applied sciences

MDPI

Article

STA-Net: A Spatial–Temporal Joint Attention Network for Driver Maneuver Recognition, Based on In-Cabin and Driving Scene Monitoring

Bin He [1], Ningmei Yu [1,*], Zhiyong Wang [2,3] and Xudong Chen [2]

[1] School of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China; 1210311008@stu.xaut.edu.cn
[2] National Key Laboratory of Human–Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an 710049, China; zhywang@stu.xjtu.edu.cn (Z.W.); 2223515328@stu.xjtu.edu.cn (X.C.)
[3] School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China
* Correspondence: yunm@xaut.edu.cn

Abstract: Next-generation advanced driver-assistance systems (ADASs) are a promising direction for intelligent transportation systems. To achieve intelligent security monitoring, it is imperative that vehicles possess the ability to accurately comprehend driver maneuvers amidst diverse driver behaviors and complex driving scenarios. Existing CNN-based and transformer-based driver maneuver recognition methods face challenges in effectively capturing global and local features across temporal and spatial dimensions. This paper proposes a Spatial–Temporal Joint Attention Network (STA-Net) to realize high-efficient temporal and spatial feature extractions in driver maneuver recognition. First, we introduce a two-stream architecture for a concurrent analysis of in-cabin driver behaviors and out-cabin environmental information. Second, we propose a Multi-Scale Transposed Attention (MSTA) module and Multi-Scale Feedforward Network (MSFN) to extract features at multiple scales, addressing receptive field inadequacies and combining high-level and low-level information. Third, to address the information redundancy in multi-scale features, we propose a Cross-Spatial Attention Module (CSAM) and Multi-Scale Cross-Spatial Fusion Module (MCFM) to select essential features. Additionally, we introduce an asymmetric loss function to effectively tackle the issue of sample imbalance across diverse categories of driving maneuvers. The proposed method demonstrates a remarkable accuracy of 90.97% and an F1 score of 89.37% on the Brain4Cars dataset, surpassing the performance of the methods compared. These results substantiate the fact that our approach effectively enhances driver maneuver recognition.

Keywords: driver maneuver recognition; deep learning; multi-scale spatial–temporal attention

Citation: He, B.; Yu, N.; Wang, Z.; Chen, X. STA-Net: A Spatial–Temporal Joint Attention Network for Driver Maneuver Recognition, Based on In-Cabin and Driving Scene Monitoring. *Appl. Sci.* **2024**, *14*, 2460. https://doi.org/10.3390/app14062460

Academic Editor: Andrea Prati

Received: 30 November 2023
Revised: 21 February 2024
Accepted: 24 February 2024
Published: 14 March 2024

1. Introduction

Research on road safety indicates that the majority of traffic accidents stem from improper driver maneuvers. Despite achieving satisfactory performance in specific scenarios, fully autonomous driving remains a long-term objective due to the ongoing necessity for comprehensive legislation, regulations, and infrastructure development [1]. Consequently, human–machine cooperative driving continues to be a crucial research direction in intelligent transportation systems. Understanding the driver's intent is an essential prerequisite for effective human–machine interaction and facilitating autonomous vehicle decision-making that aligns with drivers' preferences in uncertain environments as well as alerting drivers during hazardous situations is necessary [2]. However, predicting human intent poses challenges, which are attributed to factors affecting human drivers, such as distraction, the driver's emotional state, and lack of concentration, thereby leading to potential road hazards.

In recent years, research has focused on exploring new technologies for comprehensive perception in intelligent vehicles, making it possible to predict driver intent over time. Jain et al. [3] introduced the Autoregressive Input–Output HMM (AIO-HMM) model, which processes both internal (2D facial features) and external features, predicting potential driver actions seconds before driving maneuvers. They also provided the Brain4Cars dataset, consisting of 1180 miles of natural highway and city driving, for method evaluation.

The dataset's video frames contain dynamic information on driver maneuver patterns and road traffic conditions. Gebert [4] and Xing [5], among others, conducted statistical analyses on different driving intent expressions, finding a high correlation between driving intent and driver maneuvers. When drivers are about to change their maneuver, they exhibit corresponding actions, such as head posture [3,6–8], specific maneuvers, and eye movements while checking the rearview mirror [9,10], providing crucial evidence for intent inference.

This work is significant, as it lays the foundation for driver assistance systems and proposes a method for predicting driver actions using dynamic visual data inside and outside the vehicle. It addresses the challenge of predicting driver operations seconds in advance, allowing for timely warnings to drivers and contributing to the development of next-generation advanced driver-assistance systems (ADASs), reducing road hazard risks.

Several researchers have proposed improvements to this pipeline. Jain et al. [11] suggested a deep learning architecture based on recurrent neural networks (RNN-LSTMs), which upgrades internal driver features from 2D to 3D facial features, enhancing the accuracy of driver maneuver prediction by fusing information from multiple sensors. Moussaid et al. [12] presented a method using driver facial information to predict lane-changing actions, implementing a model based on CNN-LSTMs for analyzing driver actions before lane changes. Tonutti et al. [8] introduced a method based on Domain-Adversarial Recurrent Neural Networks (DA-RNNs), improving the generalization capability of driving manipulation prediction. Gebert et al. [4] combined 3D-ResNet with an LSTM to predict driver intent by analyzing driver motion and external vehicle video data. Rong et al. [13] proposed a driver intent prediction method based on monitoring internal and external scenes, achieving better prediction performance with fewer parameters.

Analyzing Gebert et al. [4] and Rong et al. [13], with both using two branches for internal and external video processing, reveals their distinct approach. While Gebert et al. [4] computed optical flow from internal videos, Rong et al. [13] calculated it from external videos. These studies differ from previous approaches that use numerical data (e.g., lane numbers, speed) as external features. Instead, they directly extract external features from external videos by using CNN models.

Previously, LSTMs played a crucial role in driver maneuver recognition due to their ability to capture long-distance dependencies. However, LSTMs face challenges in capturing extended dependencies, and they present other challenges, such as high computational complexity, susceptibility to video noise, and interpretability issues. Recent studies favor 3D-CNN models for spatiotemporal feature extraction, addressing LSTMs' limitations. However, learning effective spatiotemporal representations remains a challenge due to local redundancy and global dependence issues.

A combination of 3D convolutional neural networks (3D-CNNs) and spatiotemporal transformers has emerged as a promising solution for better driver intent inference. However, both have limitations. While 3D-CNNs reduce spatiotemporal redundancy, their finite receptive fields make learning long-term dependencies difficult. Spatiotemporal transformers excel at capturing global dependencies but introduce redundancy in shallow layers when encoding local spatiotemporal features.

Additionally, two challenges affect driver intent inference accuracy. First, inadequate utilization of external video information limits the perception and understanding of the surrounding environment. Rong et al. [13] demonstrated that external videos complement internal driver videos, providing essential information. This external information is necessary to avoid misidentification in challenging situations. Second, the imbalance in training data samples, with straight driving maneuvers being more common than turns and lane

changes, poses difficulties in machine learning model training. This imbalance may cause the model to favor dominant classes, reducing accuracy in predicting minority classes.

Inspired by transformer models, we propose the Spatial–Temporal Joint Attention Network (STA-Net), combining CNN and transformers in a dual-stream framework. The contributions of this study can be summarized as follows:

(1) We propose a two-stream network to extract in-cabin driver behavior and out-cabin environmental information, addressing spatiotemporal redundancy and insufficient use of driving scene information.
(2) We employ the joint learning of the CNN and the transformer to fuse spatiotemporal information at different levels. CNN focuses on low-level local features to reduce redundancy, while the transformer captures high-level global information to address long-term dependencies.
(3) We introduce an asymmetric loss function to tackle the problem of imbalanced training data, reducing the negative impact of sample imbalance on model optimization.

2. Related Works

2.1. Driver Maneuver Recognition

Driver Maneuver Recognition is primarily about intelligent vehicles monitoring the driver's maneuver in real-time. Using machine learning methods, the vehicle analyzes this maneuver data to determine the current actions of the driver and anticipate their future intentions. This enables the provision of various driver assistance features or timely alerts to the driver, thereby enhancing driving safety and efficiency. Current research includes traditional machine learning methods, combinations of 2D CNNs and RNNs, methods combining the 3D CNN and optical flow estimation, and dual-stream frameworks. Here are some representative works:

Combination of 2D CNNs and RNN Techniques: Xing et al. [14] proposed a method that combines RNN technology with 2D CNN in video processing to handle spatial and temporal information. The 2D CNN is used as an encoder to extract spatial features from the driver's maneuver sequence, and RNN technology serves as a decoder for time modeling to infer the driver's intentions. This method leverages both advantages, improving the efficiency of feature extraction and temporal processing. However, the combined model of 2D CNNs and RNNs is often more complex than using either method alone, leading to challenges in model complexity, increased parameter count, and higher computational resource requirements.

Combining 3D CNN and Optical Flow Estimation: Gebert et al. [4] proposed a vision-based 3D convolutional residual learning method using optical flow images from the driver's cabin to predict driver intentions. While this method has advantages in capturing spatiotemporal information, accurate maneuver recognition, efficiency, and flexibility in video understanding, it suffers from drawbacks such as numerous parameters, high hardware requirements, large data volume requirements, and sensitivity to lighting and occlusion conditions.

Dual-Stream Framework: Rong et al. [13] introduced a ConvLSTM-based autoencoder for extracting vehicle motion information from traffic scenes. They proposed a deep network framework to simultaneously study features from two directions (inside and outside the vehicle) without manual encoding or handcrafted features. This structure achieves advanced driver maneuver prediction performance with fewer parameters than previous works. However, it requires more data to train a finer decoder to interpret long-term motion better, presenting challenges in predicting longer-term motion and computational efficiency due to optical flow estimation. Chen et al. [15] introduce an intelligent vehicle driving intention inference method based on spatiotemporal feature enhancement (STEDII-GRU), aiming to improve the accuracy of driving intention inference. The method first utilizes a pre-trained dual-stream network (SlowFast network) as the backbone for feature extraction. Subsequently, the low frame rate and high frame rate paths are employed to process internal driver behavior video data and external forward traffic scene data. Finally, the

joint spatiotemporal features are input into the GRU to obtain the most probable intention. The proposed STEDII-GRU method demonstrates high accuracy in driving intention inference, providing a promising solution for enhancing intelligent vehicles' safety and driving performance. Ma et al. [16] introduce a novel framework for driver intention prediction. The framework, CEMFormer, employs spatial-temporal transformers to unify memory representations for improved prediction accuracy. It integrates data from both in-cabin and external cameras, enhancing the prediction through historical data fusion and a novel context-consistency loss. Bonyani et al. [17] explore a deep neural network framework to anticipate driver maneuvers and enhance takeover readiness in automated driving. Utilizing the Brain4Cars dataset, the model integrates DenseNet, LSTM, attention mechanisms, and FlowNet2 to predict driver intentions up to 4 s in advance. The study assesses driver readiness through in-cabin and out-cabin video data and demonstrates improved prediction accuracy and performance against existing models.

Other latest methods: Zhang et al. [18] present a novel method for recognizing driver lane-changing intention (LCI) in a connected environment. Utilizing a driving simulator, it determines LCI time windows and feature parameters, including yaw rate, vehicle speed, and driver gaze. A new LCI model using phase-space reconstruction and Swin Transformer for classification is proposed, surpassing classical machine learning algorithms in accuracy and addressing vanishing gradient issues in long time-series data. This method is significant for lane-changing assistance and human–machine co-driving systems, enhancing traffic safety and efficiency. Li et al. [19] present a driving behavior prediction model that blends gradient boosting decision tree (GBDT), convolutional neural networks (CNN), and long short-term memory networks (LSTM) in a wide-deep framework. This model aims to extract comprehensive driving behavior characteristics and enhance the interpretability of CNN-LSTM models. The integration of GBDT allows for the quantitative analysis of vehicular interactions. Chen et al. [20] introduce a transfer learning-based approach for recognizing various driving behaviors using a convolutional neural network (CNN) model. This method utilizes vehicle kinematic data and drivers' facial expressions, enhancing recognition accuracy for patterns like acceleration, deceleration, turning, lane changing, and lane keeping. The transfer learning technique effectively refined pre-trained models with limited data, significantly improving performance and training cost efficiency. This approach is particularly valuable for challenging data collection scenarios, such as with heavy-duty freight vehicles. Wu et al. [1] present a model for identifying lane-changing maneuvers using the HighD dataset. Focusing on acceleration and velocity as physical data, the research implements a k-nearest neighbor (KNN) classification model. The findings indicate high classification accuracy, suggesting the model's potential utility in advanced driver-assistance systems to improve road safety.

2.2. Video Understanding

The key to understanding driver intent based on visual information lies in accurately interpreting both in-car and out-of-car visual data. In recent years, various deep learning methods for video understanding have been proposed. The purpose of video understanding is to enable computers to interpret and understand video content like humans. Models process videos into sequentially ordered frames, and current video understanding primarily employs recurrent neural networks (RNNs), convolutional neural networks (CNNs), and transformer-based architectures to extract and analyze spatiotemporal features from image sequences, ultimately making predictions based on these data.

Recurrent Neural Networks (RNNs): Sun et al. [21] proposed a novel dual-stream LSTM architecture called L2STM for human action recognition in videos, addressing non-stationary dynamics in long-term motion. Li et al. [22] introduced the RTPR network architecture for video action detection, utilizing action proposals from previous frames to assist detection in the current frame through a recurrent neural network. In summary, RNNs in video understanding can capture temporal information, handle sequence data, facilitate end-to-end learning, and exhibit high flexibility. However, they also face challenges in

training, handling variable-length sequences, dealing with multimodal data, and capturing long-term dependencies.

Convolutional Neural Networks (CNNs): Carreira et al. [23] proposed a novel Two-Stream Inflated 3D ConvNets model (I3D) that extends 2D ConvNets to 3D ConvNets, addressing spatiotemporal modeling issues in video understanding. Feichtenhofer et al. [24] introduced a new video recognition model, the SlowFast network, which incorporates two different paths (Slow Path and Fast Path) to handle slow and fast video information separately. CNNs in video understanding can effectively capture local patterns and structural information, automatically learn features, and perform parallel computation on large-scale data. However, they also have limitations in capturing global information, dealing with variable-length sequences, having many parameters, and facing imbalances in spatial and temporal information.

Transformer-based Architectures: Xu et al. [25] presented an algorithm called Long Short-Term Transformer (LSTR) for online action detection, addressing the effective modeling of long-time sequence data and online action detection in videos. Li et al. [26] proposed a multi-scale visual transformer (MViT) for video and image recognition, connecting the basic idea of a multi-scale feature hierarchy with transformer models. Bertasius et al. [27] introduced a video classification model, TimeSformer, based on self-attention mechanisms, incorporating Divided Space-Time Attention (T + S) to calculate time and space self-attention scores separately. Arnab et al. [28] presented a pure transformer-based video classification model, ViViT, introducing a method called Tubelet Embedding to capture spatiotemporal information in videos effectively.

While transformer-based models in video understanding have advantages such as capturing long-term dependencies, strong parallel computation capabilities, applicability to multiple tasks, and end-to-end learning, they also come with disadvantages, including high computational resource requirements, limitations on sequence length, high training time, and cost, and a high number of parameters.

3. Methods

The spatiotemporal joint reasoning process of driving intention is a solution to the sequence image classification problem. In our work, we propose a novel driving intention inference framework, Spatial–Temporal Joint Attention Network (STA-NET), which simultaneously utilizes two input sources: internal and external videos, as shown in Figure 1. One branch learns spatial semantic features in traffic videos. In contrast, the other branch learns spatial semantic features in driver videos, thereby addressing the deficiency of relying solely on in-vehicle driver spatiotemporal features for driving intention inference. As illustrated in Figure 1, the backbone network is a parallel dual-branch network. The basic structure of the backbone network is mainly composed of Spatial–Temporal Joint Attention Block (STA Block) and Cross-Spatial Attention Module (CSAM), where the STA Block consists of Multi-Scale Transposed Attention (MSTA) and Multi-Scale Feedforward Network (MSFN). The STA Block adopts a joint CNN and transformer approach to simultaneously extract driver maneuver features and spatiotemporal features of the traffic scene. MSTA and MSFN alleviate the insufficient receptive field at different levels and enhance the richness of spatiotemporal feature information. At each stage, features extracted by the STA Block at the same scale are aggregated through CSAM. The MCFM (Multi-CSAM Fusion Module) aggregates different scales of in-vehicle driver features and traffic scene features at different stages for driving intention inference.

More specifically, we hierarchically stack STA Block units to construct our network for spatiotemporal learning. As shown in Figure 1, our network comprises four stages with channel numbers 64, 128, 256, and 512, respectively. We build the backbone network of the STA framework based on the quantities of STA Block units in each stage, which are [5,7,8,20]. We employ MSTA (Equation (1)) at each STA Block to reduce spatiotemporal redundancy. We normalize the data using LN [29]. Before the first stage, we apply a $3 \times 4 \times 4$ convolution with a stride of $2 \times 4 \times 4$, meaning both spatial and temporal

dimensions are downsampled. Before the other stages, we use a $1 \times 1 \times 2$ convolution with a stride of $1 \times 1 \times 2$. Finally, the spatiotemporal average pooling and fully connected layers are employed for the ultimate prediction. In this way, our STA-Net, with an insightful unified framework, addresses video redundancy and dependencies. Each module is detailed as follows.

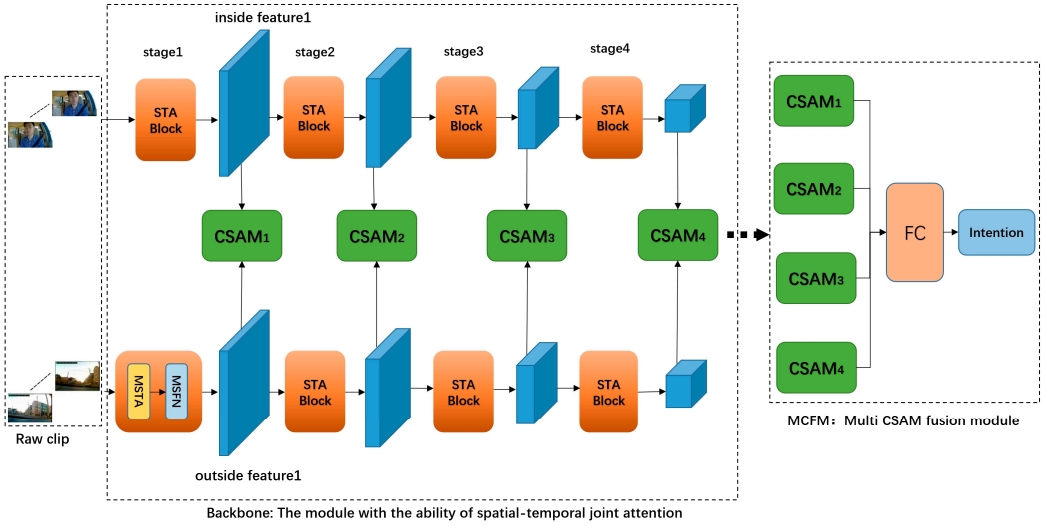

Figure 1. Spatial–Temporal Joint Attention Network (STA-NET).

3.1. Framework for Spatiotemporal Feature Extraction Based on Dual-Stream Networks

As previously mentioned, the innovative driving intent inference framework, STA-NET, simultaneously handles video data inside and outside the vehicle. One branch learns spatial semantic information from traffic videos, while the other learns from driver videos. The specific structures are described below.

3.1.1. STA-Block

To overcome spatiotemporal redundancy and dependency issues, we propose a novel module called Spatial–Temporal Joint Attention Block (STA-Block), as illustrated in Figure 1. We leverage the fundamental Transformer architecture [30] and tailor it specifically for efficient and effective spatiotemporal representation learning. Specifically, the STA-Block comprises two key modules: the Multi-Scale Transposed Attention (MSTA) and the Multi-Scale Feedforward Network (MSFN). Our MSTA adeptly addresses local video redundancy and global video dependencies by extracting features at different scales in both shallow and deep layers. Finally, we introduce a Feedforward Network (FFN) with two linear layers to enhance each token pointwise.

a. Multi-Scale Transposed Attention

As mentioned above, we aim to address two main challenges: significant local redundancy and intricate global dependencies, aiming for efficient and effective spatiotemporal representation learning. However, existing methods, such as popular 3D CNNs and spatiotemporal transformers, often focus solely on one of these challenges. Therefore, we introduce a novel approach called Multi-Scale Transpose Attention (MSTA). Designed in a concise transformer format, MSTA seamlessly unifies 3D convolution and spatiotemporal self-attention, adeptly tackling video redundancy and dependencies at different levels in both shallow and deep layers.

Due to the substantial computational overhead of transformers, primarily from the self-attention layer, applying the traditional self-attention mechanism (SA) [30,31] becomes impractical for most video-understanding tasks. In the conventional self-attention mechanism, the time and memory complexity of key-query dot-product interactions grows quadratically with the spatial resolution of the input. To address this issue, we propose

Multi-Scale Transpose Attention (MSTA), which exhibits linear complexity, as depicted in Figure 2. The key distinction lies in MSTA applying self-attention across channels, calculating cross-covariance across channels to generate an attention map implicitly encoding global context. As another integral component of MSTA, we introduce depthwise convolution to emphasize 3D local context, performing this operation before computing feature covariance to generate a global attention map.

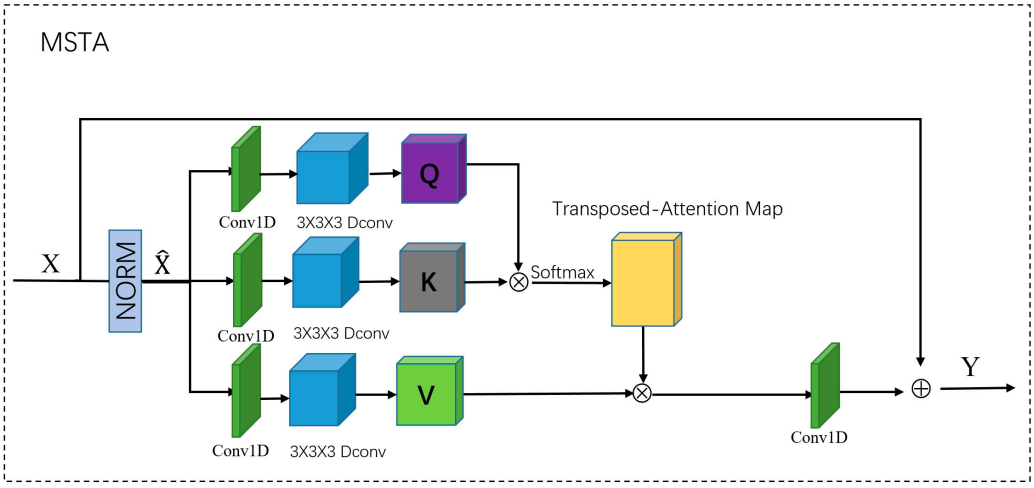

Figure 2. Multi-Scale Transpose Attention (MSTA) module.

From the Layer Normalization tensor $\hat{\mathbf{X}} \in \mathbb{R}^{H\times W\times C\times T}$, our Multi-Scale Transpose Attention (MSTA) initially generates query (**Q**, key (**K**), and value (**V**) projections, enriching local contexts. This is achieved by applying a $1 \times 1 \times 1$ convolution to aggregate spatiotemporal cross-channel context, followed by a $3 \times 3 \times 3$ depthwise convolution to encode channel-level spatiotemporal context, resulting in $\mathbf{Q} = \mathbf{W}_d^Q \mathbf{W}_p^Q \hat{\mathbf{X}}$, $\mathbf{K} = \mathbf{W}_d^K \mathbf{W}_p^K \hat{\mathbf{X}}$, and $\mathbf{V} = \mathbf{W}_d^V \mathbf{W}_p^V \hat{\mathbf{X}}$. Here, $\mathbf{W}_p^{(\cdot)}$ represents a $1 \times 1 \times 1$ pointwise convolution, and $\mathbf{W}_d^{(\cdot)}$ represents a $3 \times 3 \times 3$ depthwise convolution. In summary, the MSTA process is defined as follows:

$$\begin{aligned} \mathbf{Y} &= \mathbf{W}_p \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) + \hat{\mathbf{X}}, \\ \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) &= \mathbf{V} \cdot \text{Soft max}(\mathbf{K} \cdot \mathbf{Q}/\alpha), \end{aligned} \tag{1}$$

Here, $\hat{\mathbf{X}}$ and $\mathbf{Y}$ are the input and output feature maps, respectively. In this context, α is a learnable scaling parameter used to control the magnitude of the dot product between **K** and **Q** before applying the SoftMax function. Similar to traditional multi-head self-attention [31], we divide the number of channels into "heads" and independently learn attention maps in parallel.

b. Multi-Scale Feedforward Network

A conventional Feedforward Network (FN) [30,31] performs the same operation at each spatiotemporal position to transform features. This network utilizes two $1 \times 1 \times 1$ convolutions, where the first convolution layer is employed to expand feature channels (typically expanded by a factor of $\gamma = 4$), and the second convolution layer reduces the channel count back to the original input dimensions.

The relationship between these two operations lies in that the former aims to increase the spatial dimension of features to capture advanced features of spatiotemporal characteristics more effectively. The latter's task is to map these advanced features back to the original input dimensions to fuse them with the outputs of other layers. Although the goals of these two convolutional layers differ, they operate in the same space, namely the dimensions of the original input. Thus, they can perform dot product calculations in the same space, effectively merging the outputs of the two convolutional layers. This design

allows the network to operate in different feature spaces and merge these feature spaces when necessary.

In this work, we made two fundamental modifications to the Feedforward Network (FN) to enhance representation learning: (1) introducing a gating mechanism and (2) adopting depthwise convolutions. The Multi-Scale Feedforward Network (MSFN) structure we designed is shown in Figure 3.

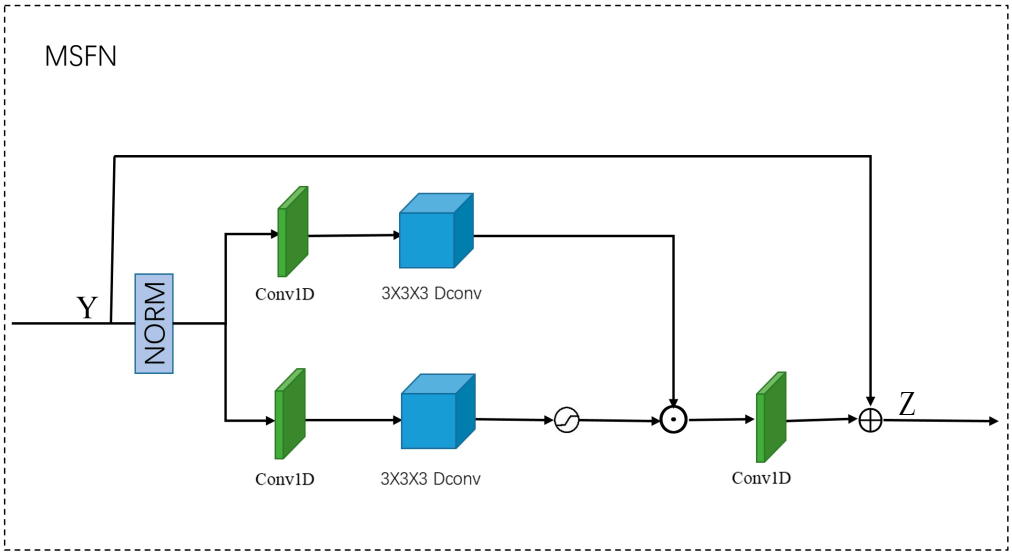

Figure 3. Multi-Scale Feedforward Network (MSFN).

The gating mechanism helps regulate the flow of information in the network hierarchy, enabling each layer to focus on finer image attributes. This mechanism is achieved through the element-wise product of the two parallel paths of the linear transformation layer, with one path passing through the GELU non-linearity [32] activation. Similar to Multi-Scale Transpose Attention (MSTA), we also introduced depth-wise convolutions in Multi-Scale Feedforward Network (MSFN) to encode information from spatially adjacent positions, allowing the model to capture channel-specific information better. This helps improve the model's ability to distinguish between different features, enabling it to learn more discriminative feature representations more effectively. Given that the input tensor $\mathbf{Y} \in \mathbb{R}^{H \times W \times C \times T}$, the representation of MSFN is formulated as:

$$\begin{aligned} \mathbf{Z} &= \mathbf{W}_p^0 \text{Gating}(\mathbf{Y}) + \mathbf{Y}, \\ \text{Gating}(\mathbf{Y}) &= \phi(\mathbf{W}_d^1 \mathbf{W}_p^1(\text{LN}(\mathbf{Y}))) \odot \mathbf{W}_d^2 \mathbf{W}_p^2(\text{LN}(\mathbf{Y})), \end{aligned} \tag{2}$$

where $\odot$ represents element-wise multiplication, ϕ represents the GELU non-linearity, and LN is Layer Normalization [29]. Overall, MSFN controls the flow of information at each hierarchical level in our pipeline, allowing each level to focus on complementary fine details with other levels. In other words, compared to MSTA, MSFN provides different roles, focused on enriching features with contextual information). In summary, MSFN can further blend token context at each spatiotemporal position to improve classification accuracy.

3.1.2. Cross-Spatial Attention Module

To address the challenge of integrating the features of in-car driver maneuvers and the traffic motion scene features at the same scale, we propose a novel module called the Cross-Spatial Attention Module (CSAM), as illustrated in Figure 4.

As illustrated in Figure 4, within each of the four stages of the STA backbone, the features of the in-car driver maneuver and the traffic motion scene are aggregated separately at the same scale. Specifically, we initially aggregate the in-car driver maneuver sequence feature (inside feature1) and the traffic motion scene sequence feature (outside feature1)

using 3D-CNN. This allows the simultaneous consideration of different aspects of the input data in the spatiotemporal dimensions. As the motion features inside and outside the car often contain information at different levels and types, aggregating these features comprehensively captures the spatiotemporal relationships in the input data. By leveraging their complementarity, the model gains a more comprehensive understanding of the data's characteristics, enhancing its expressive power. Furthermore, feature fusion enables the model to better adapt to various input variations and noise, contributing to improved robustness across different in-car driver environments and external traffic conditions.

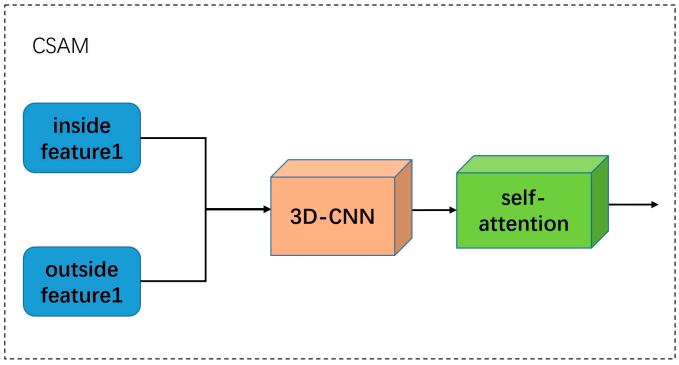

Figure 4. Cross-Spatial Attention Module (CSAM).

To further explore and aggregate the extracted features after the fusion by 3D-CNN, we introduce a self-attention mechanism in CSAM. This allows more effective capture of long-distance dependencies between different positions in spatiotemporal data, enhancing the model's understanding of the global structure. By combating positional biases and dynamically focusing on the importance of different spatiotemporal points, the model's performance and expressive capabilities are improved when dealing with sequential or volumetric data.

3.1.3. Multi-CSAM Fusion Module

To comprehensively capture information, enhance expressiveness, improve robustness, and mitigate overfitting risks, we performed a secondary fusion of features extracted from different stages and scales, as depicted in the Multi-CSAM Fusion Module (MCFM) in Figure 1.

Specifically, we started by downsampling the CSAM1, CSAM2, and CSAM3, fused by the CSAM module at different stages, to match the scale of CSAM4. Subsequently, we applied the same spatial dimension reduction operation to each scale of features, using spatiotemporal average pooling. The dimension-reduced features from all scales were then concatenated to form a single feature vector. A dropout layer was also introduced after the final average pooling layer to prevent overfitting. Finally, the concatenated feature vector was input into fully connected layers to output the ultimate prediction for driver intent recognition.

This process of making different-scale features consistent facilitates parameter sharing, improving computational efficiency, ensuring dimension consistency, and avoiding information loss. It simplifies the model's learning task, reduces the risk of overfitting, and enhances the model's generalization ability and performance across various tasks.

By fusing features from different scales and stages for driver intent recognition, the model comprehensively captures different levels and details of input data, improving its global understanding of data features. Fusing features from different scales enhances the model's expressive power, enabling it to better learn and represent complex data patterns. Furthermore, the fusion of features from multiple scales helps improve the model's robustness to scale and structural variations, making it more resilient in different environments. Multi-scale feature fusion also aids in reducing the model's parameter count, lowering the risk of overfitting, and enhancing its generalization performance. In summary, the multi-scale feature fusion followed by classification through fully connected layers is well-suited for driver intent recognition.

3.2. Asymmetric Loss

According to statistics on driving paths, the straight driving maneuver is more common compared to turning and lane-changing, resulting in the issue of class imbalance across various samples. This imbalance challenges machine learning models during training, as the dominant class samples are more abundant in the training set. The model may tend to learn features and patterns of the dominant class more strongly, leading to insufficient learning for minority classes and increased difficulty in training the model.

We introduce a weighted cross-entropy loss function to address the problem of imbalanced training data in driver intent inference and to ensure that the model adapts better to the sample distribution of different classes while ensuring a more balanced contribution of the loss function to each class. In this context, the loss for each class is multiplied by weight, and we adjust the loss for each sample by setting the weight to the reciprocal of the number of samples in the training set, followed by calculating the average loss. This method helps prevent overfitting to classes with a more significant number of samples, ensuring that each class appropriately impacts the overall loss for more effective model training. Specifically, for the 5-class classification of driver intent inference, the weighted cross-entropy loss function can be defined as follows:

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{5}\omega_j \cdot y_{ij} \cdot \log(\hat{y}_{ij}) \tag{3}$$

where N is the total number of training samples. ω_j is the weight for class j, set as the reciprocal of the number of samples for that class in the training set. y_{ij} is whether the i-th sample in the actual labels belongs to class j. $\hat{y}_{ij}$ is the model's predicted output, representing the probability that sample i belongs to class j.

Setting the weights in this manner helps address the issue of imbalanced samples and improves the prediction accuracy for each class. The specific weight adjustments should be tuned and experimented with based on the dataset's characteristics and the task of finding the optimal weight configuration.

Our experiments showed that identifying certain maneuvers, such as left and right lane changes or turns, can be particularly challenging. This difficulty arises from the dataset's distribution and the inherent characteristics of these maneuvers. Therefore, we adjusted the weights in our model to allocate more significance to these harder-to-recognize categories. Our approach involves increasing the weights for categories based on their difficulty level, with the current coefficients refined through continuous hyperparameter tuning.

Here, we provide an example weight set for the cross-entropy loss function. Based on the distribution of driving actions in our training dataset, as well as finer adjustments made according to the difficulty level of model recognition for different categories, we arrived at these parameter weights as follows (assuming a hypothetical distribution for illustration purposes):

Go Straight: weight = 0.3
Left Lane Change: weight = 1.5
Left Turn: weight = 1.2
Right Lane Change: weight = 1.0
Right Turn: weight = 1.0

We found that applying these weights enhanced the model's prediction accuracy for minority classes and those more challenging to recognize, without significantly impacting overall performance.

4. Experiments and Analysis

4.1. Datasets and Evaluation Metrics

This section delineates the datasets utilized in the experiments and the evaluation criteria employed.

We evaluate the performance of our proposed maneuver prediction method using the publicly available Brain4Cars dataset [33]. This dataset comprises 594 video segments, The

Brain4Cars [3] dataset includes driver observation videos (1088 px × 1920 px, 25 fps) and videos of the outside scenes (480 px × 720 px, 30 fps) recorded simultaneously. The dataset consists of five driving maneuver categories: go straight, left lane change, left turn, right lane change, and right turn. Moreover, samples with no simultaneous recordings of the inside and outside view are considered invalid and not further used in our study.

We use a 5-fold cross-validation for all the experiments in this work, aligning with previous works using the Brain4Cars dataset [3,4,6–8]. The final evaluation metrics include average accuracy and F1 score, along with their standard deviations.

In this research, we utilize accuracy, F1 score, and confusion matrix to evaluate the driver intent recognition performance of both the proposed model and other models. Accuracy (Acc) and F1 score are computed using Equations (4) and (7) [34,35], outlined as follows:

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{4}$$

$$\text{Pr} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{5}$$

$$\text{Re} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

$$\text{F1} = \frac{2\text{Pr} \cdot \text{Re}}{\text{Pr} + \text{Re}} \tag{7}$$

In the equations, TP stands for true positive, indicating cases where both the true label and the predicted label are positive; TN stands for true negative, signifying instances where both the true label and the predicted label are negative; FP corresponds to false positive, representing situations where the true label is negative, but the predicted label is positive; and FN denotes false negative, indicating scenarios where the true label is positive, but the predicted label is negative. Pr and Re refer to precision and recall, computed using Equations (5) and (6), respectively, while the F1 score is the harmonic mean.

4.2. Experiment Environment

The training process of STA-NET adopts a transfer learning strategy. The backbone feature extraction network is initialized with weights from Kinetics-400 [36] (a human action dataset). Subsequently, the entire framework is trained. The proposed method, implemented using the PyTorch deep learning framework, The version of PyTorch is 2.1.0, employs AdamW [37] as the optimizer with a weight decay of 0.05. A cosine learning rate scheduler [38] is utilized, setting the base learning rate to 1×10^{-5}. The resolutions of in-cabin and external vehicle camera streams are both set to 224 × 224. The model is trained for 200 epochs on an NVIDIA RTX 3090Ti GPU with 24 GB of memory. The Brain4Cars dataset [3] records video sequences of driving maneuvers. A total of 625 samples are collected, comprising 234 forward samples, 124 left lane change samples, 58 left turn samples, 123 right lane change samples, and 55 right turn samples. Among these, 80% of the video sequences are used for training, and the remaining 20% for testing.

4.3. Pre-Processing and Data Augmentation

To pre-process the data, we first extracted frames from the videos and resized all inputs to a uniform resolution of 224 by 224 pixels. Subsequently, we applied several data augmentation techniques to enhance the dataset and increased the robustness of our model. These techniques include translating the images by 6 pixels in each direction, applying a flip-left-to-right (flipLR), which necessitates a corresponding label change (e.g., a 'turning left' label becomes 'turning right', or the driver's behavior on the left side becomes that on the right side, which covers a broader range of usage scenarios.), and implementing cutout [39]—a method that randomly masks out square regions of the image. Additionally, we employed Augmix [40], which combines various augmentations such as auto contrast, equalization, posterization, and solarization to create a diverse set of training

examples. This comprehensive augmentation approach not only amplified our dataset but also ensured broader scenario coverage. Further details of these techniques are discussed in the referenced literature.

4.4. Ablation Experiments

To thoroughly evaluate the practical impact of the modules in improving the performance of the baseline method in real-world scenarios, this study conducted ablation experiments.

Two ablation experiments were performed: (1) Comparison between the improved cross-entropy loss function (ICELF) and the regular cross-entropy loss function, assessing the accuracy of driver intent recognition results under these two scenarios. (2) Comparison of the accuracy of driver intent recognition results obtained by adding the CSAM and MCFM against directly using CSAM4. as shown in Table 1.

Table 1. The results of the ablation experiments on the Brain4Cars dataset.

Module		Accuracy ± SD (%)	F1 ± SD (%)
CSAM and MCFM	ICELF		
		81.35 ± 0.52	80.39 ± 1.63
	√	82.16 ± 5.18	83.73 ± 2.96
√		88.67 ± 0.63	88.32 ± 3.72
√	√	**90.97 ± 0.72**	**89.37 ± 1.56**

√ indicates that this module was added, the bold indicates the final result of this work after adding all modules.

We systematically tested the impacts of CSAM and MCFM and the improved cross-entropy loss function on the accuracy of driver intent recognition by modifying one component at a time while keeping the other unchanged. The experiments utilized both in-cabin and external camera views. According to the results in Table 1, CSAM and MCFM increased the ACC score by 7.32% compared to the baseline model. Simultaneously, the improved cross-entropy loss function raised the ACC score by 0.81%. These results indicate that these two components complement each other in terms of performance and variance, playing crucial roles in enhancing the accuracy of driver intent recognition. When used together, their accuracy is 90.97%, with an F1 score of 89.37%.

4.5. Comparative Experiments of Existing Methods

In Table 2, we summarized and compared our work with other relevant studies, utilizing the most commonly used metrics: precision, recall, and time-to-maneuver (TTM). Time-to-maneuver is defined as the time interval between the time of the model's prediction with the greatest confidence and the actual start of the maneuver. It is observed that our model's precision and recall are on par with the average levels.

Table 2. The summary of the performance of related works on driver intention prediction.

Paper	Data Source	Method	Precision	Recall	TTM (s)
Jain et al. [3]	in-cabin and external	AIO-HMM	77.4%	71.2%	3.53
Jain et al. [11]	in-cabin and external	RNN-LSTM	84.5%	77.1%	3.58
Tonutti et al. [8]	in-cabin and external	DA-RNN	92.3%	90.8%	3.98
Zhou et al. [7]	in-cabin and external	LSTM	91.7%	90.7%	3.30
Rekabdar et al. [41]	in-cabin and external	Dilated CNN	91.8%	92.5%	3.76
Ours	in-cabin and external	CNN + Trans	90.8%	91.1%	0

Table 3 presents the comparative results of the Brain4Cars test set. We compared STA-NET with three widely used end-to-end studies [4,13] because they have demonstrated better performance in driver intent recognition tasks than traditional machine learning

methods. In [4,13], researchers applied video action recognition methods to driver intent prediction, such as 3DResNet and ConvLSTM + 3DResNet. All models were pretrained using the Kinetics-400 dataset [36], and all video data with a duration of 5 s were used as input. The table shows results for three different inputs, including only in-cabin driver maneuver videos, only external traffic scene videos, and both in-cabin and external videos. Average accuracy and F1 scores based on five-fold cross-validation are used to illustrate the performance of different methods, where "SD" denotes standard deviation.

Table 3. The comparison of different algorithms with different data inputs.

Algorithms	Data Source	Param. (M)	Accuracy ± SD (%)	F1 ± SD (%)
3DResNet [4]	in-cabin only	240.26	83.10 ± 2.5	81.7 ± 2.6
	external only	240.26	53.20 ± 0.5	43.4 ± 0.9
	in-cabin and external	480.52	75.50 ± 2.4	73.2 ± 2.2
ConvLSTM + 3DResNet [13]	in-cabin only	46.22	77.40 ± 0.02	75.49 ± 0.02
	external only	160.41	60.87 ± 0.01	66.38 ± 0.03
	in-cabin and external	212.92	83.98 ± 0.01	84.3 ± 0.01
STA-Net (Ours)	in-cabin only	**60.56**	**87.31 ± 0.56**	**85.32 ± 1.71**
	external only	**60.56**	**71.12 ± 0.58**	**73.10 ± 0.63**
	in-cabin and external	**137.25**	**90.97 ± 0.72**	**89.37 ± 1.56**

The bold indicates the final result of this work.

Based on Table 3, when exclusively using in-cabin driver maneuver data, all algorithms achieved satisfactory results, with accuracy ranging from 77 to 84%, making further improvements challenging. STA-NET achieved the highest accuracy of 90.97% and an F1 score of 89.37% when provided with dual perspectives—both inside and outside the vehicle. The experiment indicates that the accuracy of intent recognition increased by approximately 3.66% when the STA-NET model used both in-cabin driver maneuver and external traffic scene as inputs. This result strongly suggests that external traffic scene features and in-cabin driver maneuver features contain complementary information for driver intent recognition. Moreover, our model significantly reduces the number of parameters compared to previous methods. Given the potential computational costs associated with model complexity, models with low resource requirements are preferred for automotive applications. Our model achieves the extraction of valuable features with fewer parameters, facilitating easier deployment in resource-constrained environments such as onboard vehicle systems.

The confusion matrix for the proposed method is illustrated in Figure 5, showcasing the classification performance for the five intents. The results indicate that lane-keeping and right lane change intent recognition achieved the most accurate results, with accuracy reaching 91.3% and 90.9%, respectively. The ability to recognize the intent to change lanes to the left is relatively lower, with an accuracy of approximately 70.8%, often confused with the intent to go straight. The accuracy for recognizing the intent to turn left is around 83.3%, and it is also prone to confusion with the intent to go straight. The accuracy for recognizing the intent to turn right is approximately 87.5%, with potential confusion with the intent to go straight and change lanes to the right.

Through an analysis of misclassified samples, three main reasons are proposed. Firstly, for some lane-keeping maneuvers, drivers may be more inclined to perform left-check actions to ensure safe driving, making it easy to infer them as left lane change or left turn. Secondly, some right lane change intents are similar to maneuvers during lane-keeping, suggesting that drivers might occasionally perform right lane changes while maintaining their lane, although infrequently, potentially confusing the model. Thirdly, some right turn intents are very similar to maneuvers during right lane changes, which can confuse the model. These inferences suggest that in more complex traffic scenarios, drivers' maneuvers may adjust frequently based on the traffic conditions on both sides, posing a challenge for driver intent recognition.

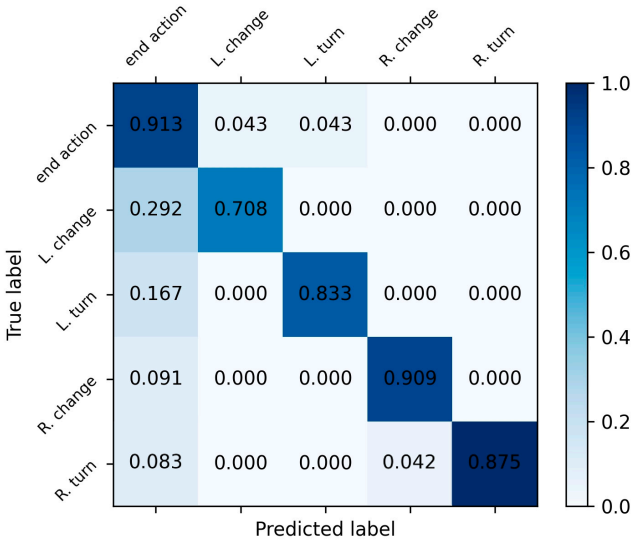

Figure 5. The confusion matrices for five driver intentions are based on the proposed models. The abscissa and the ordinate represent the probability of predicted and true labels.

Upon comparing our STA-NET's performance with the method described in [13], we acknowledge the variation in classification accuracy across different maneuvers. Specifically, the model in [13] exhibits superior performance in identifying left lane changes, left turns, and right turns, whereas STA-NET struggles with right lane change predictions. This discrepancy raises a pertinent discussion on the feasibility and potential advantages of employing a multi-method approach for driver maneuver prediction.

Rationale for Multi-Method Approach:

The diversity in driving behaviors and the complexity of road scenarios necessitate a nuanced approach to maneuver recognition. A single model may not optimally capture the intricacies associated with various maneuvers due to differences in visual cues, driver intentions, and environmental contexts. Therefore, leveraging the strengths of different predictive models based on the predicted maneuver could enhance overall performance and reliability.

Methodological Considerations:

To explore this possibility, we propose a framework where the predictive model dynamically selects between STA-NET and alternative methods, like the one presented in [13], based on the specific maneuver scenario. This selection could be informed by pre-defined criteria, such as the maneuver type, confidence levels of the predictions, or contextual factors like traffic density and road type.

Potential Benefits:

(1) Enhanced Accuracy: By aligning model selection with the maneuver's characteristics, we anticipate improvements in prediction accuracy, particularly for maneuvers where STA-NET's performance is currently lacking.
(2) Reduced False Positives/Negatives: A more tailored approach allows for finer discrimination between maneuvers, potentially reducing misclassifications and enhancing the system's reliability.
(3) Adaptability: This strategy introduces a layer of adaptability, enabling the system to evolve and incorporate new methods or findings from ongoing research.

5. Conclusions

In this study, we proposed an end-to-end driver intent inference framework named STA-NET based on joint spatiotemporal analysis. Firstly, a dual-branch network with spatiotemporal joint extraction modules was employed to extract features related to in-cabin driver maneuvers and external traffic scenes separately. Subsequently, the features were aggregated using fusion at different scales. Finally, the aggregated spatiotemporal features

were utilized to obtain the most probable driver maneuver intent. The model was validated on Brain4Cars, a natural dataset containing highway and urban road driving information. Results demonstrate that our model achieved favorable performance compared to other action recognition algorithms and classical methods, with an overall accuracy of 90.97% for driver intent recognition. In conclusion, our work contributes to ongoing efforts to enhance traffic safety and paves the way for further advancements in driver intent prediction and personalized driving assistance systems.

Author Contributions: Conceptualization, B.H.; methodology, B.H.; validation, Z.W. and X.C.; investigation, Z.W. and X.C.; writing—original draft preparation, B.H.; writing—review and editing, B.H.; supervision, N.Y.; project administration, N.Y.; funding acquisition, N.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under grant 62271388.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The Brain4Cars dataset presented in this study is openly available in [3]. The kinetics-400 dataset presented in this study is openly available in [18,31].

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Wu, Y.; Zhang, L.; Lou, R.; Li, X. Recognition of Lane Changing Maneuvers for Vehicle Driving Safety. *Electronics* **2023**, *12*, 1456. [CrossRef]
2. David, R.; Rothe, S.; Söffker, D. State Machine Approach for Lane Changing Driving Behavior Recognition. *Automation* **2020**, *1*, 68–79. [CrossRef]
3. Jain, A.; Koppula, H.S.; Raghavan, B.; Soh, S.; Saxena, A. Car that knows before you do: Anticipating maneuvers via learning temporal driving models. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 3182–3190.
4. Gebert, P.; Roitberg, A.; Haurilet, M.; Stiefelhagen, R. End-to-end Prediction of Driver Intention using 3D Convolutional Neural Networks. In Proceedings of the IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; pp. 969–974.
5. Xing, Y.; Lv, C.; Wang, H.; Cao, D.; Velenis, B. An ensemble deep learning approach for driver lane change intention inference. *Transp. Res. Part C Emerg. Technol.* **2020**, *115*, 102615. [CrossRef]
6. Jain, A.; Soh, S.; Raghavan, B.; Singh, A.; Koppula, H.S.; Saxena, A. Brain4Cars: Sensory-Fusion Recurrent Neural Models for Driver Activity Anticipation. Available online: http://brain4cars.com/pdfs/baylearn.pdf (accessed on 29 November 2023).
7. Zhou, D.; Ma, H.; Dong, Y. Driving maneuvers prediction based on cognition-driven and data-driven method. In Proceedings of the 2018 IEEE Visual Communications and Image Processing (VCIP), Taichung, Taiwan, 9–12 December 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–4.
8. Tonutti, M.; Ruffaldi, E.; Cattaneo, A.; Avizzano, C.A. Robust and subject-independent driving manoeuvre anticipation through Domain-Adversarial Recurrent Neural Networks. *Robot. Auton. Syst.* **2019**, *115*, 162–173. [CrossRef]
9. Braunagel, C.; Kasneci, E.; Stolzmann, W.; Rosenstiel, W. Driver-activity recognition in the context of conditionally autonomous driving. In Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems, Gran Canaria, Spain, 15–18 September 2015; pp. 1652–1657.
10. Braunagel, C.; Geisler, D.; Rosenstiel, W.; Kasneci, E. Online recognition of driver-activity based on visual scanpath classification. *IEEE Intell. Transp. Syst. Mag.* **2017**, *9*, 23–36. [CrossRef]
11. Jain, A.; Singh, A.; Koppula, H.S.; Soh, S.; Saxena, A. Recurrent neural networks for driver activity anticipation via sensory-fusion architecture. In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 16–21 May 2016; pp. 3118–3125.
12. Moussaid, A.; Berrada, I.; El Kamili, M.; Fardousse, K. Predicting driver lane change maneuvers using driver's face. In Proceedings of the International Conference on Wireless Networks and Mobile Communications, (WINCOM), Fez, Morocco, 29 October–1 November 2019; pp. 1–7.
13. Rong, Y.; Akata, Z.; Kasneci, E. Driver intention anticipation based on in-cabin and driving scene monitoring. In Proceedings of the IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–8.
14. Xing, Y.; Tian, B.; Lv, C.; Cao, D. A Two-Stage Learning Framework for Driver Lane Change Intention Inference. *IFAC PapersOnLine* **2020**, *53*, 638–643. [CrossRef]

15. Chen, H.; Chen, H.; Liu, H.; Feng, X. Spatiotemporal Feature Enhancement Aids the Driving Intention Inference of Intelligent Vehicles. *Int. J. Environ. Res. Public Health* **2022**, *19*, 11819. [CrossRef] [PubMed]
16. Ma, Y.; Ye, W.; Cao, X.; Abdelraouf, A.; Han, K.; Gupta, R.; Wang, Z. CEMFormer: Learning to Predict Driver Intentions from In-Cabin and External Cameras via Spatial-Temporal Transformers. *arXiv* **2023**, arXiv:2305.07840.
17. Bonyani, M.; Rahmanian, M.; Jahangard, S.; Rezaei, M. DIPNet: Driver intention prediction for a safe takeover transition in autonomous vehicles. *IET Intell. Transp. Syst.* **2023**, *17*, 1769–1783. [CrossRef]
18. Zhang, H.; Guo, D.; Guo, Y.; Wu, F.; Gao, S. A Novel Method for the Driver Lane-Changing Intention Recognition. *IEEE Sens. J.* **2023**, *23*, 20437–20451. [CrossRef]
19. Li, R.; Shu, X.; Li, C. Driving Behavior Prediction Based on Combined Neural Network Model. *IEEE Trans. Comput. Soc. Syst.* **2024**. [CrossRef]
20. Chen, S.; Yao, H.; Qiao, F.; Ma, Y.; Wu, Y.; Lu, J. Vehicles driving behavior recognition based on transfer learning. *Expert Syst. Appl.* **2023**, *213*, 119254. [CrossRef]
21. Sun, L.; Jia, K.; Chen, K.; Yeung, D.Y.; Shi, B.E.; Savarese, S. Lattice Long Short-Term Memory for Human Action Recognition. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2147–2156.
22. Li, D.; Qiu, Z.; Dai, Q.; Yao, T.; Mei, T. Recurrent Tubelet Proposal and Recognition Networks for Action Detection. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 303–318.
23. Carreira, J.; Zisserman, A. Quo vadis, action recognition? A new model and the kinetics dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6299–6308.
24. Feichtenhofer, C.; Fan, H.; Malik, J.; He, K. SlowFast Networks for Video Recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6202–6211.
25. Xu, M.; Xiong, Y.; Chen, H.; Li, X.; Xia, W.; Tu, Z.; Soatto, S. Long Short-Term Transformer for Online Action Detection. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 1086–1099.
26. Li, Y.; Wu, C.Y.; Fan, H.; Mangalam, K.; Xiong, B.; Malik, J.; Feichtenhofer, C. Feichtenhofer, MViTv2: Improved Multiscale Vision Transformers for Classification and Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 4804–4814.
27. Bertasius, G.; Wang, H.; Torresani, L. Is Space-Time Attention All You Need for Video Understanding? In Proceedings of the 38th International Conference on Machine Learning, PMLR, Virtual, 18–24 July 2021; Volume 139, pp. 813–824.
28. Arnab, A.; Dehghani, M.; Heigold, G.; Sun, C.; Lučić, M.; Schmid, C. Vivit: A video vision transformer. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021.
29. Ba, J.L.; Kiros, J.R.; Hinton, G.E. Layer normalization. *arXiv* **2016**, arXiv:1607.06450.
30. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 11.
31. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16 × 16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
32. Hendrycks, D.; Gimpel, K. Gaussian error linear units (GELUs). *arXiv* **2016**, arXiv:1606.08415.
33. Jain, A.; Koppula, H.S.; Soh, S.; Raghavan, B.; Singh, A.; Saxena, A. Brain4Cars: Car That Knows Before You Do via Sensory-Fusion Deep Learning Architecture. *arXiv* **2016**, arXiv:1601.00740.
34. Wu, Z.; Liang, K.; Liu, D.; Zhao, Z. Driver Lane Change Intention Recognition Based on Attention Enhanced Residual-MBi-LSTM Network. *IEEE Access* **2022**, *10*, 58050–58061. [CrossRef]
35. Yu, B.; Bao, S.; Zhang, Y.; Sullivan, J.; Flannagan, M. Measurement and prediction of driver trust in automated vehicle technologies: An application of hand position transition probability matrix. *Transp. Res. Part C Emerg. Technol.* **2021**, *124*, 102957. [CrossRef]
36. Kay, W.; Carreira, J.; Simonyan, K.; Zhang, B.; Hillier, C.; Vijayanarasimhan, S.; Viola, F.; Green, T.; Back, T.; Natsev, P.; et al. The kinetics human action video dataset. *arXiv* **2017**, arXiv:1705.06950.
37. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. In Proceedings of the 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, 6–9 May 2019.
38. Loshchilov, I.; Hutter, F. SGDR: Stochastic Gradient Descent with Warm Restarts. In Proceedings of the 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, 24–26 April 2017.
39. De Vries, T.; Taylor, G.W. Improved regularization of convolutional neural networks with cutout. *arXiv* **2017**, arXiv:1708.04552.
40. Hendrycks, D.; Mu, N.; Cubuk, E.D.; Zoph, B.; Gilmer, J.; Lakshminarayanan, B. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv* **2019**, arXiv:1912.02781.
41. Rekabdar, B.; Mousas, C. Dilated convolutional neural network for predicting driver's activity. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 3245–3250.