LiDAR-Based Multi-Task Road Perception Network for Autonomous Vehicles

FUWU YAN1,2,3, KEWEI WANG1,2,3, BIN ZOU1,2,3, LUQI TANG1,2,3, WENBO LI1,2,3, AND CHEN LV4, (Member, IEEE)

1Hubei Key Laboratory of Advanced Technology for Automotive Components, Wuhan University of Technology, Wuhan 430070, China
2Hubei Collaborative Innovation Center for Automotive Components Technology, Wuhan University of Technology, Wuhan 430070, China
3Hubei Research Center for New Energy and Intelligent Connected Vehicle, Wuhan 430070, China
4School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore 639798

Corresponding author: Bin Zou (zoubin@whut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 51975434, in part by the Overseas Expertise Introduction Project for Discipline Innovation under Grant B17034, in part by the Innovative Research Team in University of Ministry of Education of China under Grant IRT_17R83, in part by the Department of Science and Technology, and in part by the Hubei Provincial People’s Government under Grant 2019AEA169.

ABSTRACT For autonomous vehicles, it is an important requirement to obtain integrate static road information in real-time in dynamic driving environment. A comprehensive perception of the surrounding road should cover the accurate detection of the entire road area despite occlusion, the 3D geometry and the types of road topology in order to facilitate the practical applications in autonomous driving. To this end, we propose a lightweight and efficient LiDAR-based multi-task road perception network (LMRoadNet) to conduct occlusion-free road segmentation, road ground height estimation, and road topology recognition simultaneously. To optimize the proposed network, a corresponding multi-task dataset, named MultiRoad, is built semi-automatically based on the public SemanticKITTI dataset. Specifically, our network architecture uses road segmentation as the main task, and the remaining two tasks are directly decoded on a concentrated 1/4 scale feature map derived from the main task’s feature maps of different scales and phases, which significantly reduces the complexity of the overall network while achieves high performance. In addition, a loss function with learnable weight of each task is adopted to train the neural network, which effectively balances the loss of each task and improves performance of the individual tasks. Extensive experiments on the test set show that the proposed network achieves great performance of the three tasks in real-time, outperforms the conventional multi-task architecture and is comparable to the state-of-the-art efficient methods. Finally, a fusion strategy is proposed to combine results on different directions to expand the field of view for practical applications.

INDEX TERMS Autonomous vehicles, dense height estimation, multi-task learning, occlusion-free road segmentation, road topology recognition, 3D LiDAR.

I. INTRODUCTION

Understanding the layout and shape of the road near the ego vehicle is the basis of safe autonomous driving. A comprehensive perception of the surrounding road not only includes the accurate detection of road area, but also involves the global semantic information of road topology, such as the presence and type of an intersection, as it defines the scenario, provides context information and constrains the future motion of traffic participants. Moreover, to facilitate the practical applications in autonomous vehicles, 3D information of the road is also necessary due to the fact that the ground is not always flat. The problem of road perception has been investigated for many years and a large variety of approaches can be found in the literature. As to road areas detection, free space road detection and road boundaries detection are two popular fields. The free space road detection methods [1]–[7] focus on the obstacle-free road region that vehicles could drive on, while its representation form somehow confuses the static road areas and on-road dynamic objects, which is not sufficient for planning in complex driving scenes. The road boundaries [8]–[14] are commonly represented by curves like Bezier Splines and Cubic Splines, which is not capable of describing the complex road shapes such as intersections.
On the other hand, the 3D information of the road is an important factor to control a vehicle. Some researchers [15], [16] pay attention to the lateral and longitudinal road slopes. Others [17], [18] are interested in reconstructing the road surface by various models. At the global level, the road topology recognition methods [19]–[23] aim to understand what type of road the ego vehicle is approaching. However, most of the literature mentioned above focus on only a single aspect of road perception, even if the single tasks are not time-cost, conducting all the necessary tasks for autonomous driving is time consuming.

In recent years, vision-based methods have been significantly benefit from deep learning. However, vision-based road detection suffers from ambiguity caused by the loss of 3D information in real autonomous driving applications. To tackle this, some methods [2], [5], [7], [22], [24] have been proposed to transform LiDAR point cloud to front-view or top-view representation, which could keep the 3D information while take advantage of DCNN technique. In [7], the LiDAR data are represented as a multi-channel 2D signal where the horizontal axis corresponds to the rotation angle and the vertical axis represents channels – laser beams for ground segmentation using a convolutional neural network. In [5], a deep learning approach has been developed to carry out road detection using only LiDAR data of top-view representation. In [24], to enhance the object detection with geometric and semantic priors, the authors firstly predict road height and segmentation in a top-view 3D occupancy grid of LiDAR point cloud. Although the above transformed representations are both able to increase the accuracy and efficiency, a top-view representation is, in our opinion, more appropriate than the front-view representation given that both path planning and vehicle control are easy to conduct on the top-view representation.

In addition, multi-task deep learning [10], [23], [25]–[27] has attracted the attention of researchers due to its potential to boost the performance of each individual task and improve the efficiency of the total network. In RBNet [10], a Bayesian model is implemented, and the RBNet can learn to estimate the road and the road boundary simultaneously. MultiNet [26] combines classification, detection and segmentation in a single architecture, which is based on ResNet [28] and consists of three shared encoding layers followed by task independent decoding layers. Reference [29] introduces an efficient approach for simultaneous object detection, depth estimation and pixel-level semantic segmentation using a shared convolutional architecture. Reference [27] proposes a unified neural network to detect drivable areas, lane lines, and traffic objects simultaneously, in which the three tasks are most important for autonomous driving. Besides the mechanism of parameters sharing, the most challenge problem in multi-task learning is how to balance the loss weight of each task in training phase. Early works always use a weighted sum of the individual task losses, which may even decrease the performance when the loss weights are not properly selected. Recently, [30] proposes a novel and principled multi-task loss to simultaneously learn various classification and regression losses of varying quantities and units using homoscedastic task uncertainty, demonstrating the superiority and effectiveness of homoscedastic uncertainty for multi-task learning.

To cope with the challenge road perception problems mentioned above and obtain an integrated road areas detection despite occlusion, as well as 3D information and the high level understanding of the road types, we proposed a multi-task network performing occlusion-free road segmentation (ORS), dense road ground height estimation (DHE) and road topology recognition (RTR) simultaneously, as shown in Figure 1. The proposed multi-task network named LMRoadNet, taking the top-view representation of LiDAR point cloud.

The main contributions of our work are as follows:
1) A lightweight and efficient multi-task road perception network is proposed to perform occlusion-free road segmentation, dense road height estimation and road topology recognition simultaneously in real-time, the architecture of which is elaborately designed to obtain a trade-off between accuracy and runtime.
2) To train and test the network, a multi-task road perception dataset denoted MultiRoad was built semi-automatically based on the public SemanticKITTI dataset.
3) A fusion strategy is proposed to combine the results of the proposed model on different directions, which could flexibly and efficiently expand the field of view depending on the complexity of the driving scenario and onboard computing power in real application.

The remainder of this paper is organized as follows. Section II describes the overview of our method for multi-task road perception. Section III describes the procedure of building the multi-task dataset in detail. Section IV evaluates the proposed method through comprehensive experiments. Section V summarizes the contributions of this paper.

II. METHODOLOGY
A. TASK DEFINITION
In our work, we perform three related and complementary tasks from single LiDAR input in road perception field. The first task is occlusion-free road segmentation (ORS), which aims to obtain full road area despite the occlusion. The second task is dense road height estimation (DHE), which estimates each road cell’s height in the grid map. The third task intends to understand the global property of the road shape, named road topology recognition (RTR). The input is a grid-based top-view representation of the unstructured point cloud. In our work, 5 basic statistics are computed for each cell in the grid $\text{Grid} \in \{(N_i, z_{\text{min}}, z_{\text{max}}, z_{\text{mean}}, I_i) | 0 < i \leq NG\}$, where $N_i$ is the number of points, $z_{\text{min}}, z_{\text{max}}, z_{\text{mean}}$ the minimum, mean, and maximum height of points, $I_i$ the mean reflectivity of points in i-th cell, $NG$ the number of cells in the grid. The output of the network $\text{Grid} \in \mathbb{R}^{H \times W \times 5}$, the output of the ORS task $RS \in \mathbb{R}^{H \times W \times 2}$, the output of the DHE task $DH \in \mathbb{R}^{H \times W \times 1}$, the output of the RTR task $RT \in \mathbb{R}^{7}$, where $H$ the height of the grid and $W$ its width.
B. NETWORK ARCHITECTURE

With the purpose of efficiency and effectiveness in mind, the proposed network named LMRoadNet was designed to get the best possible trade-off between accuracy and runtime. The network jointly reasons about the ORS, DHE and RTR tasks. LMRoadNet can be trained end-to-end and joint inference over all tasks in less than 15ms. Unlike usual multi-task learning architecture that contains an encoder and some task-specific decoders, the proposed architecture decodes the other tasks on an intermediate scale feature map comes from a main task’s feature maps of different scales and phases, which significantly reduces number of parameters and computational complexity while retains high accuracy. The proposed model is visualized in Figure 2.

1) OCCLUSION-FREE ROAD SEGMENTATION

In our work, the ORS task is chose as the main task due to two aspects. One is that the ORS task extracts more detailed features in the process. Another is that the other two tasks are highly related with and could benefit from the ORS task. To make it more efficient to capture useful features, we use MixNet [31] as the backbone, whose basic component is mixed depth-wise convolution (MixConv) that naturally mixes up multiple kernel sizes in a single convolution, such that it can easily capture different patterns with various resolutions. To further reduce the compute complexity, only the first 4 stage of the MixNet is used to get a 1/16 scale feature map.

To performing the ORS task based on the backbone, we proposed a Joint Up-sampling Module (JUM) that performing up-sampling on two feature maps from different stages. The feature map from earlier stage with bigger resolution and fewer channels carries sufficient details in spatial and the feature map from later stage with smaller resolution and more channels contains necessary facts in context. Thus the JUM takes advantage of both detailed and global information to infer not only the free road cells but also the occupancy and occluded road cells. As shown in Figure 2, the JUM up-samples the small scale feature map to that of the larger resolution features, followed by an $1 \times 1$ convolutional layer. The larger resolution feature map takes a $1 \times 1$ convolution before the two branch concatenated together. Then a Squeeze-and-Excitation block (SE) [32] is used to adaptively recalibrate channel-wise feature responses by explicitly modelling interdependencies between channels, followed by two $3 \times 3$ convolutional layers. The JPU works in a cascading way to obtain 1/2 scale feature map, and finally a deconvolution layer is used to transform the features to the original resolution.

2) DENSE ROAD HEIGHT ESTIMATION

To efficiently utilize the features from the main task, the DHE task performs height estimation on the concatenated features come from a Feature Sharing Module (FSM), which merges features of different scales and stages of the main task and could provide suitable features for the other tasks. For that reason, only a proper simple decoder is able to obtain the accurate estimation of the height of each cell. As shown in
figure 2, the feature sharing module consist of four inputs $F_1$, $F_2$, $F_3$, $F_4$. $F_1$ and $F_2$ are the 1/4 and 1/16 scale features in encoder phase, $F_3$ is the 1/4 scale features in decoder phase and $F_4$ is the final predicted road segmentation mask. All the features are transformed to 1/4 scale with proper operations before concatenation if they are not with the same resolution, for example, $F_2$ is processed with up-sampling and 1×1 convolution, $F_4$ with max-pooling. The DHE decoder firstly uses 1×1 convolutional layer to capture more task-specific useful features, and then up-samples the features to original resolution. Finally, two 1×1 convolutional layer with 32 and 1 channels respectively are applied to obtain the height estimation result.

3) ROAD TOPOLOGY RECOGNITION
The RTR task shares the same feature sharing module with the DHE task. It also takes a 1×1 convolutional layer to extract task-specific features firstly, an average pooling layer is adopted then, after which two 1×1 convolutional layers are used as fully convolutional layers to obtain prediction of the road topology types. The final output channel is 7, as to the number of classes of the road topology.

C. LOSS FUNCTION
For the ORS and RTR task, we use the cross-entropy loss for these pixel-level and image-level classification tasks, denoted $L_{ors}$ and $L_{rtr}$ respectively. For the DHE task, the L1 loss is applied on the road area, the non-road area are ignored in training phase, the loss is referred to $L_{dhe}$.

$$L_{dhe}(p, q) = \frac{1}{N} \sum_{i}^{N} \| p_i - q_i \|_1$$

where, $L_{dhe}$ is the loss of DHE task, $N$ the number of valid cells belong to road area, $p$ the prediction of height, $q$ the ground truth height, $i$ is the cell index in the grid.

For multi-task learning, it is essential to appropriately balance the loss of each task to obtain a total loss when training the network jointly. Usually, the total loss is a weighted linear sum of the losses for each individual task, as (2), which is refer to fixed loss weight strategy in the paper. However, it is expensive and difficult to manually searching for an optimal weighting. Thus, instead of manually tuning the weight of each task to account for the differing variances and offsets amongst the single-task losses, we follow [30] to add the loss weight of each task to the learnable network parameters in our work, as (3). To avoid gradient exploding due to a potential division by zero, our network predicts $s = \log \sigma^2$ instead of $\sigma$ in practice, the total loss is defined as (4), which is applied in the learnable loss weight strategy. In practice, the loss weight of RTR task doesn’t converge as the other two tasks, owing to that the RTR task is a global classification task that should be better to optimize later after the other tasks. However, the rapid change of the loss weight of the RTR task in the last training phase hurts the accuracy of the other tasks, so we fix the weights of the tasks in the last few epochs, which makes
the training results more robust.
\[
L_{\text{total}} = \sum_i \lambda_i L_i \quad (2)
\]
where, \( L_i \) is the loss of \( i \)-th task, \( \lambda_i \) its loss weight.
\[
L_{\text{total}} = \frac{1}{\sigma_i^2} L_{\text{ors}} + \frac{1}{2} \log \sigma_i^2 + \frac{1}{2} \log \sigma_d^2 + \frac{1}{2} \log \sigma_{\text{dhe}}^2 + \frac{1}{2} \log \sigma_{\text{attr}}^2 + \frac{1}{2} \log \sigma_{\text{refl}}^2 \quad (3)
\]
where, \( \sigma_i \) is the task loss uncertainty, \( i=1,2,3 \).
\[
L_{\text{total}} = e^{-s_1} L_{\text{ors}} + 0.5s_1 + 0.5e^{-s_2} L_{\text{dhe}} + 0.5s_2 + e^{-s_3} L_{\text{attr}} + 0.5s_3 \quad (4)
\]
where, \( s_i \) is the learnable parameter of each task to balance the losses of the individual tasks.

### III. DATASET DESCRIPTION

#### A. BUILDING DATASET

To training and test our algorithm, we built a multi-task dataset named MultiRoad based on the SemanticKITTI dataset [33], which provides dense semantic annotations for each individual LiDAR scan of sequences 00-10 of the odometry task of the large-scale KITTI Vision Benchmark [2]. It enables the usage of multiple sequential scans for semantic scene interpretation, like semantic segmentation and semantic scene completion. The labeled point clouds are recorded at a rate of 10 Hz in sequences, which enables the usage of temporal information for semantic scene understanding and aggregation of information over multiple scans.

Instead of the unstructured point cloud, we transformed it to a grid-based top-view representation for the proposed multi-task learning method to utilize the powerful convolutional natural network in computer vision field. The first step of the procedure is to create a grid in the \( x-y \) plane of the LiDAR and to assign each element of the point cloud to one of its cells. The grid covers a region which is 30 meters wide, \( y \in [-15, 15] \), and 46 meters long, \( x \in [0, 46] \); its cells are squares of size 0.10 \( \times \) 0.10 meters. Some basic statistics are then computed for each grid cell: number of points; mean reflectivity; minimum, mean, and maximum height of points in the cell. Finally, five images, one for each of the above statistics, are generated by viewing the grid cells as pixels. Given the chosen cell size and grid range, these top-view images have a resolution of 460 \( \times \) 300 pixels.

We labeled the samples with an interval of 20 frames. There will be only a few empty cells containing none points in the grid when generated using the aggregated multiple scans. For the annotation of the ORS task, the course annotations were obtained from aggregated multiple scans around the selected scan frame. The cells containing points of road class are assigned to the road category and other cells to the none-road category. Then we manually refined the road segmentation annotation by complementing the empty road cells (road cells containing none points) and occupied road cells (road cells occluded by on-road objects).

For the annotation of DHE task, the key idea is to find the seed cells and derive the height value of other cells from its spatially neighboring seed cells. The seed cell is defined as there are high probability of none objects standing on the ground, whose height could be directly assigned to the mean height of the points in the cell. To find the seed cells, the feature of height difference \( \Delta H_i \) and the density \( DS_i \) of the cell is used to calculate the probability.

The height difference \( \Delta H_i \) is defined as:
\[
\Delta H_i = z_{i}^{\text{max}} - z_{i}^{\text{min}} \quad (5)
\]
where, \( z_{i}^{\text{max}} \) is the maximum height of the points in the cell and \( z_{i}^{\text{min}} \) the minimum height.

The density of the cell is obtained by normalizing the number of points in the cell:
\[
DS_i = \min(1.0, \frac{\log(N_i + 1)}{\log 150}) \quad (6)
\]
where, \( N_i \) is the number of points in the cell.

The two features are transformed to probabilities of free road cell \( p_1(C_i) \) and \( p_2(C_i) \) by Gauss kernels(4).

\[
p_1(C_i) = \exp\left(-\frac{(\Delta H_i - \mu_H)^2}{2\sigma_H^2}\right) \quad (7)
\]
where, \( \Delta H_i \) is the height different of the cell, \( \mu_H \) its standard deviation, and \( \sigma_H^2 \) its variance.

\[
p_2(C_i) = \exp\left(-\frac{(\Delta S_i - \mu_DS)^2}{2\sigma_{DS}^2}\right) \quad (8)
\]
where, \( DS_i \) is the density of the cell, \( \mu_DS \) its standard deviation and \( \sigma_{DS}^2 \) its variance.

Then they are fused by binary Bayes filter to obtain the final probability of the cell being a seed cell \( p(C_i) \).

\[
p(C_i) = \text{odds}^{-1}\left(\prod_{i=1}^{2} \frac{p(C_i)}{1-p(C_i)}\right) \quad (9)
\]
where, \( \text{odds}^{-1} \) is inverse odds function.

In practice, the cell with probability higher than 0.7 is set to seed cell. The height of other cells is calculated by a weighted sum of the height of the surrounding seed cells.

\[
DH(C_i) = \left\{ \begin{array}{ll}
\sum_{C_j \in NE_i} \alpha_{i,j} \cdot z_j^{\text{mean}}, & \text{if } p(C_i) > 0 \\
0, & \text{otherwise}
\end{array} \right. \quad (10)
\]
where, \( p(C_i) \) is the probability of the cell being seed cell, the \( z_j^{\text{mean}} \) the mean height of the points in the cell, \( NE_i \) the set of surrounding seed cells of \( C_i \) and \( \alpha_{i,j} \) the weight of each seed cell.

Each surrounding seed cell’s weight is measured in distance and probability in Gaussian metrics, as followed.

\[
\alpha_{i,j} = \exp\left(-\frac{(d_{i,j} - \mu_d)^2}{2\sigma_d^2}\right) \cdot \exp\left(-\frac{(p(C_i) - \mu_p)^2}{2\sigma_p^2}\right) \quad (11)
\]
where, \( d_{i,j} \) the distance between the target and seed cell, \( \mu_d \) its standard deviation and \( \sigma_d^2 \) its variance, \( p(C_i) \) the probability of the cell being seed cell, \( \mu_p \) its standard deviation and \( \sigma_p^2 \) its variance.
F. Yan et al.: LMRoadNet for Autonomous Vehicles

FIGURE 3. Procedure of the proposed multi-task dataset. (a) aggregation of point cloud over multiple scans with labels encoded in different colors. (b) extracted road points. (c) raw annotation of the road segmentation task. (d) fine annotation of the road segmentation task. (e) raw annotation of the dense road height task. (f) fine annotation of the dense road height task. (g) annotation of the road topology recognition task.

Specially, we use a dynamic neighbor region for the non-seed cells, which could handle the sparsity of the non-empty cells in distant area. In details, we initialize the neighbor region radius $\varepsilon$ as 5 cells, and expand the radius until the sum of the number of points of the seed cells in such neighbor exceed the defined threshold $T_n$.

$$NE_i = \{C_j | d_{i,j} < \varepsilon, p(C_j) > 0.7\}$$  \hspace{1cm} (12)

where, $d_{i,j}$ is the distance between the target and seed cell, $\varepsilon$ the dynamic threshold of the region radius, which is calculated by:

$$\varepsilon = \min(T_\varepsilon, \arg\min_\varepsilon \text{Card}(NE_i) > T_n)$$  \hspace{1cm} (13)

where, $T_\varepsilon$ is minimum the threshold of $\varepsilon$, $T_n$ the minimum threshold of the number of seed cells of its neighbor.

The total procedure could refer to Algorithm 1.

For the annotation of RTR task, we classified the common road layouts to 7 categories, straight road, left turn, right turn, left side road, right side road, T intersection and crossroad, as shown in Figure 3 (g). The samples are manually annotated with road topology ground truth labels in the view of region of 0 to 46 meters in front of the ego vehicle.

Finally, we got 912 train samples and 320 test samples. Each sample contains a grid-based LiDAR Representation, a road segmentation label, a dense road height label and a road topology label.

B. DATA PREPROCESS AND AUGMENTATION

In the training and test phase, the samples are preprocessed before feed to the network. The number of points $N_i$ is transformed to density feature by (14). The height values are normalized to $[-1, 1]$ by (15), thus the predicted height value in test phase should multiplied by 5 to obtain the real height value. In the training phase, the training data is augmented with random flipping left to right technique, which extends the training samples and reduces overfitting of the network.

$$density = \min(1.0, \frac{\log(N_i + 1)}{\log 20})$$  \hspace{1cm} (14)

$$f(x) = \begin{cases}  \min(0.2x, 1) & \text{if } x = 0 \\ \max(0.2x, -1) & \text{otherwise} \end{cases}$$  \hspace{1cm} (15)

IV. EXPERIMENTS AND RESULTS

In this section, we evaluate the effectiveness of the proposed LMRoadNet on the proposed multi-task dataset MultiRoad. All the experiments are trained on the train set and evaluated on the test set. The results on the tasks of ORS, DHE and RTR are reported in both qualitative and quantitative aspects, the quantitative evaluation uses interpretable error metric for each task. We also report various ablation studies to shed light on the effects of the design decisions. In addition, we compare with other state-of-art efficient methods in the terms of accuracy and runtime. Finally, we show the strategy of extending
Algorithm 1 Computing Dense Road Height Ground Truth

Input: Grid \(\{(N_i, z^\text{max}_{i1}, z^\text{max}_{i2}, z^\text{mean}_{i1}, I_i)|0 < i \leq UNG\}\) 
Output: Dense Road Height \(DH\)

1: for \(i = 1\) to \(NG\) do 
2: \(\Delta H_i = z^\text{max}_{i1} - z^\text{min}_{i1}\) 
3: \(DS_i = \min(1.0, \frac{\log(N_i+1)}{\log 150})\) 
4: \(p_1(C_i) = \exp(-\frac{(\Delta H_i - \mu)^2}{2\sigma^2})\) 
5: \(p_2(C_i) = \exp(-\frac{(DS_i - \mu)^2}{2\sigma^2})\) 
6: \(p(C_i) = \frac{1}{\sum_{i=1}^{N_i} p_i(C_i)}\) 
7: end for 

8: for \(i = 1\) to \(NG\) do 
9: if \(p(C_i) > 0.5\) then 
10: \(DH(C_i) = z^\text{mean}_{i1}\) 
11: else 
12: Find the valid neighbor cells \(NE_i = \{C_j|d_{ij} < \varepsilon, p(C_j) > 0.7\}\) 
13: \(\alpha_{i,j} = \exp(-\frac{(d_{ij} - \mu)^2}{2\sigma^2}) \cdot \exp(-\frac{(p(C_j) - \mu)^2}{2\sigma^2})\) 
14: \(DH(C_i) = \sum_{C_j \in NE_i} \alpha_{i,j} \cdot z^\text{mean}_{j}\) 
15: end if 
16: end for

the field of view by directly acting on and fusing different directions simultaneously.

A. EVALUATION METRICS
For quantitative evaluation of the ORS task, precision (PRE), recall (REC), F1 score and average precision (ACC), were used as the metrics. The metrics are calculated as in (16)-(19), where TP, TN, FP, FN are, respectively, the number of true positives, true negatives, false positives, and false negatives at the pixel level. For quantitative evaluation of the DHE task, the L1 error is used as the metric, and only the cells in road area are included. For quantitative evaluation of the RTR task, we use mean intersection-over-union (mIoU) as the metric. Our experiments considered an assessment that demonstrates the effectiveness of our approach for multi-task road perception.

\[
\begin{align*}
ACC &= \frac{TP + TN}{(TP + FN + FP + TN)} \quad (16) \\
PRE &= \frac{TP}{(TP + FP)} \quad (17) \\
REC &= \frac{TP}{(TP + FN)} \quad (18) \\
F1 &= \frac{2 \cdot PRE \cdot REC}{(PRE + REC)} \quad (19)
\end{align*}
\]

B. IMPLEMENTATION DETAILS
We perform all our experiments in PyTorch [34] with CUDA 10.0 and cuDNN backend. All experiments are run on a single NVIDIA GTX-1080Ti GPU. Due to GPU memory limitation, we have a maximum batch size of 4. During optimization, we use the SGD optimizer with a weight decay of 0.0001 and momentum of 0.9. The learning rate is set using poly strategy with start value of 0.01 and power of 0.9. The total epochs are set to 80. When performing multi-task training with learned loss weights of each task, the learned loss weights are fixed in the last 20 epochs to obtain a more robust training results. For fair comparison of runtime and accuracy with other state-of-art efficient models, we re-implement them in PyTorch. We average the forward passes cost time of total 320 test samples and transform the results to frame per second (FPS) as the runtime metric.

C. EXPERIMENTAL RESULTS OF THE PROPOSED LMRoadNet ALGORITHM
The performance of the network was evaluated on the test set using metrics for each task as described above. A summary of the results is given in Table 1, which shows the single-task baseline in the first part and the multi-task results in the second part. As shown in Table 1, our method achieves significant gains over the single-task baseline, illustrating that the tasks are benefit from each other when trained jointly. When trained in duple-task mode, the performance of the ORS and DHE task are both improved as that they are more complementary tasks, as well as the ORS and RTR task. However, the DHE and RTR task don’t follow this trend due to that the features they need are not related so much. When trained in triple mode, all the tasks obtain performance gains compared to single-task baselines, especially the RTR task improves 27.5% of mIoU. Qualitative results are shown in Figure 4, where we observe high quality results of the triple tasks on different situations. In the first row of the figure, red denotes false negatives, blue areas correspond to false positives and green represents true positives. The samples show that the road region is well detected in the ORS task even in occupancy road area that occluded by on-road objects (e.g., vehicles, pedestrians, etc.) and empty road area due to occlusion or sparsity of point cloud. The second row of the figure visualizes the results of the DHE task, in which different colors encode different heights in vehicle coordinate. The estimated height values are very smooth and consistent with actual road conditions. The last row in the figure are the predictions of road topology, showing that our method is capable of handling various driving scenes.

D. COMPARISON OF DIFFERENT LOSS WEIGHTS STRATEGIES
In this experiment, we trained the proposed LMRoadNet with different loss functions varies in terms of whether the loss...
As shown in Table 2, when trained with learnable loss weights strategy, performance of the proposed model improves 1.6% of ACC and 3.4% of F1 score in the ORS task, decreases 8.8 cm of L1 error in the DHE task and increases 13.3% of mIoU in the RTR task over the model trained with fixed weight of each task is learnable or fixed in training phase. As shown in Table 2, when trained with learnable loss weights strategy, performance of the proposed model improves 1.6%
loss weights. Moreover, referring to Table 1, in the fixed loss weights training mode, inappropriate loss weights may hurt the performance of the tasks even worse than that of single-task baselines.

E. COMPARISON OF DIFFERENT VARIANTS OF MULTI-TASK ARCHITECTURE

To evaluate the efficiency of our multi-task network architecture, we implement two variants of the network and compare their performance on the proposed MultiRoad dataset. The first variant shares a same encoder and three task-specific decoders as the conventional multi-task architecture, which is referred to base-1. The decoder of the DHE task is almost the same as the ORS task except the output channel. To obtain a larger receptive field for the RTR task, the branch of the task adopts the fifth phase of MixNet before the decoder. The other variant, referred to base-2, drops the last two feature maps in the decoder phase as to LMRoadNet. The results are shown in Table 3. Compared to base-1, the proposed LMRoadNet dramatically decreases the number of parameters and GFLOPs while significantly increase the performance of the tasks, indicating that our network architecture is superior to the conventional multi-task architecture. Base-2 saw a significant performance decrease of the DHE and RTR task, demonstrating that the last two feature maps of the main task are beneficial to the two tasks, which proves the effectiveness of our approach.

F. COMPARISON WITH OTHER STATE-OF-ART EFFICIENT METHODS

For fair comparison of accuracy and runtime with other state-of-art efficient models on the same tasks, we re-implement them in PyTorch, and train and evaluate them on the proposed multi-task dataset. In [5], a deep learning approach named LoDNN has been developed to carry out road detection using only LiDAR data that encoded several basic statistics in top-view representation, which is very similar with our ORS task except that our method aims to detect full road area despite occlusion. Reference [24] shows that road priors (e.g., road segmentation and ground height) can boost the performance.
and robustness of modern 3D object detectors and propose a map prediction module that predict the road segmentation and estimate ground height separately with a same network, which are refer to UNet-seg and UNet-hei respectively in our paper. In order to take advantage of both LiDAR and camera sensors for lane detection in 3D space, [35] proposes a fast convolutional neural net (CNN) based on ResNet50 [28] to predict a dense ground height from the LiDAR input.

As shown in Table 4, our model has more competitive performance on the ORS and DHE tasks considering the significant fewer parameters and GFLOPs than the baseline models.

With regard to the ORS task, our model outperforms UNet-seg by 0.3% of ACC and 0.6% of F1 score, and LoDNN by 0.5% of ACC and 0.9% of F1 score. The qualitative results of these models on the test set are shown in Figure 5, from which we can see that LMRoadNet detected road area more precisely both in overall and detailed aspect. Moreover, the performance of the models on different distance from the ego vehicle are calculated and shown in Table 5. The performance gaps between LMRoadNet and the baseline models become larger and larger as distance increases, indicating that the proposed LMRoadNet outperforms the others in the difficult distant area where the LiDAR points become more and more sparse.

As to the DHE task, although our model is a little lower in performance, it is competitive to the other models given that the parameters and GFLOPs of our model is dramatically smaller, as shown in Table 4. In addition, the models shares the same trend of the ORS task as the distance increases as can be seen from Figure 6, the proposed model achieves amazing less than 6 cm of L1 error in within 30 meters.

### G. EXPANDING FIELD OF VIEW IN REAL APPLICATION

In practice, the mere view in front of the ego vehicle may not be sufficient for complex autonomous driving tasks. Therefore, we propose a fusion strategy to process multiple grids

| range (m) | 0-16 | 16-26 | 26-36 | 36-46 |
|-----------|------|-------|-------|-------|
| UNet-seg  | 96.1 | 94.9  | 92.5  | 88.4  |
| LoDNN     | 95.4 | 94.7  | 92.1  | 88.8  |
| LMRoadNet | 95.8 | 95.4  | 93.2  | 90.7  |

**TABLE 5. Evaluation results of models with different distance.**
in different directions. Compared to directly processing a larger grid, this fusion strategy can not only effectively and flexibly expand the field of view, but also avoid processing some distant areas that require less attention. Specifically, it takes about 42ms when processing a 920 × 920 grid and only 26ms when processing four 460 × 300 grids in the front, back, left and right directions, gaining a speed boost of nearly 1.5×. The far corner areas contain vast empty cells that could be ignored in most of the time. Moreover, fusion the results from different predictions in the cross region may boost the performance. In the cross regions, for the ORS task, a binary Bayesian filter is applied to fuse predictions of different directions; for the DHE task, we use the average of estimates in each direction as the final result. Figure 7 show the combined detection results of some samples in two directions of the front and back and in four directions of front, back, left and right, respectively. It can be seen that our model performs well on different directions and obtain a larger field of view around the ego vehicle, which accurately represents the surrounding road both in semantic and geometry. Some unstructured road are well detected that shows the effectiveness of the proposed method. The field of view could be expanded depending on the computing power of the on-board processor and the complexity of the environment, which is highly flexibly and effective for practical autonomous driving applications.

V. CONCLUSIONS

In the work, we propose a dedicated multi-task algorithm named LMRoadNet for comprehensive road perception from a single LiDAR swap in real time. The tasks cover both semantic and geometry information of the road surrounding the ego vehicle, which are essential for autonomous driving. After jointly optimized on the proposed corresponding multi-task dataset denoted MultiRoad using a combined loss function with learnable loss weights of each task, the LMRoadNet achieves 97.4% of accuracy and 94.2% of F1 score on the ORS task, 6.4 cm of L1 error on the DHE task and 84.1% of mIoU on the RTR task. The experiment results show that the LMRoadNet is capable of detecting the integrate road region despite occlusion, estimating the dense road height accurately and recognizing the type of the road in various driving scenes. In addition, substantial experiments demonstrating that our elaborated designed architecture outperforms the conventional multi-task architecture and is comparable to state-of-the-art efficient models. More importantly, the high effectiveness and real-time performance of the proposed model and the fusion strategy to efficiently and flexibly expand field of view make our method greatly applicable to practical autonomous driving.

REFERENCES

[1] L. Xiao, B. Dai, D. Liu, D. Zhao, and T. Wu, “Monocular road detection using structured random forest,” Int. J. Adv. Robotic Syst., vol. 13, no. 3, p. 101, Jun. 2016.

[2] K. Liu, W. Wang, R. Tharmarasa, J. Wang, and Y. Zhuo, “Ground surface filtering of 3D point clouds based on hybrid regression technique,” IEEE Access, vol. 7, pp. 23270–23284, 2019.

[3] X. Song, T. Rui, S. Zhang, J. Fei, and X. Wang, “A road segmentation method based on the deep auto-encoder with supervised learning,” Comput. Electr. Eng., vol. 68, pp. 381–388, May 2018.

[4] S. Gu, Y. Zhang, X. Yuan, J. Yang, T. Wu, and H. Kong, “Histograms of the normalized inverse depth and line scanning for urban road detection,” IEEE Trans. Intell. Transp. Syst., vol. 20, no. 8, pp. 3070–3080, Aug. 2019.

[5] L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde, “Fast LiDAR-based road detection using fully convolutional neural networks,” in Proc. IEEE Intell. Vehicles Symp. (IV), Jun. 2017, pp. 1019–1024.

[6] G. L. Oliveira, W. Burgard, and T. Brox, “Efficient deep models for monocular road segmentation,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Oct. 2016, pp. 4885–4891.

[7] M. Velas, M. Spanel, M. Hradis, and A. Herout, “CNN for very fast ground segmentation in Velodyne LiDAR data,” in Proc. IEEE Int. Conf. Auto. Robot Syst. Competitions (ICARSC), Apr. 2018, pp. 97–103.

[8] D. Zai, J. Li, Y. Guo, M. Cheng, Y. Lin, H. Luo, and C. Wang, “3-D road boundary extraction from mobile laser scanning data via supervoxels and graph cuts,” IEEE Trans. Intell. Transp. Syst., vol. 19, no. 3, pp. 802–813, Mar. 2018.

[9] K. Mano, H. Masuzawa, J. Miura, and I. Ardiyanto, “Road boundary estimation for mobile robot using deep learning and particle filter,” in Proc. IEEE Int. Conf. Robot. Biomimetics. New York, NY, USA: IEEE, Dec. 2018, pp. 1545–1550.

[10] Z. Chen and Z. Chen, “RBNet: A deep neural network for unified road and road boundary detection,” in Neural Information Processing, Cham, Switzerland: Springer, 2017, pp. 677–687.

[11] J. Zhao, X. He, J. Li, T. Feng, C. Ye, and L. Xiong, “Automatic vector-based road structure mapping using multibeam LiDAR,” Remote Sens., vol. 11, no. 14, p. 1726, 2019.

[12] P. Sun, X. Zhao, Z. Xu, R. Wang, and H. Min, “A 3D LiDAR data-based dedicated road boundary detection algorithm for autonomous vehicles,” IEEE Access, vol. 7, pp. 29623–29638, 2019.

[13] T. Suleymanov, P. Amayo, and P. Newman, “Inferring road boundaries through and despite traffic,” in Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC), Nov. 2018, pp. 409–416.

[14] G. Wang, J. Wu, R. He, and S. Yang, “A point cloud-based robust curb detection and tracking method,” IEEE Access, vol. 7, pp. 24611–24625, 2019.

[15] E. Ustunel and E. Masazade, “Vision-based road slope estimation methods using road lines or local features from instant images,” IET Intell. Transp. Syst., vol. 13, no. 10, pp. 1590–1602, Oct. 2019.

[16] X. Liao, Q. Huang, D. Sun, W. Liu, and W. Han, “Real-time road slope estimation based on adaptive extended Kalman filter algorithm with in-vehicle data,” in Proc. 29th Chin. Control Decis. Conf. (CCDC), May 2017, pp. 6689–6894.

[17] B. Jia, J. Chen, and K. Zhang, “Drivable road reconstruction for intelligent vehicles based on two-view geometry,” IEEE Trans. Ind. Electron., vol. 64, no. 5, pp. 3696–3706, May 2017.

[18] R. Fan, X. Ai, and N. Dahoun, “Road surface 3D reconstruction based on dense subpixel disparity map estimation,” IEEE Trans. Image Process., vol. 27, no. 6, pp. 3025–3035, Jun. 2018.

[19] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, “3D traffic scene understanding from movable platforms,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 36, no. 5, pp. 1012–1025, May 2014.

[20] Q. Li, L. Chen, M. Li, Q. Zhu, Q. Zhang, and S. S. Ge, “Intersection detection and recognition for autonomous urban driving using a virtual cylindrical scanner,” IET Intell. Transp. Syst., vol. 8, no. 3, pp. 244–254, May 2014.

[21] A. Y. Hata, D. Habermann, F. S. Osorio, and D. F. Wolf, “Road geometry classification using ANN,” in Proc. IEEE Intell. Vehicles Symp., Jun. 2014, pp. 1319–1324.

[22] U. Baumann, Y.-Y. Huang, C. Glaser, M. Herman, H. Banzhaf, and J. M. Zollner, “Classifying road intersections using transfer-learning on a deep neural network,” in Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC), Nov. 2018, pp. 683–690.

[23] M. Oeljeklaus, F. Hoffmann, and T. Bertram, “A combined recognition and segmentation model for urban traffic scene understanding,” in Proc. 20th Int. Conf. Intell. Transp. Syst. (ITSC), Oct. 2017, pp. 1–6.

[24] B. Yang, M. Liang, and R. Urtasun, “HDNET: Exploiting HD maps for 3D object detection,” in Proc. 2nd Conf. Robot Learn., Proc. Mach. Learn. Res., 2018, pp. 146–155. [Online]. Available: http://proceedings.mlr.press

[25] D. O. Pop, A. Rogozan, C. Chatelain, F. Nashashibi, and A. Benshahir, “Multi-task deep learning for pedestrian detection, action recognition and time to cross prediction,” IEEE Access, vol. 7, pp. 149318–149327, 2019.
[26] M. Teichmann, M. Weber, M. Zollner, R. Cipolla, and R. Urtasun, “Multi-Net: Real-time joint semantic reasoning for autonomous driving,” in Proc. IEEE Intell. Vehicles Symp. (IV), Jun. 2018, pp. 1013–1020.

[27] Y. Qian, J. M. Dolan, and M. Yang, “DLT-net: Joint detection of drivable areas, lane lines, and traffic objects,” IEEE Trans. Intell. Transp. Syst., early access, Dec. 20, 2019, doi: 10.1109/TITS.2019.2943777.

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

[29] L. Chen, Z. Yang, J. Ma, and Z. Luo, “Driving scene perception network: Real-time joint detection, depth estimation and semantic segmentation,” in Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV), Mar. 2018, pp. 1283–1291.

[30] R. Cipolla, Y. Gal, and A. Kendall, “Multi-task learning using uncertainty to weigh losses for scene geometry and semantics,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 7482–7491.

[31] M. Tan and Q. V. Le, “MixConv: Mixed depthwise convolutional kernels,” 2019, arXiv:1907.09595. [Online]. Available: http://arxiv.org/abs/1907.09595

[32] 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.

[33] J. Behley, M. Garbade, A. Milioti, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, “SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 9297–9307.

[34] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, “Automatic differentiation in PyTorch," in Proc. NIPS-W, 2017.

[35] M. Bai, G. Mattyus, N. Homayounfar, S. Wang, S. K. Lakshmi, and R. Urtasun, “Deep multi-sensor lane detection,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Oct. 2018, pp. 3102–3109.

**FUWU YAN** is currently the Dean of the School of Automotive Engineering, Wuhan University of Technology, where he is also a Chief Professor of automotive engineering. He is the Director of Hubei Research Center for New Energy and Intelligent Connected Vehicle. His research interests include electrification of vehicles, powertrain control, and autonomous vehicles.

**KEWEI WANG** received the B.S. degree in automotive engineering from the School of Automotive Engineering, Wuhan University of Technology, China, in 2013, where he is currently pursuing the Ph.D. degree in automotive engineering. His research interests include environment perception and multisensor fusion in autonomous driving.

**CHEN LV** (Member, IEEE) received the Ph.D. degree from the Department of Automotive Engineering, Tsinghua University, Beijing, China, in 2016. He was a Research Fellow with the Advanced Vehicle Engineering Center, Cranfield University, Cranfield, U.K., from 2016 to 2018, and a joint Ph.D. Researcher with the Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA, USA, from 2014 to 2015. He is currently an Assistant Professor with the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore. His research interests include cyber-physical systems and advanced vehicle control and intelligence, in which he has contributed more than 60 articles and obtained 11 granted China patents. He was a recipient of the Highly Commended Paper Award of IMechE U.K., in 2012, the NSK Outstanding Mechanical Engineering Paper Award, in 2014, the China SAE Outstanding Paper Award, in 2015, the First Class Award of China Automotive Industry Scientific and Technological Invention, in 2015, and the Tsinghua University Outstanding Doctoral Thesis Award, in 2016. He serves as a Guest Editor for the IEEE/ASME TRANSACTIONS ON MECHATRONICS and the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS and an Associate Editor for the International Journal of Electric and Hybrid Vehicles and the International Journal of Vehicle Systems Modelling and Testing.

**BIN ZOU** received the Ph.D. degree in power machinery and engineering from the Wuhan University of Technology, Wuhan, China, in 2013. He is currently an Associate Professor with the School of Automotive Engineering, Wuhan University of Technology. His current research interests include automobile electronic control and autonomous vehicles.

**LUQI TANG** received the B.S. degree in thermal energy and power engineering and the M.S. degree in power machinery and engineering from the School of Automotive Engineering, Wuhan University of Technology, China, in 2012 and 2015, respectively, where he is currently pursuing the Ph.D. degree in automotive engineering. His research interests include decision making, vehicle dynamics, and control for autonomous vehicles.

**WENBO LI** received the B.E. degree in thermal energy and power engineering from the College of Power and Energy Engineering, Harbin Engineering University, China, in 2016. He is currently pursuing the Ph.D. degree in power machinery and engineering with the School of Automotive Engineering, Wuhan University of Technology, China. His research interests include environment perception, trajectory tracking, and prediction in autonomous driving.