Video Semantic Segmentation With Distortion-Aware Feature Correction

Jiafan Zhuang, Zilei Wang, *Member, IEEE*, and Bingke Wang

***Abstract*— Video semantic segmentation aims to generate an accurate semantic map for each frame in a video. For such a task, conducting per-frame image segmentation is generally unacceptable in practice due to high computation cost. To address this issue, many works perform the flow-based feature propagation to reuse the features of previous frames, which essentially exploits the content continuity of consecutive frames. However, the estimated optical flow would inevitably suffer inaccuracy and then make the propagated features distorted. In this article, we propose a distortion-aware feature correction method with the goal of improving video segmentation performance at a low price. Our core idea is to correct the features on distorted regions using the current frame while reserving the propagated features for other regions. In this way, a lightweight network is enough for achieving promising segmentation results. In particular, we propose to predict the distorted regions by utilizing the consistency of distortion patterns in images and features, such that the high-cost feature extraction from current frames can be avoided. We conduct extensive experiments on Cityscapes, CamVid, and UAVid, and the results show that our proposed method significantly outperforms previous methods and achieves the state-of-the-art performance on both segmentation accuracy and speed. Code and pretrained models are available at https://github.com/jfzhuang/DAVSS.**

***Index Terms*— Video semantic segmentation, feature propagation, distortion prediction, feature correction.**

I. Introduction

SEMANTIC segmentation is to assign each pixel in the scene a semantic class, which is currently an active research topic in computer vision. In recent years, image semantic segmentation has achieved unprecedented accuracy benefited from the great progress of deep convolutional neural networks (DCNN) [1] and various datasets (*e.g.*, Cityscapes [2], CamVid [3], and UAVid [4]). However, many real-world applications have strong demands for fast and accurate video semantic segmentation, *e.g.*, robotics [5], autonomous driving [6], and video surveillance [7]. Compared to images, videos involve much larger volume of data, and thus always require more efficient segmentation algorithms.

A naive approach for video segmentation is to directly apply the image segmentation model in a per-frame way. But such deployment is generally unacceptable in practice due to too heavy computational burden. Actually, the consecutive frames of a video are often similar in a large portion of content, and it is unnecessary to reprocess every pixel of the video frame using an image segmentation model [9]. Then an intuitive idea for video semantic segmentation is to reuse the features extracted from the previous frames when segmenting the current frame [8]. Naturally, the feature propagation is proposed to reduce the overall computational complexity.

In recent years, some feature propagation based methods have been proposed for video semantic segmentation, *e.g.*, DFF [8], NetWarp [10], DVSNet [9], and Accel [11]. These methods first compute the optical flow between the key frame and the current frame, and then produce the features of the current frame by propagating the features of the key frame under guidance of optical flow. Here the bilinear interpolation is usually used as the feature warping operator. The CNN-based optical flow estimation methods (*e.g.*, FlowNet [12], [13], FlowNet2.0 [13]) are preferred since they are easy to be embedded into the video segmentation framework for end-to-end training. Evidently, the accuracy of optical flow estimation would determine the quality of propagated features and performance of semantic segmentation.

Despite the great progress in the past decades, accurate optical flow estimation remains a challenging problem [14]. In particular, the occlusion caused by scene motion makes the optical flow estimation ill-posed since no visual correspondence exists for the occluded pixels [15]. When the inaccurate optical flow is used in feature propagation, the produced features would get distorted and incorrect segmentation results may be further generated. In addition, for small or slender areas of a single class (*e.g.*, pedestrian, pole), a slight offset of predicted optical flow would cause sensible distortion, which is especially serious for long-distance propagation. We show the typical distortion phenomenon in Fig. 1. The distortion of feature propagation needs to be carefully tackled in video semantic segmentation.

Some existing methods can alleviate feature distortion by modulating the propagated features. For example, DFF [8] attaches a scale field to optical flow estimation and adjusts the propagated features via element-wise multiplication. Accel [11] proposes to extract features from the current frame with a lightweight model and then fuse the extracted

Manuscript received July 16, 2020; revised September 23, 2020 and October 20, 2020; accepted November 4, 2020. Date of publication November 10, 2020; date of current version August 4, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61673362 and Grant 61836008; in part by the Youth Innovation Promotion Association CAS under Grant 2017496; and in part by the Fundamental Research Funds for the Central Universities. This article was recommended by Associate Editor Q. Wang. *(Corresponding author: Zilei Wang.)*

The authors are with the National Engineering Laboratory for Brain-inspired Intelligence Technology and Application (NEL-BITA), University of Science and Technology of China, Hefei 230027, China (e-mail: jfzhuang@mail.ustc.edu.cn; zlwang@ustc.edu.cn; wbkup@mail.ustc.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSVT.2020.3037234.

Digital Object Identifier 10.1109/TCSVT.2020.3037234

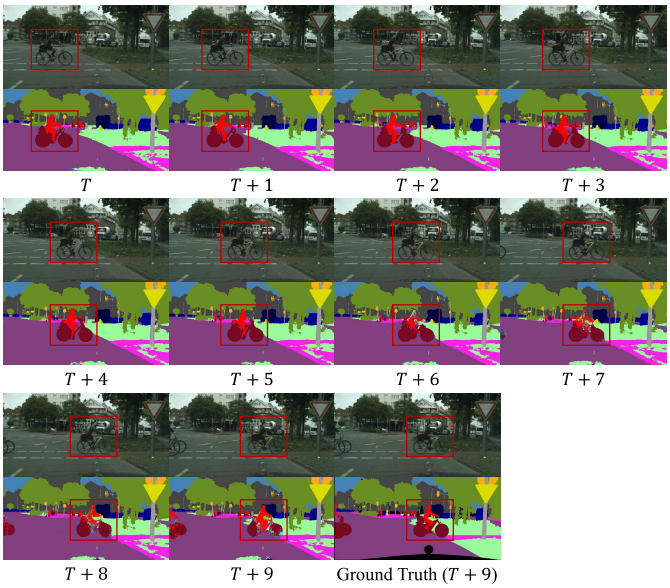

Fig. 1. **Illustration of distortion phenomenon in feature propagation.** The segmentation results of an example video produced by DFF [8] are demonstrated, where $T+i$ denotes the i-th frame from the key frame T. In particular, we also give the ground-truth segmentation of the frame $T+9$ for comparison. Red rectangles highlight the distorted regions caused by inaccurate optic flow estimation. Best viewed in color and zoom in.

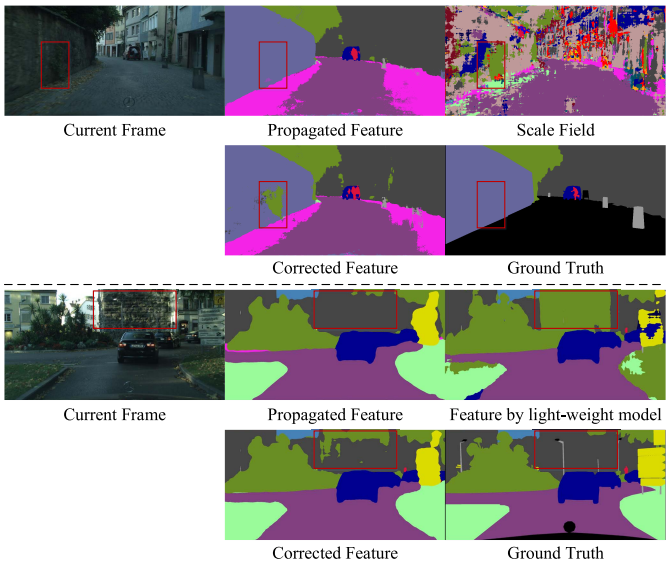

Fig. 2. **Visualization of false correction.** The propagated and corrected features are visualized and represented by their segmentation results. The upper case is from DFF and the blow one is from Accel50. Red rectangles highlight the areas corrected wrongly. Best viewed in color.

and propagated features to perform semantic segmentation. However, these works equally treat every pixel of the current frame without distinguishing the quality of propagated features among different pixels. Consequently, the regions where the features are correctly propagated may be modulated to be wrong, namely, false correction may occur. We show typical cases of such a phenomenon in Fig. 2. Besides, we check the ratio of the number of pixels wrongly and rightly rectified, and the statistics of different methods on *Cityscapes val subset* are shown in Fig. 3. Evidently, many

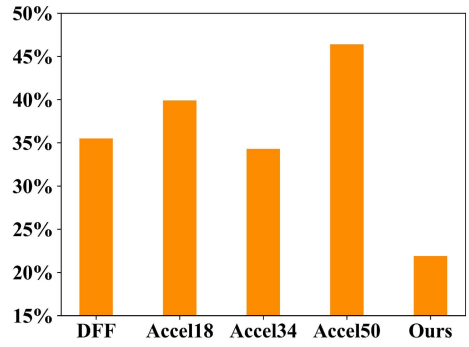

Fig. 3. **Statistics of false correction on Cityscapes val subset.** Here the ratio of the number of pixels wrongly and rightly rectified is particularly calculated for different methods. It can be seen that some correct propagated features would be wrongly rectified, and our proposed method can achieve the best result.

correct features are wrongly rectified, even for models equipped with a heavy network (*e.g.*, Accel50).

In this work, we propose a novel distortion-aware feature correction method to rectify the propagated features, aiming at improving the accuracy of video semantic segmentation at a low price. Our key idea is to correct the features on the distorted regions while reserving the propagated features for other regions. With such a design, a lightweight network can be enough to perform the rectification. To this end, we need first to identify the distorted regions. An intuitive approach is to extract features from the current frame using an image segmentation model, and then get the misalignment regions by comparing the extracted and propagated features. However, extracting the features would involve too high computation cost. To tackle the issue, we propose to get the distorted regions through image comparison of video frames. Actually, the distortion is mainly caused by inaccurate optical flow, *i.e.*, the distorted regions are essentially the regions where the optical flow is miscalculated. So we propose to concurrently propagate video frames with the same optical flow as in feature propagation, and then compare the propagated frame and current frame in the image space to get the distorted regions. Our proposed method essentially utilizes the consistency of the distortion patterns for images and features, as shown in Fig. 4. Following this idea, we propose a very lightweight network to predict the distortion maps.

Then we propose a feature correction module (FCM) to perform distortion correction on the propagated features. Here the predicted distortion maps are utilized in two folds. First, we propose a CFNet to extract the correction cues from the current frame, and enforce it to focus on the distorted regions by applying the distortion map to the calculation of training loss. CFNet can be designed very lightweight since only the capacity to process the distorted regions is required. Second, FCM uses the distortion maps to identify the important regions on which the propagated features need to be rectified greatly by correction cues. Consequently, the correction cues from the current frame dominate the distorted regions while the propagated features dominate other regions. Finally, we conduct semantic segmentation on the corrected features.

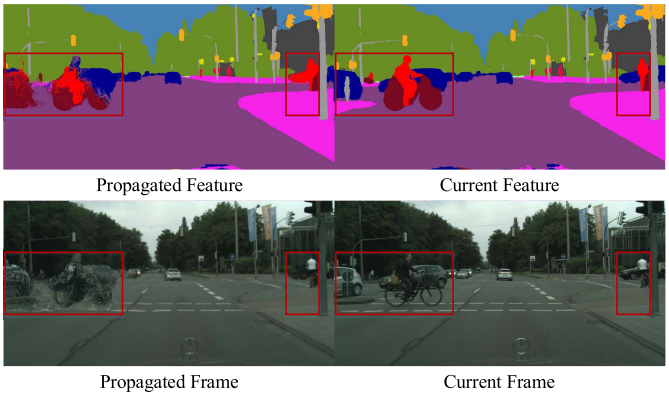

Fig. 4. **Illustration of distortion consistency for images and features.** We provide the segmentation results of the propagated and extracted features in the first row, and the propagated image from previous frame and current frame in the second row. Red rectangles highlight the main distorted regions. It can be seen that similar distortion patterns present for images and features. Best viewed in color and zoom in.

The contributions of this work are summarized as

- We propose a novel distortion-aware feature correction method for video semantic segmentation, which can effectively boost the segmentation performance at a low price by focusing on the distorted regions.
- We propose a lightweight network to predict the distorted regions of propagated features, which works in the image space and can effectively guide feature correction.
- We experimentally evaluate the effectiveness of our proposed method, and the results on Cityscapes, CamVid, and UAVid demonstrate the superiority of our method to previous state-of-the-art methods, especially for long-distance feature propagation.

The rest of this article is organized as follows. We review the related works on image and video semantic segmentation, optical flow estimation, and attention mechanism in Section II. Section III provides the details of our approach, and Section IV experimentally evaluates the proposed method. Finally, we conclude the work in Section V.

II. Related Work

A. Image Semantic Segmentation

Benefited from the rapid development of DCNN [16]–[20], more and more semantic segmentation networks spring up. Specifically, the fully convolutional network (FCN) [1] firstly uses the convolutional layers to replace fully-connected layers, and better performance is achieved. Inspired by FCN, many extensions [21]–[23] have been proposed, which together advance image semantic segmentation. The dilated layers [24], [25] are also introduced to replace the pooling layers, which can better balance the computational cost and size of receptive fields. In addition, [24], [26], [27] propose to use the conditional random field (CRF) to refine the results of image segmentation. Recently, spatial pyramid pooling [28] and atrous spatial pyramid pooling (ASPP) [24], [29] are respectively used in PSPNet [21] and DeepLab [24] to capture multi-scale contextual information. MPF [30] uses a new structural context descriptor and a self-weighted multiview clustering method for robust group detection. Priori s-CNNs [31] learns priori location information at superpixel level and adopts a soft restricted MRF energy function to reduce over smoothness. CCNet [32] contains a criss-cross attention module to harvest the contextual information. HRNet [33] maintains the high resolution feature in the whole process and fuses multi-resolution features repeatedly for reliable and discriminative representations. SANet [34] applies the pixel-group attention to capture spatial-channel inter-dependencies. Li *et al*. [35] propose to generate data-dependent routes for adapting to the scale distribution of each image. Lin *et al*. [36] propose to use skeleton representation to effectively bridge the synthesis and real domains and achieve comparable performance on multi-person part segmentation without any human-annotated labels. Ca-crfs Net [37] introduces cascaded CRFs into the decoder to learn boundary information and enhance the ability of object boundary location. The great progress of image semantic segmentation offers foundation for video semantic segmentation.

B. Optical Flow Estimation

Optical flow is a representative pattern describing the apparent motion of objects in the video. Optical flow estimation is a fundamental task in the video analysis domain. Classical variational approaches formulate optical flow estimation as an energy minimization problem [39], [40]. Such methods are effective for small motion, but tend to fail when displacements are relatively larger. Recent works use convolutional neural networks (CNNs) to improve sparse matching by learning an effective feature embedding [12], [13], [41], [42].

Although current methods can generate satisfactory optical flow in most common cases, it is still a challenging problem to calculate accurate optical flow for occlusion areas. Most methods detect occlusion by consistency check on the estimated forward and backward optical flow [43], [44], and then extrapolating the occluded areas. But the used optical flow would be adversely affected by the occlusion. Evidently, the propagated features under the guidance of inaccurate optical flow would be severely distorted, especially for occlusion areas.

Actually, most video semantic segmentation methods prefer the current state-of-the-art CNN networks for predicting the optical flow [12], [13], [41], [45] because they are easily embedded for end-to-end training. However, these methods do not explicitly deal with occlusion, and consequently video segmentation would suffer from severe feature distortion. Thus how to deal with the feature distortion efficiently and effectively is crucial for the optical flow based video segmentation methods.

C. Attention Mechanism

Attention mechanism has been widely used in computer vision. It can effectively increases the representational power of neural networks by selectively weighting the feature maps. For example, [46] proposes a squeeze-and-excitation (SE) module to adaptively recalibrate channel-wise feature responses by explicitly modeling interdependencies

between channels. OCNet [47] and DANet [48] utilize self-attention mechanism to harvest the contextual information. Chen *et al*. [50] propose the reverse attention to guide side-output residual learning in a top-down manner. Chen *et al*. [51] propose a visual attention mechanism which can bridge high-level semantic information to help the shallow layers locate salient objects and filter out noisy response in the background region. BiANet [52] introduces a bilateral attention module to focus on the foreground region with a gradual refinement style and recover potentially useful salient information in the background region. Fan *et al*. [53] proposes a parallel reverse attention network to aggregate the features in high-level layers and mine the boundary cues using the reverse attention module. In this article, we propose to only focus on the distorted regions of propagated features under the guidance of predicted distortion maps.

D. Video Semantic Segmentation

Different from static images, videos embody useful temporal information that can be exploited. So many previous works focus on modeling cross-frame relations to integrate the information from different frames to boost the semantic segmentation accuracy. STFCN [54] utilizes a spatial-temporal LSTM over per-frame CNN features. Nilsson and Sminchisescu [55] propose to use the gated recurrent units to propagate semantic labels. Gadde *et al*. [10] propose to fuse the features warped from the key frame and those from the current frame. V2V [56] utilizes a 3D CNN to perform a voxel-level prediction. Wang *et al*. [57] propose a metadata-based global projection model with the coordinate transformation to estimate motion information between frames.

On the other hand, many works reduce the overall computation cost of video semantic segmentation by utilizing the content continuity of consecutive frames. Clockwork Net [58] updates different levels of feature maps with different frequencies. DFF [8] estimates the optical flow fields from the key frame to other frames and then propagates the high-level features using the predicted optic flows. DVSNet [9] builds a decision model to dynamically choose the key frames, which can achieve better balance between quality and efficiency. Li *et al*. [38] propose spatially variant convolution to adaptively fuse the features over time. Accel [11] proposes a reference branch to extract high-quality segmentation from the key frames and an update branch to efficiently extract low-quality segmentation from the current frames, and then fuses them to improve the segmentation accuracy. TDNet [59] distributes several sub-networks over sequential frames and then recomposes the extracted features for segmentation via an attention propagation module.

A related task to video semantic segmentation is video object segmentation. Several works [60]–[62] reduce the structural complexity of the graphical model with spatio-temporal superpixels. Chen *et al*. [63] propose a two-stage framework of integrating motion and appearance cues for foreground object segmentation in unconstrained videos. Liu *et al*. [64] propose a guided co-segmentation network to simultaneously incorporate the short-term, middle-term, and long-term temporal inter-frame relationships.

In this work, we follow the route of feature propagation for video semantic segmentation. Different from previous works that equally treat every pixel of a video frame, we propose to focus on the distorted regions when rectifying the propagated features. In this way, the semantic segmentation results can be more effectively enhanced, and the used network can be more lightweight for high efficiency.

III. Our Approach

In this work, we try to boost the accuracy of video semantic segmentation on the non-key frames effectively and efficiently under the framework of optical flow based feature propagation. To this end, we propose a distortion-aware feature correction method, and the core idea is to correct the features on the distorted regions while reserving the propagated features for other regions. For such an idea, we need to design an elegant solution to address the following issues, namely, 1) how to identify the distorted regions, 2) how to extract the correction cues from the current frame, and 3) how to effectively perform feature correction. In the following, we first introduce the framework of our proposed method. Then we elaborate on two main components of the proposed method: distortion map prediction and feature correction. Finally, we provide training details of our proposed network.

A. Framework

The framework of our proposed approach is illustrated in Fig. 5, where semantic segmentation is performed on the feature of each frame individually. To be specific, each of the video frames is treated as the key or non-key frame. For the key frames, we directly conduct image semantic segmentation to get the results using an off-the-shelf network, and the intermediate features are propagated to subsequent non-key frames. In particular, we propagate the features in a frame-by-frame way during inference. That is, the feature of the current frame is first produced by propagating that of the previous frame, in which the predicted optical flow is used as the guidance and the bilinear interpolation is usually adopted as the warping operator. Along with feature propagation, we also propagate the video frame with the same optical flow, resulting in the propagated frame. For the non-key frames, we first feed the propagated frame and current frame into our proposed distortion map network (DMNet) to predict a distortion map, which actually represents the distortion pattern of the propagated feature. Then we use the current frame to rectify the propagated feature under the guidance of the predicted distortion map. We complete such feature rectification in the proposed feature correction module (FCM). Finally, we conduct semantic segmentation on the corrected feature to get the segmentation result of the current frame.

The key components of our proposed framework are the distortion map prediction and feature correction. In our implementation, we particularly adopt DeepLabv3+ [65] as the image semantic segmentation model due to its great performance in both accuracy and efficiency. The modified FlowNet2-S [13] is used for optical flow estimation.

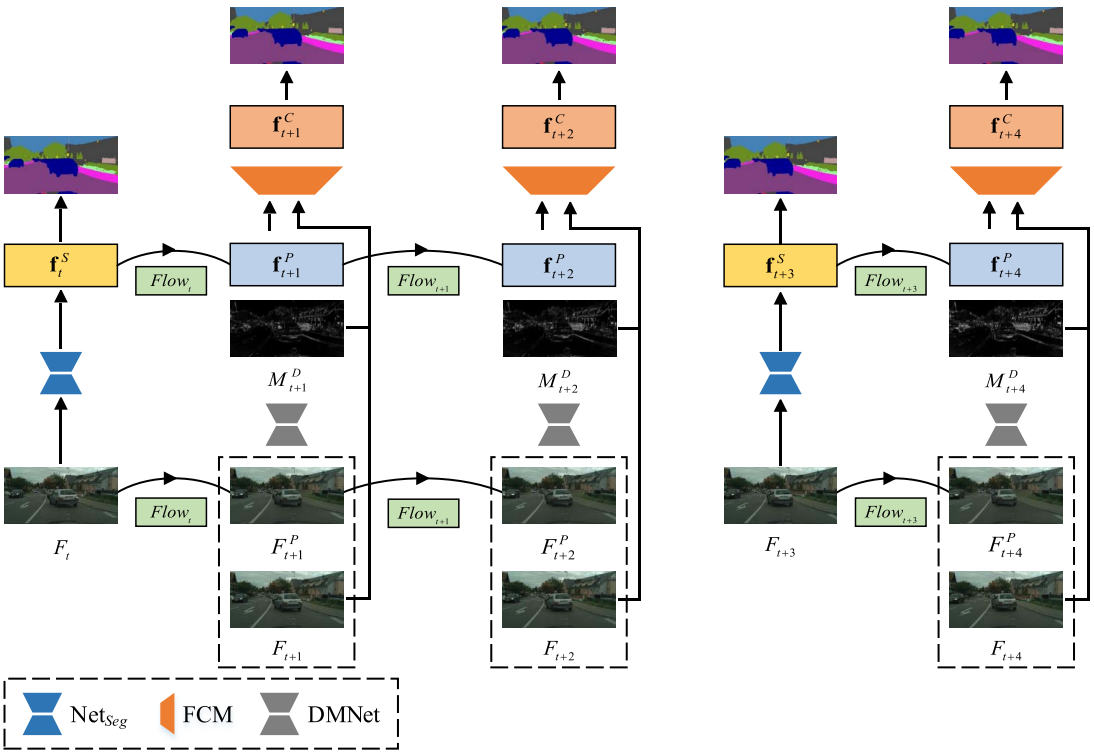

Fig. 5. **Framework of our proposed approach**. F and F^P represent the original video frame and propagated one from previous frame, respectively. Particularly, the frames F_t and F_{t+3} are selected as the key frames for illustration. In real deployment, the key frames can be selected by a fixed-interval schedule like in [8] or an adaptive schedule like in [9] and [38]. For the key frames, the feature $\mathbf{f}^S$ is extracted via an image segmentation network Net$_{seg}$. For the non-key frames, the propagated feature $\mathbf{f}^P$ is first produced through frame-by-frame propagation, and then is rectified into $\mathbf{f}^C$ by feature correction module (FCM) that combines the correction cues extracted from the current frame under the guidance of the distortion map M^D. Here M^D is predicted by a lightweight network DMNet taking as input the propagated and current frames. Best viewed in color.

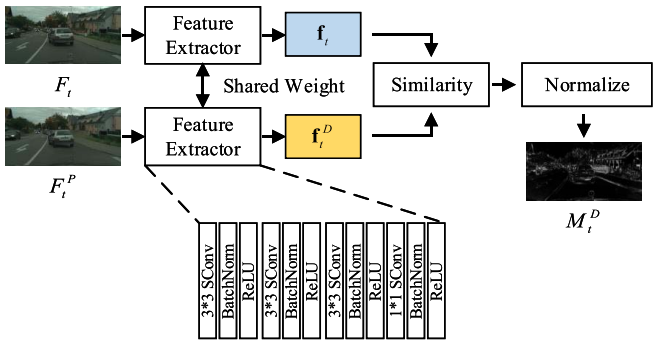

Fig. 6. **Illustration of our proposed DMNet**. Following the design of siamese networks, DMNet takes in the propagated frame and current frame to perform feature extraction and similarity computation.

B. Distortion Map Prediction

In this work, we propose a distortion map network (DMNet) to predict the distorted area of propagated features. Here we do not extract any high-level feature from the current frame since it involves too high computation cost. Instead, we compare the propagated frame and the current frame to get the distorted regions, which actually exploits the consistency of distortion pattern for images and features. To be specific, we follow the design of siamese networks to build DMNet that calculates the difference between the propagated frame and the current frame, as shown in Fig. 6. To achieve high computational efficiency, the feature extractor is designed to only comprise four separable convolutional layers interlaced with BatchNorm and ReLU layers. Consequently, the involved computation cost is nearly negligible. Then we can calculate the cosine similarity of two features, resulting a similarity map S. Formally, let $\mathbf{f}_t$ and $\mathbf{f}_t^D$ denote the features from the current frame F_t and propagated frame F_t^P, respectively. Then

$$S_t(p) = \langle \bar{\mathbf{f}}_t(p), \bar{\mathbf{f}}_t^D(p) \rangle = \bar{\mathbf{f}}_t^D(p) \bar{\mathbf{f}}_t^T(p), \tag{1}$$

where p denotes the spatial position, $\bar{\mathbf{f}} = \mathbf{f}/\|\mathbf{f}\|_2$ denotes the ℓ_2-normalized feature, and $\bar{\mathbf{f}}^T$ is the transpose of $\bar{\mathbf{f}}$. Obviously, the distorted regions would have lower value on the similarity map. To obtain the distortion map M_t^D, we normalize the similarity map as

$$M_t^D = (-S_t + 1)/2. \tag{2}$$

In our implementation, we use the supervised learning to train DMNet, as shown in Fig. 7. Here the ground truth of distortion maps is obtained by calculating the difference of segmentation results between the propagated feature and extracted feature from the current frame. To be specific, we propagate the feature of F_t to F_{t+k} to get the propagated feature $\mathbf{f}_{t+k}^P$, where F_{t+k} denotes the k-th frame from the frame F_t. Then we get the segmentation result of $\mathbf{f}_{t+k}^P$ via an argmax operation. Meanwhile, we get the segmentation result of F_{t+k} using the image segmentation model. Finally, we produce the

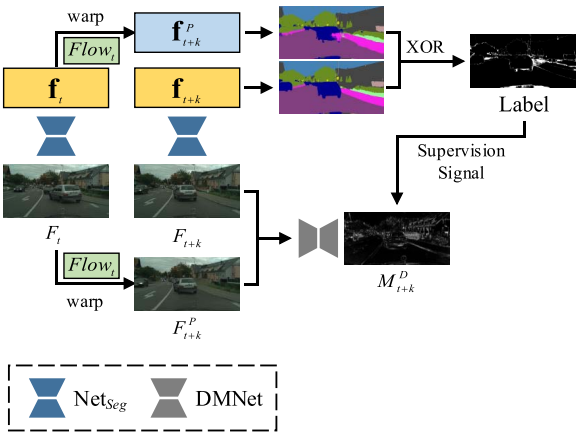

Fig. 7. **Illustration of training DMNet**. $Flow_t$ represents the predicted optical flow from F_t to F_{t+k}. The distortion map for training DMNet is obtained by calculating the difference between the segmentation results of the propagated features and current frame.

distortion map for DMNet by applying the XOR operator on two segmentation results.

C. Feature Correction Module

In this work, we explicitly rectify the propagated feature using the information of the current frame. And we propose a feature correction module (FCM) to complete such feature rectification. Specifically, we consider two key goals of video semantic segmentation: high segmentation accuracy and low computation cost. To compromise two goals, we particularly propose to utilize the predicted distortion map to guide the feature correction. In FCM, we utilize the distortion map in two folds, as shown in Fig. 8.

First, it is used to guide the extraction of correction cues from the current frame. Here it is expected that the correction cue can produce accurate segmentation results over distorted regions and meanwhile the used network is lightweight enough for efficient computation. To this end, we propose CFNet in this work that mainly consists of ten convolutional layers interlaced with batchnorm and LReLU layers for feature encoding and four deconvolutional layers interlaced with LReLU layer for feature decoding. Then we use the distortion map to weight the *CrossEntropy* loss when training CFNet, namely, distortion-guide feature learning (DGFL) is constructed. The loss function is

$$L_{DGFL} = -\frac{1}{HW} \sum_{h \in H, w \in W} M_t^D(h,w) \log p_t^{CC}(h,w), \quad (3)$$

where M_t^D is the distortion map and p_t^{CC} is the predicted probability towards the ground truth from $\mathbf{f}_t^{CC}$. Through such loss weighting, CFNet would pay more attention on the distorted regions than others, and a lightweight model is enough to effectively extract discriminative features.

Second, the distortion map is used to determine how to rectify the propagated feature. Let $\mathbf{f}_t^P$ denote the propagated feature from the key frame to the current frame F_t, $\mathbf{f}_t^{CC}$ be the extracted correction cue, and $\mathbf{f}_t^C$ be the corrected feature. FCM adopts the weighted sum to perform the feature correction,

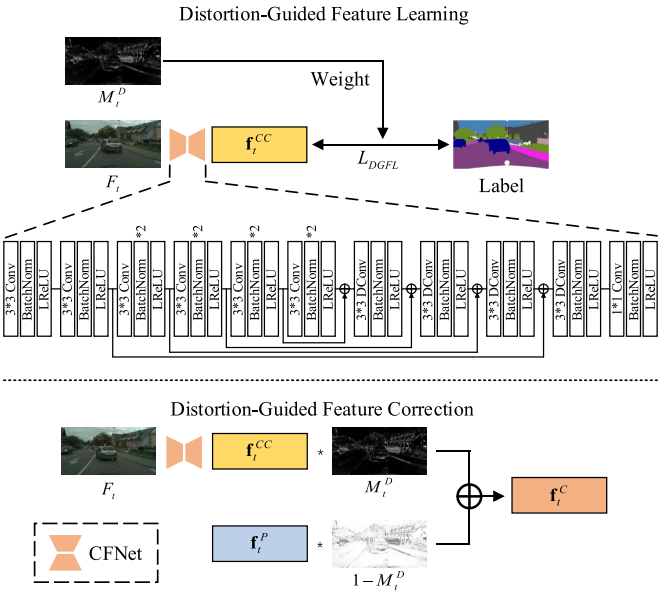

Fig. 8. **Illustration of feature correction module (FCM)**. Here the predicted distortion map M_t^D is used in two folds. First, it guides the training of the network to extract correction cues from the current frame by weighting the loss of different regions. Second, it determines how to fuse the propagated feature $\mathbf{f}_t^P$ and extracted correction cue $\mathbf{f}_t^{CC}$ to get the corrected feature $\mathbf{f}_t^C$. Best viewed in color.

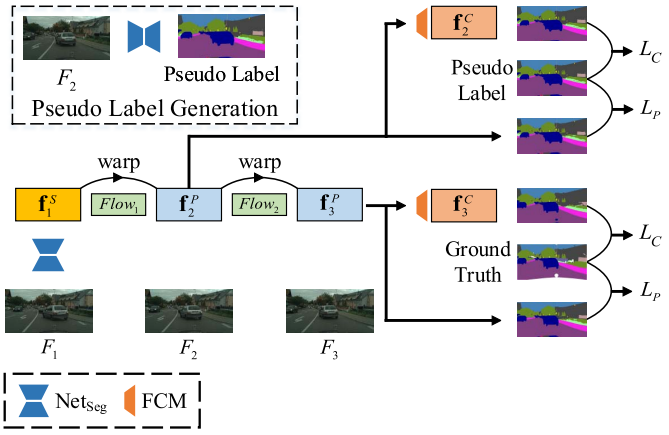

Fig. 9. **Illustration of training strategy**. We propose dual deep supervision (DDS) to improve the training of the network. Here L_P and L_C denote the loss calculated for feature propagation and correction, respectively. Best viewed in color.

in which the features on the distorted regions are mainly rectified, namely, distortion-guided feature correction (DGFC) is constructed. Then we have

$$\mathbf{f}_t^C = \mathbf{f}_t^P \odot (1 - M_t^D) + \mathbf{f}_t^{CC} \odot M_t^D, \quad (4)$$

where $\odot$ represents the spatially element-wise multiplication. It can be seen that the features on the distorted regions are dominated by the correction cue $\mathbf{f}_t^{CC}$ while the features on other regions are dominated by the propagated features $\mathbf{f}_t^P$.

D. Training Strategy

Here we explain the training strategy of our proposed method, which is illustrated in Fig. 9. Before elaborating on

the details, we briefly introduce the training procedure [8] widely used in previous works. For video semantic segmentation, let (F_1, F_3, GT) denote one training sample, where F_1 and F_3 are the key frame and current frame respectively, and GT is the segmentation ground truth of F_3. During training, F_1 is fed into the image segmentation model to extract features, and meanwhile the optical flow between F_1 and F_3 is estimated with FlowNet. Then the extracted features are propagated to F_3, and the *CrossEntropy* loss at F_3 is calculated to train networks. In practice, F_1 is randomly selected from a 10 frames video clip and F_3 is always the last one with ground truth, which can enrich the diversity of training samples.

However, the above training procedure may be unstable due to inaccuracy of optical flow estimation, especially for long-distance propagation (*e.g.*, larger than 5 frames). In this work, we propose *dual deep supervision* (DDS) to improve network training by providing more supervision. Specifically, we add an intermediate frame for each training sample, denoted by F_2, to reduce the propagation distance, and meanwhile impose the supervision signal on F_2. Note that our method propagates the features frame-by-frame in the inference phase, and thus two-warp operation in the training phase is more appropriate than the original one.

In our experiments, F_2 is randomly selected to ensure the diversity of training samples. To be specific, we extract the features of F_1, and conduct feature propagation twice ($F_1 \rightarrow F_2 \rightarrow F_3$). Then we produce the pseudo label of F_2 using the image segmentation model for more supervision. Actually, using the pseudo label has been a natural and popular way to improve the segmentation quality in domain adaptation [66] and semi-supervised learning [67]. Finally, we use the generated pseudo label and ground truth to supervise both the feature propagation and correction procedures on F_2 and F_3, as shown in Fig. 9. In particular, the propagation loss L_P works on the warped features $\mathbf{f}_t^P$ for improving the quality of optical flow, and the correction loss L_C works on the rectified features $\mathbf{f}_t^C$ for enhancing the ability of feature correction. The two losses can be written as:

$$L_P = -\frac{1}{HW} \sum_{h \in H, w \in W} \log p_t^P(h, w), \tag{5}$$

$$L_C = -\frac{1}{HW} \sum_{h \in H, w \in W} \log p_t^C(h, w), \tag{6}$$

where p_t^P and p_t^C are the predicted probability towards the ground truth from $\mathbf{f}_t^P$ and $\mathbf{f}_t^C$, respectively. Taking the loss of feature learning in FCM L_{DGFL}, our final loss for one single frame is

$$L = (L_P + L_C + L_{DGFL})/3. \tag{7}$$

Our method consists of four main components, *i.e.*, Net_{seg}, FlowNet, DMNet, and CFNet. Here we explain how they are trained. Net_{seg} is pretrained on ImageNet and then finetuned on a particular segmentation dataset (*e.g.*, Cityscapes, CamVid, and UAVid). DMNet is trained with the generated ground-truth distortion maps. Net_{seg} and DMNet would keep fixed in the following training procedure. FlowNet is pretrained on the synthetic Flying Chairs dataset [12] and then jointly trained with the randomly initialized CFNet by following the proposed DDS training strategy. For each step of training, we adopt the Adam optimizer [68] with $\beta_1 = 0.9$ and $\beta_2 = 0.99$. The learning rate is set to 10^{-4} for the first 50 epochs and then fixed to 10^{-5} for the rest 50 epochs.

IV. Experiment

In this section, we experimentally evaluate our proposed method on three challenging datasets, namely, Cityscapes [2], CamVid [3], and UAVid [4], and compare it with some state-of-the-art methods. We conduct all of the experiments on the NVIDIA GTX 1080Ti GPUs.

A. Datasets

Cityscapes [2]: is a popular dataset in semantic segmentation and autonomous driving domain. It focuses on semantic understanding of urban street scenes. The training and validation subsets contain 2,975 and 500 video clips, respectively, and each video clip contains 30 frames. The 20th frame in each clip is annotated by pixel-level semantic labels with 19 categories.

CamVid [3]: also focuses on the semantic understanding of urban street scenes, but it contains less data than Cityscapes. It only has 701 color images with annotations of 11 semantic classes. CamVid is divided into the trainval set with 468 samples and test set with 233 samples. All samples are extracted from driving videos captured at daytime and dusk, and have pixel-level semantic annotations. Each CamVid video contains 3,600 to 11,000 frames at a resolution of 720×960.

UAVid [4]: is a high-resolution Unmanned Aerial Vehicle (UAV) semantic segmentation dataset, which brings new challenges, including large scale variation, moving object recognition and temporal consistency preservation. The training and validation subsets contain 20 and 7 video clips, respectively, and each video clip contains 900 frames at a resolution of 2160×3840. Every 100 frames in each clip are annotated by pixel-level semantic labels with 8 categories.

B. Evaluation Metrics

We experimentally evaluate different video semantic segmentation methods by measuring the segmentation accuracy and computational efficiency.

For segmentation accuracy, we propose to use *propagation distance vs. accuracy curve* (PDA Curve), which indicates how the segmentation accuracy changes along different propagation distances. Some previous works [8], [11] use the average accuracy among different propagation distances, which is inconvenient to figure out the actual performance. For computational efficiency, we propose to use *computation cost vs. accuracy curve* (CCA Curve). CCA Curve is an important metric for model deployment, which indicates how the segmentation accuracy changes along different average computation cost. Note that the PDA and CCA represent similar information on the segmentation performance since different computation costs are actually obtained by setting different propagation distances.

TABLE I

CALCULATION OF FLOPS FOR DIFFERENT OPERATORS. H_i, W_i, AND C_i ARE THE HEIGHT, WIDTH, AND CHANNEL OF THE INPUT FEATURE MAP, AND H_o, W_o, AND C_o CORRESPOND TO THE OUTPUT FEATURE MAP. K_h AND K_w REPRESENT THE SIZE OF CONVOLUTIONAL KERNEL

Layer	FLOPs
Convolution	$2H_oW_o(C_iK_hK_w+1)C_o$
Bilinear Upsampling	$11H_oW_oC_o$
Batch Normalization	$2H_iW_iC_i$
ReLU or LReLU	$H_iW_iC_i$

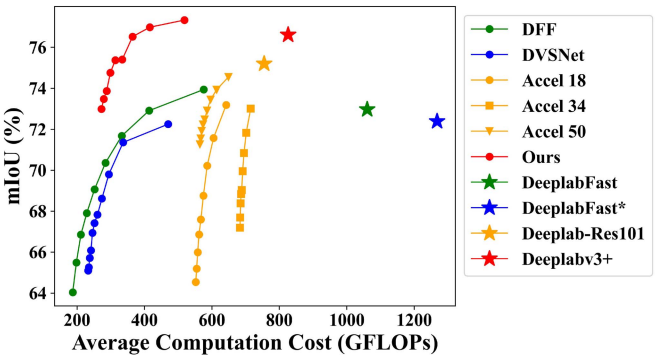

Fig. 10. **Performance comparison of different methods on Cityscapes val subset with CCA Curve.** Here ★ denotes the results of per-frame image segmentation model. In particular, "DeeplabFast*" represents the segmentation model used in DVSNet, which processes the regions of frame multiple times and thus has higher computation cost. Best viewed in color.

In the experiments on Cityscapes, we set the 11th to 19th frames as the key frame candidates, and propagate the feature of the selected key frame to the annotated 20th frame, which is used to measure the segmentation accuracy for each video clip. That is, the propagation distance (denoted by D_P) ranges from 1 to 9 for plotting the PDA Curve. When plotting the CCA Curve, we first calculate the computation cost of components used on the key frames (*i.e.*, Net_{seg}) and non-key frames (*i.e.*, FlowNet, CFNet, and DMNet), which are denoted by C_{seg} and C_{warp}, respectively. The average computation cost is calculated by

$$C_{mean} = (C_{seg} + C_{warp} * D_P)/(D_P + 1). \tag{8}$$

The evaluation on CamVid and UAVid is similar to Cityscapes. Here the mean intersection over union (mIoU) is adopted to measure the segmentation accuracy, and floating point operations (FLOPs) is used for the computation cost. Following common practices [69], [70], we calculate the FLOPs of convolutional layer, batch normalization layer, activation layer, and bilinear upsampling operator, of which the formulas are provided in Table I.

C. Performance Comparison

We compare our proposed method with recent state-of-the-art methods, including DFF [8], DVSNet [9], and Accel [11], and the CCA Curve is used for evaluation. Considering the baseline methods only provide the model

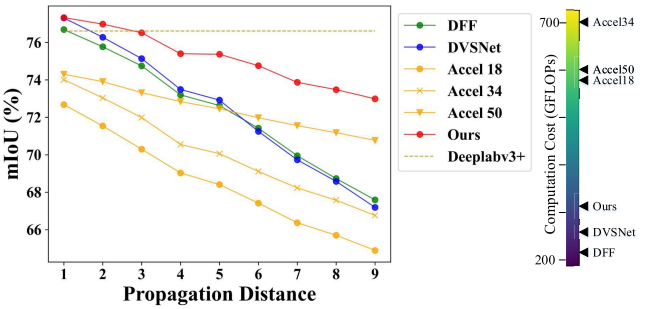

Fig. 11. **Performance evaluation on Cityscapes val subset with PDA Curve.** All methods are equipped with Deeplabv3+ as the backbone of segmentation network for fair comparison. The colorbar represents the computation cost of different methods for the propagation distance $D_P = 5$, in which lighter color indicates higher computation cost. Best viewed in color.

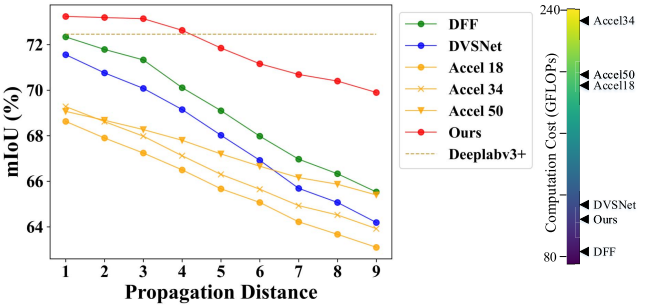

Fig. 12. **Performance evaluation on CamVid with PDA Curve.** All methods are equipped with Deeplabv3+ as the backbone of segmentation network for fair comparison. The colorbar represents the computation cost of different methods for the propagation distance $D_P = 5$, in which lighter color indicates higher computation cost. Best viewed in color.

on Cityscapes, here we only give the results on Cityscapes for fair comparison (the results on other datasets using our implementation will be presented in ablation study). To be specific, DFF and DVSNet use the same network DeeplabFast as the segmentation backbone. But DVSNet splits the input frames into four overlapped regions to perform multiple times of segmentation, which is obviously more time-consuming. As for Accel, Deeplab with deformable ResNet-101 is used for image segmentation, and multiple versions of ResNets with different depths are adopted to process the current frame. Fig. 10 shows the results of different methods on Cityscapes val subset. It can be seen that our proposed method significantly outperforms other method in both accuracy and efficiency.

D. Ablation Study

1) Effectiveness of Our Method: Here we verify the effectiveness of our method on Cityscapes, CamVid, and UAVid, and the results are shown in Fig. 11, Fig. 12, and Fig. 13, respectively. For fair comparison, we reimplement the baseline methods with DeepLabv3+ as the backbone of segmentation networks and same FlowNet as in our proposed method. In particular, our implemented DeepLabv3+ achieves a mIoU score of 76.61% on Cityscapes, 72.46% on CamVid, and 69.30% on UAVid for per-frame image segmentation. From

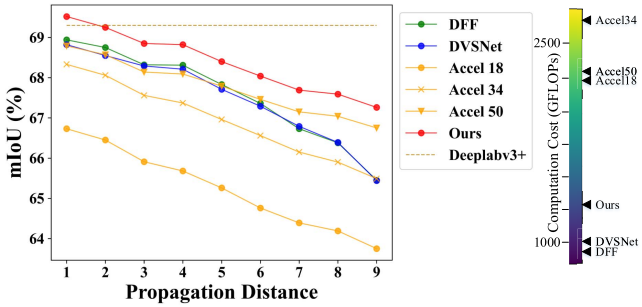

Fig. 13. **Performance evaluation on UAVid with PDA Curve.** All methods are equipped with Deeplabv3+ as the backbone of segmentation network for fair comparison. The colorbar represents the computation cost of different methods for the propagation distance $D_P = 5$, in which lighter color indicates higher computation cost. Best viewed in color.

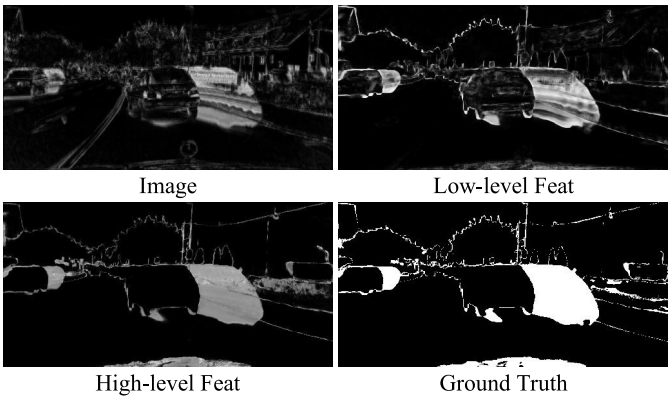

Fig. 14. **The predicted distortion maps using different features.** It can be seen that the higher-level feature can get better distortion maps.

the results, it can be seen that our proposed method significantly outperforms other state-of-the-art methods, especially for long-distance feature propagation.

Besides, we calculate the average computation cost by fixing the propagation distance as 5 for all methods. The results are shown in Fig. 11, Fig. 12, and Fig. 13 with color bars, in which lighter color represents higher computation cost. Note that the computation cost of Accel34 is higher than that of Accel 50 because an extra deconvolutional layer is involved in Accel34 for feature upsampling. Each component in our proposed framework has a complexity of $O(HW)$, where H and W are height and width of the input images. Moreover, we analyze the computation cost of the main components and overall framework for key and non-key frames, and the statistics are provided in Table II. Note that the computation cost for key frames is that of the image segmentation model. It can be seen that the image segmentation network dominates the computation cost. As shown in Fig. 11, Fig. 12, and Fig. 13, our method has slightly higher computation cost than DFF and DVSNet, but gets significant accuracy improvement.

Actually, the key of the proposed method getting accuracy improvement and efficient computation is our designed distortion-aware mechanism. Benefited from such a design, both the feature extraction from current frames and feature correction can be completed by a lightweight network (e.g., DMNet and CFNet) due to only handling part of an image. Moreover, only correcting features in distorted regions can effectively avoid false correction and further boost segmentation accuracy. Therefore, our method can outperform state-of-the-art methods in terms of accuracy and computation cost.

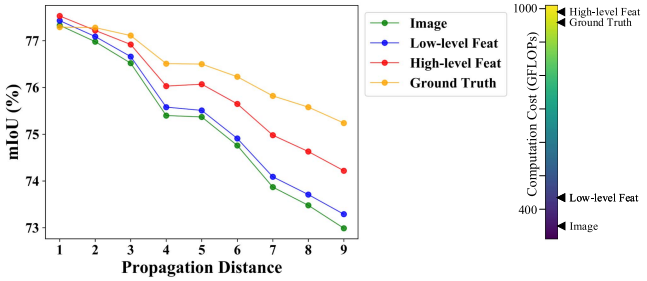

Fig. 15. **Comparison of different data to predict distortion maps.** Here the segmentation accuracy on Cityscapes val subset is adopted for evaluation. It can be seen that higher-level feature can generate higher-quality distortion maps but would involve higher computation cost. Best viewed in color.

TABLE II

COMPUTATION COST OF DIFFERENT MODULES (GFLOPS). THE RESOLUTION OF INPUT IMAGES IS 1024×2048 ON CITYSCAPES, 720×960 ON CAMVID, AND 2160×3840 ON UAVID

Module	Cityscapes	CamVid	UAVid
FlowNet	86.918	29.219	341.441
CFNet	123.775	41.741	484.319
DMNet	0.400	0.132	1.584
Overall for key	826.378	272.189	3265.457
Overall for non-key	212.910	71.448	830.515

TABLE III

EFFECT OF DIFFERENT COMPONENTS IN OUR METHOD ON CITYSCAPES VAL SUBSET. "MIOU" IS USED AS THE METRIC

DDS	FCM		Distance			
	DGFL	DGFC	1	5	9	Mean
√	√	√	**77.33**	**75.37**	**72.99**	**75.19**
	√	√	77.30	74.30	71.37	74.26 (-0.93)
√		√	77.09	74.34	71.16	74.17 (-1.02)
√	√		74.98	73.65	71.92	73.56 (-1.63)
√			76.64	72.45	67.53	72.21 (-2.98)

It is notable that our method can yield higher segmentation accuracy than per-frame image segmentation for short-distance feature propagation. It is because our proposed feature propagation can well exploit the information from multiple frames. That is, the segmentation of the current frame would benefit from the feature combination of the previous and current frames.

2) Effect of Different Components: Here we investigate the contribution of each proposed component to the segmentation performance by removing them (*i.e.*, DDS, DGFL, and DGFC) one by one. Table III gives the results, in which the propagation distances $\{1, 5, 9\}$ are particularly used and the mean segmentation accuracy over all distances is also provided. From the results, we have the following observations. (1) DDS can improve the network training of our proposed method and

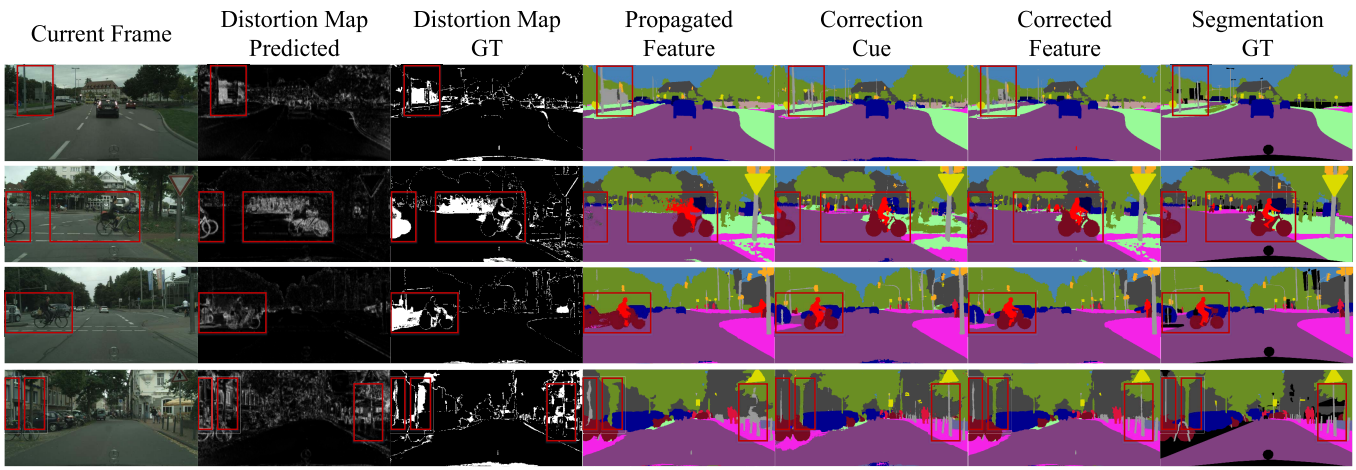

Fig. 16. **Visualization of some samples from Cityscapes.** It can be seen that the predicted distortion map can represent the distortion pattern of propagated features, and our proposed method can effectively correct the distorted features. Red rectangles highlight the main distorted regions. Best viewed in color.

TABLE IV

UPPER BOUND ANALYSIS OF DIFFERENT METHODS ON CITYSCAPES VAL SUBSET. MIOU IS USED AS THE METRIC. * DENOTES THE UPPER BOUND, AND ↑ REPRESENTS THE CORRESPONDING GAP

Distance	1	5	9	Mean	↑
DFF	73.94	69.06	64.04	69.14	
DFF*	75.47	71.16	66.31	71.15	2.01
Accel18	73.19	67.59	64.54	68.21	
Accel18*	77.37	70.44	66.92	71.26	3.05
Accel34	73.01	69.03	67.19	69.64	
Accel34*	78.33	73.08	70.76	73.82	4.18
Accel50	74.55	72.48	71.27	72.70	
Accel50*	78.97	75.98	74.52	76.36	3.66
Ours	**77.33**	**75.37**	**72.99**	**75.19**	
Ours*	**77.99**	**76.59**	**74.50**	**76.37**	**1.18**

further bring accuracy increase, as shown in the first two rows. (2) FCM is the main source of performance gains, especially for long-distance feature propagation that usually would cause serious distortion. (3) DGFL and DGFC in FCM are both important. In particular, CFNet without DGFL cannot effectively extract the correction cues, and without DGFC, the false correction would become severe, especially for short-distance propagation. Considering these results, it is convinced that our proposed distortion-aware feature correction is very effective for boosting the performance of propagation-based video semantic segmentation.

3) Upper Bound Analysis: Similar to our proposed method, DFF [8] and Accel [8] also rectify the propagated features. Here we explore the upper bound of segmentation accuracy of these methods, in which only the wrongly predicted regions are rectified during inference (the ground truth of semantic segmentation is used). Table IV shows the results. It can be seen that DFF and Accel have a larger gap corresponding to their upper bounds while our method can achieve a smaller one. Such results imply that our method can effectively alleviate false correction with our proposed distortion prediction.

4) Design of Distortion Map Prediction: In this work, we propose to predict the distortion maps using images rather than features in order to achieve low computation cost. Here we compare different data to predict distortion maps regardless of the computational price. In particular, we take the propagated features after classifier (high-level feature) and features after entry flow block2 in DeepLabv3+ (low-level feature) as the inputs of DMNet. Besides, we use the ground truth of distortion maps to test the upper bound of segmentation performance, which can well demonstrate the effectiveness of our idea to exploit the distortion map in video semantic segmentation.

Fig. 14 provides the visual comparison of predicted distortion maps for different features, and Fig. 15 gives the corresponding segmentation performance. We have the following observations. First, the higher-level feature can bring higher segmentation accuracy for getting more consistent distortion maps with the ground truth, but would involve higher computation cost. Thus we need to find a good trade-off between the segmentation accuracy and computation cost. Second, from the results in Table III and Fig. 15, it can be seen that our proposed method can get significant performance improvement if the ground truths of distortion maps are used, which shows the rationality of focusing on the distortion regions in this work.

5) Visualization: To intuitively illustrate our proposed method, we provide the visualizations of four samples from the Cityscapes datasets in Fig. 16, where the intermediate features are demonstrated by applying the segmentation head to them. For each sample, we extract the feature from the key frame, and then propagate it to the current frame (the 9th one from the key frame). First, we can see that the predicted distortion maps can represent the distorted regions of propagated features. Second, under the guidance of distortion maps, our proposed method can accurately extract the correction cues, especially on the distorted regions, and then effectively correct the propagated features.

V. CONCLUSION

We present a novel video semantic segmentation method in this article, aiming at achieving high segmentation accuracy and competitive real-time performance simultaneously

by tackling the feature distortion problem in propagation. Specifically, we propose DMNet to predict distorted regions of the propagated features, and then propose FCM to correct the distorted features with a lightweight model. Our experimental results on Cityscapes, CamVid, and UAVid show that the proposed method outperforms the state-of-the-art methods in both precision and speed.

ACKNOWLEDGMENT

The authors acknowledge the support of GPU cluster built by the MCC Laboratory of Information Science and Technology Institution, USTC.

REFERENCES

[1] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[2] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[3] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognit. Lett.*, vol. 30, no. 2, pp. 88–97, Jan. 2009.

[4] Y. Lyu, G. Vosselman, G.-S. Xia, A. Yilmaz, and M. Y. Yang, "UAVid: A semantic segmentation dataset for UAV imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 165, pp. 108–119, Jul. 2020.

[5] I. Kostavelis and A. Gasteratos, "Semantic mapping for mobile robotics tasks: A survey," *Robot. Auto. Syst.*, vol. 66, pp. 86–103, Apr. 2015.

[6] M. Teichmann, M. Weber, M. Zollner, R. Cipolla, and R. Urtasun, "MultiNet: Real-time joint semantic reasoning for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1013–1020.

[7] S. Liu, C. Wang, R. Qian, H. Yu, R. Bao, and Y. Sun, "Surveillance video parsing with single frame supervision," in *Proc. CVPR*, 2017, pp. 413–421.

[8] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei, "Deep feature flow for video recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2349–2358.

[9] Y.-S. Xu, T.-J. Fu, H.-K. Yang, and C.-Y. Lee, "Dynamic video segmentation network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6556–6565.

[10] R. Gadde, V. Jampani, and P. V. Gehler, "Semantic video CNNs through representation warping," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4453–4462.

[11] S. Jain, X. Wang, and J. E. Gonzalez, "Accel: A corrective fusion network for efficient semantic segmentation on video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8866–8875.

[12] A. Dosovitskiy *et al.*, "FlowNet: Learning optical flow with convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2758–2766.

[13] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "FlowNet 2.0: Evolution of optical flow estimation with deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2462–2470.

[14] P. Liu, M. Lyu, I. King, and J. Xu, "SelFlow: Self-supervised learning of optical flow," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4571–4580.

[15] M. Neoral, J. Šochman, and J. Matas, "Continual occlusion and optical flow estimation," in *Proc. ACCV*, 2018, pp. 159–174.

[16] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and 0.5 mb model size," 2016, *arXiv:1602.07360*. [Online]. Available: http://arxiv.org/abs/1602.07360

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[19] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[21] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[22] Z. Wu, C. Shen, and A. van den Hengel, "Wider or deeper: Revisiting the ResNet model for visual recognition," *Pattern Recognit.*, vol. 90, pp. 119–133, Jun. 2019.

[23] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1925–1934.

[24] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, Atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[25] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*. [Online]. Available: http://arxiv.org/abs/1511.07122

[26] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," 2014, *arXiv:1412.7062*. [Online]. Available: http://arxiv.org/abs/1412.7062

[27] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1529–1537.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[29] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: http://arxiv.org/abs/1706.05587

[30] Q. Wang, M. Chen, F. Nie, and X. Li, "Detecting coherent groups in crowd scenes by multiview clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 1, pp. 46–58, Jan. 2020.

[31] Q. Wang, J. Gao, and Y. Yuan, "A joint convolutional neural networks and context transfer for street scenes labeling," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1457–1470, May 2018.

[32] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 603–612

[33] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5693–5703.

[34] Z. Zhong *et al.*, "Squeeze-and-Attention networks for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13065–13074.

[35] Y. Li *et al.*, "Learning dynamic routing for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8553–8562.

[36] K. Lin, L. Wang, K. Luo, Y. Chen, Z. Liu, and M.-T. Sun, "Cross-domain complementary learning using pose for multi-person part segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, early access, May 15, 2020, doi: 10.1109/TCSVT.2020.2995122.

[37] J. Ji, R. Shi, S. Li, P. Chen, and Q. Miao, "Encoder-decoder with cascaded crfs for semantic segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Aug. 11, 2020, doi: 10.1109/TCSVT.2020.3015866.

[38] Y. Li, J. Shi, and D. Lin, "Low-latency video semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5997–6005.

[39] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artif. Intell.*, vol. 17, nos. 1–3, pp. 185–203, Aug. 1981.

[40] M. Anguita, J. Diaz, E. Ros, and F. J. Fernandez-Baldomero, "Optimization strategies for high-performance computing of optical-flow in general-purpose processors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 10, pp. 1475–1488, Oct. 2009.

[41] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-net: CNNs for optical flow using pyramid, warping, and cost volume," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8934–8943.

[42] M. Zhai, X. Xiang, R. Zhang, N. Lv, and A. El Saddik, "Optical flow estimation using dual self-attention pyramid networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 10, pp. 3663–3674, Oct. 2020.

[43] Q. Chen and V. Koltun, "Full flow: Optical flow estimation by global optimization over regular grids," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 4706–4714.
[44] N. Sundaram, T. Brox, and K. Keutzer, "Dense point trajectories by GPU-accelerated large displacement optical flow," in *Proc. ECCV*, 2010, pp. 438–451.
[45] A. Ranjan and M. J. Black, "Optical flow estimation using a spatial pyramid network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4161–4170.
[46] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.
[47] Y. Yuan and J. Wang, "OCNet: Object context network for scene parsing," 2018, *arXiv:1809.00916*. [Online]. Available: http://arxiv.org/abs/1809.00916
[48] J. Fu *et al.*, "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3146–3154.
[49] H. Zhao *et al.*, "Psanet: Point-wise spatial attention network for scene parsing," in *Proc. ECCV*, 2018, pp. 267–283.
[50] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Proc. ECCV*, 2018, pp. 234–250.
[51] S. Chen, B. Wang, X. Tan, and X. Hu, "Embedding attention and residual network for accurate salient object detection," *IEEE Trans. Cybern.*, vol. 50, no. 5, pp. 2050–2062, May 2020.
[52] Z. Zhang, Z. Lin, J. Xu, W. Jin, S.-P. Lu, and D.-P. Fan, "Bilateral attention network for RGB-D salient object detection," 2020, *arXiv:2004.14582*. [Online]. Available: http://arxiv.org/abs/2004.14582
[53] D.-P. Fan *et al.*, "Pranet: Parallel reverse attention network for polyp segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2020, pp. 263–273.
[54] M. Fayyaz, M. H. Saffar, M. Sabokrou, M. Fathy, F. Huang, and R. Klette, "STFCN: Spatio-temporal fully convolutional neural network for semantic segmentation of street scenes," in *Proc. ACCV*, 2016, pp. 439–509.
[55] D. Nilsson and C. Sminchisescu, "Semantic video segmentation by gated recurrent flow propagation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6819–6828.
[56] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Deep End2End Voxel2 Voxel prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 1–5.
[57] Y. Wang, W. Ding, B. Zhang, H. Li, and S. Liu, "Superpixel labeling priors and mrf for aerial video segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 8, pp. 2590–2603, Aug. 2019.
[58] E. Shelhamer, K. Rakelly, J. Hoffman, and T. Darrell, "Clockwork convnets for video semantic segmentation," in *Proc. ECCV*, 2016, pp. 852–868.
[59] P. Hu, F. Caba, O. Wang, Z. Lin, S. Sclaroff, and F. Perazzi, "Temporally distributed networks for fast video semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8818–8827.
[60] J. Wang, P. Bhat, R. A. Colburn, M. Agrawala, and M. F. Cohen, "Interactive video cutout," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 585–594, Jul. 2005.
[61] Y. Li, J. Sun, and H.-Y. Shum, "Video object cut and paste," in *Proc. ACM SIGGRAPH Papers*, 2005, pp. 595–600.
[62] B. L. Price, B. S. Morse, and S. Cohen, "LIVEcut: Learning-based interactive video segmentation by evaluation of multiple propagated cues," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 779–786.
[63] Z. Chen, C. Guo, J. Lai, and X. Xie, "Motion-appearance interactive encoding for object segmentation in unconstrained videos," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1613–1624, Jun. 2020.
[64] W. Liu, G. Lin, T. Zhang, and Z. Liu, "Guided co-segmentation network for fast video object segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, early access, Jul. 20, 2020, doi: 10.1109/TCSVT.2020.3010293.
[65] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018, pp. 801–818.
[66] Y. Zou, Z. Yu, B. Vijaya Kumar, and J. Wang, "Unsupervised domain adaptation for semantic segmentation via class-balanced self-training," in *Proc. ECCV*, 2018, pp. 289–305.
[67] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou34, Y.-Y. Lin, and M.-H. Yang, "Adversarial learning for semi-supervised semantic segmentation," in *Proc. BMVC*, 2018, pp. 1–5.
[68] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015, pp. 1–15.
[69] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," in *Proc. ICLR*, 2019, pp. 1–17.
[70] A. G. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861

Jiafan Zhuang received the B.S. degree in control science and engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2017, where he is currently pursuing the Ph.D. degree in control science and engineering.

His current research interests include semantic segmentation and video analysis.

Zilei Wang (Member, IEEE) received the B.S. and Ph.D. degrees in control science and engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2002 and 2007, respectively.

He is currently an Associate Professor with the Department of Automation, USTC, and the Founding Lead of the Vision and Multimedia Research Group. His research interests include computer vision, multimedia, and deep learning.

Prof. Wang is a Member of the Youth Innovation Promotion Association and the Chinese Academy of Sciences.

Bingke Wang received the B.S. degree in control science and engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2018, where he is currently pursuing the M.S. degree in control science and engineering.

His current research interests include lane detection and segmentation.