FuseSeg: Semantic Segmentation of Urban Scenes Based on RGB and Thermal Data Fusion

Yuxiang Sun, *Member, IEEE*, Weixun Zuo, Peng Yun, Hengli Wang, and Ming Liu, *Senior Member, IEEE*

***Abstract*— Semantic segmentation of urban scenes is an essential component in various applications of autonomous driving. It makes great progress with the rise of deep learning technologies. Most of the current semantic segmentation networks use single-modal sensory data, which are usually the RGB images produced by visible cameras. However, the segmentation performance of these networks is prone to be degraded when lighting conditions are not satisfied, such as dim light or darkness. We find that thermal images produced by thermal imaging cameras are robust to challenging lighting conditions. Therefore, in this article, we propose a novel RGB and thermal data fusion network named FuseSeg to achieve superior performance of semantic segmentation in urban scenes. The experimental results demonstrate that our network outperforms the state-of-the-art networks.**

***Note to Practitioners*—This article investigates the problem of semantic segmentation of urban scenes when lighting conditions are not satisfied. We provide a solution to this problem via information fusion with RGB and thermal data. We build an end-to-end deep neural network, which takes as input a pair of RGB and thermal images and outputs pixel-wise semantic labels. Our network could be used for urban scene understanding, which serves as a fundamental component of many autonomous driving tasks, such as environment modeling, obstacle avoidance, motion prediction, and planning. Moreover, the simple design of our network allows it to be easily implemented using various deep learning frameworks, which facilitates the applications on different hardware or software platforms.**

***Index Terms*— Autonomous driving, information fusion, semantic segmentation, thermal images, urban scenes.**

I. Introduction

SEMANTIC image segmentation generally refers to densely label each pixel in an image with a category. Recent years have witnessed a great trend for semantic segmentation shifting from traditional computer vision algorithms to deep learning-based approaches, in which convolutional neural networks (CNNs) have been proven to be really effective in tackling the semantic segmentation problem. With the popularities of autonomous vehicles [1]–[5] and human-assistant driving [6]–[9], semantic segmentation of urban scenes has attracted great attention. It has become a fundamental component for autonomous driving. For example, it provides contributive information to improve point-cloud registration [10]–[12], which is the backbone of many localization and mapping algorithms [13]–[17]. Note that the type of urban scenes we are considering is the street scene because we use the public data set released in [18] and the data set is recorded in urban street scenes.

Currently, most of the deep learning-based semantic segmentation networks are designed using single-modal sensory data, which are usually RGB images generated by visible cameras. However, RGB images could become less informative when lighting conditions are not satisfied, such as dim light or total darkness. We found that thermal images are robust to challenging lighting conditions. They are transformed from thermal radiations by thermal imaging cameras. Virtually, any matter with temperature above absolute zero could be seen with thermal [21]. The spectrum of thermal radiations ranges from 0.1 to 100 μm, whereas the visible light ranges from 0.4 to 0.76 μm. Most of the thermal radiations are invisible to human eyes or imaging sensors (e.g., CCD or CMOS) but visible to thermal imaging cameras. Therefore, thermal images could be helpful to detect and segment objects when lighting conditions are not satisfactory.

Note that Lidars can also work in unsatisfactory lighting conditions. The advantages of using thermal imaging cameras lie in fourfold. First, thermal imaging cameras are expensive than visible cameras, but they are still much cheaper than Lidars. For price-sensitive applications, such as driver assistance systems, solutions based on thermal imaging cameras would be more attractive. Second, thermal images are grayscale visual images in nature. Therefore, technology advancements in computer vision could directly benefit thermal imaging applications. For example, successful CNNs could be directly used on thermal images to extract features without any modification. While Lidar point clouds have different data structures from images, they are sparse point lists instead of dense arrays [22]–[24]. Computer vision techniques might not be directly used on Lidar point clouds [23]. Third, thermal imaging cameras can provide real-time dense images, such as visible cameras. For instance, the FLIR automotive

Manuscript received January 9, 2020; revised April 1, 2020; accepted May 4, 2020. This article was recommended for publication by Associate Editor C. Yang and Editor D. O. Popa upon evaluation of the reviewers' comments. This work was supported in part by the National Natural Science Foundation of China under Project U1713211, and in part by the Research Grant Council of Hong Kong under Project 11210017. *(Corresponding author: Ming Liu.)*

Yuxiang Sun, Weixun Zuo, Hengli Wang, and Ming Liu are with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: eeyxsun@ust.hk, sun.yuxiang@outlook.com; wzuo@connect.ust.hk; hwangdf@connect.ust.hk; eelium@ust.hk).

Peng Yun is with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong (e-mail: pyun@connect.ust.hk).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2020.2993143

thermal cameras[1] could stream thermal images with the resolution of 512×640 and can run at 60 Hz. However, Lidar point clouds are much sparser than thermal images, and the frame rates are slow. For example, the Velodyne HDL-64E S3 can only rotate up to 20 Hz [25]. As a semantic understanding device, the sparse measurements (64 lines) may overlook object details or far-distance small objects, and the slow frame rate may introduce artifacts or motion distortions that may hinder the perception. Finally, current spinning Lidars are mechanically complex, which mainly stems from the optical beam deflection unit. The mechanical parts, such as motors and gears, are subject to friction and abrasion, making Lidars less durable in long-term operation. In addition, autonomous vehicles usually require Lidars to be installed outside, which may directly expose them under adverse weather conditions and hence shorten the life expectancy, whereas thermal imaging cameras are only electronic devices and could be placed inside vehicles, such as visible cameras. They could work in long term without extra maintenance.

Many researchers resort to Lidar-camera fusion to overcome the limitations of solely using visible cameras. For example, Gao *et al.* [26] proposed a CNN-based method for object classification with Lidar-camera fusion. They convert the sparse Lidar point clouds to front-view depth images and upsample the depth images to dense ones. Then, the depth images and RGB images can be registered and processed by CNN. Qi *et al.* [27] proposed a cascade Lidar-camera fusion pipeline, in which 2-D region proposals are extracted from front-view RGB images with a 2-D image-based object detector, and then, the region proposals are projected to 3-D frustums in point clouds. The points in the frustums are processed by PointNet to get the instance segmentation results. Despite the success of Lidar-camera fusion methods, we still think that RGB-thermal fusion would be more suitable than Lidar-camera fusion for semantic reasoning in autonomous driving. Because vulnerable road users, such as pedestrians, normally have higher temperatures than surrounding environments, they are more discernible in thermal images, which could provide strong signals for segmentation. In addition, thermal imaging cameras can work at 60 Hz or higher, which allows semantic reasoning to be performed space intensively. Taking 70-km/h vehicle speed as an example, the vehicle moving distance between two consecutive images from a 60-Hz camera is around $(70 \times 10^3/60 \times 3600) \approx 0.3$ m. Such a distance between two times of semantic reasoning would be sufficient for most cases.

In this article, we fuse both the RGB and thermal data in a novel deep neural network to achieve superior performance in urban scenes. In particular, from the probabilistic data fusion theory [28], we have to find $\mathbb{P}(\text{Seg}|x_1, x_2) = \mathbb{P}(x_2|\text{Seg}, x_1)\mathbb{P}(x_1|\text{Seg})\mathbb{P}(\text{Seg})/\mathbb{P}(x_1\ x_2)$, where $\mathbb{P}(\cdot)$ represents the probability functions, Seg represents the segmentation results, x_1 and x_2 represent the RGB and thermal data, respectively, and $\mathbb{P}(x_1\ x_2)$ is usually the constant normalization term. The main novelty of this article lies in the network architecture, especially the data fusion strategy and the proposed decoder. The example in Fig. 1 shows that a person is clearly visible in the thermal image even the environment is with almost total darkness. We can see that our FuseSeg provides an acceptable segmentation result for the person, whereas the other two networks fail to segment the person. The example demonstrates that the networks relying only on RGB data could be degraded when lighting conditions are not satisfied, and our data fusion-based network could be a solution to address the problem. The contributions of this article are listed as follows.

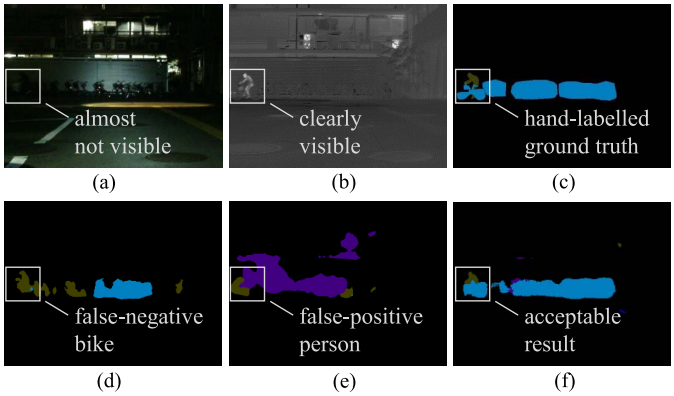


Fig. 1. Qualitative comparison with two state-of-the-art networks in an almost total darkness lighting condition. A person on a bike is almost invisible in the RGB image but can be clearly seen in the thermal image. We can see that both the SegHRNet [19] and DFN [20] fail to correctly segment the objects, whereas our FuseSeg can give an acceptable result. The yellow and blue colors in the mask images represent person and bike, respectively. The other colors represent other classes. The figure is best viewed in color. (a) RGB image. (b) Thermal image. (c) Ground truth. (d) SegHRNet. (e) DFN. (f) Our FuseSeg.

1) We develop a novel RGB-thermal data fusion network for semantic segmentation in urban scenes. The network can be used to get accurate results when lighting conditions are not satisfied, for instance, dim light, total darkness, or on-coming headlights, which is an advantage over the single-modal networks.
2) We construct our Bayesian FuseSeg using the Monte Carlo (MC) dropout technique [29] to analyze the uncertainty for the semantic segmentation results. The performance with different dropout rates is compared.
3) We evaluate our network on a public data set released in [18]. The results demonstrate our superiority over the state of the arts. We also evaluate our network on the SUN-RGBD v1 data set [30]. The results demonstrate our generalization capability to RGB-D data.

The remainder of this article is organized as follows. In Section II, we review the related work. In Section III, we describe our network in detail. Sections IV–VI present the experimental results and discussions. Conclusions and future work are drawn in Section VII.

II. Related Work

The related work to this article includes single-modal and data fusion semantic segmentation networks, as well as computer vision applications using thermal imaging. We review several representative works in each field.

[1]https://www.flir.com/products/adk

A. Single-Modal Semantic Segmentation Networks

The first work addressing the semantic segmentation problem end-to-end was the fully convolutional networks (FCNs) proposed by Shelhamer *et al.* [31]. They modified image classification networks, such as VGG-16 [32], into the fully convolutional form to achieve pixel-wise image segmentation. Noh *et al.* [33] developed DeconvNet that consists of a convolutional module for feature extraction and a deconvolutional module for resolution restoration. Badrinarayanan *et al.* [34] introduced the encoder–decoder concept in SegNet. The functionalities of encoder and decoder are analogous to those of the convolutional and deconvolutional modules in DeconvNet. Ronneberger *et al.* [35] developed UNet by introducing skip connections between the encoder and the decoder. It was proven to be effective to keep the spatial information by the skip connections. Although UNet was initially designed for biomedical imaging, it generalizes well to other domains. Paszke *et al.* [36] designed ENet for efficient semantic segmentation by speeding up the inference process of the initial block. They proposed a pooling operation in parallel with a convolutional operation with a stride of 2. Moreover, the asymmetric convolutions were employed in its bottleneck module to reduce the redundancy of convolutional weights. Zhao *et al.* [37] observed that context information could be helpful to improve semantic segmentation performance. Based on this observation, they introduced the pyramid pooling module (PPM) in PSPNet to extract local and global context information at different scales. Wang *et al.* [38] designed the dense upsampling convolution (DUC) and the hybrid dilated convolution (HDC) for decoder and encoder, respectively. Compared with bilinear upsampling and deconvolution networks, DUC is learnable and free of zero padding. HDC can alleviate the gridding issue during downsampling. Pohlen *et al.* [39] proposed FRRN for semantic segmentation, which consists of two processing streams. One stream maintains the feature map resolution at the input level. The other one performs pooling operations to increase the size of the receptive field. The two streams are coupled in the proposed FRRU block. Yu *et al.* [20] proposed DFN to address two common challenges in semantic segmentation: the intraclass inconsistency problem and the interclass indistinction problem. It mainly consists of a smooth network to capture the multiscale and global context information, as well as a border network to discriminate the adjacent patches with similar appearances but different class labels. ERFNet was developed by Romera *et al.* [40] for efficient semantic segmentation. The core components of ERFNet are the proposed 1-D convolutional layers with the kernel sizes of 3×1 and 1×3. The 1-D convolutional layers are combined with skip connections to form a residual block, which is integrated with an encoder–decoder architecture. Yu *et al.* [41] proposed BiSeNet that mainly consists of a spatial path and a context path. The spatial path was designed to preserve the spatial information. It contains three sequential downsampling convolutional operations, which reduces the feature map resolution to $1/8$ of the original input size. The context path was designed to provide a sizeable receptive field. An attention refinement module was developed in the context path for performance refinement. Sun *et al.* [42] developed HRNet that was able to keep high-resolution representations through the whole encoding process. The network was designed for human pose estimation but can be utilized as a general CNN backbone for other computer vision tasks. They improved HRNet by upsampling low-resolution representations to high resolution [19], with which semantic segmentation maps could be estimated.

B. Data Fusion Semantic Segmentation Networks

Apart from using the single-modal RGB data, depth data from RGB-D cameras [43] have been exploited for semantic segmentation. Hazirbas *et al.* [44] proposed FuseNet by fusing RGB and depth data in an encoder–decoder structure. In FuseNet, two encoders using VGG-16 as backbone were designed to take as inputs the RGB and depth data, respectively. The feature maps from the depth encoder were gradually fused into the RGB encoder. Wang and Neumann [45] fused RGB and depth information for semantic segmentation by introducing the depth-aware convolution and depth-aware average pooling operations, which incorporate geometry information in conventional CNN. They computed the depth similarities between the center pixel and neighboring pixels. The neighboring pixels with close depth values were weighted to contribute more in the operations. For semantic segmentation of urban scenes, MFNet [18] and RTFNet [46] were both proposed to use RGB and thermal data. Ha *et al.* [18] designed MFNet by fusing RGB and thermal data in an encoder–decoder structure. Two identical encoders were employed to extract features from RGB and thermal data, respectively. A mini-inception block was designed for the encoder. RTFNet [46] was also designed with two encoders and one decoder. In the decoder of RTFNet, two types of upception blocks were designed to extract features and gradually restore the resolution.

C. Computer Vision Applications Using Thermal Imaging

Apart from semantic segmentation, thermal imaging has been used in other computer vision applications, such as facial expression recognition [48]. Wang *et al.* [49] proposed a thermal-augmented facial expression recognition method. They designed a similarity constraint to jointly train the visible and thermal expression classifiers. During the testing stage, only visible images are used, which could reduce the cost of the system. Yoon *et al.* [50] utilized thermal images for drivable road detection at nighttime. A Gabor filter was applied to thermal images to find textureless areas that were considered as the rough detection results for the drivable road. Superpixel algorithms were employed on thermal images to smooth the segmentation results. Knapik and Cyganek [51] developed a yawn detection-based fatigue recognition method using thermal imaging for driver assistance systems. The method consists of a face detection module, an eye-corner localization module, and a yawn detection module. The yawn

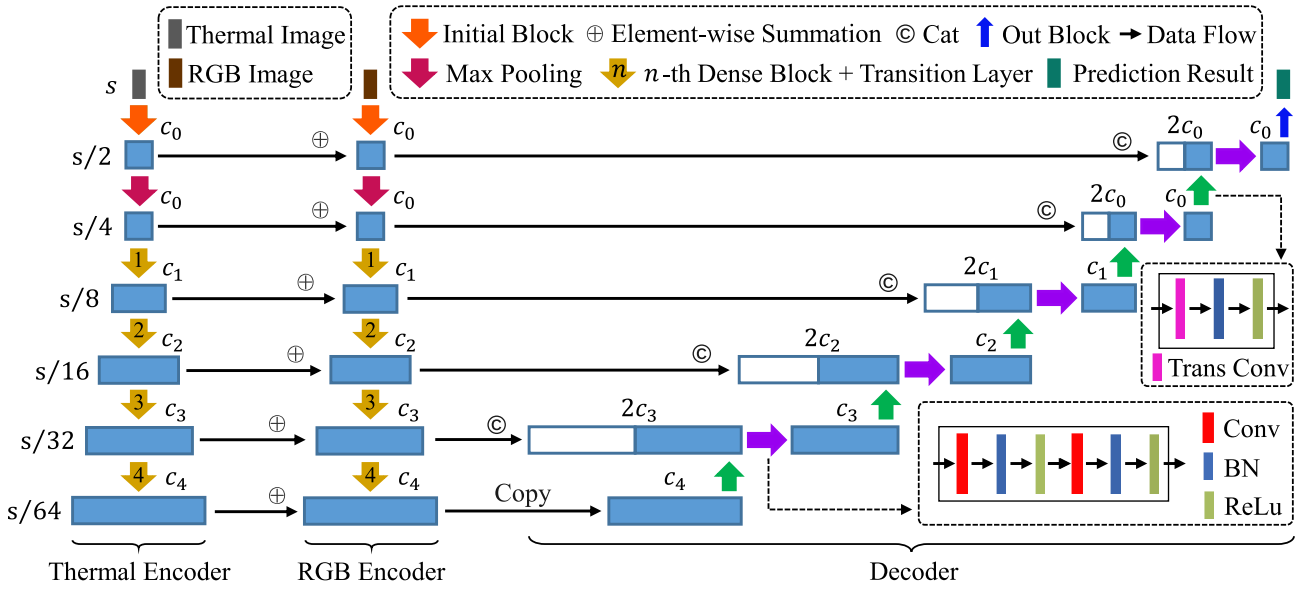


Fig. 2. Overall architecture of our FuseSeg. It consists of an RGB encoder, a thermal encoder, and a decoder. We employ DenseNet [47] as the backbone of the encoders. In the first stage of two-stage fusion (TSF), the thermal feature maps are hierarchically added with the RGB feature maps in the RGB encoder. The fused feature maps are then concatenated with the corresponding decoder feature maps in the second fusion stage. The blue rectangles represent the feature maps. The white rectangles represent the fused features maps copied from the RGB encoder. The purple and green arrows represent the feature extractor and the upsampler in the decoder, respectively. s represents the input resolution of the RGB and thermal images. $s = 480 \times 640$ in this article. The feature maps at the same level share the same resolution. c_n represents the number of channels of the feature maps at different levels. *Cat*, *Conv*, *Trans Conv*, and *BN* are short for concatenation, convolution, transposed convolution, and batch normalization. The figure is best viewed in color.

detection is inferred from a thermal-anomaly detection model, which is based on the temperature change measurement from the thermal imaging camera.

III. Proposed Network

A. Overall Architecture

We propose a novel data fusion network named FuseSeg. It generally consists of two encoders to extract features from input images and one decoder to restore the resolution. The two encoders take as input the three-channel RGB and one-channel thermal images, respectively. Fig. 2 shows the overall structure of our FuseSeg. We employ DenseNet [47] as the backbone of the encoders. We innovatively propose a TSF strategy in our network. As shown in Fig. 2, in the first stage, we hierarchically fuse the corresponding thermal and RGB feature maps through elementwise summation in the RGB encoder. Inspired by [35], the fused feature maps except the bottom one are then fused again in the second stage with the corresponding feature maps in the decoder through tensor concatenation. The bottom one is directly copied to the decoder instead of concatenation. With our TSF strategy, the loss of spatial information through the intensive downsampling could be recovered.

B. Encoders

The RGB and thermal encoders are designed with the same structure except for the input dimension because the input data are with different channels. As aforementioned, we use DenseNet as the backbone. We first delete the classification layer in DenseNet to avoid excessive loss of spatial information. Then, we add a transition layer that is similar to other transition layers after the fourth dense block. The dense blocks in DenseNet keep the feature map resolution unchanged, whereas the initial block, the max-pooling layer, and the transition layers reduce the feature map resolution by a factor of 2. Note that the feature map resolution has been reduced to 15×20 (given the input resolution of 480×640) before the final transition layer. Because we disable the ceiling mode of the average pooling operation in the final transition layer, the feature map resolution after the final transition layer is reduced to 7×10 (not 8×10). There are four architectures for DenseNet: DenseNet-121, DenseNet-169, DenseNet-201, and DenseNet-161. The complexity increases from 121 to 161. DenseNet-161 possesses the most number of parameters because it is grown with the largest rate of 48, whereas the others share the growth rate of 32. We refer readers to [47] for the details of DenseNet. Our FuseSeg follows the same naming rule of DenseNet. The number of channels c_n in Fig. 2 varies with different DenseNet architectures. Detailed numbers are listed in Table I.

C. Decoder

The decoder is designed to gradually restore the feature map resolution to the original. We design a decoder that mainly consists of three modules: a feature extractor that sequentially contains two convolutional layers, an upsampler, and an out block that both contain one transposed convolutional layer. Note that there are a batch normalization layer and a ReLu activation layer followed by the convolutional and transposed convolutional layers in the feature extractor and the upsampler. The detailed configurations for the convolutional and transposed convolutional layers are displayed in Table II.

TABLE I

NUMBER OF FEATURE MAP CHANNELS c_n AT DIFFERENT LEVELS ACCORDING TO DIFFERENT DENSENET ARCHITECTURES

	FuseSeg-121	FuseSeg-169	FuseSeg-201	FuseSeg-161
c_0	64	64	64	96
c_1	128	128	128	192
c_2	256	256	256	384
c_3	512	640	896	1056
c_4	512	832	960	1104

TABLE II

CONFIGURATIONS FOR THE CONVOLUTIONAL (CONV) AND TRANSPOSED CONVOLUTIONAL (TRANS CONV) LAYERS IN THE INDIVIDUAL MODULES OF THE DECODER

Modules	Name	Size	Stride	Padding
Upsampler	Trans Conv	2×2	2	0
Feature Extractor	Conv 1	3×3	1	1
	Conv 2	3×3	1	1
Out Block	Trans Conv	2×2	2	0

The feature extractor is employed to extract features from the fused feature maps. It keeps the resolution of the feature maps unchanged. Both the upsampler and out block increase the resolution by a factor of 2. The out block outputs the final prediction results with the channel number of 9, which is the number of classes. We add a softmax layer after the output to get the probability map for the segmentation results. As aforementioned, the feature map resolution is 7×10 at the end of the encoder. To restore it to 15×20, we employ a padding technique at this level of upsampler. The upsampled feature map is concatenated with the one from the RGB encoder during the second stage of our TSF. The number of feature channels doubles after the concatenation.

IV. EXPERIMENTAL SETUP

A. Data Set

In this article, we use the public data set released by Ha *et al.* [18]. It was recorded in urban street scenes, which contains common objects: car, person, bike, curve (road lanes), car stop, guardrail, color cone, and bump. The images are captured at the 480×640 resolution by an InfReC R500 camera, which can provide RGB and thermal images simultaneously. There are 1569 registered RGB and thermal images in the data set, among which 749 are taken at nighttime and 820 are taken at daytime. The data set was provided with hand-labeled pixel-wise ground truth, including the aforementioned eight classes of common objects and one unlabeled background class.

B. Training Details

We train the networks on a PC with an Intel i7 CPU and an NVIDIA 1080 Ti graphics card, including 11-GB graphics memories. We accordingly adjust the batch sizes for the networks to fit the graphics memories. We employ the data set splitting scheme used in [18]. The training set consists of 50% of the daytime images and 50% of the nighttime images, whereas the validation and test sets both consist of 25% of the daytime images and 25% of the nighttime images. The training set is augmented with the flip technique. Our FuseSeg is implemented with the PyTorch. The convolutional and transposed convolutional layers in the decoder are initialized using the Xavier scheme [52]. The encoder layers are initialized using the pretrained weights provided by PyTorch. We use the stochastic gradient descent (SGD) optimization solver and the cross-entropy loss for training. The learning rate is decayed exponentially. The networks are trained until no further decrease in the loss is observed.

C. Evaluation Metrics

For the quantitative evaluation, we use the same metrics from [46]: Accuracy (Acc) and intersection over union (IoU). Let Acc_i and IoU_i denote the Acc and IoU for class i. They are computed in the formulas

$$\text{Acc}_i = \frac{\sum_{k=1}^{K} \theta_{ii}^k}{\sum_{k=1}^{K} \theta_{ii}^k + \sum_{k=1}^{K} \sum_{j=1, j\neq i}^{N} \theta_{ij}^k}, \tag{1}$$

$$\text{IoU}_i = \frac{\sum_{k=1}^{K} \theta_{ii}^k}{\sum_{k=1}^{K} \theta_{ii}^k + \sum_{k=1}^{K} \sum_{j=1, j\neq i}^{N} \theta_{ji}^k + \sum_{k=1}^{K} \sum_{j=1, j\neq i}^{N} \theta_{ij}^k}, \tag{2}$$

where θ_{ii}^k, θ_{ij}^k, and θ_{ji}^k represent in the image k the number of pixels of class i that are correctly classified as class i, the number of pixels of class i that are wrongly classified as class j, and the number of pixels of class j that are wrongly classified as class i, respectively. K and N represent the number of test images and the number of classes, respectively. $N = 9$ in this article. We use mAcc and mIoU to represent the arithmetic average values of Acc and IoU across the nine classes.

V. ABLATION STUDY

A. Ablation for Encoders

1) Encoder Backbone: Since ResNet [53], WideResNet [54], ResNext [55], and HourglassNet [56] have similar structures as DenseNet, we replace DenseNet with these networks and compare their performance with ours. The quantitative results are listed in Table III. As we can see, using DenseNet-161 achieves the best performance, which confirms the effectiveness of our choice.

2) Single-Modal Performance: We delete the thermal encoder of FuseSeg to see the performance without using the thermal information. We name this variant as no thermal encoder (NTE). Similarly, we delete the RGB encoder to see how the network performs given only the thermal information. The variant is termed no RGB encoder (NRE). In these two variants, the first-stage fusion in our TSF strategy is canceled since there is only one encoder in the networks. We display the results with respect to different DenseNet architectures in Table IV. We can see that all the networks using DenseNet-161 gain more accuracy than the others. The superior performance is expected because DenseNet-161 presents the best image classification performance among the four DenseNet

TABLE III

RESULTS (%) OF ABLATION STUDY FOR ENCODERS USING DIFFERENT BACKBONES ON THE TEST SET. WE USE DENSENET-161 IN OUR NETWORK. BOLD FONT HIGHLIGHTS THE BEST RESULTS

Variants	Ours	ResNet-50	ResNet-101	ResNext-50	ResNext-101	Wide ResNet-50	Wide ResNet-101	HourglassNet
mAcc	**70.6**	54.1	65.8	65.7	68.4	67.2	66.7	48.6
mIoU	**54.5**	45.1	53.1	51.6	53.4	51.9	53.9	43.6

TABLE IV

RESULTS (%) OF ABLATION STUDY FOR ENCODERS ON THE TEST SET. OURS DISPLAYS THE RESULTS OF OUR FUSESEG. BOLD FONT HIGHLIGHTS THE BEST RESULTS

Variants	Ours		NTE		NRE	
	mAcc	mIoU	mAcc	mIoU	mAcc	mIoU
DenseNet-121	**59.9**	**49.1**	45.5	38.4	46.2	39.1
DenseNet-169	**66.6**	**50.5**	50.1	40.7	50.2	41.9
DenseNet-201	**63.4**	**51.4**	48.3	41.3	51.7	43.4
DenseNet-161	**70.6**	**54.5**	59.0	46.5	61.9	47.4

architectures. Moreover, our FuseSeg outperforms NTE and NRE, proving that the data fusion is a benefit here. Comparing NTE and NRE, we find that all the NRE results are better than those of NTE. This indicates that thermal information plays a significant role in our network.

B. Ablation for Fusion Strategy

For the ablation of fusion strategy, we compare the TSF proposed in our FuseSeg with seven variants. The former four variants modify the first stage of our TSF strategy. The next two modify the second stage. The last one modifies both the first stage and the second stage. The detailed descriptions for the variants are listed as follows.

1) *OEF:* This variant deletes all the fusion connections between the two encoders and keeps only one encoder. The encoder is fed with four-channel RGB-thermal data, so it is a version of only early fusion (OEF).
2) *HEF:* This variant keeps network unchanged, except that the RGB encoder is fed with four-channel RGB-thermal data, so it has early fusion (HEF).
3) *OLF:* This variant deletes the fusion connections between the two encoders except the last one, so the encoder feature maps are performed with only late fusion (OLF). Since there is no fusion between RGB and thermal at other levels, only the RGB feature maps are fused to the decoder at those levels.
4) *RCF:* The summation fusion between the encoders is replaced with concatenation fusion (RCF). To keep the number of channels unchanged, the concatenated feature maps are processed with a 1×1 convolution layer to reduce the number of channels.
5) *NSC:* This variant deletes all the fusion connections between the encoder and the decoder. Therefore, the variant has no skip connection NSC) between the encoder and the decoder except at the bottom level.
6) *RSF:* The concatenation fusion between the encoder and the decoder is replaced with the summation fusion (RSF). The input dimension of the feature extractor in the decoder is correspondingly modified to take as input the summarized feature map.
7) *CSF:* This variant combines RSF and RCF. Therefore, it is performed with the concatenation and summation fusion (CSF) at the first and second stages, respectively.

TABLE V

RESULTS (%) OF ABLATION STUDY FOR FUSION STRATEGY ON THE TEST SET. ALL THE VARIANTS USE THE DENSENET-161 AS THE ENCODER BACKBONE. OURS DISPLAYS THE RESULTS OF OUR FUSESEG-161. BOLD FONT HIGHLIGHTS THE BEST RESULTS

Variants	Ours	OEF	HEF	OLF	RCF	NSC	RSF	CSF
mAcc	**70.6**	61.2	66.7	63.2	66.5	60.9	67.8	63.1
mIoU	**54.5**	49.5	53.3	50.7	51.0	50.2	54.0	51.1

The results are displayed in Table V. Our FuseSeg with the proposed TSF strategy presents the best performance, which confirms the effectiveness of TSF. We find that OEF and NSC both provide low performance. The reason for the OEF performance could be that the features are not well extracted with only one encoder even it is fed with four-channel data. The inferior performance of NSC proves that the second-stage fusion between the encoder and the decoder in our TSF strategy is critical to improve the performance. We find from the HEF results that having the early fusion at the input level could degrade the performance. The OLF results show that the fusions between the two encoders at different levels are necessary for our network. From the results of RCF, RSF, and CSF, we could find that using summation for the first stage and concatenation for the second stage would be a superior choice here.

C. Ablation for Decoder

In our FuseSeg, the feature extractor in the decoder consists of two sequential *Conv-BN-ReLu* blocks, which is shown in Fig. 2. We compare our FuseSeg with five variants that have different feature extractors in the decoder. We list the detailed information as follows.

1) *TPC:* The feature extractor mainly consists of two parallel convolutional (TPC) layers.
2) *OC:* The feature extractor consists of only One *Conv-BN-ReLu* (OC) block.
3) *TSC:* The feature extractor consists of three sequential *Conv-BN-ReLu* (TSC) blocks.
4) *THC:* The feature extractor mainly consists of three hybrid-organized convolutional (THC) layers.
5) *FSC:* The feature extractor consists of four sequential *Conv-BN-ReLu* (FSC) blocks.

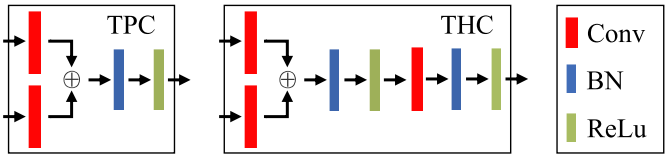


Fig. 3. Detailed structures for TPC and THC. The figure is best viewed in color.

The detailed structures for TPC and THC are shown in Fig. 3. All the convolutional layers in the different feature extractors share the same kernel size, stride, and padding with ours. We also build three variants that replace the *TranConv-BN-ReLu* block in the upsampler with different structures. The detailed descriptions are listed as follows.

1) *OCOI:* The upsampler sequentially consists of One *Conv-BN-ReLu* block and One Interpolation function (OCOI). The stride of the convolutional layer in the *Conv-BN-ReLu* block is 1. The scale factor for the interpolation function is 2.
2) *TPOI:* The upsampler sequentially consists of two parallel convolutional layers and one interpolation function (TPOI). The two parallel convolutional layers are similar to those in TPC. The stride for the convolutional layers is 1. The scale factor for the interpolation function is 2.
3) *OCOT:* The upsampler sequentially consists of One *Conv-BN-ReLu* block and One *TranConv-BN-ReLu* block (OCOT). The strides of the convolutional layers in the *Conv-BN-ReLu* block and the *TranConv-BN-ReLu* block are 1 and 2, respectively.

Table VI displays the results. For the feature extractor, our FuseSeg with the simple two sequential *Conv-BN-ReLu* blocks presents the best performance. OC, TSC, and FSC also have a sequential structure. Their results show that the performance decreases with the increasing number of layers in the sequential structure. We find that the THC results are close to ours. The reason could be that the summation of the two parallel convolutional layers in THC actually resembles the one convolutional layer in ours. It can be imagined as breaking one convolutional layer into two convolutional layers and then adding them together. This increases the number of parameters, but the results show that it could not increase the performance. A similar case happens to TPC and OC. The two parallel convolutional layers in TPC resemble the one convolutional layer in OC, so they share a similar performance, but TPC is slightly worse than OC. For the upsampler, we find that using the interpolation function (i.e., OCOI and TPOI) to increase the feature map resolution presents inferior performance. Comparing the results of ours and OCOT, we find that only using one transposed convolutional layer to simultaneously change the feature map dimension and increase the feature map resolution would be sufficient for our network.

VI. Comparative Study

We compare our FuseSeg with FRRN [39], BiSeNet [41], DFN [20], SegHRNet [19], MFNet [18], FuseNet [44], DepthAwareCNN [45], and RTFNet [46] in this section. The results of MFNet [18], FuseNet [44], and RTFNet [46] are directly imported from [46] to facilitate comparison. We use RTFNet-152, FRRN model B and HRNetV2-W48 here. The results of SegNet [34], UNet [35], ENet [36], PSPNet [37], DUC-HDC [38], and ERFNet [40] can be found in [46]. Our FuseSeg outperforms these networks. As FuseSeg uses four-channel RGB-thermal data, to make fair comparisons, we modify the input layers of the single-modal networks to take as input the four-channel data. We train and compare them using the three- and four-channel data, respectively.

TABLE VI
Results (%) of Ablation Study for the Decoder on the Test Set. All the Variants Use the DenseNet-161 as the Encoder Backbone. Ours Displays the Results of Our FuseSeg-161. Bold Font Highlights the Best Results

Variants	Ours	TPC	OC	TSC	THC	FSC	OCOI	TPOI	OCOT
mAcc	**70.6**	67.1	69.0	66.7	69.2	65.4	65.2	62.3	66.0
mIoU	**54.5**	52.0	53.7	54.0	53.8	52.0	52.0	53.4	53.8

A. Overall Results

Table VII displays the quantitative results for the comparison. We can see that our FuseSeg-161 outperforms the other networks in terms of mAcc and mIoU. Among the single-modal networks, both the DFN and SegHRNet present relatively good results, which shows the generalization capabilities of the networks. Comparing the three- and four-channel results of the single-modal networks, we find that almost all the four-channel results are better than the three-channel ones. This demonstrates that using thermal information is beneficial to the overall performance.

B. Daytime and Nighttime Results

We evaluate the networks under the daytime and nighttime lighting conditions, respectively. The comparative results are displayed in Table VIII. We find that FuseSeg outperforms most of the other networks. For the daytime condition, some of the single-modal networks using the three-channel data are better than those using four-channel data. We conjecture that the reason is that the registration errors [18] between the RGB and thermal images confuse the prediction. In the daytime, both RGB and thermal images encode strong features, so temporal or spatial misalignments between the two-modal data would give contradict information and thus degrade the performance. For the nighttime condition, almost all the single-modal networks provide superior performance when using the four-channel data. This is expected because RGB images are less informative when lighting conditions are not well satisfied. Incorporating visible thermal images could help the segmentation.

C. Inference Speed

Table IX displays the approximate number of parameters and the inference speed for each network. The speed is evaluated on an NVIDIA GTX 1080 Ti and an NVIDIA Jetson TX2 (Tegra X2). For the single-modal networks, we only test with four-channel data. We find that almost all the networks

TABLE VII

COMPARATIVE RESULTS (%) ON THE TEST SET. 3*c* AND 4*c* REPRESENT THAT THE NETWORKS ARE TESTED WITH THE THREE-CHANNEL RGB DATA AND FOUR-CHANNEL RGB-THERMAL DATA, RESPECTIVELY. NOTE THAT THE MACC AND MIOU ARE CALCULATED WITH THE UNLABELED CLASS, BUT THE RESULTS FOR THE UNLABELED CLASS ARE NOT DISPLAYED. THE BOLD FONT HIGHLIGHTS THE BEST RESULT IN EACH COLUMN

Method	Car		Person		Bike		Curve		Car Stop		Guardrail		Color Cone		Bump		mAcc	mIoU
	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU	Acc	IoU		
FRRN (4c)	81.9	74.7	66.2	60.8	62.8	50.3	41.2	35.0	12.5	11.5	0.0	0.0	37.2	34.0	35.2	34.6	48.5	44.2
FRRN (3c)	80.0	71.2	53.0	46.1	65.1	53.0	34.0	27.1	21.6	19.1	0.0	0.0	34.7	32.5	36.2	30.5	47.1	41.8
BiSeNet (4c)	89.7	84.1	72.0	63.2	74.1	60.1	45.1	36.7	34.2	25.3	18.2	5.0	47.4	42.2	39.8	35.9	57.7	50.0
BiSeNet (3c)	90.0	84.5	65.0	54.3	75.0	61.4	32.1	25.7	32.3	26.2	3.2	0.9	49.6	43.3	48.1	40.5	54.9	48.2
DFN (4c)	90.0	84.4	73.2	65.0	75.5	60.9	54.0	40.4	38.9	25.7	10.2	4.0	48.3	42.5	55.8	47.4	60.5	52.0
DFN (3c)	90.7	81.4	67.7	52.8	71.5	57.5	49.2	34.9	35.1	23.8	4.1	0.9	44.2	31.0	54.6	47.5	57.3	47.5
SegHRNet (4c)	92.8	87.6	79.3	71.0	78.3	63.4	59.8	42.5	25.7	19.1	18.8	2.7	**56.5**	**49.8**	63.5	44.5	63.7	53.2
SegHRNet (3c)	92.2	86.6	73.1	59.8	74.9	61.3	47.0	33.2	38.3	28.7	7.3	1.4	54.6	47.2	61.5	46.2	60.9	51.3
MFNet	77.2	65.9	67.0	58.9	53.9	42.9	36.2	29.9	12.5	9.9	0.1	0.0	30.3	25.2	30.0	27.7	45.1	39.7
FuseNet	81.0	75.6	75.2	66.3	64.5	51.9	51.0	37.8	17.4	15.0	0.0	0.0	31.1	21.4	51.9	45.0	52.4	45.6
DepthAwareCNN	85.2	77.0	61.7	53.4	76.0	56.5	40.2	30.9	**41.3**	29.3	22.8	**8.5**	32.9	30.1	36.5	32.3	55.1	46.1
RTFNet	93.0	87.4	79.3	70.3	76.8	62.7	60.7	**45.3**	38.5	**29.8**	0.0	0.0	45.5	29.1	**74.7**	**55.7**	63.1	53.2
FuseSeg-161 (Ours)	**93.1**	**87.9**	**81.4**	**71.7**	**78.5**	**64.6**	**68.4**	44.8	29.1	22.7	**63.7**	6.4	55.8	46.9	66.4	47.9	**70.6**	**54.5**

TABLE VIII

COMPARATIVE RESULTS (%) IN DAYTIME AND NIGHTTIME. THE BOLD FONT HIGHLIGHTS THE BEST RESULT IN EACH COLUMN

Methods	Daytime		Nighttime	
	mAcc	mIoU	mAcc	mIoU
FRRN (4c)	42.4	38.0	46.2	42.3
FRRN (3c)	45.1	40.0	41.6	37.3
BiSeNet (4c)	52.9	44.8	53.1	47.7
BiSeNet (3c)	52.1	44.5	50.3	45.0
DFN (4c)	53.4	43.9	57.4	51.8
DFN (3c)	53.7	42.2	52.4	44.6
SegHRNet (4c)	50.0	41.4	50.2	44.9
SegHRNet (3c)	59.7	47.2	55.7	49.1
MFNet	42.6	36.1	41.4	36.8
FuseNet	49.5	41.0	48.9	43.9
DepthAwareCNN	50.6	42.4	50.7	43.2
RTFNet	60.0	45.8	60.7	**54.8**
FuseSeg-161 (ours)	**62.1**	**47.8**	**67.3**	54.6

TABLE IX

NUMBER OF PARAMETERS AND INFERENCE SPEED FOR EACH NETWORK. ms AND FPS REPRESENT MILLISECOND AND FRAMES PER SECOND, RESPECTIVELY

Methods	# params	GTX 1080 Ti		Jetson TX2	
		ms	FPS	ms	FPS
FRRN (4c)	2.4×10^7	10.16	98.39	50.38	19.85
BiSeNet (4c)	8.9×10^7	10.43	95.92	59.21	16.89
DFN (4c)	1.3×10^8	14.87	67.23	81.02	12.34
SegHRNet (4c)	6.6×10^7	36.14	27.67	319.85	3.13
MFNet	7.4×10^5	4.35	229.86	25.74	38.85
FuseNet	4.4×10^7	3.92	255.27	22.25	44.94
DepthAwareCNN	2.1×10^7	3.48	287.32	19.22	52.04
RTFNet	2.5×10^8	29.35	34.07	508.71	1.97
FuseSeg-161 (Ours)	1.0×10^8	33.32	30.01	589.58	1.70

run real-timely on 1080 Ti (i.e., greater than 30 Hz), but most of them cannot run real-timely on TX2. Our FuseSeg reaches only 1.7 Hz on TX2, making it not practical for real-time applications on such low-level computing devices. In addition, it would also be not practical to run our network on the low-cost NVIDIA Jetson Nano and Intel Movidius, which are weaker than TX2 [57]. We think that the double processing (two-branch encoder) of images using complex backbones (e.g., in the table ResNet-152 for RTFNet, DenseNet-161 for ours) might be the major factor leading to the low speed.

D. Qualitative Demonstrations

Fig. 4 shows sample qualitative results for the data fusion networks. We can see that our FuseSeg can provide superior results under various challenging lighting conditions. Specifically, in the second column, two persons behind the far bikes are almost invisible in the RGB image due to the limited dynamic range of the RGB camera, but they can be seen in the thermal image. Our FuseSeg could take advantage of the contributive thermal information to correctly segment the two persons. In the seventh column, the bikes are almost invisible in the thermal image, which may be caused by a similar temperature with the environment. They can be seen a little in the RGB image. Our FuseSeg could make use of the two-modal information and correctly find the three bikes.

We also find that the results of FuseSeg and RTFNet are very close to each other, but FuseSeg performs better because it provides sharper object boundaries, especially in the first column. By comparing FuseSeg and RTFNet, we conjecture that this may be benefited from our connections between

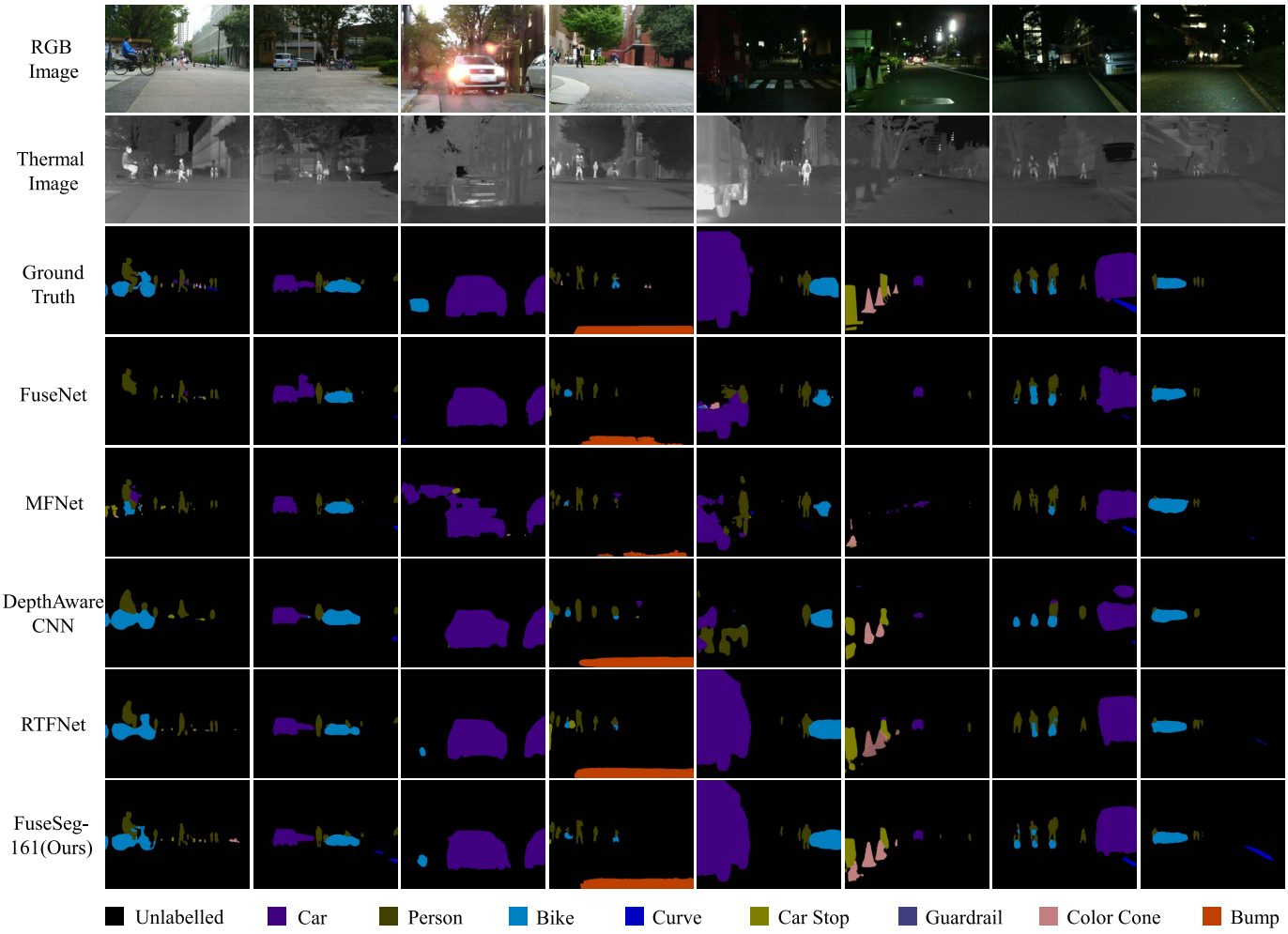


Fig. 4. Qualitative demonstrations for the fusion networks in typical daytime and nighttime scenarios, which are shown in the left four and right four columns, respectively. We can see that our network can provide acceptable results in various lighting conditions. The comparative results demonstrate our superiority. The figure is best viewed in color.

the encoder and the decoder. The detailed spatial information could be retained through the short connections at each level. This can also explain the unsatisfactory performance of FuseNet because both FuseNet and RTFNet have no such short connections. Note that although MFNet has such connections, it still presents inferior performance compared with FuseSeg. We think that the reason may stem from the tiny and weak decoder of MFNet, in which only one convolutional layer is contained at each stage. In addition, they use concatenation for the encoder fusion and summation for the encoder–decoder fusion, which we have proven inferior to our fusion strategy (see the CSF variant in the ablation study). DepthAwareCNN assumes that the pixels on the same object share similar depth (replaced by temperature) values. However, this assumption is violated here, which may explain its inferiority. For example, the temperature of the car in the fifth column does not distribute evenly so that the car cannot be completely segmented.

E. Uncertainty Estimation

Estimating uncertainty for semantic segmentation can help to know how much the predictions could be trusted. It is an important capability to ensure safe decision-making for autonomous vehicles. MC dropout has been successfully employed to infer posterior distributions for the model parameters of Bayesian networks. This article adopts the MC dropout technique for uncertainty estimation [29]. We construct the Bayesian FuseSeg by inserting dropout layers after the initial blocks, max-pooling layers, and No.$1-4$ transition layers of the RGB and thermal encoders. During runtime, we sample the model T times, and here, we set $T = 50$. The uncertainty ζ for each pixel is calculated by

$$\zeta = -\frac{1}{N}\sum_{n=1}^{N} p(l_n|\mathcal{I},\theta)\log p(l_n|\mathcal{I},\theta) \tag{3}$$

where $\mathcal{I}$, θ, and l_n represent the input image, network parameters, and label for the nth class, N is the number of classes ($N = 9$), and $p(\cdot)$ is the average softmax output of the network for each pixel over T times. The uncertainty ζ actually calculates the entropy that measures the disorder of different class probabilities at a pixel [58]. Large entropy means large disorder and, hence, large uncertainty.

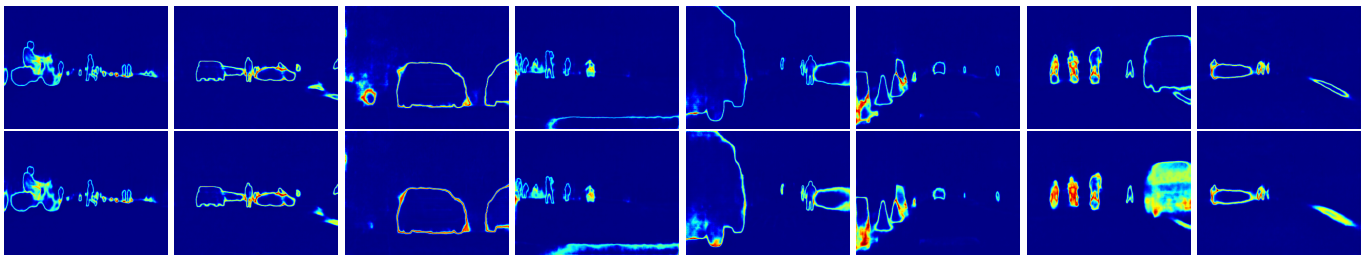


Fig. 5. Uncertainty maps of the Bayesian FuseSeg-161 for the results shown in Fig. 4. The first and second rows are with dropout rates 10^{-4} and 10^{-2}, respectively. Uncertainties increase from blue to red. The figure is best viewed in color.

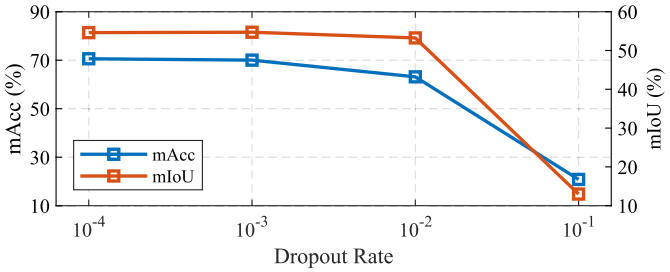


Fig. 6. Performance of Bayesian FuseSeg-161 according to different dropout rates. We find that the semantic segmentation performance severely degrades when the dropout rate is larger than 10^{-2}.

TABLE X

QUANTITATIVE RESULTS (%) ON THE *test* SET OF THE SUN-RGBD V1 DATA SET. BEST RESULTS ARE HIGHLIGHTED WITH BOLD FONT

Methods	mAcc	mIoU
FuseNet	38.0	**29.4**
DepthAwareCNN	36.9	27.2
FuseSeg-161 (Ours)	**38.3**	28.8

Fig. 6 plots the semantic segmentation performance of the Bayesian FuseSeg-161 according to different dropout rates. We find that the performance degrades severely when the dropout rate is larger than 0.01. The reason could be that a large dropout rate could dramatically change the structure of the network and hence severely influence the performance. Fig. 5 shows the uncertainty maps of our Bayesian FuseSeg-161 for different dropout rates. We observe that most of the large uncertainties concentrate on object boundaries. This indicates the ambiguities around the areas where the semantic labels change from one to another. We also find that when the model predicts wrong labels or objects are visually difficult to identify, the uncertainties at these pixels are larger, for example, the left person in the seventh column. Moreover, the uncertainties for the 10^{-2} dropout rate are generally larger than the 10^{-4} dropout rate, indicating that uncertainties increase when the segmentation accuracy decreases.

F. Generalization to RGB-D Data

In order to validate the generalization capability of our FuseSeg, we train and test the networks using the SUN-RGBD v1 scene parsing benchmark data set [30]. We split the data set into the train, validation, and test sets, which account for around 51.14%, 24.43%, and 24.43%, respectively. All the images are resized to 400×528 to increase training efficiency. The thermal images are replaced by the depth images in the experiment. Table X displays the results. We can see that our FuseSeg-161 also achieves better performance, indicating that FuseSeg could generalize well to RGB-D data.

VII. CONCLUSION AND FUTURE WORK

This article proposed a novel deep neural network for RGB and thermal data fusion. We aimed to achieve superior semantic segmentation performance under various lighting conditions, and the experimental results confirmed the superiority over the state of the arts. We performed intensive ablation studies, which showed that the data fusion was a benefit here. The ablation also proved the effectiveness of our network design, including the encoder, decoder, and fusion strategy. We also estimated the uncertainties of our network predictions using the MC dropout technique. As aforementioned, our inference speed on low-level computing platforms, such as NVIDIA TX2, is slow. This may restrict the moving speed of autonomous vehicles that are equipped with such platforms. We consider it as our major limitation. In the future, we would like to boost the runtime speed of the network using weight pruning techniques. We will also design encoder backbones that are more efficient and powerful than the general-purpose backbones for our data fusion network. In addition, the data set that we use is class imbalanced. We will tackle this problem using focal-loss techniques [23] to improve our results.

To enable further studies, we list three promising research directions. First, current fusion operations are not aware of the image quality. For the case that one modal of data is more informative than the other, fusion should give more considerations for the data that are more informative. Thus, how to determine the image quality and smartly do the fusion is an open question. Second, the data set that we use is not recorded as video sequences. We believe that previous frames in a video sequence could provide stronger signals to correct wrong segmentations and lower the uncertainties of the segmentation in the current frame because they are visually similar. Therefore, recording a new data set as video sequences and improving the overall performance of networks given as input more than one image is a research direction. Finally, current low-cost off-the-shelf RGB-D cameras, such as Intel RealSense D435, can work in outdoor environments, so they can be used for autonomous vehicles. Different from thermal imaging cameras that discriminate objects with temperature, depth cameras differentiate objects by the measured pixelwise distances to the camera. They can provide a totally different

modality of information. Therefore, recording a new data set together with an RGB-D camera and a thermal imaging camera, and fusing RGB, thermal, as well as depth images in a network to improve the segmentation performance is also a research direction.

REFERENCES

[1] D. Li and H. Gao, "A hardware platform framework for an intelligent vehicle based on a driving brain," *Engineering*, vol. 4, no. 4, pp. 464–470, Aug. 2018.

[2] P. Cai, X. Mei, L. Tai, Y. Sun, and M. Liu, "High-speed autonomous drifting with deep reinforcement learning," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1247–1254, Apr. 2020.

[3] H. Wang, Y. Sun, and M. Liu, "Self-supervised drivable area and road anomaly segmentation using RGB-D data for robotic wheelchairs," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4386–4393, Oct. 2019.

[4] P. Cai, Y. Sun, Y. Chen, and M. Liu, "Vision-based trajectory planning via imitation learning for autonomous vehicles," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 2736–2742.

[5] H. Chen, C. Xue, S. Liu, Y. Sun, and Y. Chen, "Multiple-object tracking based on monocular camera and 3-D lidar fusion for autonomous vehicles," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2019, pp. 456–460.

[6] X. Wu, Z. Li, Z. Kan, and H. Gao, "Reference trajectory reshaping optimization and control of robotic exoskeletons for human-robot comanipulation," *IEEE Trans. Cybern.*, early access, Aug. 30, 2019, doi: 10.1109/TCYB.2019.2933019.

[7] Z. Li, B. Huang, A. Ajoudani, C. Yang, C.-Y. Su, and A. Bicchi, "Asymmetric bimanual control of dual-arm exoskeletons for human-cooperative manipulations," *IEEE Trans. Robot.*, vol. 34, no. 1, pp. 264–271, Feb. 2018.

[8] Y. Sun, W. Zuo, and M. Liu, "See the future: A semantic segmentation network predicting ego-vehicle trajectory with a single monocular camera," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3066–3073, Apr. 2020.

[9] Y. Sun, L. Wang, Y. Chen, and M. Liu, "Accurate lane detection with atrous convolution and spatial pyramid pooling for autonomous driving," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2019, pp. 642–647.

[10] A. Zaganidis, L. Sun, T. Duckett, and G. Cielniak, "Integrating deep semantic segmentation into 3-D point cloud registration," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 2942–2949, Oct. 2018.

[11] Z. Min, H. Ren, and M. Q.-H. Meng, "Statistical model of total target registration error in image-guided surgery," *IEEE Trans. Autom. Sci. Eng.*, vol. 17, no. 1, pp. 151–165, Jan. 2020.

[12] Y. Sun, M. Liu, and M. Q.-H. Meng, "Improving RGB-D SLAM in dynamic environments: A motion removal approach," *Robot. Auto. Syst.*, vol. 89, pp. 110–122, Mar. 2017.

[13] W. S. Grant, R. C. Voorhies, and L. Itti, "Efficient velodyne SLAM with point and plane features," *Auto. Robots*, vol. 43, no. 5, pp. 1207–1224, Jun. 2019.

[14] J. Cheng, Y. Sun, and M. Q.-H. Meng, "Robust semantic mapping in challenging environments," *Robotica*, vol. 38, no. 2, pp. 256–270, Feb. 2020.

[15] H. Huang, Y. Sun, H. Ye, and M. Liu, "Metric monocular localization using signed distance fields," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 1195–1201.

[16] J. Cheng, Y. Sun, and M. Q.-H. Meng, "Improving monocular visual SLAM in dynamic environments: An optical-flow-based approach," *Adv. Robot.*, vol. 33, no. 12, pp. 576–589, Jun. 2019.

[17] Y. Sun, M. Liu, and M. Q.-H. Meng, "Motion removal for reliable RGB-D SLAM in dynamic environments," *Robot. Auto. Syst.*, vol. 108, pp. 115–128, Oct. 2018.

[18] Q. Ha, K. Watanabe, T. Karasawa, Y. Ushiku, and T. Harada, "MFNet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 5108–5115.

[19] K. Sun *et al.*, "High-resolution representations for labeling pixels and regions," 2019, *arXiv:1904.04514*. [Online]. Available: https://arxiv.org/abs/1904.04514

[20] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Learning a discriminative feature network for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1857–1866.

[21] M. Vollmer *et al.*, *Infrared Thermal Imaging: Fundamentals, Research and Applications*. Berlin, Germany: Wiley, 2017.

[22] X. Sun, H. Ma, Y. Sun, and M. Liu, "A novel point cloud compression algorithm based on clustering," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 2132–2139, Apr. 2019.

[23] P. Yun, L. Tai, Y. Wang, C. Liu, and M. Liu, "Focal loss in 3D object detection," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1263–1270, Apr. 2019.

[24] S. Wang, Y. Sun, C. Liu, and M. Liu, "PointTrackNet: An End-to-End network for 3-D object detection and tracking from point clouds," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 3206–3212, Apr. 2020.

[25] F. Wu, B. He, L. Zhang, S. Chen, and J. Zhang, "Vision-and-Lidar based real-time outdoor localization for unmanned ground vehicles without GPS," in *Proc. IEEE Int. Conf. Inf. Autom. (ICIA)*, Aug. 2018, pp. 232–237.

[26] H. Gao, B. Cheng, J. Wang, K. Li, J. Zhao, and D. Li, "Object classification using CNN-based fusion of vision and LIDAR in autonomous vehicle environment," *IEEE Trans. Ind. Informat.*, vol. 14, no. 9, pp. 4224–4231, Sep. 2018.

[27] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum PointNets for 3D object detection from RGB-D data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 918–927.

[28] I. Bloch, "Information combination operators for data fusion: A comparative review with classification," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 26, no. 1, pp. 52–67, 1996.

[29] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian SegNet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," 2015, *arXiv:1511.02680*. [Online]. Available: https://arxiv.org/abs/1511.02680

[30] S. Song, S. P. Lichtenberg, and J. Xiao, "SUN RGB-D: A RGB-D scene understanding benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 567–576.

[31] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

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

[33] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1520–1528.

[34] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[35] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention—(MICCAI)*. Cham, Switzerland: Springer, 2015, pp. 234–241.

[36] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, "ENet: A deep neural network architecture for real-time semantic segmentation," 2017, *arXiv:1606.02147*. [Online]. Available:https://arxiv.org/abs/1606.02147

[37] 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. 6230–6239.

[38] P. Wang *et al.*, "Understanding convolution for semantic segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1451–1460.

[39] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe, "Full-resolution residual networks for semantic segmentation in street scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3309–3318.

[40] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "ERFNet: Efficient residual factorized ConvNet for real-time semantic segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 263–272, Jan. 2018.

[41] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: Bilateral segmentation network for real-time semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 325–341.

[42] 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.

[43] Y. Sun, M. Liu, and M. Q.-H. Meng, "Active perception for foreground segmentation: An RGB-D data-based background modeling method," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 4, pp. 1596–1609, Oct. 2019.

[44] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "FuseNet: Incorporating depth into semantic segmentation via fusion-based CNN architecture," in *Computer Vision—ACCV*. Cham, Switzerland: Springer, 2017, pp. 213–228.
[45] W. Wang and U. Neumann, "Depth-aware CNN for RGB-D segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 135–150.
[46] Y. Sun, W. Zuo, and M. Liu, "RTFNet: RGB-thermal fusion network for semantic segmentation of urban scenes," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2576–2583, Jul. 2019.
[47] 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.
[48] T. T. D. Pham, S. Kim, Y. Lu, S.-W. Jung, and C.-S. Won, "Facial action units-based image retrieval for facial expression recognition," *IEEE Access*, vol. 7, pp. 5200–5207, 2019.
[49] S. Wang, B. Pan, H. Chen, and Q. Ji, "Thermal augmented expression recognition," *IEEE Trans. Cybern.*, vol. 48, no. 7, pp. 2203–2214, Jul. 2018.
[50] J. S. Yoon *et al.*, "Thermal-infrared based drivable region detection," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 978–985.
[51] M. Knapik and B. Cyganek, "Driver's fatigue recognition based on yawn detection in thermal images," *Neurocomputing*, vol. 338, pp. 274–292, Apr. 2019.
[52] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, Mar. 2010, pp. 249–256.
[53] 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.
[54] S. Zagoruyko and N. Komodakis, "Wide residual networks," 2016, *arXiv:1605.07146*. [Online]. Available: https://arxiv.org/abs/1605.07146
[55] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1492–1500.
[56] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 483–499.
[57] M. Modasshir, A. Quattrini Li, and I. Rekleitis, "Deep neural networks: A comparison on different computing platforms," in *Proc. 15th Conf. Comput. Robot Vis. (CRV)*, May 2018, pp. 383–389.
[58] P.-Y. Huang, W.-T. Hsu, C.-Y. Chiu, T.-F. Wu, and M. Sun, "Efficient uncertainty estimation for semantic segmentation in videos," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 520–535.

Yuxiang Sun (Member, IEEE) received the bachelor's degree from the Hefei University of Technology, Hefei, China, in 2009, the master's degree from the University of Science and Technology of China, Hefei, in 2012, and the Ph.D. degree from The Chinese University of Hong Kong, Hong Kong, in 2017.

He is currently a Research Associate with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong. His current research interests include autonomous driving, deep learning, robotics and autonomous systems, and semantic scene understanding.

Dr. Sun was a recipient of the Best Paper Award in Robotics at the IEEE ROBIO 2019 and the Best Student Paper Finalist Award at the IEEE ROBIO 2015.

Weixun Zuo received the bachelor's degree from Anhui University, Hefei, China, in 2016, and the master's degree from The Hong Kong University of Science and Technology, Hong Kong, in 2017.

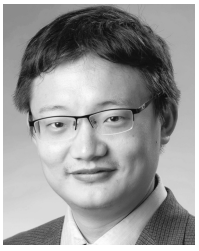
He is currently a Research Assistant with the Department of Electronic and Computer Engineering, Robotics Institute, The Hong Kong University of Science and Technology. His current research interests include mobile robots, semantic segmentation, deep learning, and autonomous vehicles.

Peng Yun received the B.Sc. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2017. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong.

His current research interests include computer vision, machine learning, and autonomous driving.

Hengli Wang received the B.E. degree from Zhejiang University, Hangzhou, China, in 2018. He is currently pursuing the Ph.D. degree with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong.

His current research interests include robot navigation, autonomous driving, computer vision, and deep learning.



Ming Liu (Senior Member, IEEE) received the B.A. degree from Tongji University, Shanghai, China, in 2005, and the Ph.D. degree from ETH Zürich, Zürich, Switzerland, in 2013.

He stayed one year at the University of Erlangen-Nuremberg, Erlangen, Germany, and Fraunhofer Institute IISB, Erlangen, as Visiting Scholar. He is involved in several NSF projects, and National 863-Hi-Tech-Plan projects in China. He is a Principal Investigator of over 20 projects, including projects funded by RGC, NSFC, ITC, SZSTI, and so on. His current research interests include dynamic environment modeling, 3-D mapping, machine learning, and visual control.

Dr. Liu was the General Chair of ICVS 2017, the Program Chair of the IEEE RCAR 2016, and the Program Chair of the International Robotic Alliance Conference 2017.