传播时所进行的浮点运算的数量，它反映了模型的计算复杂度。

最后，为了评估模型的计算速度，本文采用了每秒处理的图片数（Frame Per Second，FPS）以及每秒处理的视频片段数（clips/s）这两个指标。FPS 表示模型每秒能够处理的图片数量，而 clips/s 则表示模型每秒能够处理的视频片段数量。这两个指标能够直观地反映模型在实际应用中的处理速度，对于实时性要求较高的应用场景尤为重要。这些指标共同构成了本文对模型性能的全面评价体系。

2.3.2 驾驶员行为识别数据集

下面简要介绍现有的驾驶员行为识别数据集。

（1）StateFarm 数据集

Kaggle 数据平台在 2016 年开源了 StateFarm 数据集。该数据集汇聚了众多在真实驾驶环境下捕捉的静态图片，总量超过十万张，全面反映了驾驶过程中可能出现的多种场景。这些图片均通过车载摄像头捕捉，确保了数据的真实性和准确性。在数据集中，带有明确标签的训练图片共有 22424 张，为模型训练提供了丰富的素材。同时，未带标签的测试图片公有 79726 张，可用于评估模型的泛化性能。每张图片的分辨率均达到了 640×480，既保证了图像的清晰度，又满足了模型训练对图像质量的高要求。根据驾驶员的行为和动作特征，将数据集划分为十个类别（C0 安全驾驶、C1 右手玩手机、C2 右手接电话、C3 左手玩手机、C4 左手接电话、C5 调试电台、C6 喝水、C7 向后伸手、C8 打扮、C9 与乘客交谈）。

图 2-12 StateFarm 数据集样例

Fig 2-12 Sample of the StateFarm dataset

（2）数据集

开罗美国大学制作了一个分心驾驶行为识别的 AUC 数据集，分类体系沿用了 StateFarm 数据集，涵盖了从 C0 至 C9 的十类驾驶行为，其中 C0 代表正常驾驶，而 C1 至 C9 则分别指代了各种异常驾驶行为。该数据集共收录了 11678 张静态图像，这些图像全部由 31 名志愿者参与录制，确保了数据的真实性和多样性。在数据的划分上，AUC 数据集将图像划分为训练集和测试集两个部分，其中训练集部分包含 10555 张图像，用于模型的训练和优化；测试集部分包含 1123 张图像，用于模型的评估。每张图像的像素尺寸达到了 1920×1080，保证了图像质量的细腻度和清晰度。

图 2-13 AUC 数据集样例

Fig 2-13 Sample of the AUC datasets

（3）AHPU-SET 数据集

以上两种数据集均为图像数据集，只能提供静态的瞬间画面，无法反映驾驶行为的连续性和动态变化。本文参考 StateFarm 数据集设计了一个遵循相似行为的类似视频数据集并命名为 AHPU-SET，包括 6 类驾驶员行为分别为 C0 正常驾驶、C1 喝水、C2 化妆、C3 脱离方向盘、C4 吸烟、C5 接电话。AHPU-SET 数据

集其中的视频是在车内记录的，光照条件不同，视角变化，90 位志愿者参与采集了 8301 段视频，每段视频的帧率为 30FPS，平均时长为 10 秒。

表 2-1 AHPU-SET 数据集每个驾驶员行为类别的视频个数

Table2-1 Number of videos per driver behavior category in the AHPU-SET dataset

类别	训练	测试
正常驾驶	848	560
喝水	963	592
化妆	937	623
脱离方向盘	633	431
吸烟	735	456
打电话	902	621
总计	5018	3283

图 2-14 AHPU-SET 数据集样例

Fig 2-14 Sample of the AHPU-SET dataset

2.4 嵌入式部署

英伟达推出的 Jetson TX2 是一款高性能的嵌入式人工智能计算机，专为边缘计算和人工智能项目设计。其卓越的计算能力和丰富的存储资源，使得它能够轻松应对复杂的深度学习任务，并支持多传感器处理和高性能计算。Jetson TX2 的核心模块在尺寸上进行了极致的压缩，仅50×87毫米的尺寸使其在小型化方面达到了新的高度。同时，其重量也控制得相当出色，仅为 85 克，轻巧便携的特点使其能够灵活部署在各种环境中。此外，其低功耗设计也是一大亮点，7.5 瓦的功率消耗使得 Jetson TX2 在长时间运行时能够保持稳定的性能，同时降低能源消耗。英伟达在 TX2 上预装了 Linux 开发环境，为开发者提供了一个稳定且强大的工作平台。利用 JetPack3.1 安装包，TX2 的配置环境得到了进一步优化，其中包含 TensorRT、CUDA 等工具包。下表 2-2 展示了 Jetson TX2 开发板参数。

表 2-2 Jetson TX2 开发板参数

Table2-2 Parameters of Jetson TX2 development board

参数	Jetson TX2
图形处理单元	NVIDIA Pascal™ 架构、配有 256 个 NVIDIA CUDA 核心
中央处理器	HMP Dual Denver 2/2MB L2+ Quad ARM® A57/2MB L2
内存	8GB 128 位 LPDDR4 59.7GB/s
数据存储	32GB eMMC、SDIO、SATA
视频	4K×2K 60Hz 编码（HEVC）、4K×2K 60Hz 编码（支持 12 位）
显示控制器	2 个 DP1.2 接口/HDMI 2.0 接口/eDP 1.4 接口、2 个 DSI 接口
摄像头	最多 6 个摄像头（双通道）CSIS D-PHY 1.2（2.5Gbps/通道）
网络连接	1 千兆以太网、802.11acWLAN、蓝牙

2.5 本章小结

本章首先介绍了卷积神经网络相关原理，其次介绍了经典轻量化卷积神经网络结构，最后介绍了本文使用的评价指标、数据集 StateFarm、AUC 和 AHPU-SET 以及嵌入式部署。

第 3 章 基于图卷积的驾驶员行为识别方法研究

3.1 引言

在驾驶员监控系统中，一些高性能的嵌入式设备被有效地应用于驾驶座舱环境之中，这些设备之所以备受青睐，主要得益于其高度的集成度和出色的便捷性，然而它们同样存在着一些局限，如内存空间相对较小以及处理速度有待提升等缺陷。目前主流的行为识别解决方案虽在功能强大的服务器平台上表现出色，但计算复杂度高的问题限制了其在实时嵌入式系统中的应用。故针对部署于嵌入式平台的实际需求，驾驶员行为识别的算法必须在精度与效率之间取得平衡。近年来，随着技术的不断进步，人体姿态估计算法的性能得到了显著提升，这使得基于骨架的行为识别任务逐渐成为研究领域的热点。骨架数据以其独特的非欧几里得结构特性而备受关注，仅包含二维或三维的人体关节点位置坐标。这种低维度的数据表示方式不仅大幅降低了运算的复杂度和计算量，还满足了嵌入式设备对高效处理的迫切需求。

在本章中，鉴于驾驶员动作主要集中在上半身的手部和头部，设计了一种驾驶员的姿态时空图，并将数据集构建为一系列驾驶姿态时空图序列。为了实现更精确的驾驶员行为识别，提出了一种创新的基于图卷积的识别方法。该方法的核心在于利用图卷积神经网络对驾驶员行为视频进行特征提取，在此基础上，通过引入全连接层对提取到的特征进行进一步的精细化处理，从而实现更为精确的行为分类。考虑到实际应用中需要部署在 Jetson TX2 嵌入式设备上的限制，结合了 MobileNet 与空洞卷积的技术对 Openpose 算法进行了轻量化处理，实现了算法的高效运行和内存占用优化。在实验部分，验证了该算法在准确性、实时性和鲁棒性方面的出色表现，证明了其在驾驶员行为识别领域的实际应用价值。

3.2 基于骨架信息的图卷积神经网络模型构建

本章所构建模型的流程图如下图 3-1 所示。首先，以包含驾驶员的视频片段作为输入数据，通过应用 Openpose 算法对视频中的驾驶员姿态进行详尽的估计。

这一过程中，Openpose 算法能够精确地提取每个视频帧中驾驶员的关节点信息，这些信息不仅包括关节点的位置坐标，还涵盖了对应的置信度。基于此，按照驾驶员的时空姿态运动规律，构建出驾驶行为时空图。这一步骤对于捕捉驾驶员的动态行为模式至关重要，能够有效地反映出驾驶员在驾驶过程中的姿态变化和运动轨迹。为了从构建的驾驶行为时空图中提取出有用的特征，进一步采用了图卷积网络（Graph Convolutional Network，GCN）和时空卷积网络（Temporal Convolutional Network，TCN）的交替使用。GCN 擅长处理非欧几里得结构的数据，能够有效地捕捉驾驶员姿态之间的空间关系，而 TCN 则擅长捕捉序列数据中的时间依赖关系。通过交替使用这两种网络，对空间和时间维度进行特征变换，提取出丰富的特征信息。在特征提取完成后，进一步通过平均池化操作对特征信息进行压缩，以减少冗余信息并突出关键特征。最后，利用全连接层进行驾驶员行为识别，将提取的特征映射到具体的行为类别上。为了提升模型的整体运行速度并满足嵌入式设备的部署需求，采用了 MobileNet 与空洞卷积对 Openpose 网络结构进行轻量化处理。MobileNet 以其轻量级的网络结构和高效的运行速度，提供了理想的模型基础；而空洞卷积能够在不增加参数数量的同时，扩大模型的感受野，进一步提升模型的性能。算法整体的框架如下图所示：

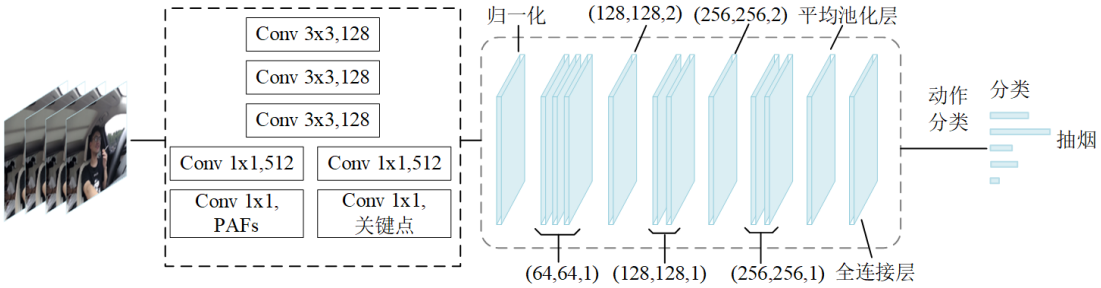


图 3-1 算法流程图

Fig 3-1 Overview diagram of algorithm

3.2.1 驾驶员姿态时空图

Openpose 算法是一种自下而上的人体姿态估计算法，其独特之处在于先识别出人体所有关键点的位置，随后通过连接这些关键点形成完整的人体骨架。这种算法能够保持实时性能，即使在多人场景下也不会受到人数增多的影响。关键在于 Openpose 采用了名为 Part Affinity Fields（PAFs）的创新方法。PAFs 实际上

是一种矢量场，它精确表示了不同关键点之间的位置和方向关系，为关键点的正确连接提供了有力支持。Openpose 的网络结构能够同时输出关键点的位置信息和与之对应的 PAFs 信息。这种设计显著减少了在构建人体骨架过程中由关键点连接所产生的误差，进而增强了姿态估计的精确性与稳定性。当视频帧被输入系统后，算法通过两个并行的卷积神经网络分支进行处理，这两个分支分别用于捕获关键点的置信度分布和部分亲和字段分布。接着，算法依据部分亲和字段分布中的信息，以二分图匹配的方式对关键点进行精准配对，从而精细地描绘出人体姿态。

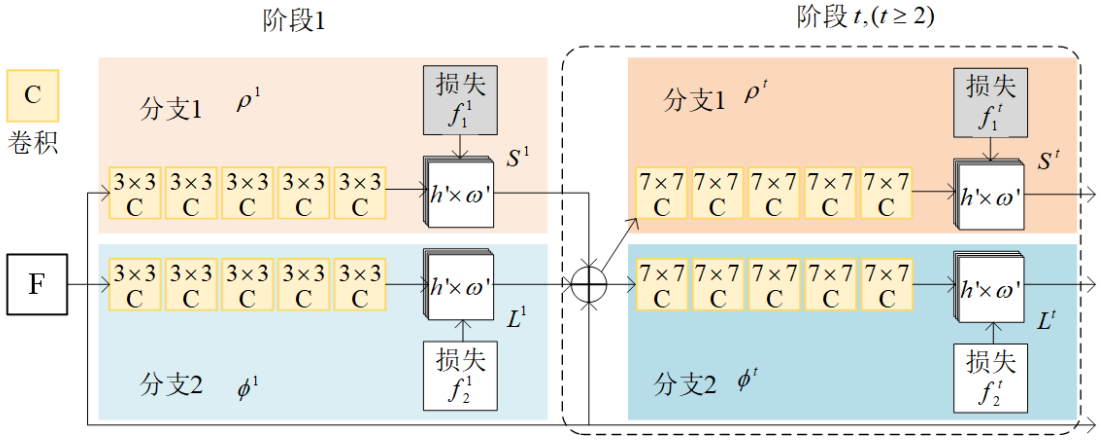


图 3-2 Openpose 双分支网络结构图

Fig 3-2 Structure diagram of Openpose twin-branch network

两个分支的 Openpose 网络结构如上图 3-2。橙色分支预测关节点置信度图，蓝色分支预测人体部分亲和字段。这两个分支共同构成了 Openpose 网络模型的第一阶段。在第一阶段中，输入视频帧首先通过卷积神经网络提取像素特征，生成特征映射 F，然后作为两个分支的输入。两个分支分别利用这些特征进行关节点和亲和度向量场的预测，输出关节点置信度图和人体部分亲和字段。

定义的关节点置信度图由 J 个子图组成，表示为 $S=(S_1,S_2,...,S_J)$，数量即为人体骨骼关节点个数。每个子图均表示特定关节在图像像素点中出现的概率分布。当视频帧中仅存在单一人物时，每个关节点置信度图中将呈现单一的峰值；若存在 k 个人物，则关节点 j 的置信度图中将出现 k 个峰值。根据标注的关节点位置生成这些置信度图 S^* 作为监督学习的标签，其中第 k 个人第 j 个关节点的置信图 $S_{j,k}^*$ 表示为：

$$S_{j,k}^{*}(p)=\exp(-\frac{\| p-x_{j,k} \|_2^2}{\sigma^2}) \tag{3-1}$$

$x_{j,k}\in\mathbb{R}^2$ 表示视频帧中第 k 个人第 j 个关节点的像素位置，使用参数 σ 控制峰值在关键点置信图内的分布范围，通过取各个单独关键点置信图的最大值进行聚合得到整体的置信图，公式表示为：

$$S_j^{*}(p)=\max_k S_{j,k}^{*}(p) \tag{3-2}$$

定义的人体部分亲和字段，用来衡量每对人体部分之间的相关性。该字段保存了肢体支持区域中的位置和方向信息，由 C 个向量图组成的 $L=(L_1,L_2,...,L_C)$，与骨骼连接数相等，其中的每个向量图都记录了一个特定骨骼连接的二维空间方向。在像素点 p 表示亲和域（PAFs）为：

$$L_{c,k}^{*}(p)=\begin{cases} v & \text{p落在第k个人的第c肢体上} \\ 0 & \text{其他} \end{cases} \tag{3-3}$$

上式中像素点 p 所在肢体方向上的单位向量为 $v=(x_{j_2,k}-x_{j_1,k})/\| x_{j_2,k}-x_{j_1,k} \|_2$，若点 p 在肢体 c 上，点 p 的 PAFs 是两个关键点之间的一个单位向量。像素点 p 的取值范围为 $0\le v\cdot(p-x_{j_1,k})\le l_{c,k}$ 和 $|v_{\perp}\cdot(p-x_{j_1,k})|\le\sigma_l$，通过取所有个体亲和域的平均值得到每张视频帧中的亲和域，表示为：

$$L_c^{*}(p)=\frac{1}{n_c(p)}\sum_k L_{c,k}^{*}(p) \tag{3-4}$$

上式中 $n_c(p)$ 是在像素点 p 处第 k 个人第 c 肢体的非零向量。为了评估两个候选关键点 d_{j_1} 和 d_{j2} 之间的相关性进行如下线积分运算：

$$E=\int_{u=0}^{u=1} L_c(p(u))\cdot\frac{d_{j_2}-d_{j_1}}{\| d_{j_2}-d_{j_1} \|_2}du \tag{3-5}$$

上式中 $p(u)=(1-u)d_{j_1}+ud_{j2}$ 表示两个关键点 d_{j_1}、d_{j2} 之间插入的位置。

Openpose 第一阶段中，双分支网络预测的关节点置信度图 $S^1=\rho^1(F)$ 和人体部分亲和字段 $L^1=\phi^1(F)$ 将作为后续阶段的输入，其中 ρ^1、ϕ^1 代表第一阶段的卷积前向运算。整个 Openpose 网络模型由 t 个阶段组成，在每个阶段，网络都会输出更新后的关节点置信度图和人体部分亲和字段，后续每个阶段的输入都是原始图像特征与前两个分支预测结果的融合，输出 S^t、L^t 分别表示为：

$$S^t = \rho^1(F, S^{t-1}, L^{t-1}), \forall t \geq 2 \quad (3\text{-}6)$$

$$L^t = \phi^t(F, S^{t-1}, L^{t-1}), \forall t \geq 2 \quad (3\text{-}7)$$

为了优化网络预测的准确性，在每个阶段都引入了两种 L_2 损失函数，这两种损失函数分别针对两个分支预测进行迭代优化，在第 t 阶段分别为：

$$Z_S^t = \sum_{j=1}^{J}\sum_{p} W(p)\cdot \| S_j^t(p) - S_j^*(p) \|_2^2 \quad (3\text{-}8)$$

$$Z_L^t = \sum_{c=1}^{C}\sum_{p} W(p)\cdot \| L_C^t(p) - L_C^*(p) \|_2^2 \quad (3\text{-}9)$$

其中 S_j^*、L_C^* 分别是标注的关节点置信度图和人体部分亲和字段。此外，针对数据集中存在的人体姿态漏标问题，引入空间像素加权机制。当像素点 p 未标注的时，令权重 $W(p)=0$，以避免因漏标而导致的错误惩罚。最终各阶段的损失组成整个网络的损失表示如下：

$$Z = \sum_{t=1}^{T}(Z_S^t + Z_L^t) \quad (3\text{-}10)$$

在驾驶员姿态估计的过程中，由于驾驶员所处空间的特殊性以及摄像头视角的限制，某些关键点的检测存在较大的不稳定性。特别是左胯、右跨和右膝等下半身关键点的检测，往往受到驾驶员坐姿、衣物遮挡以及摄像头角度等因素的影响，导致检测结果不够准确和稳定，从而影响了 Openpose 对驾驶员行为的准确分析，本章特别设计了一种专注于上半身 12 个关键关节点的驾驶员姿态估计图，这些关键点包括头部、颈部、肩部、肘部以及手部等关键部位，它们对于描述驾驶员上半身姿态具有重要意义，而下半身如右跨、左胯、右膝、左膝、右脚和左脚等关键点的检测则被暂时忽略。Openpose 关键点置信图集合表示如下：

$$S_{上} = \{S_{j,k}^* \in S_k^* \mid 1 \leq j \leq 12\} \quad (3\text{-}11)$$

在实际应用中，车内摄像头捕捉的视频帧中可能不仅包含驾驶员，还可能有乘客或外部行人。这些多余人员的姿态估计图会产生冗余人体信息，对驾驶员行为的识别造成干扰。如下图 3-3，当使用 Openpose 对车内环境进行人体姿态估计时，摄像头图像中后座出现非驾驶员人员，产生了多余的姿态估计图，从而引入冗余的人体信息并导致识别错误。

图 3-3 驾驶员姿态估计图

Fig 3-3 Estimation diagram of driver attitude

驾驶员作为一个关键的角色，往往因其距离摄像头的最近性和在视频中的显著占比，使得其姿态估计的精准度通常达到最高水平。鉴于驾驶员姿态估计的高精准度，采用了一种基于平均置信度的筛选方法，核心思想是通过计算每个人体姿态估计图的平均置信度，选取其中置信度最高的图像作为驾驶员的姿态估计图。驾驶员关键点集合可以表示为：

$$S_{驾} = \max(\frac{1}{j} \cdot \sum_{j} S_{上}) \tag{3-12}$$

第二章所提及的数据集主要聚焦于原始的视频剪辑，然而未包含对于算法至关重要的骨架数据。为了方便后续的数据处理与分析工作，将视频片段的数据集转换为以 JSON 格式存储的驾驶员姿态时空图。这种格式的转换使得数据更加直观、易于处理，并且便于在不同平台和工具之间进行数据共享和交换。转换过程首先从视频剪辑的每一帧开始，利用 Openpose 算法精准提取驾驶员的姿态估计图，这些姿态估计图包含了驾驶员在驾驶过程中的各种动作和姿态信息。本章主要关注驾驶员的上半身，包含 12 个关键点，其中每个关键点都包含二维位置信息 (x, y) 和置信度 acc 三个方面的信息。故每位驾驶员的姿态估计图可以表示为一个 $(3,12,1)$ 维度的张量形式，这样的数据结构既直观又易于处理。然而，仅仅提取出驾驶员的姿态估计图还不足以构建一个完整的驾驶员姿态时空图。为了确保数据的一致性和可比性，需要对所有的驾驶员姿态估计图进行归一化处理，能够消除不同视频片段中可能存在的尺度差异，使得算法能够更加稳健地应对各种驾驶场景。归一化处理之后，包含 T 帧的驾驶员行为视频片段便可以表示为一个 $(3,T,12,1)$ 维度的张量。该张量不仅包含了驾驶员在时间序列上的姿态变化，还

融入了空间上的位置信息，提供了一个全面的驾驶员姿态时空图。在转换过程中，每个视频片段的 JSON 文件都由多帧组成，每帧都详细记录了帧索引和骨架信息。骨架信息则包括经过归一化处理的二维坐标以及对应的置信度。值得注意的是，若某一关键点在某一帧中未被检测到，则会在对应位置留空，以确保数据的完整性和真实性。数据采集完成后，需要进一步对数据集进行标注工作。第一步，将以 JSON 格式存储的驾驶员姿态时空图按照的7:3比例随机分成训练集和测试集。这样的划分既保证了训练集的充足性，又确保了测试集的有效性，使得算法能够在不同的数据集上得到充分的验证。第二步，进行相应训练集和测试集标签文件的编制工作。每个视频片段都被分配了唯一的文件名，并根据视频片段的内容，为其分配了相应的行为标签。通过比对文件名和标签，可以将每个视频片段与其对应的行为标签进行匹配，从而确保后续数据处理和分析的准确性。在本章中，标签 C0—C5 分别对应正常驾驶、喝水、化妆、脱离方向盘、吸烟和打电话等六种不同的驾驶行为。

3.2.2 轻量化 Openpose 网络

传统的 Openpose 网络规模庞大，消耗的内存和计算资源显著，因此并不适宜直接部署在资源有限的嵌入式设备上。鉴于此，对 Openpose 进行轻量化处理成为必要步骤，以适应嵌入式设备的运行环境。在 Openpose 中，网络的核心在于其主干特征提取部分，该部分主要借鉴了 VGG-19 网络的前四个模块，并在其后附加了两个卷积层，以增强特征的提取能力。随后，进入了一个连续的六个阶段处理过程，这些阶段包括一个初始化阶段以及五个逐步细化的阶段。每个细化阶段内部设计了两个分支，一个分支专注于生成关键点的置信图，用以定位图像中关键点的位置；而另一个分支则负责生成亲和域的置信图，用以描绘关键点之间的连接关系。轻量化 Openpose 的核心思想在于精简流程，去除不必要的处理阶段，仅保留最为关键和有效的部分。根据下表 3-1 数据得知，即便仅采用一个细化阶段，Openpose 的性能表现已能达到 43.4%的较高水平。然而，当细化阶段增加至五个时，虽然性能有所提升，达到 48.6%，但这一提升幅度并不显著。随着细化阶段的增加，运算量几乎翻了一番，这无疑增加了模型的计算负担。因此，在轻量化 Openpose（lightweight Openpose）的设计中，仅保留初始化阶段和第一

个细化阶段，实现了运算量的减少和运算速度的提升。初始化阶段负责从输入图像中提取基本的姿态特征，为后续处理奠定基础；而细化阶段则负责对初始特征进行进一步的优化和调整，以得到更为准确和精细的姿态估计结果。

表 3-1 不同个数细化阶段对精度和运算量的影响

Table 3-1 Impact of different number of refinement stages on accuracy and arithmetic volume

	精度	运算量	总运算量
主干网络	n/a	37.8 GFLOPs	37.8 GFLOPs
Conv4_3	n/a	2.5 GFLOPs	40.8 GFLOPs
Conv4_4	n/a	0.6 GFLOPs	40.9 GFLOPs
初始化阶段	35.5%	2.2 GFLOPs	43.1 GFLOPs
细化阶段 1	43.4%	18.6 GFLOPs	61.7 GFLOPs
细化阶段 2	46.2%	18.6 GFLOPs	80.3 GFLOPs
细化阶段 3	47.4%	18.6 GFLOPs	98.9 GFLOPs
细化阶段 4	48.1%	18.6 GFLOPs	117.5 GFLOPs
细化阶段 5	48.6%	18.6 GFLOPs	136.1 GFLOPs

为了进一步实现网络结构的轻量化，采用 MobileNet 替代 VGG-19 作为网络主干，进一步减小模型规模。然而，MobileNet 的网络结构相对较浅，导致感受野较小。在姿态估计任务中，为了准确回归骨骼点，不仅需要关注点附近的局部信息，还需要捕捉更大范围的全局信息，以减少误差。此外，骨骼点的回归过程本身能够学习到一定的骨骼连接结构，因此感受野的大小对姿态估计算法的性能具有重要影响。

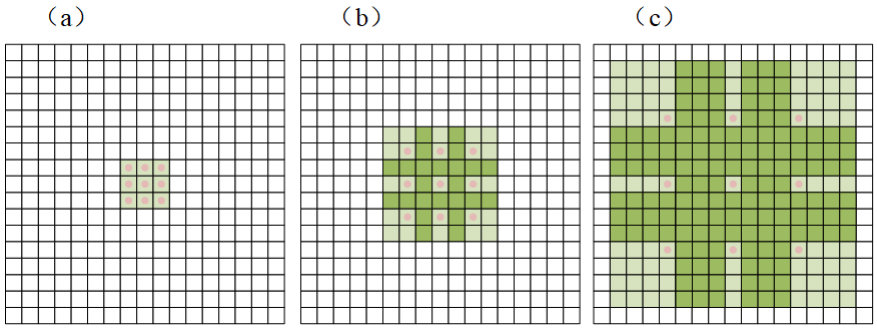


图 3-4 空洞卷积

Fig 3-4 Dilated convolution

为了解决感受野过小的问题，本章引入了空洞卷积这一技术，用以扩大模型

的感受野。空洞卷积通过在卷积核内部设置空洞，有效地在不增加参数数量的前提下增大了感受野，从而提升了模型的特征提取能力。该方法避免了通过卷积和池化减小图像尺寸以增大感受野时，常规卷积可能产生的信息丢失问题。空洞卷积能够保持图像大小的同时，确保信息的完整性，这对于保持图像中的细节信息和空间结构至关重要。与正常的卷积相比，空洞卷积增加了一个重要的超参数，即空洞率。空洞率是指卷积核中元素之间的间隔数，它决定了感受野的扩大程度。不同空洞率的卷积核产生了不同的感受野大小，如图 3-4 所示，三个不同的空洞卷积从左到右连续进行卷积，其中图 3-4 中（a）为标准的3×3卷积核，空洞率为 1（正常卷积的默认空洞率）；图 3-4 中（b）卷积核空洞率设置为 2，感受野扩大到了7×7；图 3-4 中（c）卷积核的空洞率进一步增加为 4，感受野进一步扩大到了15×15。这种通过调整空洞率来灵活控制感受野大小的方法，使得空洞卷积在不增加卷积核参数量的情况下，有效地扩大了模型的感受野，从而提高了模型的特征提取能力和性能。

最终，采用“空洞卷积+MobileNet”的组合作为新型的网络主干，以替代原有的 VGG-19。在精细化处理的阶段，对轻量化 Openpose 进行了两方面的优化。一方面是将原先的两个预测分支整合成一个统一的预测分支。具体来说，如图 3-5 左侧所示，原始 Openpose 模型在每个阶段都设置了两个结构相似的预测分支，仅在输出时产生不同数量的结果。为了降低网络的运算量，提出共享权值的策略，将两个分支合并为一个分支。如图 3-5 右侧所示，合并后的单一分支在输出阶段才通过1×1卷积操作分为两个子分支，分别输出关键点的置信度分布和部分亲和字段分布。这一改进在保持模型性能的同时，有效地降低了模型的计算复杂度。

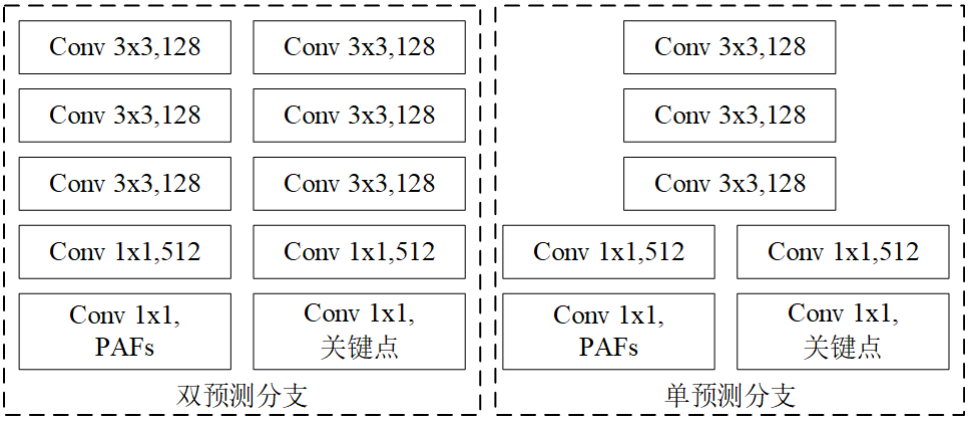


图 3-5 原始预测分支和合并预测分支

Fig 3-5 Original and combined forecast branches

另一方面是使用带有空洞卷积的模块替代原有的大卷积核，虽然大卷积核具有较大的感受野，但计算量也相应增加。通过采用级联的小卷积核并结合空洞卷积，可以在保持相同感受野的同时显著降低计算量。如图 3-6 所示，在最后一个 3×3 的卷积中使用空洞率为 2 的空洞卷积，使这个级联模块的感受野与 7×7 的卷积核相同，且计算量小了很多。同时在每个模块中都使用了残差连接结构，以进一步提升网络的性能。

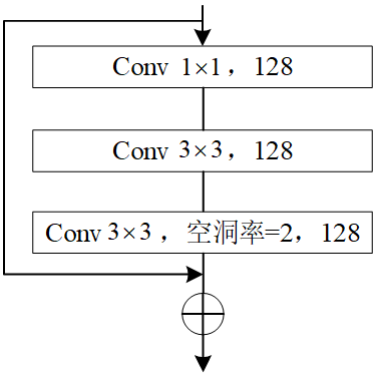


图 3-6 代替 7×7 的卷积

Fig 3-6 Instead of a 7×7 convolution

3.2.3 基于 GCN 的驾驶员行为识别方法

Sijie Yan 等人[11]从图卷积神经网络角度出发，结合了时间卷积与空间图卷积，构建了时空图卷积网络（Spatial Temporal Graph Convolutional Network，ST-GCN）。在空间维度上，为了更有效地提取驾驶员姿态中的关键信息，采用了 GCN 来捕捉相邻关节之间的局部特征，充分利用姿态数据中的空间结构信息，通过构建关节点之间的连接关系，实现局部特征的有效提取。在时间维度上，为了捕捉关节随时间变化的局部特征，运用了 TCN，能够处理序列数据，通过卷积操作捕捉关节点在不同时间点的状态变化，理解驾驶员姿态随时间的动态演变过程。

具体而言，空间图卷积主要聚焦于视频单帧中的节点所构成的空间图，并对其进行图卷积运算。这种操作与针对欧式数据的二维卷积操作在某种程度上相似，其核心思想在于通过定义合适的卷积核大小，对空间位置上的节点进行局部特征聚合。假设卷积核大小为 $K \times K$，那么在空间位置 x 处的卷积操作可表达为对邻域内节点特征的加权求和，这一过程充分考虑了节点间的空间关系，有助于提取出更加丰富的局部特征，公式可表达为：

$$f_{out}(x)=\sum_{h=1}^{K}\sum_{w=1}^{K}f_{in}(p(x,h,w))\cdot w(h,w) \tag{3-13}$$

其中 p 为采样函数，w 为权重函数，f_{in} 表示输入特征图，f_{out} 表示输出特征图，具体表示如下：

$$p(x,h,w)=x+p^{'}(h,w) \tag{3-14}$$

上式中，$p^{'}$ 表示以点 x 为中心与卷积核大小相匹配的邻域内的像素点集合，尺寸为 $K\times K$。权重函数与输入位置无直接相关性，表明卷积核在图像上是共享使用的。由于采用图卷积方法处理由二维矩阵网格组成的图像时，每个节点的邻点数量并不恒定。故以空间位置 x 为中心，定义该点在视频帧空间上的邻点集合，该集合包含了与位置 x 相邻的所有节点，这些节点的数量因位置而异。该点在该视频帧空间上的邻点集合可定义为：

$$B(v_{ti})=\{v_{tj}\mid d(v_{tj},v_{ti})\le D\} \tag{3-15}$$

其中 $d(v_{tj},v_{ti})$ 表示该视频帧内节点由 v_{tj} 到 v_{ti} 的最短路径，为简化计算，将采样函数 p 简化为 $p(v_{ti},v_{tj})=v_{tj}$。在本章的研究中，选取距离 $D=1$ 的邻域集，即只考虑与 v_{ti} 直接相邻的节点，有助于更精准地捕捉节点间的空间关系，进而提升时空图卷积网络对骨架序列数据的处理能力。

图片的构成基础为二维的矩阵网格，每个像素点都拥有其独特的位置，并与周围的像素点形成了一个明确的邻域关系，在空间上呈现出一种固定的排列顺序，在 CNN 中，依据这种空间顺序的索引来设定权重函数。而在图卷积的情境中，权重函数的定义方式发生了显著变化，不再仅仅依赖于像素点在空间上的排列顺序来设定权重函数，而是更多地考虑了像素点之间的拓扑关系以及在整个视频帧结构中的位置。具体而言，不再简单地为每个邻域关键点分配单一的标签，而是将中心点 v_{ti} 的邻域集合 $B(v_{ti})$ 划分为 K 个带有数字标签的子集。这种划分方式使得权重函数能够根据子集的标签对张量进行空间索引。其公式表达为：

$$w(v_{ti},v_{tj})=w^{'}(l_{ti}(v_{tj})) \tag{3-16}$$

在视频帧的驾驶员图卷积模型中，引入了卷积神经网络模型的思想，并基于改进的采样函数和权重函数进行了优化，表示为：

$$f_{out}(v_{ti})=\sum_{v_{ti}\in B(v_{ti})}\frac{1}{Z_{ti}(v_{tj})}f_{in}(v_{tj})\cdot w(l_{ti}(v_{tj})) \tag{3-17}$$

上式中，$Z_{ti}(v_{tj})=|\{v_{tk}\,|\,l_{ti}(v_{tk})=l_{ti}(v_{tj})\}|$ 代表对应子集的基数，$l_{ti}(v_{tj})=(v_{ti},v_{tj})$ 代表不同的子集。

骨架图的时空联系主要体现在相同类型的关节点在连续帧之间的时间连接上，通过对节点 v_{ti} 空间邻域进行时空拓展来构建这种联系，其表达式为：

$$B(v_{ti})=\{v_{qj}\,|\,d(v_{tj},v_{ti})\le 3,|q-t|\le|\Gamma/2|\} \tag{3-18}$$

其中 Γ 为相关联的帧数范围，q 为相邻的某一帧，由于节点时间演化具有顺序性，因此节点时空邻域中的子集标签映射遵循以下规则：

$$l_{ST}(v_{qj})=l_{ti}(v_{tj})+(q-t+|\Gamma/2|)\times K \tag{3-19}$$

本章采用 GCN 单层最终形式：

$$f_{out}=\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}f_{in}W \tag{3-20}$$

在上式中，邻接矩阵被表示为 A，详尽地反映了图中各个节点之间的连接情况；此外，还引入了自连接矩阵 $\tilde{A}=A+I$，得以捕捉并整合节点自身的信息；同时，$\tilde{D}$ 为度矩阵，其内部元素对应着每个节点所连接的边的数量，提供了关于图中节点连接密度的直观信息；最后，W 代表了待训练的参数集合，这些参数与上述矩阵共同构成了图卷积神经网络的核心组件，从而确保了模型能够精准地处理图结构数据。这样不仅能够充分利用图中的结构信息，还能通过训练参数来优化模型的性能，以实现更准确的图数据分析和处理。

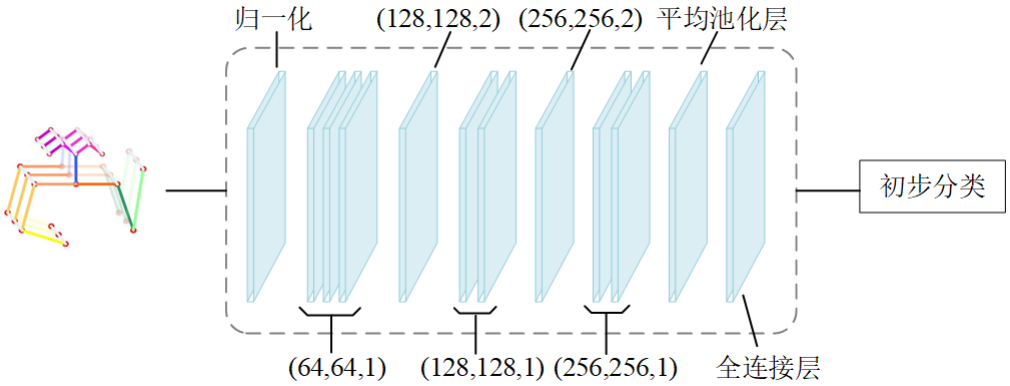


图 3-7 基于图卷积的驾驶员行为识别网络结构

Fig 3-7 Graph convolution based network architecture for driver behavior recognition

图 3-7 展示了基于图卷积的驾驶员行为识别网络结构，称为 En-GCN。利用

设计的轻量化 Openpose 模型，从每个视频帧中提取了驾驶员上半身 12 个关键关节点，每个关节点都包含坐标和置信度两个属性，以三元组(x, y, S)的形式进行表示，从而得到包含 12 个三元组的数组来表示提取出的驾驶员骨架图。由于视频帧之间的关节点坐标可能存在较大的变化，为了确保数据的稳定性和一致性，对同一关节在不同视频帧中的位置进行了归一化。输入序列表示为一个三维张量(C,V,T)，其中三个维度分别代表关节点特征数C（值为 3）、视频序列长度T（取 300 帧，长度不足的视频使用循环填充的方式补齐）以及估计关节点数量V（值为 12）。在空间卷积方面，采用公式（3-20）将三维张量与驾驶员骨架图的邻接矩阵相乘，从而有效地提取了骨架数据的空间特征。而在时间卷积方面，选用大小为$1\times K_t$的卷积核进行卷积运算，以捕捉驾驶员行为的动态变化。En-GCN 网络结构共包含 9 层时空图卷积，每层均融合了 1 个 GCN 和 3 个 TCN。随着网络层数的递增，输出通道数也随之增加，前 3 层输出通道为 64，第 4 至 6 层增大至 128 通道，第 7 至 9 层增大到 256 通道。为了进一步增强网络的性能，在第 4 层和第 7 层的 TCN 中引入了步长为 2 的池化操作，并在这两层加入了残差结构。最终，En-GCN 将最后一层输出的 256 个通道的特征图通过平均池化进行聚合，并借助全连接层对聚合后的特征进行分类。分类结果以每类动作的置信度形式呈现，根据这些置信度得出最终的驾驶员行为识别结果。

3.3 实验设计与结果分析

3.3.1 实验设计

在本章中，采用第二章所述的 AHPU-SET 数据集进行模型的训练和测试工作。实验所用的硬件环境为配备有 Intel Core-I7 CPU 和 NVIDIA RTX2080TI GPU 的 Dell 工作站。训练了所提出的模型以及对比模型，评估了它们在 Dell 工作站和 Jetson TX2 嵌入式设备上的识别准确性和效率。嵌入式设备上的实现如图：

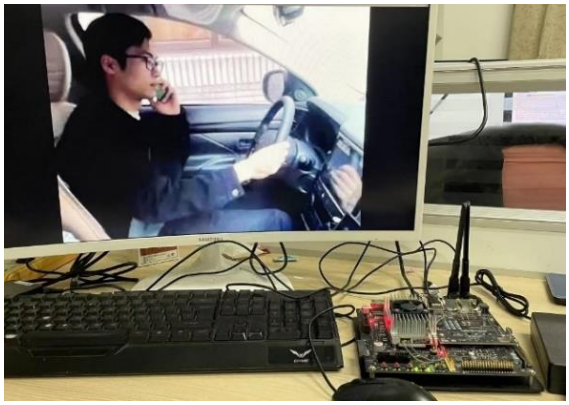


图 3-8 Jetson TX2 上的嵌入式实现

Fig 3-8 Embedded implementation on Jetson TX2

本文的实验配置与环境如下表：

表 3-2 实验配置与环境

Table 3-2 Experimental Configuration and Environment

	名称	配置
硬件	CPU	I7-10700K
	显卡	RTX 2080TI
	内存	32GB
软件	操作系统	Ubuntu 18.04
	配套环境	CUDA 11.0、cuDNN 8.0
	编译语言	Python
	深度学习框架	Pytorch

关于本章实验中的参数设置，具体细节如下所述。首先，将视频的分辨率调整至统一标准，即调整为416×416。随后，在模型训练的过程中，选用了随机梯度下降（Stochastic Gradient Descent，SGD）算法来优化网络模型。在训练过程中，设定了批量大小为 16，初始学习率定为 0.01，并将动量参数设定为 0.9。整体训练过程将历经 60 个 epoch，为了更好地调控模型的学习进度，当训练到第 20、30、40、50 个 epoch 时，对学习率进行了迭代下降，每次下降 0.1。这样的参数设置旨在确保模型训练的高效性和准确性。

为全面评估本章提出的算法在性能与效率上的表现，利用本章设计的驾驶员姿态时空数据集，系统地训练并测试了 ST-GCN 和 En-GCN 两种模型，实验包括准确率验证和运行速度测试。（1）在准确率验证环节，致力于检验两种算法在

识别驾驶员姿态时的准确性。首先，将 ST-GCN 和 En-GCN 模型在本章构建的专有数据集上进行训练，随后对训练好的模型进行测试。为更全面地反映模型性能，采用交叉验证的策略，按照 7:3 的比例将数据集随机分割为三种不同的训练/测试样本组合，即 Split1、Split2 和 Split3。每种样本组合均独立进行模型的训练与测试，确保结果的公正性和稳定性。最终，取三次识别率的平均值作为算法性能的评估结果，并以混淆矩阵的形式呈现每种算法的检测结果，以便直观地对比不同算法在识别驾驶员姿态时的优劣。（2）在运行速度测试环节，主要关注 ST-GCN 和 En-GCN 两种算法的执行效率。为了准确测量算法的运行速度，选取一个包含 300 帧动作循环的视频序列作为测试样本，记录该样本总的处理时间，通过将总处理时间除以视频帧数，计算出了每个视频帧的平均处理时间。这一指标能够直观地反映出算法在处理每一帧图片时所需的时间。进一步通过计算每秒所能处理的图片数量，即帧率（FPS），更直观地评估算法的运行速度。帧率越高，说明算法在单位时间内能够处理的图片数量越多，从而具备更好的实时处理能力。

3.3.2 实验结果与分析

表 3-3 详尽地展示了两种模型在三种不同数据分割模式下的识别率结果。经过对比分析，可以清晰地看到，相较于 ST-GCN 模型，本章提出的 En-GCN 方法在平均识别率上有了显著的提升，具体数值达到了 5.32%。为了进一步深入探讨模型在各类驾驶行为上的识别性能，采用了混淆矩阵的方式，将数据集上的检测结果进行了可视化呈现。这一矩阵直观地展示了不同种类驾驶行为的识别情况，其中矩阵的每一行代表实际的行为类别，而每一列则代表模型预测的行为类别。在混淆矩阵中，主对角线上的数值尤为重要，它们直接反映了对应行为类别的识别准确率。同时，矩阵的其他位置上的数值则代表了误检率，即模型将某一行为错误地识别为另一行为的概率。此外，本研究中涉及的标签 C0 至 C5 分别对应着六种常见的驾驶行为，分别为正常驾驶、喝水、化妆、脱离方向盘、吸烟以及打电话。

表 3-3 驾驶员行为识别在自制数据集上的准确率

Table 3-3 Accuracy of driver behavior recognition on a handmade dataset

网络模型	准确率			
	Split1	Split2	Split3	平均准确率
ST-GCN	84.20%	84.49%	85.56%	84.75%
En-GCN	92.12%	89.78%	88.31%	90.07%

图 3-9 和图 3-10 分别呈现了 ST-GCN 和 En-GCN 模型在自制数据集上的混淆矩阵。通过对比这两张矩阵，可以观察到两者在多数驾驶员行为识别任务中都展现出了较好的性能，但在处理部分相似度较高的动作时，均存在一定的识别误差。

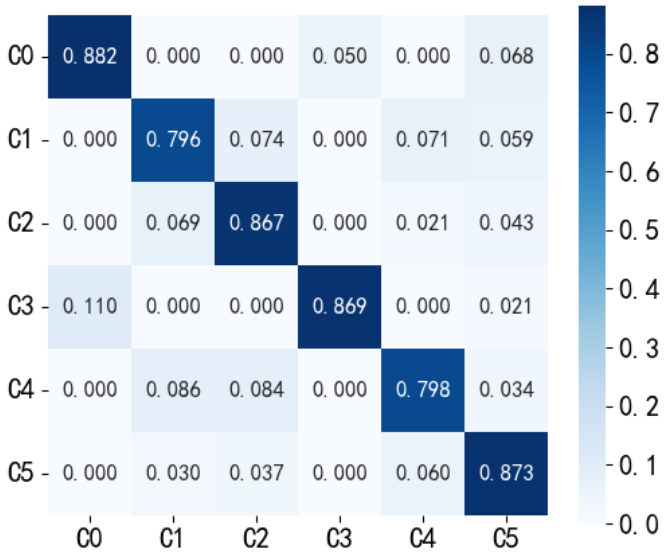


图 3-9 ST-GCN 实验结果

Fig 3-9 ST-GCN experimental results

具体而言，从图 3-9 中可以看到，ST-GCN 模型在正常驾驶行为的识别上仍存在部分细微动作被误判为其他行为的情况，正确识别率为 88.2%，错误地识别为脱离方向盘或打电话的行为仍占 5%和 6.8%。这主要是由于某些动作之间的相似性较高，而骨架信息主要聚焦于人体的行为特征，对于手部与物体的交互识别能力相对有限。因此，在涉及手部与物体交互的动作，如喝水和抽烟时，ST-GCN 模型的识别结果往往存在较大的混淆。例如，喝水的准确率仅为 79.6%，可能会被误识别为化妆、抽烟或打电话等行为；同样地，抽烟的准确率也仅为 79.8%，可能会被误识别为喝水、化妆或打电话等行为。

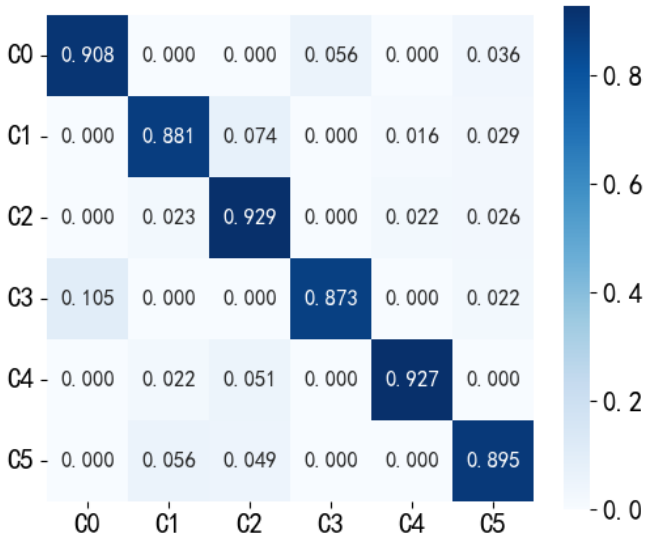


图 3-10 En-GCN 实验结果

Fig 3-10 En-GCN experimental results

通过深入分析图 3-10 所展示的数据，本章设计的 En-GCN 模型在驾驶员行为识别任务中展现出了不俗的性能，主要归功于模型对驾驶室环境及驾驶员行为特性的深入理解和精准建模。En-GCN 模型特别注重手部和头部相对于身体重心的运动，通过精确捕捉这些关键部位的运动轨迹，模型能够有效地区分驾驶员的细微动作差异。此外，本章选取置信度最高的图像作为驾驶员的姿态估计图，这一特点为 En-GCN 模型提供了丰富的信息，有助于提升行为识别的准确性。具体来说，与现有模型相比，En-GCN 在多个驾驶行为的识别率上均有所提升。正常驾驶的识别率提高了 2.6%，喝水行为的识别率提升了 8.5%，化妆行为的识别率增加了 6.2%，脱离方向盘的识别率提高了 0.4%，吸烟行为的识别率更是显著提高了 12.9%，打电话的识别率也提升了 2.2%。这些实验数据充分证明了仅利用驾驶员的姿态估计图进行行为识别的有效性，这对于提升网络的整体识别率具有显著意义。

表 3-4 算法检测速度对比

Table 3-4 Comparison of algorithm detection speed

网络模型	检测速度
Openpose	18 FPS
Lightweight Openpose	24 FPS
ST-GCN	19 FPS
En-GCN	26 FPS

在评估两种算法的运行速度时，驾驶员姿态的估计速度对本章所提出算法的检测速度具有显著影响。具体而言，在高性能计算机上 ST-GCN 和 En-GCN 运行速度达到约 22FPS 和 23FPS，这样的速度表现基本能够满足实时性检测的需求。为了更全面地验证算法的实际性能，本章将训练好的 Openpose、Lightweight Openpose、ST-GCN 以及 En-GCN 部署到 Jetson TX2 上进行实时性检测，运行速度对比结果如表 3-4，清晰地了解到不同算法在实际应用中的运行速度差异。原始的 Openpose 在 Jetson TX2 平台上虽然仍展现出良好的性能，达到了 18FPS，但其速度相较于其他方法仍稍逊一筹。相比之下，经过优化的 Lightweight Openpose 能够以 24FPS 的速度运行，充分满足了实时检测的需求。本章所提出的驾驶员行为识别算法 En-GCN，其速度更是达到了 26FPS，这一速度足以确保在移动嵌入式系统中实现驾驶行为的实时检测。此外，值得一提的是，本章设计的基于图卷积的驾驶员行为识别算法在进行了网络轻量化处理后，显著降低了算法的计算量和复杂度。同时，通过引入空洞卷积以扩大感受野，该算法在 AHPU-SET 数据集上取得了高达 90%的准确率。这一成果不仅凸显了算法的高效性，也进一步验证了其在驾驶员行为识别领域的实际应用价值。

3.4 本章小结

本章详细阐述了一种创新的驾驶员行为识别方法，该方法的核心在于基于图卷积网络对驾驶员姿态的精准分析。利用先进的 Openpose 技术提取了人体关键点，根据驾驶员动作的特殊性，对驾驶员姿态估计图进行了针对性的优化，并将原始的驾驶行为视频数据集转化为驾驶员姿态时空图数据集。通过对驾驶员姿态时空图进行图卷积运算，有效地提取出驾驶员骨架图在空间上的结构信息以及其随时间的动态演变过程，进而通过全连接层对驾驶员的行为进行精准判别。为了在嵌入式设备上实时识别，引入 MobileNet 与空洞卷积进行轻量化处理，在保证精度的同时，减少了算法的计算量，提升了运算速度。

第 4 章 基于压缩视频的多分支驾驶员行为识别方法研究

4.1 引言

本文在第三章设计了基于图卷积神经网络的驾驶员行为识别方法。根据第三章实验结果可知，由于骨架信息只关注于人体运动本身，可能受到姿势变化和遮挡的影响，骨架信息不能对运动轨迹进行正确判断，通过骨架信息进行识别容易造成误差。对于物品交互的细节，压缩视频能够提供更全面、直观、具有时空关系的信息。压缩视频中包含了更丰富的信息，能够帮助模型更好地捕捉驾驶行为的时序关系和空间关系，可以更好地理解不同驾驶行为之间的差异。

于是本章参考 Coviar 模型[55]，旨在从压缩视频中学习驾驶员的行为表示，效率上得到了提升，但该模型采用多个 ResNet 网络，参数量比较大，在计算速度上并没有很大改进。本章的目标是在时空依赖性和互补知识的指导下，为嵌入式系统实现高精度和实时的驾驶员动作识别，故提出了一种基于压缩视频的多分支轻量化卷积神经网络的驾驶员行为识别模型。首先，该模型对压缩视频数据流进行处理，利用 P 帧（Predicted Frame）中的运动矢量和残余误差代替光流图，减少了预计算时间；其次，设计了多分支时空卷积网络模型，通过轻量化卷积网络结构分别从 I 帧（Intra-coded Frame）和 P 帧提取外观信息与动作信息，旨在降低模型的参数量及计算复杂度；进一步地，设计了跨层连接模块与时空特征池化模块用于外观信息与动作信息的融合，实现人体行为时空表征；最后，为了弥补轻量化模型引起的性能退化，本章采用知识蒸馏学习策略，即引入老师模型引导设计的轻量化模型，进一步改善模型的行为表征能力。

4.2 基于压缩视频的多分支融合模型构建

本章所提出模型的流程图如下图 4-1 所示。首先对实时采集的压缩视频进行处理，采用两种策略对运动信息进行编码，即计算光流或加载压缩表示；然后，以 RGB 图像序列和堆叠光流为输入，对双流 I3D 网络进行训练和测试；最后对 RGB 帧、运动矢量和残余误差进行处理，通过师生学习策略实现高精度、实时

的驾驶员动作识别。下图 4-2 为模型整体的框架。

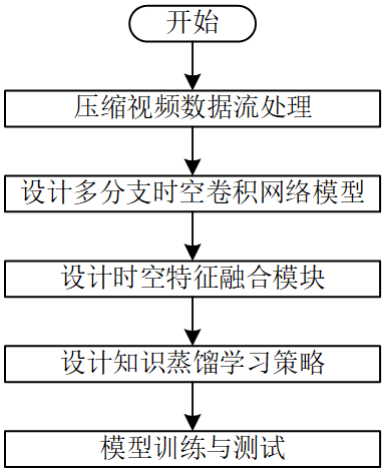


图 4-1 多分支融合驾驶员行为识别网络流程图

Fig 4-1 Flowchart of multi-branch fusion driver behavior recognition network

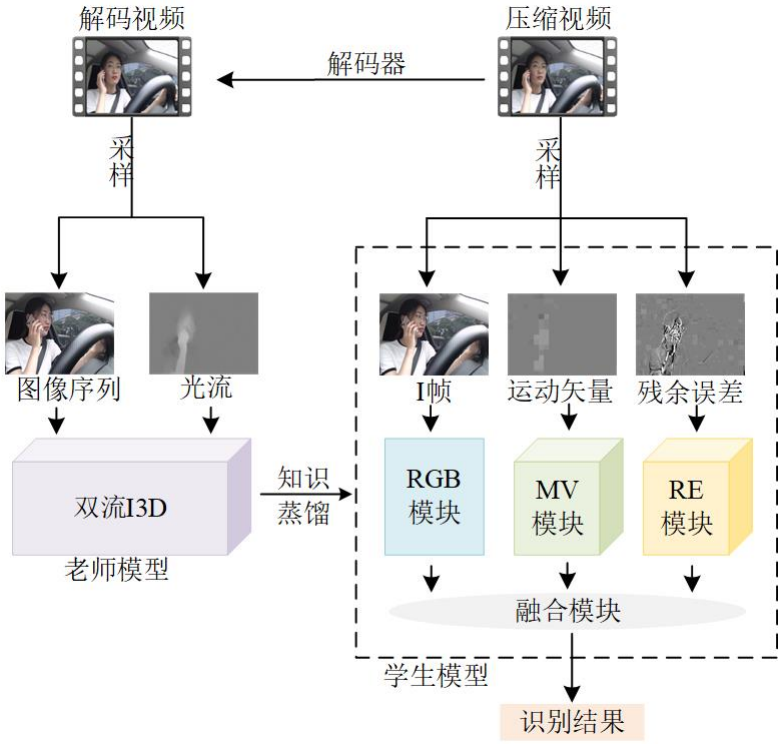


图 4-2 多分支融合驾驶员行为识别网络整体框架

Fig 4-2 An overall framework for multi-branch fusion driver behavior recognition network

4.2.1 MPEG-4 压缩视频处理

使用压缩视频进行高效动作识别的研究很少，视频压缩编解码器主要包括 HEVC、H.264、MPEG-4，实现了高效的视频传输和存储。视频压缩的原理是存储一个关键帧，通过运动表示来重建相邻的帧，压缩后的视频可以分成几组图片

（Group of Pictures，GOP），每组 GOP 包含一个内编码帧（Intra-coded Frame，I 帧），一些预测帧（Prediction-Frame，P 帧）和双向帧（Bi-direction prediction Frame，B 帧）。MPEG-4 通过画面组 GOP 进行视频序列编码，只保存每个 GOP 的初始 I 帧，而预测 P 帧是通过运动矢量与残余误差进行估计的。压缩编码后的视频占用的存储空间更小，并且运动矢量及残余误差能够直接从编码信息中读取，用于替代光流预计算，不需要预计算操作就大大减少了运算量。运动矢量记录了图像块从 I 帧到 P 帧的运动偏移，残余误差表示经过运动补偿后 P 帧与其参考 I 帧之间的像素变化。$(t+\tau)$时刻重构帧可以表示为

$$I_{(x+y)}^{t+\tau} = I_{(x,y)-M_{(x,y)}^{\tau}}^{t} + R_{(x,y)}^{\tau} \tag{4-1}$$

式中：$I_{(x+y)}^{t}$为t时刻的视频帧；$M_{(x,y)}^{\tau}$和$R_{(x,y)}^{\tau}$为τ时间段内的运动矢量和残余误差。如下图 4-3 所示，I 帧记录了视频初始帧的外观信息；运动矢量表示全局偏移，残余误差表示运动区域的细节变化。

图 4-3 压缩视频相邻帧的运动矢量与残余误差

（a）（b）视频片段中相邻的两帧；（c）（d）从 P 帧中提取的运动矢量和残差

Fig 4-3 Motion vectors and residual errors in adjacent frames of compressed videos

(a) (b) two adjacent frames in the video clip; (c) (d) motion vectors and residual errors extracted from P-frames

4.2.2 多分支轻量化网络模型

本章设计了多分支的时空卷积网络模型，其中包含三个网络分支，能够分别处理 I 帧，运动矢量及残余误差，分别命名为 RGB 帧分支，运动矢量分支和残差帧分支。该模型以 ShuffleNetV2 结构作为骨干网络,采用信道分割和信道打乱的操作来实现信道间的信息交互，整体结构如图 4-4 所示。

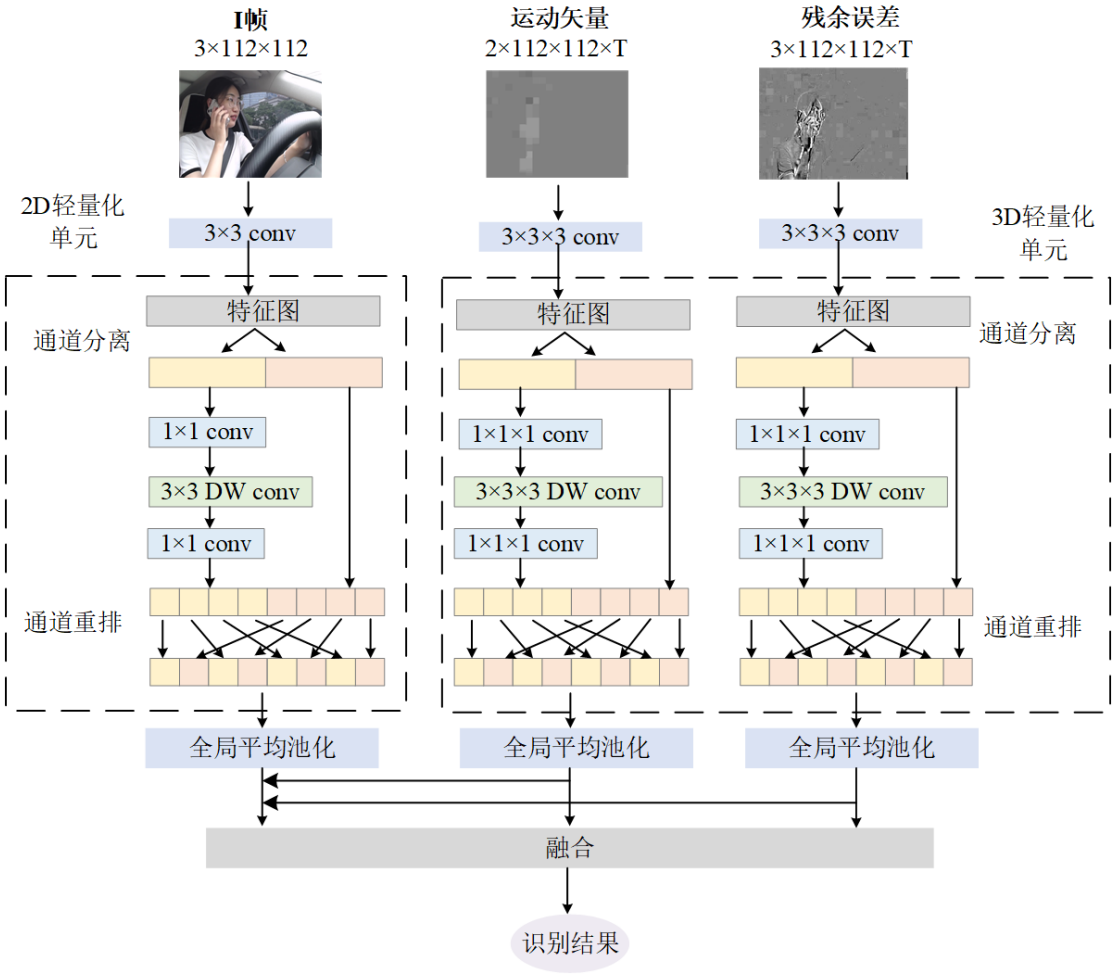


图 4-4 多分支轻量化时空卷积网络的整体结构

Fig 4-4 Overall structure of multi-branch lightweight spatio-temporal convolutional network

其中 RGB 帧分支采用 ShuffleNetV2 结构用于提取外观线索；运动矢量分支和残差帧分支将 ShuffleNetV2 中的 2D 卷积核扩展为 3D 卷积核，并对卷积通道数进行压缩，旨在进一步降低模型的参数量及计算复杂度。

RGB 帧分支采用了 ShuffleNetV2 网络结构，其输入为视频的初始帧 I 帧，尺寸为3×112×112。第一层为3×3基础卷积层，中间层采用了轻量化卷积单元如上图所示，该单元由1×1卷积层与3×3深度可分离卷积层组成，减少了可学习参

数量，此外该单元采用了通道分离与通道重排操作，促进了信息交互。RGB 帧分支的最后一层为全局均值池化层，旨在将特征图映射为 1024 维特征向量，特征学习过程 $RGBNet(\bullet|)$ 可以定义为：

$$F_{RGB}^{(l)} = RGBNet(F_{RGB}^{(l-1)} \mid \theta_{RGB}^{(l)}) = \theta_{RGB}^{(l)} * F_{RGB}^{(l)} \tag{4-2}$$

其中 $*$ 表示卷积操作；$\theta_{RGB}^{(l)}$ 为 RGB 帧分支第 l 层的卷积参数；$F_{RGB}^{(l-1)}$ 与 $F_{RGB}^{(l)}$ 分别表示 RGB 帧分支第 $(l-1)$ 层与第 l 层的输出特征图。

运动矢量分支采用了 3DShuffleNetV2 0.25x 网络结构，其输入为运动矢量序列，尺寸为 $2\times112\times112\times16$。第一层为 $3\times3\times3$ 基础卷积层，中间层采用了轻量化卷积单元，该单元由 $1\times1\times1$ 卷积层与 $3\times3\times3$ 深度可分离卷积层组成，此外该单元也采用了通道分离与通道重排操作。运动矢量分支的最后一层为全局均值池化层，旨在将特征图映射为 1024 维特征向量。相比较于 ShuffleNetV2 模型，3DShuffleNetV2 0.25x 将 2D 卷积核扩展为 3D 卷积核，并对卷积通道数进行了压缩，有效的降低了计算复杂度。特征学习过程 $MVNet(\bullet|)$ 可以定义为：

$$F_{MV}^{(l)} = MVNet(F_{MV}^{(l-1)} \mid \theta_{MV}^{(l)}) = \theta_{MV}^{(l)} * F_{MV}^{(l)} \tag{4-3}$$

其中 $*$ 表示卷积操作；$\theta_{MV}^{(l)}$ 为运动矢量分支第 l 层的卷积参数；$F_{MV}^{(l-1)}$ 与 $F_{MV}^{(l)}$ 分别表示运动矢量分支第 $(l-1)$ 层与第 l 层的输出特征图。

残差帧分支采用了 3DShuffleNetV2 0.25x 网络结构，其输入为残余误差序列，尺寸为 $3\times112\times112\times16$；特征学习过程 $RENet(\bullet|)$ 与 $MVNet(\bullet|)$ 相同，可以定义为：

$$F_{RE}^{(l)} = RENet(F_{RE}^{(l-1)} \mid \theta_{RE}^{(l)}) = \theta_{RE}^{(l)} * F_{RE}^{(l)} \tag{4-4}$$

其中 $*$ 表示卷积操作；$\theta_{RE}^{(l)}$ 为残差帧分支第 l 层的卷积参数；$F_{RE}^{(l-1)}$ 与 $F_{RE}^{(l)}$ 分别表示残差帧分支第 $(l-1)$ 层与第 l 层的输出特征图。

4.2.3 特征融合模块

接下来，本章设计了结合跨层连接单元与时空特征池化模块用于外观信息与动作信息的融合。跨层连接模块能够融合不同网络分支的中间层特征图，将运动矢量分支和残差帧分支学习的 3D 特征图经过时间卷积映射为 2D 特征图，并与 RGB 帧分支进行特征拼接；时空特征池化模块用于聚合不同网络分支的末层特征向量，通过 Conv1D 卷积学习特征向量的高维潜在信息，采用三线性池化操作

对不同分支提取的特征向量进行聚合，进而实现驾驶行为时空表征。

（1）跨层连接单元

设计跨层连接单元以融合不同网络分支的中间层特征图，如图 4-5 所示。RGB 帧分支、运动矢量分支和残差帧分支对应层输出特征图具有相同的空间尺寸，而时间尺寸不同，基于此，跨层连接单元首先采用$1\times1\times T$时间卷积将运动矢量分支和残差帧分支学习的 3D 特征图映射为 2D 特征图；接着将其与 RGB 帧分支输出的 2D 特征图进行拼接，并通过1×1卷积实现维度变换；具体地，RGB 帧分支、运动矢量分支和残差帧分支的第 l 层特征图分别为：$F_{RGB}^{(l)}\in\mathbb{R}^{C_{RGB}\times W\times H}$，$F_{MV}^{(l)}\in\mathbb{R}^{C_{MV}\times W\times H\times T}$ 和 $F_{RE}^{(l)}\in\mathbb{R}^{C_{RE}\times W\times H\times T}$，那么跨层连接单元操作 $CLCM(\cdot)$ 可以定义为：

$$\begin{aligned}\hat{F}_{RGB}^{(l)}&=CLCM(F_{RGB}^{(l)},F_{MV}^{(l)},F_{RE}^{(l)}\mid\theta^{clcm})\\&=\theta_{RGB}^{clcm}*[F_{RGB}^{(l)},\theta_{MV}^{clcm}*F_{MV}^{(l)},\theta_{RE}^{clcm}*F_{RE}^{(l)}]\end{aligned}\tag{4-5}$$

其中$*$表示卷积操作，[]为特征级联操作；θ_{MV}^{clcm} 与 θ_{RE}^{clcm} 为$1\times1\times T$时间卷积参数，θ_{RGB}^{clcm} 为1×1卷积参数；$\hat{F}_{RGB}^{(l)}\in\mathbb{R}^{\hat{C}_{RGB}\times W\times H}$ 表示输出特征图。

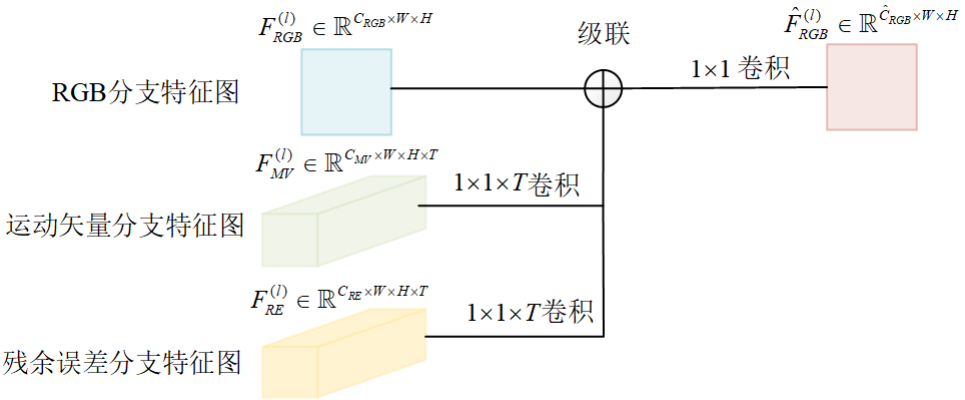


图 4-5 跨层连接单元示意图

Fig 4-5 Schematic diagram of a cross-layer connection unit

（2）时空特征池化模块

设计时空特征池化模块用于融合 RGB 帧分支、运动矢量分支和残差帧分支的顶层特征向量：$x_{RGB}\in\mathbb{R}^d$，$x_{MV}\in\mathbb{R}^d$，$x_{RE}\in\mathbb{R}^d$，如图 4-6 所示。融合特征向量可以定义为：

$$\hat{x}_{fusion}=STTPM(x_{RGB},x_{MV},x_{RE}\mid\theta^{sttpm})\tag{4-6}$$

其中 $STTPM(\cdot|)$ 表示时空特征池化模块，$\theta^{sttpm}=\{\theta_{RGB}^{sttpm},\theta_{MV}^{sttpm},\theta_{RE}^{sttpm}\}$ 表示该模

块的相关参数，$\hat{x}_{fusion}$ 为融合特征向量。

时空特征池化模块可分解为三个阶段：首先通过 Conv1D 卷积将特征向量映射到高维隐空间；其次采用三线性池化操作对不同分支提取的特征向量进行聚合；最后采用正则化操作对融合特征向量进行归一化。时空特征池化模块的三个阶段可以分别表示为：

$$\begin{aligned}\hat{X}_{RGB},\hat{X}_{MV},\hat{X}_{RE} &= Conv1d(x_{RGB},x_{MV},x_{RE}\mid\theta^{sttpm})\\ &= \theta^{sttpm}_{RGB}*x_{RGB},\theta^{sttpm}_{MV}*x_{MV},\theta^{sttpm}_{RE}*x_{RE}\end{aligned} \tag{4-7}$$

$$x_{fusion} = 1^{T}(\hat{X}_{RGB}\otimes\hat{X}_{MV}\otimes\hat{X}_{RE}) \tag{4-8}$$

$$\hat{x}_{fusion} = \frac{sign(x_{fusion}\sqrt{|x_{fusion}|})}{\|x_{fusion}\|_2} \tag{4-9}$$

其中 $\theta^{sttpm}_{RGB}\in\mathbb{R}^{n\times k}$、$\theta^{sttpm}_{MV}\in\mathbb{R}^{n\times k}$、$\theta^{sttpm}_{RE}\in\mathbb{R}^{n\times k}$ 分别表示 $Conv1d(\cdot|)$ 卷积层参数；n、k 为卷积核数与卷积核尺寸；$\hat{X}_{RGB}\in\mathbb{R}^{n\times d}$、$\hat{X}_{MV}\in\mathbb{R}^{n\times d}$、$\hat{X}_{RE}\in\mathbb{R}^{n\times d}$ 分别表示高维隐特征向量；$1^{T}\in\mathbb{R}^{n}$ 为全 1 向量用于求和池化，$\otimes$ 为矩阵对应元素乘积操作；$sign(\cdot|)$ 与 $\|\cdot\|_2$ 分别表示符号函数与矩阵二范数；$\hat{x}_{fusion}\in\mathbb{R}^{d}$ 表示归一化后的输出特征向量。

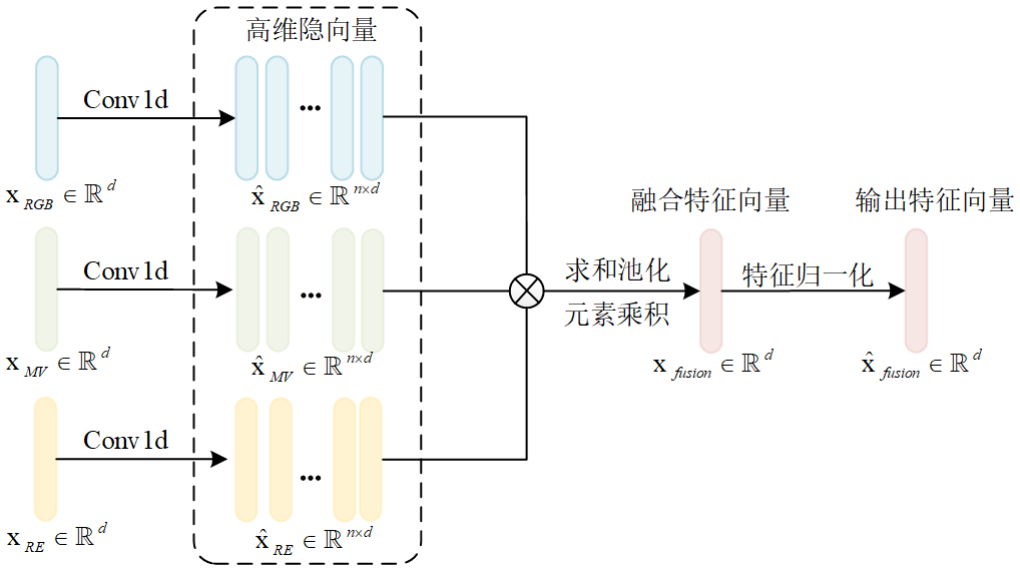


图 4-6 时空特征池化模块示意图

Fig 4-6 Schematic diagram of the spatio-temporal feature pooling module

4.2.4 知识蒸馏学习策略

在模型压缩这一领域中，2015 年 Hinton 等[56]提出了知识蒸馏的概念，作为一种典型的迁移学习技术，知识蒸馏以其独特的训练机制，在模型优化和性能提升方面展现出了显著的潜力。具体而言，知识蒸馏的核心在于构建了一个“老师—学生网络”的训练框架，扮演着关键角色。通过对大型且性能卓越的老师模型的输出结果进行深入分析，提炼出丰富的“知识”，这些知识是模型性能提升的关键。而学生模型，作为一个规模较小、复杂度较低的网络，则在老师模型的指导下进行训练。通过学习老师模型中的知识，学生模型能够在保持与老师模型相近性能的同时，实现参数数量的显著减少。

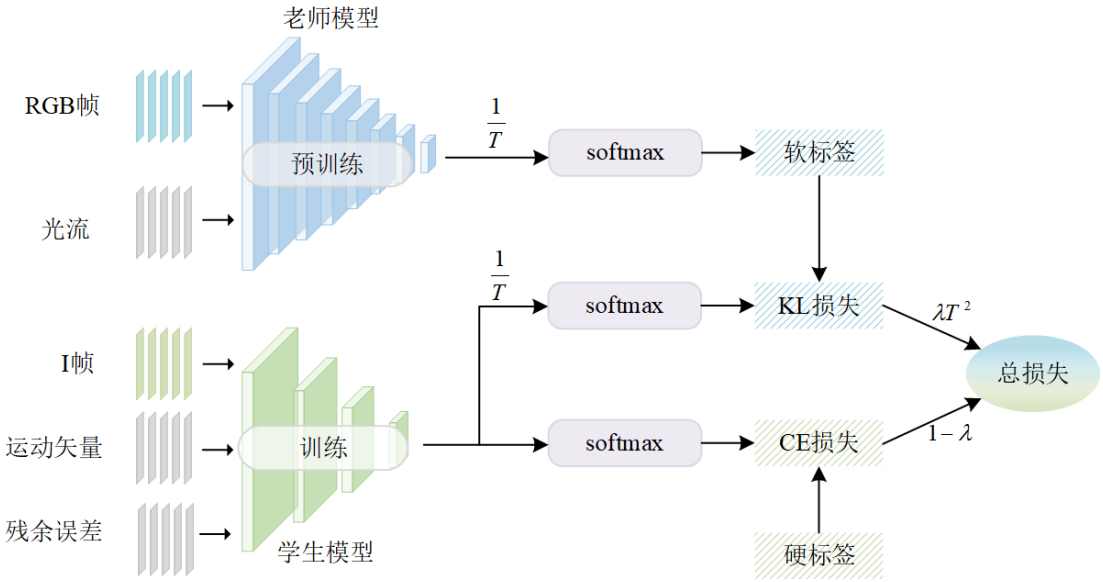


图 4-7 知识蒸馏学习策略框架

Fig 4-7 Framework of knowledge distillation learning strategy

本章采用知识蒸馏学习策略，引入 Two-stream I3D 作为老师模型，多分支轻量化时空网络模型作为学生模型。如图 4-7 所示，将老师模型输出的结果作为软标签，引导轻量化模型的训练，弥补了参数量减少引起的性能退化，进而改善模型的识别精度。

老师模型与学生模型的输出可以分别表示为 $h_t=\{(h_t)^1,(h_t)^2,...,(h_t)^k\}$ 与 $h_s=\{(h_s)^1,(h_s)^2,...,(h_s)^k\}$，其中 K 表示类别数。使用 Softmax 函数计算不同类别的概率分布，具体可定义为：

$$p(k|h)=softmax((h)^k)=\frac{e^{(h)^k}}{\sum_{k'}e^{(h)^{k'}}} \tag{4-10}$$

$$p^T(k|h)=softmax(\frac{(h)^k}{T})=\frac{e^{\frac{(h)^k}{T}}}{\sum_{k'}e^{\frac{(h)^{k'}}{T}}} \tag{4-11}$$

其中T为超参数用于调节知识蒸馏温度阈值；$p(k|h)$表示h属于第k个类别的概率；$p^T(k|h)$表示知识蒸馏获取的软标签。模型的损失函数包含两个部分，分别为交叉熵损失$\mathcal{L}_{\mathcal{CE}}$与 KL 散度损失$\mathcal{L}_{KL}$，具体可表示为：

$$\mathcal{L}_{\mathcal{CE}}=-\frac{1}{K}\sum_{k=1}^{K}(1(y^k=\hat{y}^k)\cdot\log(p(k|h_s))) \tag{4-12}$$

$$\mathcal{L}_{KL}=-\frac{1}{K}\sum_{k=1}^{K}(p^T(k|h_t)\cdot\log(\frac{p^T(k|h_s)}{p^T(k|h_t)})) \tag{4-13}$$

其中$1(\cdot)$表示指示函数；y^k与$\hat{y}^k$分别表示真实值与预测结果；$p(k|h_s)$表示学生模型的概率分布；$p^T(k|h_s)$与$p^T(k|h_t)$表示学生模型与老师模型生成的软标签。模型的最终损失是由不同学习任务的损失加权组合得到，可以表示为：

$$\mathcal{L}=\underset{\theta_{stu}}{argmin}(\sum_{i=1}^{N}((1-\lambda)\mathcal{L}_{\mathcal{CE}}+\lambda T^2\mathcal{L}_{\mathcal{KL}})+\|\theta_{stu}\|_2^2) \tag{4-14}$$

其中λ表示表示权重超参数；θ_{stu}为学生模型的相关参数，$\|\cdot\|_2^2$表示二范数距离。

4.3 实验设计与结果分析

4.3.1 实验设计

本章使用第三章介绍的实验环境。在网络训练阶段，通过三个步骤实现：首先，分别训练 RGB 帧分支、运动矢量分支和残差帧分支；其次，使用特征融合模块组合各个网络分支；最后，采用知识蒸馏策略进行表示学习。训练过程中，批量大小设置为 10；学习率初始化为 5e-4；训练过程持续 200 epoches。由于模型的目标是处理短片，为了实现视频级分类，捕捉长期依赖性，故而参考 TSN[29]使用共识聚合函数对K个短片段（$K=3$）提取的特征进行平均。在网络测试阶段，本章使用 ffmpeg[57]源代码捕获基于 MPEG-4 编解码器的实时视频流，在这种情

况下，I 帧、运动向量和残余误差都是现成的。接下来，将多模态数据输入模型，以进行最终的驾驶员动作识别。所提出超参数的参考值如下表所示。

表 4-1 超参数参考值

Table 4-1 Reference value of hyperparameter

参数	参考值
批量大小	10
最大轮次	500
初始学习率	5e-4
学习率衰减步长	100
学习率衰减比率	1e-1
权重衰减	1e-4
段数量	3
蒸馏权重	0.6
蒸馏温度	5

4.3.2 实验结果与分析

（1）多分支网络性能评估

本章提出的多分支深度学习框架，分别处理 I 帧、运动向量和残余误差。在这里，分别采用 RGB 分支(RGB)、运动矢量分支(MV，motion vector)和残余误差分支(RE，residual error)进行特征表示和驾驶员动作识别，并利用分数融合策略将各网络分支结合起来。相应模型的准确率和效率如下表 4-2 所示。RGB 分支利用 ShuffleNet-V2 学习外观线索，运动矢量分支和残余误差分支利用 3D-ShuffleNet-V2 0.25x 捕获运动信息。从表 4-2 中可以看出，RGB 分支的准确率远高于运动矢量分支和残余误差分支，因为 I 帧包含更丰富的语义信息，并且 RGB 分支具有更高的表征学习通道容量。由表可知，RGB 分支、运动矢量分支和残余误差分支的组合在 AHPU-SET 上的准确率达到了 93.6%，优于单分支网络。在计算效率方面，所提出的多分支网络可学习参数少（1.78M），计算成本低（0.239 GFLOPs），处理速度快（在 Jetson TX2 上为 32 clips/s）。本章同时研究了使用 MobileNet-V2 和 3D-MobileNet-V2 0.25x 来构建拟议的多分支网络。但结果是，

无论是识别准确率还是计算效率，它都弱于 ShuffleNet-V2 和 3D-ShuffleNet-V2 0.25x 的组合。

表 4-2 各网络分支对于驾驶员动作识别的准确率和效率

Table 4-2 Accuracy and efficiency of driver behavior recognition across network branches

网络模型	准确率	参数量	FLOPs	速度	
				NVIDAIA 2080TI	Jetson TX2
RGB	79.9%	1.36M	0.049G	358 clips/s	73 clips/s
MV	73.4%	0.25M	0.087G	547 clips/s	92 clips/s
RE	77.3%	0.24M	0.117G	446 clips/s	83 clips/s
RGB + MV	88.5%	1.66M	0.126G	235 clips/s	40 clips/s
RGB + RE	89.8%	1.67M	0.158G	224 clips/s	37 clips/s
MV + RE	82.5%	0.57M	0.198G	264 clips/s	47 clips/s
RGB + MV + RE(ShuffleNet-V2)	93.6%	1.78M	0.239G	152 clips/s	32 clips/s
RGB + MV + RE(MobileNet-V2)	92.1%	2.89M	0.246G	109 clips/s	21 clips/s

（2）特征融合模块性能评估

本章测试了跨层连接模块（CLCM）和时空三线性池化模块（STTPM）对各网络分支融合的有效性。CLCM 连接中间特征图，STTPM 用于最终特征融合。然后分别使用 CLCM 和 STTPM 以不同的融合策略来组合网络分支。定量结果如下表 4-3 所示。

表 4-3 不同融合策略下驾驶员行为识别的准确率

Table 4-3 Accuracy and efficiency of driver behavior recognition under different fusion strategies

融合策略	AHPU-SET
分数融合	91.0%
时间池化+分数融合	92.7%
CLCM+分数融合	92.9%
CLCM+求和融合	91.3%
CLCM+连接融合	92.6%
CLCM+STTPM	93.9%

首先，本章评估了 CLCM 在特征融合方面的有效性。CLCM 使用时间卷积运算将时空特征图转换为空间特征图，然后通过1×1卷积将三个网络分支串联起来。有两种可供选择的策略可供进一步探讨。第一，从运动矢量分支以及残余误差分支提取出的 4D 特征图 $F_M^{(l)} \in \mathbb{R}^{C_M \times W \times H \times T}$ 、$F_R^{(l)} \in \mathbb{R}^{C_R \times W \times H \times T}$，可以转换成 3D 特征图 $\mathbb{R}^{T \cdot C_M \times W \times H}$ 、$\mathbb{R}^{T \cdot C_R \times W \times H}$，也就是说，所有的 T 帧都可以转换为一帧的通道。第二，从运动矢量分支以及残余误差分支提取出的 4D 特征图 $F_M^{(l)} \in \mathbb{R}^{C_M \times W \times H \times T}$ 、$F_R^{(l)} \in \mathbb{R}^{C_R \times W \times H \times T}$，可以在 T 帧之间进行简单的平均。定量结果表明，时间卷积在中间层特征融合上优于其他两种，其中，CLCM 在 AHPU-SET 上的精度达到了 92.9%。

其次，本章比较了最终特征融合的不同策略，设计了 STTPM 模块来聚合外观和运动线索。三种比较融合策略如下。第一，分数融合：对每个网络分支输出的分类分数进行平均，以实现最终的驾驶员动作识别。第二，相加融合：将从各网络分支 x_I 、x_M 和 x_R 提取的特征向量相加，从形式上来说融合后的特征向量可以表达成 $\hat{x}_{fusion} = x_I + x_M + x_R$。第三，连接融合：将每个网络分支 x_I 、x_M 和 x_R 提取的特征向量连接，形式上表达成 $\hat{x}_{fusion} = [x_I, x_M, x_R]$。从表中可以发现，分数融合比连接融合和相加融合获得了更高的准确率。STTPM 使用一维卷积和三线性池化做最终特征融合，结果比另外三种特征级融合策略表现更好。而且 CLCM+STTPM 在 AHPU-SET 上的准确率达到了 93.9%。

（3）知识蒸馏策略性能评估

在训练阶段，知识蒸馏被用于进行网络的优化。知识蒸馏在模型压缩和迁移学习中扮演着重要的角色，利于提高模型的泛化能力、加速推理速度、减少资源消耗，并促进模型之间的知识传递和迁移。具体来说，高容量的老师模型可以提供补充知识，促进所提出的多分支轻量化网络实现更准确的驾驶员动作识别。本章使用不同的老师模型来实现知识提炼：双流 CNN、LRCN 和双流 I3D。从表 4-4 中可以看出，选择双流 CNN 或 LRCN 作为老师模型时，准确率没有明显提高。双流 CNN 和 LRCN 的能力弱于所提出的学生模型，只能起到网络正则化的作用，无法实现引导学习。双流 I3D 将 Inception 块的二维卷积滤波器膨胀为三维卷积滤波器，在时空表示方面具有很强的能力。选择双流 I3D 作为老师模型，在

AHPU-SET 数据集上的识别准确率高达 96.7%。

知识蒸馏的超参数有两个：蒸馏温度 T 和相关权重 λ。不同超参数下的识别性能如下图所示。当超参数 T 和 λ 设置为 5 和 0.6 时，达到了不错的性能。因此设置 $T=5$，$\lambda=0.6$ 作为知识蒸馏的参考值。

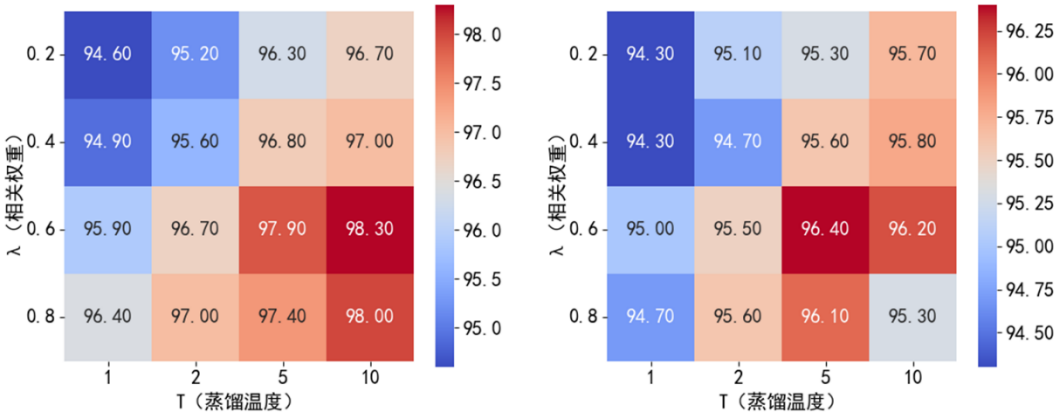


图 4-8 不同超参数的识别性能

Fig 4-8 Recognition performance for different hyperparameters

表 4-4 不同老师模型进行驾驶员行为识别的准确率

Table 4-4 Accuracy of different teacher models for driver behavior recognition

老师模型	**AHPU-SET**
无	94.7%
双流 CNN	95.9%
LRCN	95.2%
双流 I3D	96.7%

（4）与现有模型比较

在 AHPU-SET 数据集上通过与现有方法做对比，本章实验了所提出模型的准确度和识别效率，比较的方法可以分为两类。第一，基于空间深度学习方法：ResNet50，MobileNet-V2 1.0x，ShuffleNet-V2 1.0x 和 MobileViT；第二，根据时空深度学习方法：TSN，LRCN，R(2+1)D，双流 I3D，SlowFast，Vivit 和 Coviar。在 Dell 工作站和 Jetson TX2 嵌入式设备上重复上述基线模型进行实验，定量结果如下表 4-5，为不同驾驶员行为识别模型的识别精度和计算效率之间的比较。

表 4-5 不同模型进行驾驶员行为识别的准确率和计算效率

Table 4-5 Accuracy and computational efficiency of different models for driver behavior recognition

网络模型	准确率	参数量	FLOPs	速度	
				NVIDAIA 2080TI	Jetson TX2
ResNet50	89.0%	23.72M	4.124G	140 clips/s	8 clips/s
MobileNetV2 1.0x	85.0%	2.32M	0.341G	248 clips/s	46 clips/s
ShuffleNetV2 1.0x	85.6%	1.41M	0.189G	298 clips/s	58 clips/s
MobileVit	89.1%	5.84M	0.723G	176 clips/s	30 clips/s
TSN	94.2%	46.88M	3.649G	<1 clips/s	<1 clips/s
LRCN	86.1%	82.17M	1.063G	22 clips/s	<1 clips/s
R(2+1)D	93.5%	33.18M	166.772G	8 clips/s	<1 clips/s
双流 I3D	97.0%	25.82M	215.843G	<1 clips/s	<1 clips/s
SlowFast	95.3%	34.49M	36.153G	15 clips/s	<1 clips/s
Vivit	96.0%	88.99M	451.637G	3 clips/s	<1 clips/s
Coviar	93.0%	86.18M	16.274G	82 clips/s	3 clips/s
本章方法	96.5%	2.32M	0.264G	118 clips/s	25 clips/s

残差网络的优点是对驾驶员行为识别有较高的准确率，但需要较高的计算资源，不适合 JetsonTX2 嵌入式平台。MobileNet-V2 和 ShuffleNet-V2 有效地降低了计算成本并且加速了网络的推理速度，但识别性能还有待提高。最新的 MobileViT[58]是基于 transformer 的轻量级模型，但精度远达不到要求。基于时空深度学习的方法同样存在一些局限性。TSN 需要进行光流的预计算，在 Jetson TX2 上的运算速度小于 1 clips/s。LRCN[59]是以融合帧间外观信息为目标，却忽略了帧间运动信息，在 AHPU-SET 数据集上的准确率仅为 86.1%。R(2+1)D，双流 I3D 和 SlowFast 是 3D CNN 用于时空学习的三种变体。R(2+1)D 在 AHPU-SET 数据集上的准确率为 93.5%。双流 I3D 在 AHPU-SET 数据集上取得了 97.0%准确率，但是计算复杂度极高，不适用于嵌入式系统。最新 Vivit[60]是一种纯 transformer 的时空特征提取架构，在取得相同精度的同时也会占用过多的计算资源。Coviar 是一种独特的时空深度学习模型，专门用于处理压缩视频，由三个分支组成，分别使用残差网络从 RGB 帧、运动矢量和残差中提取特征，定量结果

表明，Coviar 在 NVIDAIA 2080TI 上平衡了识别精度和计算效率，但在嵌入式设备上并没有达到实时识别的效果。本章研究中，参考 Coviar 提出了用于基于压缩视频的多分支驾驶员行为识别模型，相比之下，本章所提出的模型使用轻量化的 2D/3D 卷积进行表示学习，输入帧的空间大小是 Coviar 的一半，故计算效率进一步提高。同时，本章提出 CLCM 和 STTPM 融合时空特征图以及使用知识蒸馏优化学生模型，因此识别性能能接近老师模型（双流 I3D）。本章模型在 AHPU-SET 上的准确率达到 96.5%，参数量为 2.32M，FLOPs 为 0.264G，在 Jetson TX2 上识别速度为 25 clips/s。

4.4 本章小结

本章的目标是在时空依赖性和互补知识的指导下，为嵌入式系统实现高精度和实时的驾驶员动作识别。为此，本章构建了一个新颖的深度学习模型来学习压缩视频中的驾驶员动作表示。首先，本章提出了多分支轻量化网络模型，其中一个分支使用轻量级二维卷积网络捕捉外观线索，另外两个分支使用轻量级三维卷积网络分别从运动矢量和残差误差中学习时间信息。其次，本章设计了跨层连接模块和时空三线性池化模块，用来充分利用外观线索和运动线索之间潜在的相关性。最后，采用师生学习方案，从高容量模型中提炼和获取互补知识。实验结果表明，所提出的模型在驾驶员动作识别方面同时满足了高准确度和实时性的要求。本文所提出的模型为基于车载设备的实时驾驶员动作识别提供了可行的解决方案，可促进车联网的发展，提高交通安全水平。

第 5 章 总结与展望

5.1 工作总结

驱车出行已成为现代生活中不可或缺的一部分，驾驶员的行为和决策是影响交通安全的主要因素。为了有效遏制交通事故的频发，准确识别驾驶员的危险行为并及时发出预警显得至关重要。鉴于此，本文以包含驾驶员行为的监控视频为研究对象，针对该领域的关键难题进行了深入探究，提出了可部署于嵌入式平台的轻量化驾驶员行为识别方法，实现实时检测。具体的研究成果如下：

（1）详细探讨了驾驶行为识别领域的相关研究成果，分析并总结了驾驶行为识别的研究现状和相关理论基础，对现有驾驶行为识别技术的研究难点做出了细致归纳。

（2）本文提出了一种基于图卷积的驾驶员行为识别方法，并构建了驾驶员行为数据集 AHPU-SET 以验证其有效性。利用先进的 Openpose 技术，将数据集转化为驾驶员姿态时空图形式，以便更好地捕捉驾驶员的动态行为特征。鉴于驾驶室内环境限制，通常只能检测到驾驶员的上半身姿态，特别是手部和头部的动作，设计了一种针对性的驾驶员姿态时空图表示方法，以突出关键部位的动作信息。基于这一驾驶姿态时空图，进一步采用图卷积神经网络，构建了一种高效的驾驶员行为识别方法。为了提升实时性能，运用 MobileNet 与空洞卷积技术对 Openpose 进行了轻量化处理，显著降低了模型的计算量。实验结果表明，该方法在 Jetson TX2 上能够实现实时识别，准确率方面也有所提升。

（3）本文提出了一种基于压缩视频的多分支融合驾驶员行为识别方法。为了从压缩视频中高效提取外观和运动信息，设计了多分支轻量化卷积神经网络进行时空建模。具体而言，一个分支采用轻量级的二维卷积网络来捕获外观特征；另两个分支则运用轻量级三维卷积网络，分别从运动向量和残差帧中学习并提取时间维度的信息。为进一步提高识别精度，设计了跨层连接模块和时空融合模块，用于有效融合多分支的输出结果。这些模块能够充分利用不同分支提取的特征信息，从而提升模型的判别能力。由于轻量化处理可能导致模型精度有所下降，采用知识蒸馏策略来弥补这一缺陷。通过从高容量的时空深度学习模型中提炼出互

补的知识，将其迁移到所提出的多分支轻量化模型中，从而进一步改善识别精度。该方法在 AHPU-SET 数据集上进行了消融实验，证明了其优越性。

5.2 工作展望

在驾驶员行为识别方面，本文的研究取得了一定的进展，但仍存在一些需要完善和改进的地方，需要进行下一步研究：

（1）尽管本文已构建了一个驾驶员行为识别的视频数据集，但现有数据集的规模与内容尚局限于少数几种驾驶行为的识别，远未达到满足实时监控的广泛需求，需要对驾驶行为种类的覆盖范围进行进一步的扩充。同时，考虑到驾驶行为可能发生在不同的时间段，还需要同时采集白天与夜晚的数据集，以确保模型在不同光照条件下的稳定性和准确性。

（2）尽管 Openpose 在检测人体姿态方面展现出了较强的鲁棒性，但在夜晚开车视线不佳等特定情境下，其性能仍可能受到一定影响，导致误差的产生。此外，基于二维骨架信息的驾驶员行为识别方法也存在一定的局限性。为了克服这些问题，在下一步研究中采用深度相机 Kinect 来提取驾驶员的三维骨架信息，并融合小目标检测技术，以期更精准地进行驾驶员行为识别。

参考文献

[1] Dingus T A, Guo F, Lee S, etc. Driver crash risk factors and prevalence evaluation using naturalistic driving data[J]. Proceedings of the National Academy of Sciences, 2016, 113(10): 2636-2641.

[2] Ergeneci M, Gokcesu K, Ertan E, etc. An embedded, eight channel, noise canceling, wireless, wearable sEMG data acquisition system with adaptive muscle contraction detection[J]. IEEE Transactions on Biomedical Circuits and Systems, 2018, 12(1): 68-79.

[3] Hansen J H L, Busso C, Zheng Y, etc. Driver modeling for detection and assessment of driver distraction: Examples from the UTDrive test bed[J]. IEEE Signal Processing Magazine, 2017, 34(4): 130-142.

[4] Zhao Z, Xia S, Xu X, etc. Driver distraction detection method based on continuous head pose estimation[J]. Computational Intelligence and Neuroscience, 2020, 2020: 1-10.

[5] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. science, 2006, 313(5786): 504-507.

[6] Shahroudy A, Liu J, Ng T T, etc. Ntu rgb+ d: A large scale dataset for 3d human activity analysis[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2016: 1010-1019.

[7] Li C, Zhong Q, Xie D, et al. Skeleton-based action recognition with convolutional neural networks[C]. Proceedings of the IEEE international conference on multimedia & expo workshops. Piscataway, NJ: IEEE, 2017: 597-600.

[8] Li C, Zhong Q, Xie D, etc. Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation[J]. arXiv preprint, 2018, arXiv: 1804.06055.

[9] Ke Q, Bennamoun M, An S, etc. Learning clip representations for skeleton-based 3d action recognition[J]. IEEE Transactions on Image Processing, 2018, 27(6): 2842-2855.

[10] Duan H, Zhao Y, Chen K, etc. Revisiting skeleton-based action recognition[C]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2022: 2969-2978.

[11] Yan S, Xiong Y, Lin D. Spatial temporal graph convolutional networks for skeleton-based action recognition[C]. Proceedings of the AAAI conference on artificial intelligence. Washington, DC, Menlo Park, CA: AAAI Press, 2018: 32(1).

[12] Shi L, Zhang Y, Cheng J, etc. Skeleton-based action recognition with directed graph neural networks[C]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2019: 7912-7921.

[13] Obinata Y, Yamamoto T. Temporal extension module for skeleton-based action recognition[C].

Proceedings of the 25th international conference on pattern recognition. Piscataway, NJ: IEEE, 2021: 534-540.

[14] Song Y F, Zhang Z, Shan C, etc. Stronger, faster and more explainable: A graph convolutional baseline for skeleton-based action recognition[C]. Proceedings of the 28th ACM international conference on multimedia. New York, NY: ACM, 2020: 1625-1633.

[15] Chen Y, Zhang Z, Yuan C, etc. Channel-wise topology refinement graph convolution for skeleton-based action recognition[C]. Proceedings of the IEEE/CVF international conference on computer vision. Piscataway, NJ: IEEE, 2021: 13359-13368.

[16] Plizzari C, Cannici M, Matteucci M. Skeleton-based action recognition via spatial and temporal transformer networks[J]. Computer Vision and Image Understanding, 2021, 208: 103219.

[17] Vaswani A, Shazeer N, Parmar N, etc. Attention is all you need[J]. Advances in neural information processing systems. Cambridge, MA: MIT Press, 2017, 30.

[18] Wang Q, Peng J, Shi S, etc. IIP-Transformer: Intra-Inter-Part transformer for skeletonbased action recognition[J]. arXiv preprint, 2021, arXiv: 2110.13385.

[19] Li P, Lu M, Zhang Z, et al. A novel spatial-temporal graph for skeleton-based driver action recognition[C]. Proceedings of the 2019 IEEE intelligent transportation systems conference. Piscataway, NJ: IEEE, 2019: 3243-3248.

[20] Pan C, Cao H, Zhang W, etc. Driver activity recognition using spatial-temporal graph convolutional LSTM networks with attention mechanism[J]. IET Intelligent Transport Systems, 2021, 15(2): 297-307.

[21] Martin M, Voit M, Stiefelhagen R. Dynamic interaction graphs for driver activity recognition[C]. Proceedings of the 2020 IEEE 23rd international conference on intelligent transportation systems. Piscataway, NJ: IEEE, 2020: 1-7.

[22] Weyers P, Schiebener D, Kummert A. Action and object interaction recognition for driver activity classification[C]. Proceedings of the 2019 IEEE intelligent transportation systems conference. Piscataway, NJ: IEEE, 2019: 4336-4341.

[23] Behera A, Wharton Z, Keidel A, etc. Deep cnn, body pose, and body-object interaction features for drivers' activity monitoring[J]. IEEE transactions on intelligent transportation systems, 2020, 23(3): 2874-2881.

[24] Karpathy A, Toderici G, Shetty S, etc. Large-scale video classification with convolutional neural networks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2014: 1725-1732.

[25] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos[J]. Advances in neural information processing systems. Cambridge, MA: MIT Press, 2014, 27.

[26] Horn B K P, Schunck B G. Determining optical flow[J]. Artificial intelligence, 1981, 17(1-3): 185-203.

[27] Yue-Hei Ng J, Hausknecht M, Vijayanarasimhan S, etc. Beyond short snippets: Deep networks

for video classification[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2015: 4694-4702.

[28] Feichtenhofer C, Pinz A, Zisserman A. Convolutional two-stream network fusion for video action recognition[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2016: 1933-1941.

[29] Wang L, Xiong Y, Wang Z, etc. Temporal segment networks: Towards good practices for deep action recognition[C]. Proceedings of the European conference on computer vision. Berlin: Springer, 2016: 20-36.

[30] Ji S, Xu W, Yang M, etc. 3D convolutional neural networks for human action recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 35(1): 221-231.

[31] Tran D, Bourdev L, Fergus R, etc. Learning spatiotemporal features with 3d convolutional networks[C]. Proceedings of the IEEE international conference on computer vision. Piscataway, NJ: IEEE, 2015: 4489-4497.

[32] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint , 2014, arXiv: 1409.1556.

[33] Qiu Z, Yao T, Mei T. Learning spatio-temporal representation with pseudo-3d residual networks[C]. Proceedings of the IEEE international conference on computer vision. Piscataway, NJ: IEEE, 2017: 5533-5541.

[34] Tran D, Wang H, Torresani L, etc. A closer look at spatiotemporal convolutions for action recognition[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2018: 6450-6459.

[35] Carreira J, Zisserman A. Quo vadis, action recognition? a new model and the kinetics dataset[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2017: 6299-6308.

[36] Wang X, Girshick R, Gupta A, etc. Non-local neural networks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2018: 77947803.

[37] Feichtenhofer C, Fan H, Malik J, etc. Slowfast networks for video recognition[C]. Proceedings of the IEEE international conference on computer vision, Piscataway, NJ: IEEE, 2019: 6202-6211.

[38] Lin J, Gan C, Han S. Tsm: Temporal shift module for efficient video understanding[C]. Proceedings of the IEEE/CVF international conference on computer vision, Piscataway, NJ: IEEE, 2019: 7083-7093.

[39] Shao H, Qian S, Liu Y. Temporal interlacing network[C]. Proceedings of the AAAI conference on artificial intelligence. Menlo Park, CA: AAAI Press, 2020, 34(07): 11966-11973.

[40] Li Y, Ji B, Shi X, etc. Tea: Temporal excitation and aggregation for action recognition[C]. Proceedings of the IEEE conference on computer vision, Piscataway, NJ: IEEE, 2020: 909-918.

[41] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]. Proceedings of the IEEE

conference on computer vision, Piscataway, NJ: IEEE, 2018: 7132-7141.

[42] Liu Z, Luo D, Wang Y, etc. Towards an efficient architecture for video recognition[C]. Proceedings of the AAAI conference on artificial intelligence. Menlo Park, CA: AAAI Press, 2020: 7-12.

[43] Arefin M R, Makhmudkhujaev F, Chae O, etc. Aggregating CNN and HOG features for real-time distracted driver detection[C]. Proceedings of the 2019 IEEE international conference on consumer electronics. Piscataway, NJ: IEEE, 2019: 1-3.

[44] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]. Proceedings of the Advances in neural information processing systems. Cambridge, MA: MIT Press, 2012, 25.

[45] Chuang Y, Kuo C, Sun S, etc. Driver behavior recognition using recurrent neural network in multiple depth cameras environment[J]. Electronic Imaging, 2019(15): 56-156-7.

[46] Lu M, Hu Y, Lu X. Dilated Light-Head R-CNN using tri-center loss for driving behavior recognition[J]. Image and Vision Computing, 2019, 90: 103800.

[47] Wang K, Chen X, Gao R. Dangerous driving behavior detection with attention mechanism[C]. Proceedings of the 3rd international conference on video and image processing. New York, New York: Association for Computing Machinery, 2019: 57-62.

[48] Hu Y, Lu M Q, Lu X. Spatial-temporal fusion convolutional neural network for simulated driving behavior recognition[C]. Proceedings of the 15th international conference on control, automation, robotics and vision. Piscataway, NJ: IEEE, 2018: 1271-1277.

[49] Hu Y, Lu M, Xie C, etc. Video-based driver action recognition via hybrid spatial–temporal deep learning framework[J]. Multimedia systems, 2021, 27(3): 483-501.

[50] Cao Z, Simon T, Wei S E, etc. Realtime multi-person 2d pose estimation using part affinity fields[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2017: 7291-7299.

[51] Ma N, Zhang X, Zheng H T, etc. Shufflenet v2: Practical guidelines for efficient cnn architecture design[C]. Proceedings of the European conference on computer vision, Berlin: Springer, 2018: 116-131.

[52] Howard A G, Zhu M, Chen B, etc. Mobilenets: Efficient convolutional neural networks for mobile vision applications[J]. arXiv preprint, 2017, arXiv: 1704.04861.

[53] Sandler M, Howard A, Zhu M, etc. Mobilenetv2: Inverted residuals and linear bottlenecks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2018: 4510-4520.

[54] Howard A, Sandler M, Chu G, etc. Searching for mobilenetv3[C]. Proceedings of the IEEE/CVF international conference on computer vision, Piscataway, NJ: IEEE, 2019: 1314-1324.

[55] Wu C Y, Zaheer M, Hu H, etc. Compressed video action recognition[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2018:

6026-6035.

[56] Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network[J]. arXiv preprint, 2015, arXiv: 1503.02531.

[57] Tomar S. Converting video formats with ffmpeg[J]. Linux journal,2016, 2006(146): 10.

[58] Mehta S, Rastegari M. Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer[J]. arXiv preprint, 2021, arXiv: 2110.02178.

[59] Donahue J, Anne Hendricks L, Guadarrama S, etc. Long-term recurrent convolutional networks for visual recognition and description[C]. Proceedings of the IEEE conference on computer vision and pattern recognition. Piscataway, NJ: IEEE, 2015: 2625-2634.

[60] Arnab A, Dehghani M, Heigold G, etc. Vivit: A video vision transformer[C]. Proceedings of the IEEE/CVF international conference on computer vision, Piscataway, NJ: IEEE, 2021: 6836-6846.

致谢

窗外日光弹指过，席间花影坐前移。蓦然回首，百感交集。在师长、亲友的大力支持下，七年的求学生涯走得辛苦却又收获满囊，对于这段难忘的时光我内心有道不尽的不舍。

感谢我的导师杨会成老师，传道授业解惑，师恩润物无声。由衷感谢杨老师三年来的关怀与指导，从论文选题到定稿再到最终成文，离不开老师的帮助，为我论文写作中遇到的瓶颈提供思路与方法，耐心解答我的每一个疑问，同时也感谢胡耀聪老师和徐可老师的无私帮助，祝工作顺利，幸福安康。感谢我的家人，二十多年来对我学业的支持，在生活中的关心，在失落时的关怀，永远做我坚实的避风港，祝身体健康，平安顺遂。感谢我的同门林园园和李雯婷，一起度过三年的美好时光，生活和学习中的困难都能一起解决，彻夜长谈的日子，值得一生怀念，祝前程似锦。最后，感谢所有参加本论文评审和答辩老师们的辛苦付出。

曾经的我对许多事情抱有忧虑，如今来看，已是“轻舟已过万重山”，在人生接下来的旅途中，我将乐观对待每一天，以感恩之心对待身边的人，以美好之愿祈谢世间之情。

至此，全文毕。

攻读硕士学位期间的研究成果

1 学术论文

[1] 帅真, 杨会成, 胡耀聪, 等. 基于压缩视频的驾驶行为识别方法[J]. 传感技术学报.（已录用）

[2] Hu, Y., Shuai, Z., Yang, H. et al. ESDAR-net: towards high-accuracy and real-time driver action recognition for embedded systems. Multimed Tools Appl 83, 18281–18307 (2024).

2 发明专利

[1] 胡耀聪, 帅真, 杨会成, 等. 面向车载设备的压缩视频驾驶员行为识别方法[P].（实审，申请号：202210666671 .6）