2021 IEEE International Conference on Robotics and Automation (ICRA 2021)
May 31 - June 4, 2021, Xi'an, China

Learning Interpretable End-to-End Vision-Based Motion Planning for Autonomous Driving with Optical Flow Distillation

Hengli Wang, Peide Cai, Yuxiang Sun, Lujia Wang, and Ming Liu, *Senior Member, IEEE*

2021 IEEE International Conference on Robotics and Automation (ICRA) | 978-1-7281-9077-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICRA48506.2021.9561334

Abstract— **Recently, deep-learning based approaches have achieved impressive performance for autonomous driving. However, end-to-end vision-based methods typically have limited interpretability, making the behaviors of the deep networks difficult to explain. Hence, their potential applications could be limited in practice. To address this problem, we propose an interpretable end-to-end vision-based motion planning approach for autonomous driving, referred to as IVMP. Given a set of past surrounding-view images, our IVMP first predicts future egocentric semantic maps in bird's-eye-view space, which are then employed to plan trajectories for self-driving vehicles. The predicted future semantic maps not only provide useful interpretable information, but also allow our motion planning module to handle objects with low probability, thus improving the safety of autonomous driving. Moreover, we also develop an optical flow distillation paradigm, which can effectively enhance the network while still maintaining its real-time performance. Extensive experiments on the nuScenes dataset and closed-loop simulation show that our IVMP significantly outperforms the state-of-the-art approaches in imitating human drivers with a much higher success rate. Our project page is available at `https://sites.google.com/view/ivmp`.**

I. Introduction

Motion planning is an important capability in autonomous driving, serving as a fundamental building block [1]. With the impressive advancement of deep learning technologies, many researchers have tried to develop end-to-end motion planning approaches using deep learning, which generally employ deep neural networks (DNNs) to directly map the raw sensor data (*e.g.*, point clouds and images) to planned trajectories [2]–[5] or control commands (*e.g.*, throttle and steering angle) [6]–[8]. However, end-to-end approaches are often criticized for their lack of interpretability. Here, interpretability refers to the ability to explain why the model can produce specific results [9]. Interpretability is very important for autonomous driving, since it can help people find out the limitations of the network and further improve it, especially when accidents such as collisions happen.

To improve the interpretability of end-to-end approaches, some researchers have adopted multi-task learning and developed models that can generate several interpretable representations, *e.g.*, the object detection and prediction results [10] or the egocentric semantic maps in bird's-eye-view (BEV) space [9]. These approaches are generally based on LiDARs, since motion planning is usually performed in BEV space and the point clouds provided by LiDARs inherently meet this requirement. Unfortunately, images are located in perspective-view space, and there is a large gap between perspective-view space and BEV space. To mitigate this gap, some approaches have first transformed perspective images into BEV semantic maps, which are then employed to perform motion planning [11]. However, the neglect of future environment prediction still restricts the interpretability of end-to-end vision-based approaches. Since images can provide more semantic information than point clouds, there is a strong motivation to improve the interpretability of end-to-end vision-based approaches.

In this paper, we propose an Interpretable end-to-end Vision-based Motion Planning approach, referred to as IVMP, for autonomous driving. Our IVMP, illustrated in Fig. 1, takes as input a set of past surrounding-view images. We first lift these images into three dimensions (3-D) and employ a recurrent unit to predict a set of future egocentric semantic maps in BEV space. Afterwards, our motion planning module can employ these semantic maps to plan trajectories for the self-driving vehicle (SDV). Moreover, we develop an optical flow distillation paradigm to further improve the driving performance. Specifically, we refer to the above-mentioned network as the student network (IVMP-S) and additionally propose a teacher network (IVMP-T), which adopts a similar architecture to the student network but further takes optical flow information as input. The explicit motion information provided by the optical flow can significantly improve the teacher network, but the computation and corresponding feature processing of the optical flow also seriously slows down the network [12]–[14]. We then use knowledge distillation techniques to effectively enhance the student network based on the teacher network, while still maintaining the real-time performance of the student network. To verify the effectiveness and efficiency of our approach, we perform evaluations on the popular nuScenes dataset [15]. In addition, we conduct closed-loop evaluations in the Carla simulation environment [16]. The experimental results demonstrate that our approach can imitate human trajectories more closely than existing approaches with a much higher success rate. Furthermore, our student network runs

This work was supported in part by the National Natural Science Foundation of China under grant U1713211, in part by the Collaborative Research Fund by Research Grants Council Hong Kong under Project C4063-18G, and in part by the HKUST-SJTU Joint Research Collaboration Fund under project SJTU20EG03. *(Corresponding author: Ming Liu.)*

Hengli Wang, Peide Cai and Ming Liu are with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong SAR, China (email: hwangdf@connect.ust.hk; pcaiaa@connect.ust.hk; eelium@ust.hk).

Yuxiang Sun is with the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (e-mail: yx.sun@polyu.edu.hk, sun.yuxiang@outlook.com).

Lujia Wang is with Cloud Computing Lab of Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China (email: lj.wang1@siat.ac.cn).

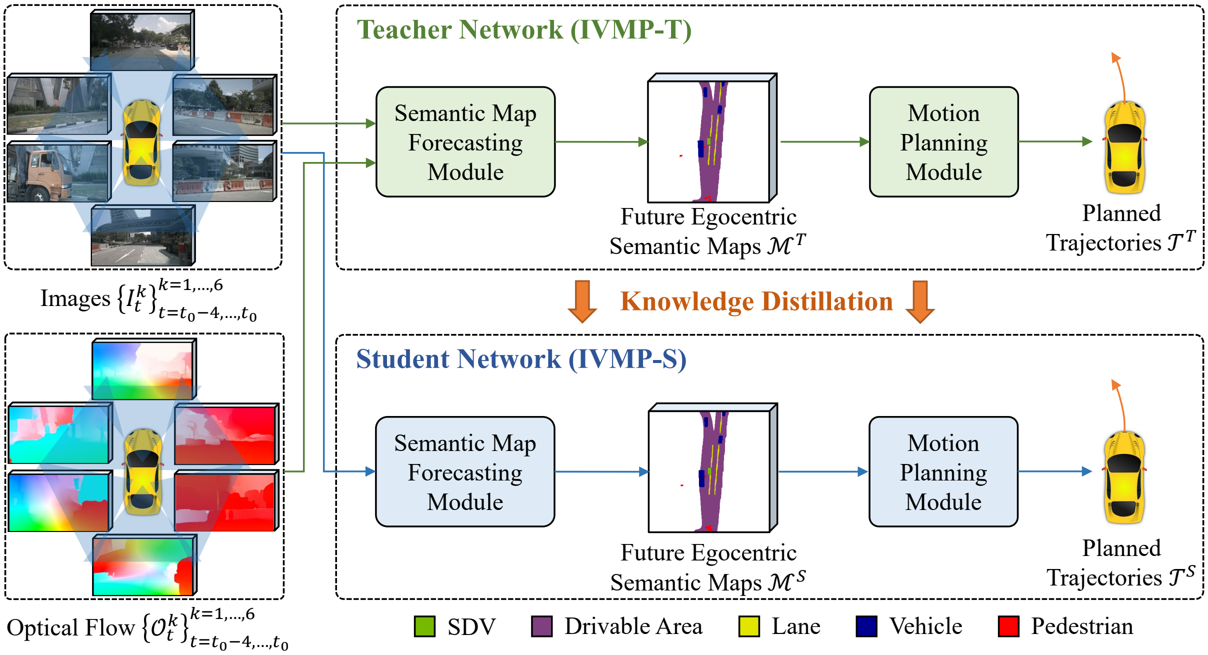
Teacher Network (IVMP-T)
Semantic Map Forecasting Module
Future Egocentric Semantic Maps $\mathcal{M}^T$
Motion Planning Module
Planned Trajectories $\mathcal{T}^T$
Images $\{I_t^k\}_{t=t_0-4,\ldots,t_0}^{k=1,\ldots,6}$
Knowledge Distillation
Student Network (IVMP-S)
Semantic Map Forecasting Module
Future Egocentric Semantic Maps $\mathcal{M}^S$
Motion Planning Module
Planned Trajectories $\mathcal{T}^S$
Optical Flow $\{O_t^k\}_{t=t_0-4,\ldots,t_0}^{k=1,\ldots,6}$
🟩 SDV
🟪 Drivable Area
🟨 Lane
🟦 Vehicle
🟥 Pedestrian

Fig. 1: An overview of our IVMP, which consists of 1) a semantic map forecasting module to predict future egocentric semantic maps in BEV space and 2) a motion planning module to generate trajectories for SDVs. In the proposed optical flow distillation paradigm, the teacher network adopts a similar architecture to the student network but further takes optical flow information as input. We then use knowledge distillation techniques to effectively enhance the student network based on the teacher network, while still maintaining the real-time performance of the student network.

much faster than the teacher network with similar driving performance thanks to the adopted optical flow distillation paradigm. The major contributions of this paper can be summarized as follows:

- We propose IVMP, an interpretable end-to-end vision-based motion planner for autonomous driving.
- We develop an optical flow distillation paradigm, which can effectively enhance the network while still maintaining its real-time performance.
- We present extensive experiments on the nuScenes dataset and closed-loop simulation that demonstrate the effectiveness and efficiency of our IVMP.

II. Related Work

A. End-to-end Approaches for Autonomous Driving

Traditional autonomous driving approaches generally perform motion planning based on the perception results [17]–[22], while end-to-end approaches directly map the raw sensor data to the planned trajectories or control commands. ALVINN was the first approach in this field, employing a 3-layer neural network to directly output control commands [23]. Recently, with the success of deep learning, end-to-end approaches have advanced with deeper network architectures and more complex sensor inputs [2]–[8]. However, these end-to-end approaches generally behave as black-box models and have limited interpretability as previously mentioned.

To improve the interpretability of end-to-end approaches, some researchers have adopted multi-task learning and developed LiDAR-based models that can generate several intermediate representations [9], [10], [24]. Specifically, Sadat *et al.* [9] employed LiDAR data to predict egocentric semantic maps in BEV space, which are then used for motion planning. Other researchers have followed this paradigm and attempted to improve the interpretability of end-to-end vision-based approaches. For example, Gupta *et al.* [11] first used a multi-layer perceptron (MLP) to transform perspective images into BEV semantic maps, which are then employed to perform motion planning. However, their approach only utilizes a monocular camera with a limited field of view (FOV) and does not predict the future environment. In contrast, our approach takes a set of past surrounding-view images as input and predicts the future semantic maps in BEV space, which improves both interpretability and driving performance.

B. Semantic Segmentation in BEV Space

Many studies have used perspective images to perform semantic segmentation in BEV space [25]–[29]. Specifically, Pan *et al.* [27] employed an MLP to conduct such transformation for surrounding-view images. Roddick *et al.* [28] and Philion *et al.* [29] incorporated the strong geometric priors of camera extrinsic parameters into the pipeline, which presents impressive performance. Moreover, Deng *et al.* [30] and Yang *et al.* [31] used fisheye images to generate semantic predictions in BEV space. The semantic map forecasting module in our approach is inspired by these works. The difference is that our approach has the capacity of predicting future semantic maps in BEV space.

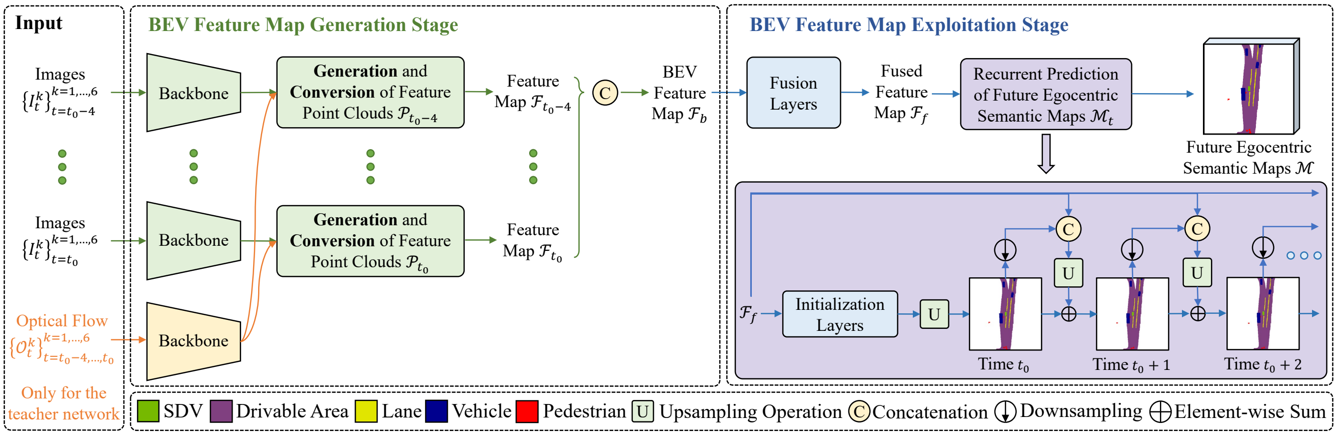
Input
Images $\{I_t^k\}_{t=t_0-4}^{k=1,\dots,6}$
Images $\{I_t^k\}_{t=t_0}^{k=1,\dots,6}$
Optical Flow $\{O_t^k\}_{t=t_0-4,\dots,t_0}^{k=1,\dots,6}$
Only for the teacher network
BEV Feature Map Generation Stage
Backbone
Backbone
Backbone
Generation and Conversion of Feature Point Clouds $\mathcal{P}_{t_0-4}$
Generation and Conversion of Feature Point Clouds $\mathcal{P}_{t_0}$
Feature Map $\mathcal{F}_{t_0-4}$
Feature Map $\mathcal{F}_{t_0}$
C
BEV Feature Map $\mathcal{F}_b$
BEV Feature Map Exploitation Stage
Fusion Layers
Fused Feature Map $\mathcal{F}_f$
Recurrent Prediction of Future Egocentric Semantic Maps $\mathcal{M}_t$
Future Egocentric Semantic Maps $\mathcal{M}$
$\mathcal{F}_f$
Initialization Layers
U
C
U
Time t_0
Time $t_0 + 1$
Time $t_0 + 2$
🟩 SDV 🟪 Drivable Area 🟨 Lane 🟦 Vehicle 🟥 Pedestrian U Upsampling Operation C Concatenation ⊕ Downsampling ⊕ Element-wise Sum

Fig. 2: The pipeline of our semantic map forecasting module, which consists of a BEV feature map generation stage and a BEV feature map exploitation stage. In the first stage, we lift each image into 3-D to generate a feature map in BEV space, which is then processed by a recurrent unit to predict future egocentric semantic maps in the second stage. Please note that we omit the visualization of the softmax operation in the recurrent unit (purple color) for brevity.

C. Knowledge Distillation

Hinton *et al.* [32] first proposed the concept of knowledge distillation, which aims at leveraging the dark knowledge of a teacher network to train a student network with fewer parameters. Since then many techniques, such as Hint Training (HT) [33] and attention distillation [34], have been developed to improve knowledge distillation, and it has been employed in many applications, *e.g.*, semantic segmentation [35] and object detection [36]. However, knowledge distillation for motion planning has not previously been explored, which is one of the major contributions of this paper.

III. Methodology

In this section, we first introduce our semantic map forecasting module and motion planning module in Section III-A and III-B, respectively. Then, we present our optical flow distillation paradigm in Section III-C. Finally, Section III-D elaborates the training phase.

A. Semantic Map Forecasting Module

Let $I_t^k \in \mathbb{R}^{H\times W\times 3}$ denote the input RGB image, where $k = 1, \dots, 6$ denotes the six cameras used in our experiments; and $t = t_0 - 4, \dots, t_0$ denotes the timestamp of the past five frames. The six cameras with known extrinsic parameters $\mathcal{E}^k$ and intrinsic parameters $\mathcal{I}^k$ roughly point in the forward, forward-left, forward-right, backward, backward-left and backward-right directions respectively. Then, given all images in the past five frames $\{I_t^k\}_{t=t_0-4,\dots,t_0}^{k=1,\dots,6}$, our semantic map forecasting module can output a set of egocentric semantic maps in the future eleven frames $\mathcal{M} = \{\mathcal{M}_t\}_{t=t_0,\dots,t_0+10}$. Fig. 2 presents the pipeline of this module, which consists of a BEV feature map generation stage and a BEV feature map exploitation stage.

1) BEV Feature Map Generation Stage: The purpose of this stage is to lift each I_t^k into 3-D to generate a feature map $\mathcal{F}_b$ in BEV space, which is the key to the prediction of $\mathcal{M}$. Inspired by [29], we achieve this by generating contextual features at all possible depths for each pixel. Specifically, we associate each pixel with a set of $|D|$ discrete depths, where $D = \{d_0 + \Delta d, \dots, d_0 + |D|\Delta d\}$. Then, based on the intrinsic parameter $\mathcal{I}^k$, we can easily generate a large point cloud $\mathcal{P}_t^k$ that contains $H\cdot W\cdot |D|$ 3-D points for each I_t^k. The contextual feature for each point in $\mathcal{P}_t^k$ is a combination of the feature for the corresponding pixel and the discrete depth inference. To be specific, our network utilizes the backbone to predict a contextual feature $\mathbf{f} \in \mathbb{R}^C$ and a distribution π over the discrete depth set D for each pixel $\mathbf{p}$. The contextual feature $\mathbf{f}_d \in \mathbb{R}^C$ for point $\mathbf{p}_d$ is then computed by

$$\mathbf{f}_d = \pi_d \cdot \mathbf{f}, \tag{1}$$

where $d \in D$ refers to any discrete depth in D.

For the teacher network, we further incorporate optical flow into $\mathcal{P}_t^k$ to enable the network to better learn the past motion for predicting future semantic maps. Specifically, given any I_t^k and I_{t-1}^k, we first utilize an off-the-shelf optical flow estimation network, PWCNet [12], to compute the backward optical flow $\mathcal{O}_t^k \in \mathbb{R}^{H\times W\times 2}$, which contains the explicit past motion information from I_{t-1}^k to I_t^k. Then, we utilize another backbone to predict a contextual feature $\mathbf{f}' \in \mathbb{R}^C$ for each pixel $\mathbf{p}$. We concatenate $\mathbf{f}'$ with $\mathbf{f}$ and then generate a new feature. For notational simplicity, we still denote this new feature as $\mathbf{f}$. We further employ (1) to compute a contextual feature $\mathbf{f}_d \in \mathbb{R}^C$ for every point $\mathbf{p}_d$ in the teacher network. Note that the following architectures are almost the same for the teacher and student networks.

Then, for each timestamp t, we can utilize the extrinsic parameters $\{\mathcal{E}^k\}^{k=1,\dots,6}$ to aggregate $\{\mathcal{P}_t^k\}^{k=1,\dots,6}$ into a large point cloud $\mathcal{P}_t$. After that, we follow [37] to convert $\mathcal{P}_t$ into "pillars", which refer to voxels with infinite height. Specifically, we assign each point to its nearest pillar and conduct pooling to construct a feature map $\mathcal{F}_t \in \mathbb{R}^{X'\times Y'\times C}$. Now $\mathcal{F}_t$ can be processed by convolutional layers to predict future egocentric semantic maps in BEV space. We then concatenate the features of all five past frames

$\{\mathcal{F}_t\}_{t=t_0-4,\ldots,t=t_0}$ to generate the BEV feature map $\mathcal{F}_b$.

2) BEV Feature Map Exploitation Stage: Given $\mathcal{F}_b$, which consists of all past information in BEV space, we will generate $\mathcal{M}$ in this stage. Note that $\mathcal{M}_t \in \mathbb{R}^{X\times Y\times|\mathcal{C}|}$, where $\mathcal{C}$ denotes the semantic classes, which include the drivable area, lane, vehicle and pedestrian in our experiments.

We first utilize several fusion layers to aggregate the spatio-temporal information of $\mathcal{F}_b$ and generate a fused feature map $\mathcal{F}_f$. The adopted fusion layers include two parallel convolutional layers with different dilation rates. Afterwards, we update the future egocentric semantic logits $\mathcal{S}_t \in \mathbb{R}^{X\times Y\times|\mathcal{C}|}$ repeatedly via a recurrent unit:

$$\mathcal{S}_t(c) = \mathcal{S}_{t-1}(c) + \mathbf{U}\left(\mathbf{C}(\mathcal{F}_f, \mathcal{S}_{t-1}(c)\downarrow)\right), \tag{2}$$

where $c \in \mathcal{C}$ denotes any semantic class; $\downarrow$ denotes $\frac{1}{2}\times$ downsampling; $\mathbf{C}(\cdot,\cdot)$ denotes concatenation; and $\mathbf{U}(\cdot)$ denotes a 2× upsampling operation, consisting of a 2× bilinear interpolation followed by convolutional layers. Please note that $\mathcal{S}_{t_0}$ is predicted from $\mathcal{F}_f$ via initialization layers, which consist of convolutional layers and the upsampling operation $\mathbf{U}(\cdot)$. Then, we perform softmax on $\mathcal{S}_t$ to generate the future egocentric semantic map (predicted distribution) $\mathcal{M}_t$. We further define a future semantic map loss $\mathcal{L}_M$:

$$\begin{aligned}\mathcal{L}_M(\widehat{\mathcal{M}}, \mathcal{M}) &= \sum_t H\left(\widehat{\mathcal{M}}_t, \mathcal{M}_t\right) \\ &= -\sum_t\sum_c\sum_{i,j}\widehat{\mathcal{M}}_t(i,j,c)\cdot\log(\mathcal{M}_t(i,j,c)),\end{aligned} \tag{3}$$

where $H(\cdot,\cdot)$ denotes the cross entropy; and $\widehat{\mathcal{M}}$ denotes the ground truth distribution.

B. Motion Planning Module

Based on $\mathcal{M}$, the current SDV state $\mathbf{s}_{t_0}$ and a given high-level route planned by a global planner, the purpose of our motion planning module is to generate a planned trajectory that contains the SDV states in the future ten frames, *i.e.*, $\mathcal{T} = \{\mathbf{s}_t\}_{t=t_0+1,\ldots,t_0+10}$. In our experiments, we adopt $\mathbf{s}_t = [x_t, y_t, \theta_t, \kappa_t, v_t, a_t]$, where x and y denote the position coordinates; and θ, κ, v and a denote the heading angle, curvature, velocity and acceleration, respectively.

To achieve motion planning, we first employ the sampling technique proposed in [38] to sample a diverse set of trajectories for the SDV based on $\mathbf{s}_{t_0}$, and then select the one with the minimal cost of a learned cost function f as follows:

$$\mathcal{T}^* = \arg\min_{\mathcal{T}} f(\mathbf{s}_{t_0}, \mathcal{M}, \mathcal{T}; \mathbf{w}), \tag{4}$$

where $\mathbf{w}$ denotes the learnable parameters of our motion planning module. f consists of two subcosts: 1) f_m, which focuses on the safety of the planned trajectory based on $\mathcal{M}$; and 2) f_o, which focuses on the comfort and the consistency between the high-level route and the planned trajectory.

The intuition for f_m is that the SDV should not collide with other objects and also should not drive on non-drivable areas. Thus, we define f_m as follows:

$$f_m = \sum_t\sum_c w_c \cdot \mathcal{M}_t(\mathcal{T}_t, c), \tag{5}$$

where $w_c \in \mathbf{w}$; $c \in \mathcal{C}'$ and $\mathcal{C}'$ includes the vehicle, pedestrian and non-drivable area (as opposed to the drivable area) classes in $\mathcal{M}$; and $\mathcal{M}_t(\mathcal{T}_t, c)$ denotes the corresponding probability for class c on $\mathcal{M}_t$ based on the position provided by the sampled trajectory $\mathcal{T}_t$. The advantage of f_m is that it employs the probability instead of the binary classification result, and thus can handle objects with low probability and further improve the safety of autonomous driving.

As for f_o, we define it as a linear combination of several cost terms. Specifically, to ensure the consistency between the high-level route and the planned trajectory, we adopt the distance between the end position of the trajectory and the given high-level route as a cost term. We also penalize the number of times the SDV changes lanes to encourage maneuvers that are consistent with the high-level route. Moreover, to encourage comfortable driving, we define several thresholds to penalize aggressive behaviors.

During training, considering that selecting the trajectory with the minimal cost in a discrete set is not differential, we follow [9] and develop a motion planning loss $\mathcal{L}_P$:

$$\mathcal{L}_P(\widehat{\mathcal{T}}, \mathcal{T}) = \max_{\mathcal{T}}\left[f(\widehat{\mathcal{T}}) - f(\mathcal{T}) + \sum_t ||\widehat{\mathcal{T}}_t - \mathcal{T}_t||_1\right]_+, \tag{6}$$

where $||\cdot||_1$ denotes the $L1$-Norm; $[\cdot]_+$ denotes the ReLU function; and $\widehat{\mathcal{T}}$ denotes the trajectory of human drivers. Please note that we omit $\mathbf{s}_{t_0}$, $\mathcal{M}$ and $\mathbf{w}$ in f for brevity. $\mathcal{L}_P$ adopts a similar formulation to the max-margin loss, which can encourage the trajectories of human drivers to have a smaller cost f than other trajectories. Moreover, $\mathcal{L}_P$ can also penalize trajectories that have a small cost but are different from the trajectories of human drivers.

C. Optical Flow Distillation Paradigm

Our teacher network and student network have been introduced in the above two subsections. The explicit motion information provided by the optical flow can significantly improve the teacher network, but the computation and corresponding feature processing of the optical flow also seriously slows down the network. In contrast, the student network can conduct motion planning in real time with a poorer performance than the teacher network. To further improve the driving performance of the student network, we develop an optical flow distillation paradigm, which distills the knowledge from a trained teacher network to the student network via knowledge distillation techniques. The distillation loss $\mathcal{L}_D$ consists of three terms, as follows:

$$\mathcal{L}_D = \lambda_{DM}\mathcal{L}_{DM} + \lambda_{DP}\mathcal{L}_{DP} + \lambda_{DF}\mathcal{L}_{DF}, \tag{7}$$

where $\mathcal{L}_{DM}$, $\mathcal{L}_{DP}$ and $\mathcal{L}_{DF}$ denote the distillation loss for $\mathcal{M}$, $\mathcal{T}$ and $\mathcal{F}_b$, respectively; and λ_{DM}, λ_{DP} and λ_{DF} are hyperparameters that scale the three loss terms.

Specifically, since the prediction of future semantic maps is a classification task, $\mathcal{L}_{DM}$ is designed based on the conventional knowledge distillation technique [32], as follows:

$$\mathcal{L}_{DM} = \mathcal{L}_M(\mathcal{M}^T, \mathcal{M}^S) = \sum_t H\left(\mathcal{M}_t^T, \mathcal{M}_t^S\right), \tag{8}$$

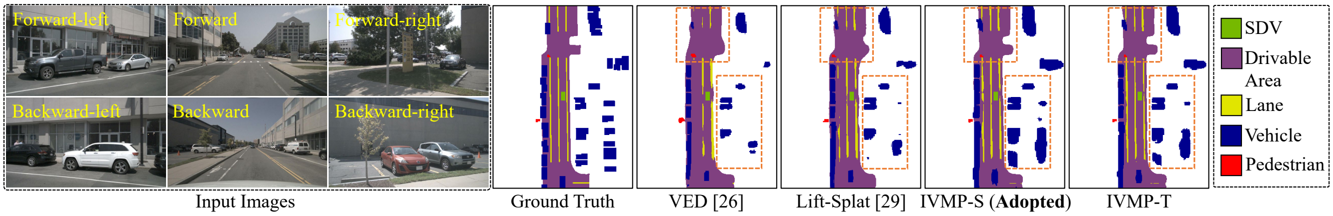

Fig. 3: An example of the BEV semantic maps on the nuScenes dataset [15]. Please note that "SDV" does not belong to the semantic classes $\mathcal{C}$ and is only used for visualization. Significantly improved regions are marked with orange dashed boxes.

where $\mathcal{M}^T$ and $\mathcal{M}^S$ denote the predicted future semantic maps of the teacher and student network, respectively. Different from $\widehat{\mathcal{M}}$ in (3) that can only provide hard information, $\mathcal{M}^T$ can provide useful soft information. For example, for a pixel that belongs to the drivable area class, $\widehat{\mathcal{M}}$ can only show that it belongs to the drivable area class and does not belong to any other classes; while $\mathcal{M}^T$ can further show that it may belong to the lane class, but it is almost impossible for it to belong to the vehicle or the pedestrian classes. The soft information provided by $\mathcal{M}^T$ can effectively improve the student network.

In addition, inspired by [36], we design $\mathcal{L}_{DP}$ as follows for motion planning:

$$\mathcal{L}_{DP} = \begin{cases} \mathcal{L}_P(\mathcal{T}^{T*}, \mathcal{T}^S), \\ \quad \text{if} \sum_t ||\widehat{\mathcal{T}}_t - \mathcal{T}_t^{S*}||_1 > \sum_t ||\widehat{\mathcal{T}}_t - \mathcal{T}_t^{T*}||_1, \\ 0, \text{ otherwise.} \end{cases}$$

where $\mathcal{T}^S$ denotes a set of sampled trajectories of the student network; $\mathcal{T}^{T*}$ and $\mathcal{T}^{S*}$ denote the trajectories of the teacher and student network with the minimal cost of f, respectively; $\widehat{\mathcal{T}}$ denotes the trajectory of human drivers; and $\mathcal{L}_P(\cdot)$ is shown in (6). $\mathcal{L}_{DP}$ encourages the student to be close to or better than the teacher, but does not push the student once it reaches the teacher's performance.

Since $\mathcal{F}_b$ of the teacher contains the explicit motion information provided by the optical flow while $\mathcal{F}_b$ of the student does not, we further develop $\mathcal{L}_{DF}$ based on HT [33]:

$$\mathcal{L}_{DF} = ||\mathcal{F}_b^T - \mathcal{F}_b^S||_1, \tag{9}$$

where $\mathcal{F}_b^T$ and $\mathcal{F}_b^S$ denote the BEV feature map $\mathcal{F}_b$ of the teacher and student network, respectively. $\mathcal{L}_{DF}$ encourages the student network to mimic $\mathcal{F}_b$ of the teacher network.

D. Training Phase

In the training phase, we first utilize the following teacher training loss $\mathcal{L}^T$ to train the teacher network:

$$\mathcal{L}^T = \lambda_M \mathcal{L}_M^T + \lambda_P \mathcal{L}_P^T, \tag{10}$$

where $\mathcal{L}_M^T = \mathcal{L}_M(\widehat{\mathcal{M}}, \mathcal{M}^T)$; $\mathcal{L}_P^T = \mathcal{L}_P(\widehat{\mathcal{T}}, \mathcal{T}^T)$; and λ_M and λ_P are hyperparameters that scale the loss terms.

Afterwards, we use the following student training loss $\mathcal{L}^S$ to train the student network based on the trained teacher network:

$$\mathcal{L}^S = \lambda_M \mathcal{L}_M^S + \lambda_P \mathcal{L}_P^S + \lambda_D \mathcal{L}_D, \tag{11}$$

where $\mathcal{L}_M^S = \mathcal{L}_M(\widehat{\mathcal{M}}, \mathcal{M}^S)$; $\mathcal{L}_P^T = \mathcal{L}_P(\widehat{\mathcal{T}}, \mathcal{T}^S)$; and λ_M, λ_P and λ_D are hyperparameters that scale the loss terms.

TABLE I: IoU (%) results of BEV semantic maps on the nuScenes dataset [15], where "D", "L", "V", "P" and "M" denote the drivable area, lane, vehicle, pedestrian and mean value, respectively. Best results are bolded.

Approach	D	L	V	P	M
VED [26]	60.82	16.74	23.28	11.93	28.19
VPN [27]	65.97	17.05	28.17	10.26	30.36
PON [28]	63.05	17.19	27.91	13.93	30.52
Lift-Splat [29]	72.23	19.98	31.22	15.02	34.61
IVMP-S-ND	71.76	18.27	33.12	16.15	34.83
IVMP-S (**Adopted**)	74.70	20.94	34.03	**17.38**	36.76
IVMP-T	**75.82**	**21.22**	**34.58**	17.29	**37.23**

IV. Experimental Results and Discussions

A. Datasets and Implementation Details

In our experiments, we first use the nuScenes dataset [15] to evaluate the performance of our approach for BEV semantic map prediction and motion planning. We split the dataset into a training, a validation and a testing set that consist of 18072, 8019 and 8033 samples, respectively. Networks are first trained on the training set, then selected on the validation set and finally evaluated on the testing set. We also conduct closed-loop evaluation in the Carla simulation environment [16]. Specifically, we first construct a large-scale driving dataset in different scenes, weather and illumination conditions. The dataset is then split into a training set with 200K samples and a validation set with 50K samples. Moreover, the closed-loop evaluation is performed in six scenes, including two unseen scenes. Each network is evaluated thoroughly with 1800 episodes (around 1000 km).

In the implementation, we adopt EfficientNet-B0 [39] as the backbone. The time interval between two consecutive frames is $0.5s$, which means that our IVMP takes the information of the past $2s$ as input and generates planned trajectories for the future $5s$. We use the Adam optimizer [40] with an initial learning rate of 10^{-4} to train our IVMP-T and IVMP-S on two NVIDIA GeForce RTX 2080 Ti GPUs. Moreover, we also train the student network without the proposed optical flow distillation paradigm, which is referred to as IVMP-S-ND, for better performance comparison.

B. BEV Semantic Map Results on the nuScenes Dataset

We adopt the intersection over union (IoU) as the evaluation metric, and the evaluation results of $\mathcal{M}_{t_0}$ are presented in Table I. We can see that the three variants of

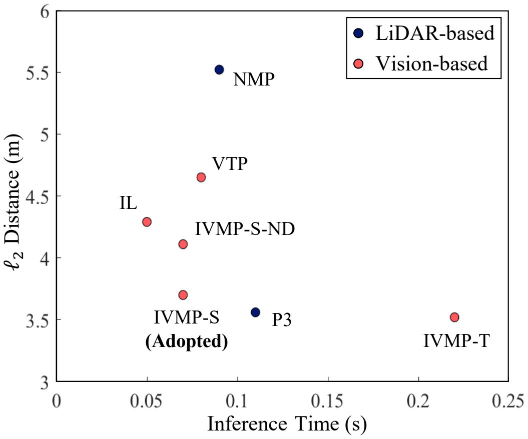

Fig. 4: Motion planning results of IL, VTP [3], P3 [9], NMP [24] and our IVMP on the nuScenes dataset [15].

our IVMP all outperform the state-of-the-art approaches, which demonstrates the effectiveness of our architecture that utilizes the past information. Moreover, IVMP-T achieves the best performance due to the explicit motion information provided by the optical flow. Furthermore, IVMP-S presents a much better performance than IVMP-S-ND and a similar performance to IVMP-T, which verifies the effectiveness of our optical flow distillation paradigm. The qualitative results in Fig. 3 also confirm the above-mentioned conclusions. Please note that we adopt IVMP-S in practice due to its real-time performance. The analysis of inference time is presented in Section IV-C.

C. Motion Planning Results on the nuScenes Dataset

Following [9], we use the ℓ_2 distance between the planned trajectory and human trajectory at $t = 5s$ for performance comparison. In addition, we also record the inference time of each approach. Fig. 4 presents the evaluation results, where IL refers to an imitation learning baseline that predicts trajectories directly from $\mathcal{M}$. Please note that IL is also an end-to-end approach. From Fig. 4, we can clearly observe that the conclusions in Section IV-B also hold for motion planning. Our IVMP-T achieves the most accurate performance due to the explicit motion information provided by the optical flow. Moreover, our IVMP-S can run in real time with a similar performance to IVMP-T thanks to the adopted optical flow distillation paradigm. In addition, one exciting fact is that our IVMP-S and IVMP-T can achieve competitive performance when compared to existing LiDAR-based approaches, which strongly demonstrates the effectiveness of our IVMP architecture with interpretable representations.

D. Closed-loop Evaluation Results in the Carla Simulator

Following [41], we adopt the success rate (SR) and right lane rate (RL) for evaluation. RL is defined as the proportion of the period in the given high-level route to the total driving time. We utilize a PID controller to transform the planned trajectories of our IVMP-S into control commands, and denote it as IVMP. We then compare the online performance of IVMP with Intention-Net [7] and CIL [8], as presented in Table II. We can observe that our IVMP achieves the best results in terms of both SR and RL. We analyze that the predicted semantic maps allow our motion planning module to handle objects with low probability, thus improving the safety of autonomous driving. Fig. 5 presents a driving scenario at intersections. Our IVMP can manuever the SDV to pass through the intersection safely and efficiently.

TABLE II: Closed-loop evaluation results in the Carla simulator [16]. Best results are bolded.

	Intention-Net [7]	CIL [8]	IVMP (**Ours**)
SR (%)	75.28	60.72	**88.67**
RL (%)	89.28	82.97	**93.16**

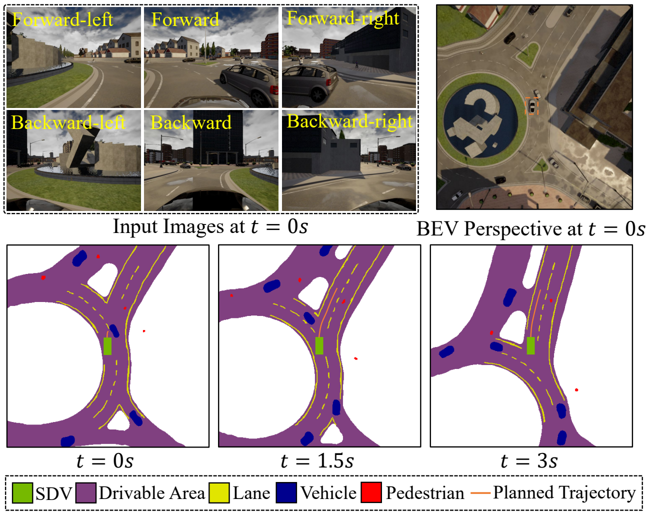

Fig. 5: An example of the closed-loop evaluation in the Carla simulator [16]. The SDV is marked with an orange dashed box in the BEV perspective.

V. Conclusions

In this paper, we proposed IVMP, an interpretable end-to-end vision-based motion planning approach for autonomous driving. Our IVMP first employs a semantic map forecasting module to predict future egocentric semantic maps in BEV space, which are then processed by a motion planning module to generate trajectories for SDVs. The predicted semantic maps not only provide useful interpretable information, but also allow our motion planning module to handle objects with low probability, thus improving the safety of autonomous driving. Moreover, we also develop an optical flow distillation paradigm, which can effectively enhance the network while still maintaining its real-time performance. Extensive experiments on the nuScenes dataset and closed-loop simulation have demonstrated the superiority of our IVMP over state-of-the-art approaches in BEV semantic map segmentation and imitating human drivers. We believe that our optical flow distillation paradigm can also be employed in other tasks related to spatio-temporal information analysis for performance improvement.

REFERENCES

[1] T. Liu, Q. Liao *et al.*, "The role of the Hercules autonomous vehicle during the Covid-19 pandemic: An autonomous logistic vehicle for contactless goods transportation," *IEEE Robotics and Automation Magazine*, 2021.

[2] M. Bansal, A. Krizhevsky, and A. Ogale, "Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst," *arXiv preprint arXiv:1812.03079*, 2018.

[3] P. Cai, Y. Sun, Y. Chen, and M. Liu, "Vision-based trajectory planning via imitation learning for autonomous vehicles," in *IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 2736–2742.

[4] H. Wang, Y. Sun, R. Fan, and M. Liu, "S2P2: Self-supervised goal-directed path planning using RGB-D data for robotic wheelchairs," in *IEEE International Conference on Robotics and Automation*, 2021.

[5] P. Cai, Y. Sun, H. Wang, and M. Liu, "VTGNet: A vision-based trajectory generation network for autonomous vehicles in urban environments," *IEEE Transactions on Intelligent Vehicles*, 2020.

[6] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang *et al.*, "End to end learning for self-driving cars," *arXiv preprint arXiv:1604.07316*, 2016.

[7] W. Gao, D. Hsu, W. S. Lee, S. Shen, and K. Subramanian, "Intention-net: Integrating planning and deep learning for goal-directed autonomous navigation," in *Conference on Robot Learning (CoRL)*, 2017.

[8] F. Codevilla, M. Miiller, A. López, V. Koltun, and A. Dosovitskiy, "End-to-end driving via conditional imitation learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–9.

[9] A. Sadat, S. Casas, M. Ren, X. Wu, P. Dhawan, and R. Urtasun, "Perceive, predict, and plan: Safe motion planning through interpretable semantic representations," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[10] W. Zeng, S. Wang, R. Liao, Y. Chen, B. Yang, and R. Urtasun, "DSDNet: Deep structured self-driving network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[11] S. Gupta, J. Davidson, S. Levine, R. Sukthankar, and J. Malik, "Cognitive mapping and planning for visual navigation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2616–2625.

[12] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 8934–8943.

[13] H. Wang, Y. Liu, H. Huang, Y. Pan, W. Yu, J. Jiang, D. Lyu, M. J. Bocus, M. Liu, I. Pitas *et al.*, "ATG-PVD: Ticketing parking violations on a drone," in *European Conference on Computer Vision*. Springer, 2020, pp. 541–557.

[14] H. Wang, R. Fan, and M. Liu, "CoT-AMFlow: Adaptive modulation network with co-teaching strategy for unsupervised optical flow estimation," *arXiv preprint arXiv:2011.02156*, 2020.

[15] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 621–11 631.

[16] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on Robot Learning (CoRL)*, 2017.

[17] H. Wang, Y. Sun, and M. Liu, "Self-supervised drivable area and road anomaly segmentation using rgb-d data for robotic wheelchairs," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4386–4393, 2019.

[18] H. Wang, R. Fan, Y. Sun, and M. Liu, "Applying surface normal information in drivable area and road anomaly detection for ground mobile robots," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[19] R. Fan, H. Wang, P. Cai, and M. Liu, "SNE-RoadSeg: Incorporating surface normal information into semantic segmentation for accurate freespace detection," in *European Conference on Computer Vision*. Springer, 2020, pp. 340–356.

[20] H. Wang, R. Fan, P. Cai, and M. Liu, "PVStereo: Pyramid voting module for end-to-end self-supervised stereo matching," *IEEE Robotics and Automation Letters*, 2021.

[21] R. Fan, H. Wang, P. Cai, J. Wu, M. J. Bocus, L. Qiao, and M. Liu, "Learning collision-free space detection from stereo images: Homography matrix brings better data augmentation," *IEEE Transactions on Mechatronics*, 2021.

[22] H. Wang, R. Fan, Y. Sun, and M. Liu, "Dynamic fusion module evolves drivable area and road anomaly detection: A benchmark and algorithms," *IEEE Transactions on Cybernetics*, 2021.

[23] D. A. Pomerleau, "Alvinn: An autonomous land vehicle in a neural network," in *Advances in Neural Information Processing Systems (NIPS)*, 1989, pp. 305–313.

[24] W. Zeng, W. Luo, S. Suo, A. Sadat, B. Yang, S. Casas, and R. Urtasun, "End-to-end interpretable neural motion planner," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8660–8669.

[25] L. Reiher, B. Lampe, and L. Eckstein, "A sim2real deep learning approach for the transformation of images from multiple vehicle-mounted cameras to a semantically segmented image in bird's eye view," in *2020 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2020.

[26] C. Lu, M. J. G. van de Molengraft, and G. Dubbelman, "Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 445–452, 2019.

[27] B. Pan, J. Sun, H. Y. T. Leung, A. Andonian, and B. Zhou, "Cross-view semantic segmentation for sensing surroundings," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4867–4873, 2020.

[28] T. Roddick and R. Cipolla, "Predicting semantic map representations from images using pyramid occupancy networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 11 138–11 147.

[29] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[30] L. Deng, M. Yang, H. Li, T. Li, B. Hu, and C. Wang, "Restricted deformable convolution-based road scene semantic segmentation using surround view cameras," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4350–4362, 2019.

[31] K. Yang, X. Hu, Y. Fang, K. Wang, and R. Stiefelhagen, "Omnisupervised omnidirectional semantic segmentation," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[32] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *Advances in Neural Information Processing Systems Workshop (NIPSW)*, 2014.

[33] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," in *International Conference on Learning Representations (ICLR)*, 2015.

[34] S. Zagoruyko and N. Komodakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," in *International Conference on Learning Representations (ICLR)*, 2017.

[35] T. He, C. Shen, Z. Tian, D. Gong, C. Sun, and Y. Yan, "Knowledge adaptation for efficient semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 578–587.

[36] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," in *Advances in Neural Information Processing Systems (NIPS)*, 2017, pp. 742–751.

[37] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 697–12 705.

[38] A. Sadat, M. Ren, A. Pokrovsky, Y.-C. Lin, E. Yumer, and R. Urtasun, "Jointly learnable behavior and trajectory planning for self-driving vehicles," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[39] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning (ICML)*, 2019.

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[41] P. Cai, S. Wang, Y. Sun, and M. Liu, "Probabilistic end-to-end vehicle navigation in complex dynamic environments with multimodal sensor fusion," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4218–4224, 2020.