IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

SPECIAL SECTION ON SCALABLE DEEP LEARNING FOR BIG DATA

Received December 1, 2019, accepted December 27, 2019, date of publication February 10, 2020, date of current version March 2, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2968965

Multi-Feature View-Based Shallow Convolutional Neural Network for Road Segmentation

MUHAMMAD JUNAID[1], MUBEEN GHAFOOR[1,2], ALI HASSAN[1], SHEHZAD KHALID[3], SYED ALI TARIQ[1], GHUFRAN AHMED[4], AND TEHSEEN ZIA[1]

[1]Department of Computer Science, COMSATS University, Islamabad 44000, Pakistan
[2]Department of Computer Science and Creative Technologies, University of the West of England, Bristol BS16 1QY, U.K.
[3]Department of Computer Engineering, Bahria University, Islamabad 44000, Pakistan
[4]Department of Computer Science, National University of Computer and Emerging Sciences, Karachi 74000, Pakistan

Corresponding author: Muhammad Junaid (mjunaid.cui@gmail.com)

ABSTRACT This study presents a shallow and robust road segmentation model. The computer-aided real-time applications, like driver assistance, require real-time and accurate processing. Current studies use Deep Convolutional Neural Networks (DCNN) for road segmentation. However, DCNN requires high computational power and lots of labeled data to learn abstract features for deeper layers. The deeper the layer is, the more abstract information it tends to learn. Moreover, the prediction time of the DCNN network is an important aspect of autonomous vehicles. To overcome these issues, a Multi-feature View-based Shallow Convolutional Neural Network (MVS-CNN) is proposed that utilizes the abstract features extracted from the explicitly derived representations of the input image. Gradient information of the input image is used as additional channels to enhance the learning process of the proposed deep learning architecture. The multi-feature views are fed to a fully-connected neural network to accurately segment the road regions. The testing accuracy demonstrates that the proposed MVS-CNN achieved an improvement of 2.7% as compared to baseline CNN consisting of only RGB inputs. Furthermore, the comparison of the proposed method with the popular semantic segmentation network (SegNet) has shown that the proposed scheme performs better while being more efficient during training and evaluation. Unlike traditional segmentation techniques, which are based on the encoder-decoder architecture, the proposed MVS-CNN consists of only the encoder network. The proposed MVS-CNN has been trained and validated with two well-known datasets: the KITTI Vision Benchmark and the Cityscapes dataset. The results have been compared with the state-of-the-art deep learning architectures. The proposed MVS-CNN outperforms and shows supremacy in terms of model accuracy, processing time, and segmentation accuracy. Based on the experimental results, the proposed architecture can be considered as an efficient road segmentation architecture for autonomous vehicle systems.

INDEX TERMS Convolutional neural network, deep learning, segmentation, road detection, autonomous vehicle, computer vision, image processing.

I. INTRODUCTION

Autonomous driving has gained great attention of many researchers recently. The key aim of Intelligent Transport Systems (ITS) is to avoid accidents while accurately guiding the vehicle through the road, along with considering traffic safety rules and avoiding obstacles in the way [1], [2]. Advancement in the field of Autonomous Vehicles (AV) has put forth enormous challenges in the automotive industry [3]. These challenges include real-time processing along with accurate and reliable segmentation. The availability of safe, secure and reliable AV will benefit humanity including visually impaired and handicapped people to commute more easily and efficiently [4].

To reduce the risk of road accidents, it is necessary to accurately distinguish the road region from the other regions. This will help autonomous vehicles to navigate correctly, as well as understand the situation of the surrounding environment including traffic signs [5], [6] and signals [7], [8], pedestrians [9], road lanes [10] and other vehicles on the road [11]. Mersky *et al.* [12] have concluded that reducing the prediction time of detection/segmentation algorithm used in

The associate editor coordinating the review of this manuscript and approving it for publication was Moayad Aloqaily.

AVs can improve fuel economy since more aspects of driving (i.e., acceleration, speed, etc.) are based on decisions made by the AV, instead of the human driver.

Typically, object detection and tracking techniques use a bounding box to locate specific objects in an image [11], [13], [14]. Road detection differs from other object detection techniques as road region is usually occluded by vehicles, cyclist and pedestrians. For road detection, semantic segmentation of perspective view can differentiate the road region from the other regions.

Researchers have proposed different techniques to address road detection and segmentation [15]–[22]. Some of these are based on Digital Image Processing (DIP) [17], [19], [22], whereas some use Machine Learning (ML) techniques [23], [24] to address the problem.

Initially, researchers [17], [19], [22] used DIP techniques to detect road lines in order to help drivers. DIP techniques can only detect roads based on structural and color information. Similarly, ML techniques have been also used in object detection [25], [26], classification [27], [28], and segmentation [29]–[31].

DIP and ML techniques work based on predefined criteria and features (such as color, edges, corner points, structures, etc.) in images. Hence, such techniques do not perform well in noisy images as well as images with different lighting conditions. Such techniques require pre-processing of images and manual fine-tuning which varies with different types of images under different circumstances [32].

Recently, Deep Learning (DL) has demonstrated excellent results in artificial intelligence and computer vision [33]–[37] applications. Many DL based applications have been proposed recently. The applications include image classification [32], [38], [39], image segmentation [40], [41], change detection in images [42], [43] and object tracking [44], [45]. While Deep Convolutional Neural Networks (DCNN) have been useful in the aforementioned DL applications, it also proves helpful in semantic segmentation [29], [46].

Semantic segmentation models classify pixels of an image into one of the predefined classes. Recently, several semantic segmentation solutions based on DCNN have been proposed [18], [40], [47], [48]. The amazing result gained by AlexNet [49] and GoogleNet [37] for classification led to an increase in the use of DCNN based computer vision applications, including segmentation.

In DCNN, each convolutional layer extracts features and learns abstract information from the input images. The deeper the network is, the more generalized information it tends to learn. Most of the state-of-the-art DL architectures, like UNet [33], ResNet [34], SegNet [35], VGGNet [36], and GoogleNet [37], deepen the network to achieve improved model accuracy and segmentation performance. However, deeper networks are data-hungry and require much more data to perform better. In situations where the availability of data is insufficient, making the networks deeper can cause overfitting. Hence, the objective of learning generalized features will not be achieved.

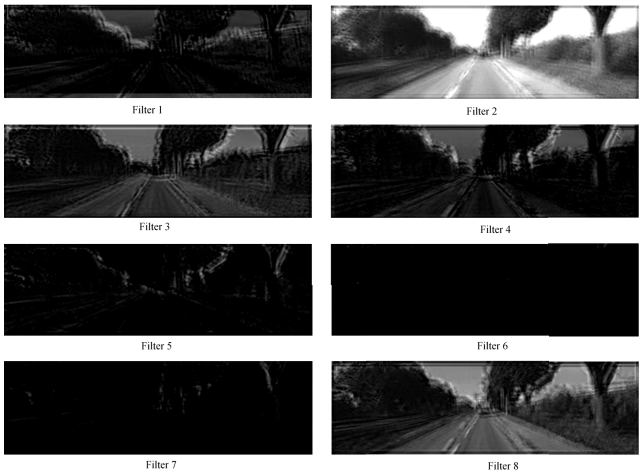


FIGURE 1. The outputs generated by each filter of the first convolutional layer of a CNN with eight filters.

Similarly, the deeper the network, the higher is the time required to obtain the results. However, in computer-aided real-time applications like driver assistance, the processing time is of key importance. Thus, there is an intense need for shallow and generalized networks that require less processing time to segment and classify images. Keeping in view the above points, it would be useful if the required abstract information are pre-calculated and passed along with the original input. This will result in a less-deeper and accurate network, which will require less data for training.

The recent state-of-the-art DL techniques use DCNN [33], [35] to achieve better results. However, deeper networks face vanishing gradient problem. The vanishing gradient problem has been handled with residual networks in [34]. However, residual connections make the network complex, which results in increasing the time required for training the network. Such networks cannot be used in real-time computer-vision based applications.

The proposed Multi-feature View-based Shallow Convolutional Neural Network (MVS-CNN) utilizes gradients information as additional features along with the input image. The gradient of an image shows the change in intensity or color in an image, which is useful to extract essential information from the images (i.e. edge, texture, etc). By examining the outputs of the convolution layers in a DCNN, shown in Fig. 1, it can be observed that the outputs are similar to a gradient image. Since convolutional layers in DCNN learn features, the gradient features along with input image will be useful for the DL model to enhance their learning process. Furthermore, they can also help with reducing the processing time of the DL architectures.

The novelty of our proposed techniques is that the gradient information is considered to enhance the learning process. It has a minimal number of convolutional layers, hence not very deep nor complex, which requires less processing time. The proposed technique is a sequential model with no complexity such as shortcut connections and encoder-decoder

networks, which makes it fast and efficient in comparison to other recently proposed DL architectures. The key contributions of this study are:

1) A novel MVS-CNN model is developed for road segmentation task and its performance is measured against recent state-of-the-art segmentation models on publicly available databases.
2) The multi-feature view based on gradient information of the images is used to the enhance the network learning process.
3) The proposed segmentation architecture consists of only the encoder network with seven (7) convolution layers, and as a result it is less complex than state-of-the-art encoder-decoder based segmentation networks.
4) The proposed scheme is evaluated against various state-of-the-art DL architectures based on training accuracy, processing time, and segmentation performance.

The rest of the paper is organized as follows: Section II discusses related studies. The proposed methodology has been discussed in section III. Results and comparisons have been discussed in section IV. Finally, section V concludes the paper.

II. LITERATURE REVIEW

Recently, researchers have proposed different techniques for road detection and segmentation. Initially, researchers proposed image processing techniques to detect road lanes and boundaries [17], [19], [22].

Aly *et al.* [17] used the Hough transform based method to recognize lane marks and to classify the road region in an image. Hough transform extracts shapes; therefore, it can result in false detection. Structures with boundaries similar to lanes, such as railings, signboards or road surface cracks, can be mis-recognized as lane markings. To accurately identify the lane, Wang *et al.* [22] proposed a novel B-Snake based lane model for detecting different types of lanes. The proposed model can detect various types of structures such as straight, curved or parabolic lanes. The authors merge the problem of side lane detection to detect the middle lane of the road.

Kim [19] introduced a lane detection algorithm based on the Random Sample Consensus (RANSAC). The algorithm is robust and able to perform in real-time. First, possible lane lines are calculated, which are then grouped by a hypothesis-verify method. The selected lines are grouped separately into right and left lanes. Since image processing techniques are designed for specific scenarios, they are not able to perform in complex and unpredictable situations. Such drawback of image processing techniques prompted researchers to discover techniques that could learn and adjust to diverse scenarios.

Unlike image processing techniques, Machine Learning (ML) techniques can extract and learn features for classification. ML techniques have been used in prior AV models [23], [24], and can perform detection and segmentation tasks in vast scenarios. A method to detect road lane boundaries has been recommended by Fang and Wang [23]. The authors propose a method named Vector Fuzzy Connectedness (VFC). From a skeleton image, the lane curves are calculated. Finally, the control points for the right and left lanes are estimated from the curves using VFC.

Zhu *et al.* [24] have studied and evaluated Extreme Learning Machines (ELM) for road and vehicle detecting. For road detection, the color histogram has been used. Similarly, gray color features and Histogram of Oriented Gradients (HOG) are used to detect vehicles. Their proposed network is proved to perform better as compared to Support Vector Machines (SVM) and Back Propagation Network (BPN). The results of road segmentation using the ELM network are not accurate because of the rectangular pattern of segmentation. Apart from the results ML provides, it is worthy to note that since the feature extraction and classification process are independent of each other, the classifiers do not provide adequate results.

DL architectures tend to extract and learn features from input images which result in better segmentation and classification results. DL requires a large dataset of images for training. Keeping this in view, AlHaija *et al.* [15] have proposed an augmented reality method to augment road scene images to obtain a large dataset for training deep networks. Augmented reality has been used to create additional images with more traffic information. The augmented images are then used to train the state-of-the-art Multi-task Network Cascade (MNC) [50]. The addition of augmented images helps in improving the segmentation performance as compared to real or synthetic images.

Liu and Deng [20] proposed fully convolutional deep residual network along with pyramid pooling. Since light exposure in images also affects the training and prediction of deep networks, the authors suggest augmentation based on exposure. Underexposure compensation has been used to augment the training images to get a larger training dataset. However, such technique requires a high knowledge of image processing to collect different images with similar illumination features to generate realistic scenes.

The impact of using a higher number of convolutional layers in the DCNN has been studied in [34]. The authors suggested that using residual layers in deep networks can result in gaining higher accuracy than conventional networks. Using ImageNet [51] dataset, they compare the results of 34-layer plain and 34-layer residual networks. They conclude that deeper networks have higher training errors due to vanishing gradient problem. However, residual connection overcomes degradation and allows us to gain accuracy from deeper networks. Their proposed network can reduce training error by 3.5% as compared to conventional DL networks.

Long *et al.* [52] discussed Fully Convolutional Neural Network (FCNN) for pixel-to-pixel prediction for semantic segmentation. The authors present a network that replaces the dense layer by an up-sampling layer to get a pixel-to-pixel segmented output. Converting the network

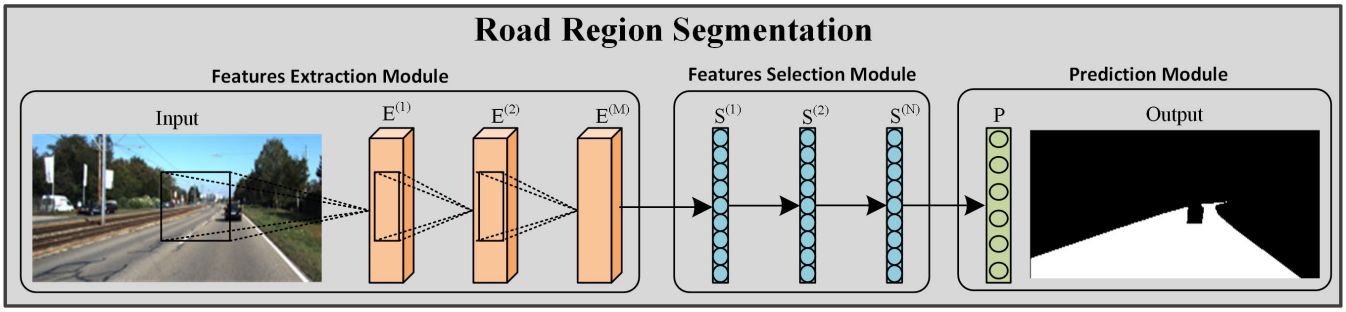


FIGURE 2. Example of DL modules with input image and ground-truth labelled image of "um_000027" file from the KITTI [57] dataset.

to fully convolutional networks results in lower inference time, which is very important in real-time applications. A network-in-network CNN model has been proposed by Mendes *et al.* [21], which is converted into FCNN to gain fast results in road detection after training. The proposed technique have achieved comparably accurate results, along with maintaining a fast inference time. Due to the minimized inference time, the method could be used effectively in real-time road detection. However, the proposed method is incapable to correctly classify between road regions due to varied lighting conditions.

Romera *et al.* [53] presented a real-time solution for road segmentation using CNN with residual connections. Residual connection overcomes the degradation problem which is faced in most architectures with a larger number of layers. In this way, their architecture allows accurate classification along with being efficiently fast. Contributing to real-time road segmentation, Romera *et al.* [54] have presented a method to redesign the residual layer. Their goal is to achieve speed efficiency while retaining the accuracy of the recently proposed techniques. Although deep and complex architectures provided competent results than DIP and ML techniques, they require high computation power and big data for training.

A self-ensembling attention network was proposed by Xu *et al.* [55]. The network consists of a student model that acts as a base network and a teacher model acting as the ensembling network. In their proposed network, the student model learns from the output of the teacher networks. As the model is trained the student model becomes accurate, hence the prediction of the teacher models also gets closer to the accurate labels in the target domain. Since the student network learns from the teacher network, the performance of the overall framework mainly depends on the teacher network. Since multi-scale features are an important part of DL, a scale-aware model is proposed in [56]. The same images of different scales are used by the network for feature extraction at different scales. The extracted features are then merged and used for the classification of every pixel in an image.

DIP based techniques [17], [19], [22] are prone to variations such as changes in lighting conditions, variations in structures, and color differences. These techniques work well in specified conditions. Methods proposed in [15], [20] require high domain expertise. These techniques require image processing knowledge to create new augmented images that look similar to the original images. Similarly, DL techniques require high computing power and large dataset. To reduce the complexity and tackle the huge data requirement, a shallow CNN which uses multi-view features is proposed. The proposed shallow CNN provides higher segmentation accuracy while minimizing the complexity of the network.

III. PROPOSED METHODOLOGY

A high-level diagram of the proposed DL architecture for road segmentation is shown in Fig. 2. In this study, segmentation is performed using MVS-CNN architecture that comprises of a feature extraction module (E), a feature selection module (S), and a prediction module (P) as can be seen in Fig. 2.

In road images, the road boundaries differentiate the road region from the background regions. The gradient of an image resembles the boundaries of the road region. Feature extraction in the segmentation architecture can be improved by adding such information. This will enhance the efficiency and efficacy of the architecture. Such views help in training the architecture without going deeper.

The network takes training data as input in the form of (X, Y). The X is the road image denoted as $I^{(r\times c\times v)}$, where r, c represent the height and width of the road image, respectively, and v represents the number of feature views (including RGB views). The Y is the corresponding segmented ground-truth of the road image such that $Y \in [0; 1]^{(r\times c)}$, where r, c represent the height and width of the ground-truth image, respectively, which are the same as the input image. The different view combinations considered in this study include horizontal gradient (G_x), vertical gradient (G_y), and the gradient magnitude (G_{Mag}) views. The horizontal and vertical gradient views are constructed by computing the respective gradients of the road image using (1) and (2), where Δ_x and Δ_y represent the gradient components computed using Gaussian filter and i and j represent row and column pixel

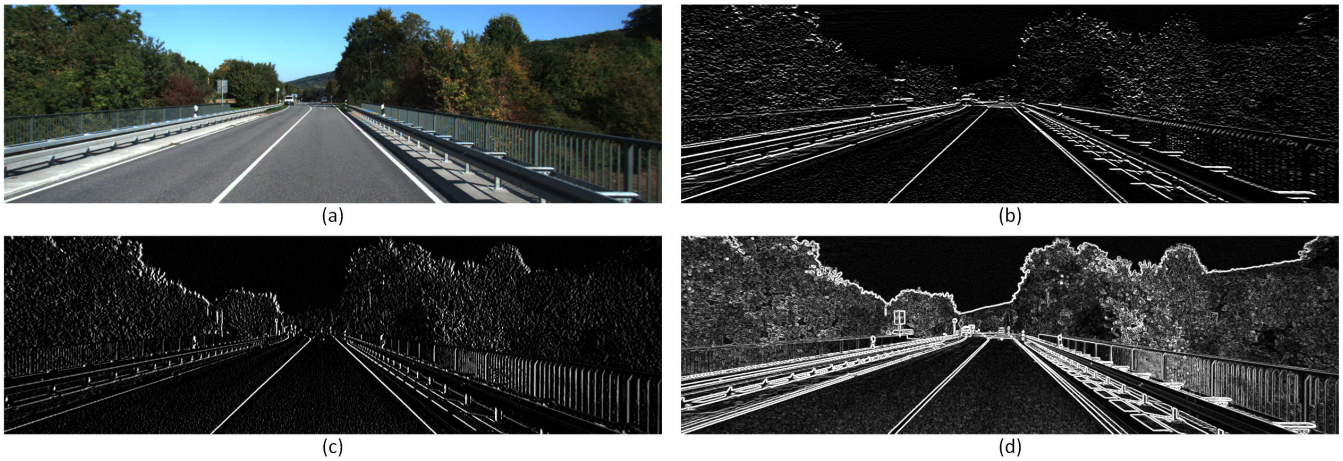


FIGURE 3. Different views created from sample road image "um_000069" from KITTI [57] dataset. (a) Input Image (I_{RGB}) (b) Horizontal gradient (G_x) (c) Vertical gradient (G_y) (d) Gradient magnitude (G_{Mag}).

coordinates respectively:

$$G_x = \sum_{u=-s}^{u=s} \sum_{v=-s}^{v=s} \Delta_x \left(x_i + u, y_i + v\right)^2 \tag{1}$$

$$G_y = \sum_{u=-s}^{u=s} \sum_{v=-s}^{v=s} \Delta_y \left(x_i + u, y_i + v\right)^2 \tag{2}$$

The gradient magnitude view G_{Mag} computed using G_x and G_y as given by (3):

$$G_{Mag} = \sqrt{G_x + G_y} \tag{3}$$

Pictorial representation of the views are shown in Fig. 3. Fig. 3(a), shows a sample image from the KITTI dataset, whereas Fig. 3(b), 3(c) and 3(d) show the gradient in horizontal and vertical along with the gradient magnitude of the sample image, respectively. The purpose of road segmentation is to learn a representation function R that can predict a binary image $\hat{y}$ relative to the ground-truth segmented image Y given input X, as given by (4):

$$\hat{y} = R\left(Y|X; W, b\right) \tag{4}$$

where W, b are parameters of R. Function R is parameterized with Sigmoid [58] function to compute the output probability of each pixel between 0 and 1:

$$\hat{y} = \text{Sigmoid}(R\left(Y|X; W, b\right)) \tag{5}$$

The threshold value of 0.5 is applied on the predicted image $\hat{y}$ to generate the final output image.

The function R, in this case, represents the proposed MVS-CNN and it can be represented as:

$$R = MVS_{CNN}(X) \tag{6}$$

Specifically,

$$MVS_{CNN}(X) = S^{(N)} \ldots (S^{(1)}(E^{(M)} \ldots (E^{(2)}(E^{(1)}(X))))) \tag{7}$$

where M and N indicate the total number of feature extraction and feature selection modules respectively. Feature extraction module consists of sequential processes that define the layer-wise operations at each layer l, given as $g^{(l)}$, and extracts features represented as $E^{(l)}$:

$$\begin{aligned} E^{(l)} &= g^{(l)}(E^{(l-1)}; W^{(l)}, b^{(l)}) \\ &= f_{norm}(P(f_{act}(W^{(l)} * E^{(l-1)} + b^{(l)}))) \end{aligned} \tag{8}$$

where $b^{(l)}$ and $W^{(l)}$ represent the l^{th} layer bias and filter weights, $*$ represents the convolution operation, f_{act} is the non-linear activation function, P defines the max-pooling operation, f_{norm} represents normalization function, and $E^{(l-1)}$ is either the input image X for the first layer ($l = 1$) or the $(l-1)^{th}$ activation for other layers ($l > 1$). In the convolution layer, the input image is processed to learn filter weights that extract useful features. To introduce non-linearity, Rectified Linear Unit (ReLU) [59] is used after the convolution layers. The pooling layer reduces the size of the feature maps by merging the locally associated features into a single feature by taking either the average or maximum of the feature values. The feature selection module at layer l represented as $h^{(l)}$, where $l > (M-1)$, is a dense neural network layer where each neuron is connected to all the neurons in the previous layer. The process at this layer involves dot product (.) operations and non-linear activations as given by (9):

$$\begin{aligned} S^{(l)} &= h^{(l)}(S^{(l-1)}; W^{(l)}, b^{(l)}) \\ &= (f_{act}(W^{(l)}.S^{(l-1)} + b^{(l)})) \end{aligned} \tag{9}$$

where $S^{(l-1)}$ defines either the $(l-1)^{th}$ layer activation for $l > M$ or it represents $E^{(M)}$ for $l = M$ connecting the feature selection module with the feature extraction module. The parameters of R are optimized based on Stochastic Gradient Descent (SGD) [60] with the cross-entropy loss function. The proposed MVS-CNN is illustrated in Fig. 4.

IV. EXPERIMENTATION AND RESULTS

The proposed MVS-CNN road segmentation network is designed to make the computer-aided road segmentation systems less complex. The primary experimental verifications

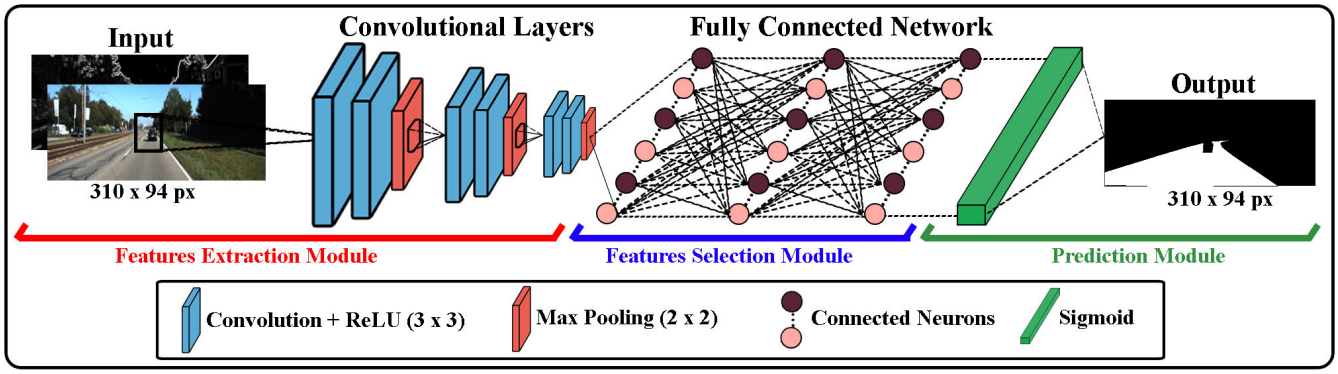


FIGURE 4. Proposed MVS-CNN architecture with input ($I_{RGB} + G_x + G_y + G_{Mag}$, 6 channel) "um_000027" image and network output is segmented road region.

required are (i) whether the inclusion of the multi-feature views improve the network performance without significant degradation of segmentation accuracy. (ii) whether the proposed network is reliable and efficient at road segmentation task and achieves improved performance as compared to recent state-of-the-art segmentation models.

A. DATASET

For evaluating the performance of DL architectures in this study, two well known dataset, KITTI Vision Benchmark Suite [57], and Cityscapes dataset [61] are utilized for experimentation, which are discussed below:

1) THE KITTI DATASET

The KITTI dataset [57] consists of 289 input images along with the lane and road labels images. The dataset comprise of different variations of roads including; marked roads (um), multiple marked roads (umm), and unmarked roads (uu). Lane labels are available for the marked and multiple marked roads only. This study focused on road segmentation; therefore, only road labels have been selected from the dataset. Since deep networks require a large number of images, these images are augmented to artificially expand the dataset to 1500 images. The parameters used for augmentation are presented in Table 1. The dataset is partitioned into 80% training and 20% validation set randomly. The same augmented dataset is used for training and validation of all DL architecture evaluated in this study for a fair comparison.

2) CITYSCAPES DATASET

The Cityscapes dataset [61] comprises of 5000 labeled images. The images have been collected from 50 cities during different months and seasons. There are 2975 training, 1525 testing and 500 validation images in the dataset. The images are classified into 19 categories in the Cityscapes dataset. Since this study focuses on the segmentation of roads, only road labels are selected and all other labels are considered as background regions.

TABLE 1. Data augmentation parameters.

Parameter	Value
Rotation range	20
Height and Width shift range	0.2
Shear and Zoom range	0.2
Re-scale	1/255
Horizontal flip	True
Fill mode	Nearest

B. PROPOSED NETWORK ARCHITECTURE

The proposed network architecture is derived by exhaustively testing various combinations of the following four network hyper-parameters to find the better performing segmentation architecture:

- The convolution layers in the network are varied between 5 and 9 and max-pooling is performed after every layer or after stacking two or three convolution layers.
- For training, different optimization algorithms such as SGD [60] and RMSProp [62] are tested.
- ReLU [59] and Leaky-ReLU [63] activation functions are evaluated after every convolution layer.
- For regularization, both Batch-Normalization (BN) [64] and Dropout [65] have been evaluated.

Based on extensive testing, the network consisting of seven (7) convolution layers performed better than the other network configurations. Amongst the tested optimizer, RMSProp exhibited faster convergence and better accuracy as compared to SGD optimizer over 50 epochs. For the tested ReLU and Leaky-ReLU activation functions, both achieved comparable accuracy but ReLU was more efficient. Furthermore, BN after each convolution layer performed better, while increasing speed, performance, and stability of the proposed architecture. The MVS-CNN architecture yielding the best results is given in Table 2. The experimentations in this study are performed using the TensorFlow [66] library on the Google Colaboratory platform [67].

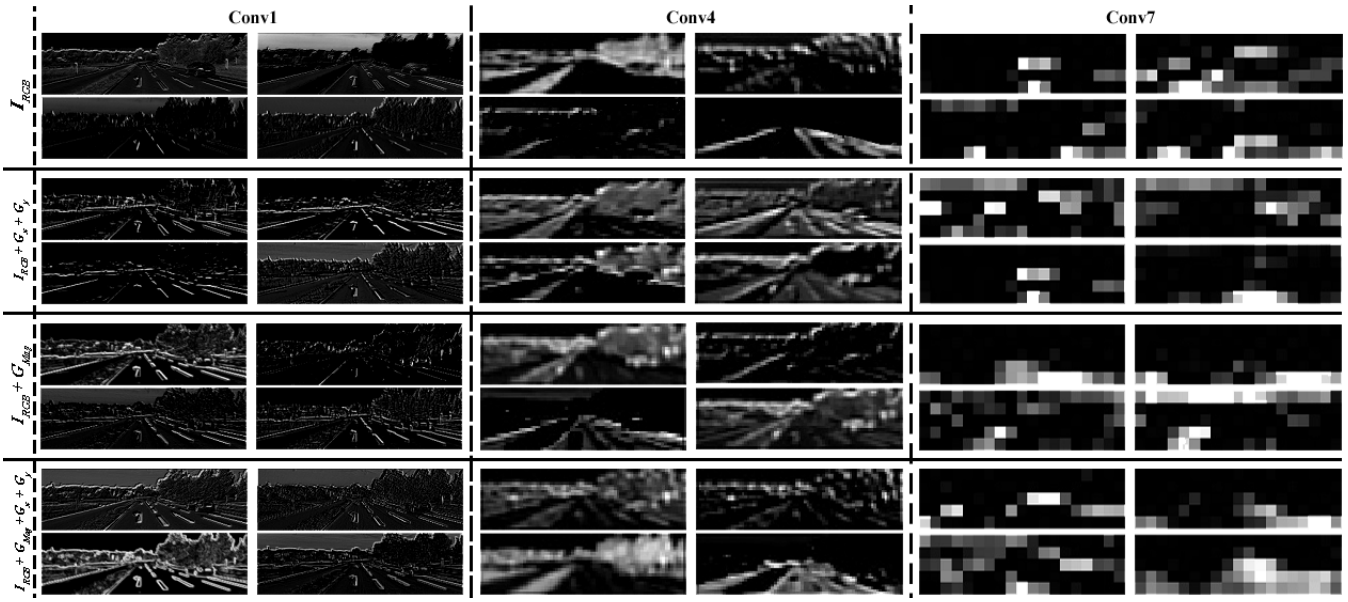


FIGURE 5. Feature maps from first, intermediate and last convolution layers of the proposed MVS-CNN on image "um_000015" from the KITTI [57] dataset.

TABLE 2. Architecture of the Proposed MVS-CNN (baseline).

Layer	Filter size	No. of filters	Input	Output	Activation
Conv-1	3×3	32	$94 \times 310 \times 3$	$94 \times 310 \times 32$	ReLU
Conv-2	3×3	64	$94 \times 310 \times 32$	$94 \times 310 \times 64$	ReLU
Pool-1	2×2	-	$94 \times 310 \times 64$	$47 \times 155 \times 64$	-
Conv-3	3×3	64	$47 \times 155 \times 64$	$47 \times 155 \times 64$	ReLU
Pool-2	2×2	-	$47 \times 155 \times 64$	$23 \times 77 \times 64$	-
Conv-4	3×3	128	$23 \times 77 \times 64$	$23 \times 77 \times 128$	ReLU
Conv-5	3×3	128	$23 \times 77 \times 128$	$23 \times 77 \times 128$	ReLU
Pool-3	2×2	-	$23 \times 77 \times 128$	$11 \times 38 \times 128$	-
Conv-6	3×3	256	$11 \times 38 \times 128$	$11 \times 38 \times 256$	ReLU
Pool-4	2×2	-	$11 \times 38 \times 256$	$5 \times 19 \times 256$	-
Conv-7	3×3	512	$5 \times 19 \times 256$	$5 \times 19 \times 512$	ReLU
Pool-5	2×2	-	$5 \times 19 \times 512$	$2 \times 9 \times 512$	-
FC-1	-	-	-	128	-
FC-2	-	-	-	29140	Sigmoid

C. PROPOSED MVS-CNN NETWORK CONFIGURATION

In this section, the MVS-CNN is evaluated by varying different multi-feature view combinations to find the optimal combination. Different views such as gradient in horizontal and vertical directions (G_x and G_y) and gradient magnitude (G_{Mag}) are derived from the road images, as discussed in Section III, with an example shown in Fig. 3 of the same section.

To study the effect of individual feature views on proposed MVS-CNN architecture, the KITTI dataset [57] is considered, and the results of the proposed network with the different feature view combinations are presented in Table 3. The baseline proposed MVS-CNN with RGB channel input achieved a training accuracy of 95.9% along with testing accuracy of 94.2%. When horizontal and vertical gradients are used with input image (i.e., $I_{RGB} + G_x + G_y$, 5-channel input), the feature learning process improve due to gradient features, hence leading to proposed MVS-CNN training and testing accuracy of 97.3% and 95.7% respectively. Similarly, another experiment is carried out by using input image with gradient magnitude (i.e., $I_{RGB} + G_{Mag}$, 4-channel input), the proposed MVS-CNN accuracy increased from baseline MVS-CNN (i.e. I_{RGB}, 3-channel input), resulting with training and testing accuracy of 96.5% and 95.1% respectively. Furthermore, when all gradient information is used along with input image (i.e. $I_{RGB} + G_x + G_y + G_{Mag}$, 6-channel input), the proposed MVS-CNN achieved 98.4% training accuracy along with testing accuracy of 96.9%. It can be observed in Table 3, that the gradient information is helpful in improving the learning process of the proposed MVS-CNN architecture. Furthermore, the effect of different multi-feature view combinations on model accuracy is shown in Fig. 5. Fig. 5 shows that the feature maps extracted using the first convolution layer (i.e. Conv1), intermediate convolution layer (i.e. Conv4) and the last convolution layer (i.e. Conv7) of the proposed MVS-CNN architecture, the proposed multi-feature view combination (i.e. $I_{RGB} + G_x + G_y + G_{Mag}$) is learning more features as compared to other multi-feature view combinations. Similarly, it can be observed that the input images along with horizontal and vertical gradient based combination (i.e. $I_{RGB} + G_x + G_y$) are retaining useful features in convolutional layers when compared to input images along with gradient magnitude combination (i.e.

TABLE 3. Model accuracy of the proposed MVS-CNN on different multi-feature views combination for KITTI [57] Vision Benchmark Suite database.

Multi-feature-view configuration	Training Accuracy (%)	Testing Accuracy (%)
I_{RGB} (baseline CNN)	95.9	94.2
$I_{RGB} + G_x + G_y$	97.3	95.7
$I_{RGB} + G_{Mag}$	96.5	95.1
$I_{RGB} + G_x + G_y + G_{Mag}$	98.8	96.9

TABLE 4. Network complexities in term of trainable parameters, training time and testing time.

Method	No. of Trainable Parameters	KITTI [57]				Cityscapes [61]			
		Training Acc. (%)	Testing Acc. (%)	Training Time (s)	Prediction Time (ms)	Training Acc. (%)	Testing Acc. (%)	Training Time (s)	Prediction Time (ms)
UNet [33]	31,032,837	95.5	93.2	4120	5.3	96.4	94.8	3816	4.9
ResNet [34]	35,211,220	96.9	94.8	1766	2.6	97.8	95.1	1328	2.2
SegNet [35]	5,467,265	98.3	96.1	853	1.1	98.5	95.8	687	0.8
MVS-CNN	6,697,460	98.8	96.9	704	0.8	99.1	96.2	643	0.6

$I_{RGB} + G_{Mag}$), thus resulting in better accuracy and output image similar to the ground-truth. Moreover, it can also be noted that the gradient information enhances the model learning process when compared to just input images (i.e. I_{RGB}) based model training approach, hence yielding higher model accuracy. Based on the analysis presented in this sub-section, further experimentation and evaluation of the proposed MVS-CNN architecture are carried out using all multi-feature view based combinations along with input image (i.e. $I_{RGB} + G_x + G_y + G_{Mag}$).

D. NETWORK PERFORMANCES

In order to carry out further experimentation, both dataset (i.e. KITTI [57] and Cityscapes [61]) are considered for the performance analysis of the different network architectures. The performances of the MVS-CNN, SegNet [35], UNet [33], and ResNet [34] models are compared in terms of model accuracy, processing time, and network complexity (number of trainable parameters) presented in Table 4. It can be seen that the UNet [33] has large number of trainable parameters (i.e. 31 millions) and slow model training and prediction time when compared to other networks. Similarly, ResNet [34] with highest trainable parameters (i.e. 35 millions) has achieved better training and validation accuracy when compared to UNet [33], along with less model processing and prediction time respectively. Similarly, the state-of-the-art SegNet [35] with fewest trainable parameters (i.e. 5.4 millions) achieved the better model accuracy and take less processing and prediction time when compared to UNet [33] and ResNet [34]. However, the proposed MVS-CNN with 6.6 million trainable parameters performs better than the state-of-the-art networks, while achieving highest training and testing accuracy of 98.8% and 96.9% on KITTI dataset, respectively. Similarly, the proposed MVS-CNN achieved training and testing accuracy of 99.1% and 96.2% on Cityscapes dataset. The proposed MVS-CNN show supremacy in terms of processing time, while achieving significant less training and prediction times respectively. It is clear from Table 4, that although SegNet [35] has fewer trainable parameters, the proposed MVS-CNN outperformed in terms of model accuracy and processing time. Similarly, in comparison to other state-of-the-art networks, our proposed architecture has a fewer number of convolution layers and can deal with the additional multi-features view while optimizing the network performance when compared to other state-of-the-art DL architectures.

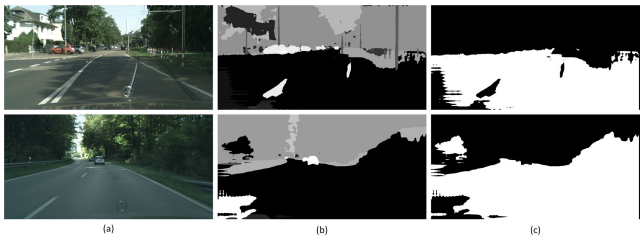


FIGURE 6. Results of utilized SEANet [55] architecture on input images from Cityscapes [61] dataset (top-to-bottom: "frankfurt_000000_004617", "lindau_000058_000019"). (a) Input image (b) SEANet [55] predicted output (c) Final output with only road region.

E. METRICS FOR SEMANTIC SEGMENTATION ACCURACY

In this study, the segmentation accuracy is calculated using the most commonly used semantic segmentation metrics [32], [52], such as mean Intersection over Union (IoU), mean accuracy, pixel accuracy, and frequency weighted (f.w.). IoU. These metrics are computed as follows [52]:

$$Pixel\ accuracy = \frac{\sum_i n_{ii}}{\sum_i t_i} \tag{10}$$

$$Mean\ accuracy = \frac{(1/n_{cl})\sum_i n_{ii}}{\sum_i t_i} \tag{11}$$

$$Mean\ IoU = \frac{(1/n_{cl})\sum_i n_{ii}}{(t_i + \sum_i n_{ji} - n_{ii})} \tag{12}$$

$$f.w.\ IoU = \frac{(\sum_k t_k)^{-1}\sum_i t_i n_{ii}}{(t_i + \sum_i n_{ji} - n_{ii})} \tag{13}$$

where n_{ii} represents the number of correctly identified pixels, t_i represents the overall pixels in class i, n_{cl} represents total classes, n_{ij} represents number of class i pixels accurately predicted as class j, and n_{ji} represents the number of pixels incorrectly rejected for class i. In addition to UNet, ResNet, and SegNet, SEANet [55] model is also utilized for comparison and evaluation of segmentation accuracy. In this study, the pre-trained model of SEANet is used for comparisons. Since the SEANet [55] model is pre-trained on different labels, the predicted output of the SEANet is refined to represent all labels other than road regions as a single background region as shown in Fig. 6. The segmentation accuracy results and comparison of proposed network with the aforementioned DL architectures are presented in Table 5, which shows that the MVS-CNN architecture performs better on KITTI [57] and Cityscapes [61]. Whereas, SegNet achieved comparable results against proposed architecture.

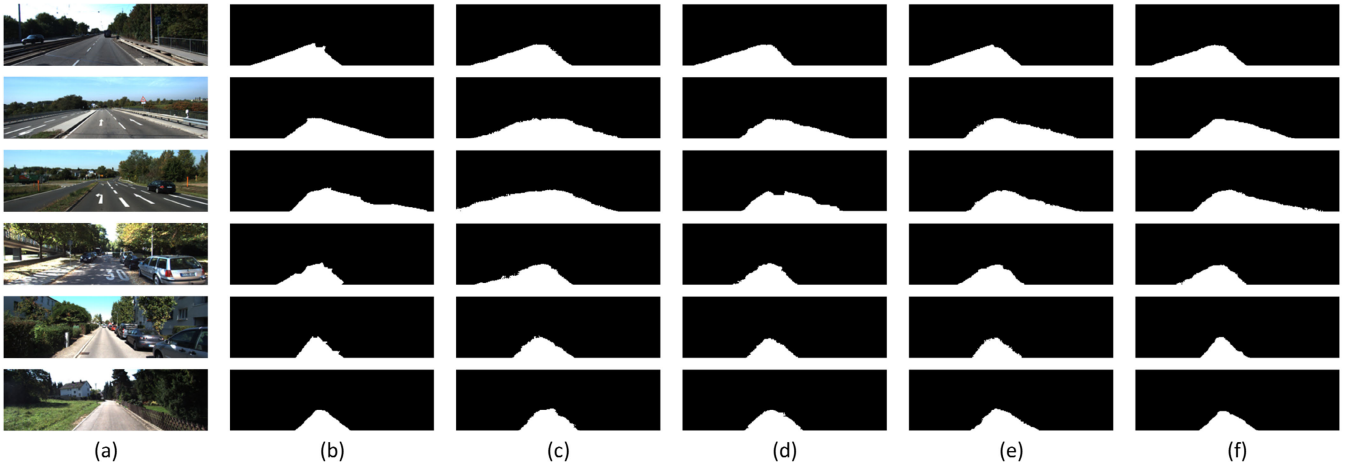


FIGURE 7. Segmentation results of all deep learning architectures on input images from KITTI [57] dataset (top-to-bottom: 'um_000083', 'umm_000011', 'umm_000015', 'uu_000063', 'uu_000066' and 'uu_000072'). (a) Input image (b) Ground truth labeled image (c) UNet [33] (d) ResNet [24](e) SegNet [35] (f) Proposed MVS-CNN.

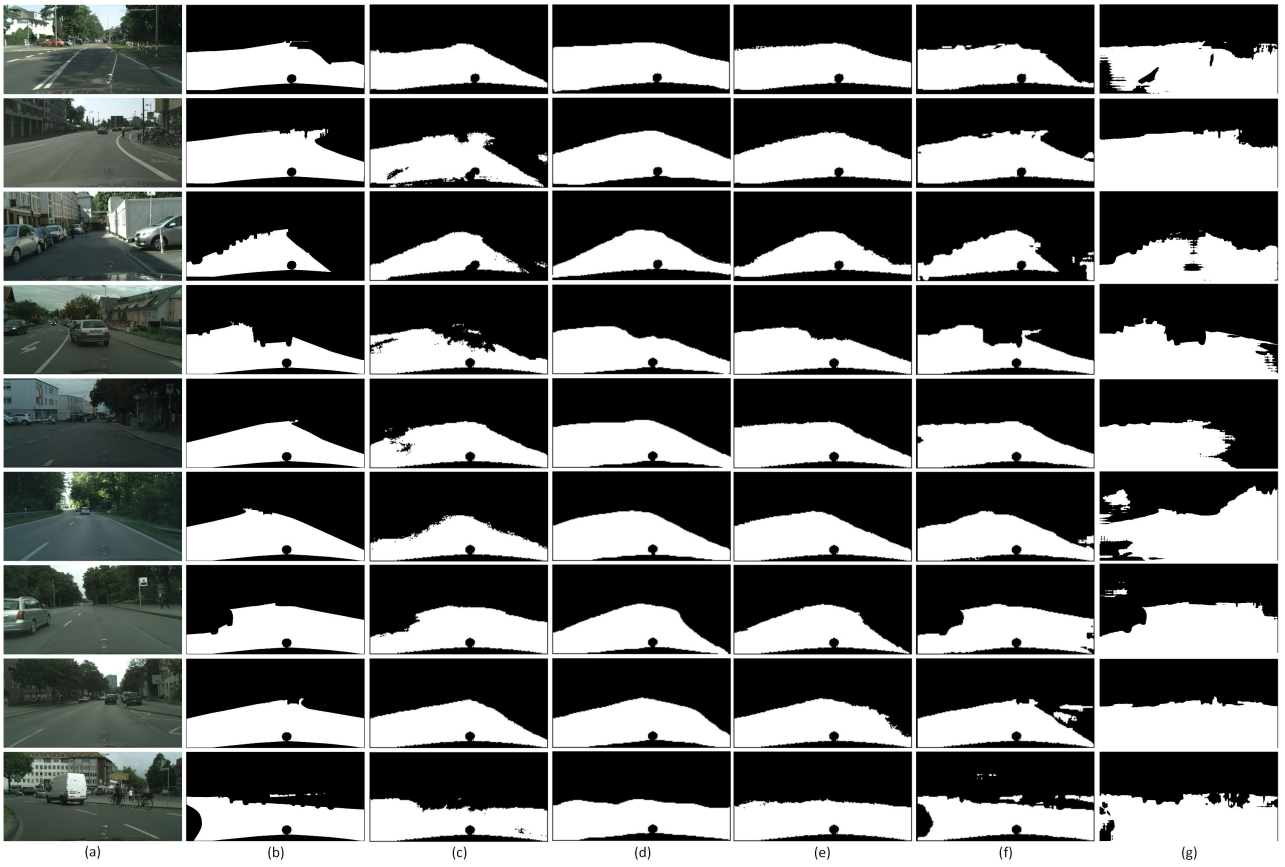


FIGURE 8. Segmentation results of all deep learning architectures on input images from Cityscapes [61] dataset (top-to-bottom: 'frankfurt_000000_004617', 'frankfurt_000000_004617', 'frankfurt_000001_015768', 'lindau_000000_000019', 'lindau_000010_000019', 'lindau_000058_000019', 'munster_000001_000019', 'munster_000022_000019' and 'munster_000038_000019'). (a) Input image (b) Ground truth labeled image (c) UNet [33] (d) Res-Net [24] (e) SegNet [35] (f) Proposed MVS-CNN (g) SEANet [55].

It is clear from the results reported in Table 5 that the proposed MVS-CNN precisely predicts the road regions in the input images compared to other DL architectures.

The predicted output of the SegNet [35], ResNet [34], UNet [33] and MVS-CNN on KITTI dataset are illustrated in Fig. 7. Similarly, the predicted results of utilized and proposed DL architectures on the Cityscapes dataset are also shown in Fig. 8. It can be observed that our proposed MVS-CNN and SegNet [35] have obtained comparable results by precisely segmenting road regions, which are very

TABLE 5. Comparison of Road Regions Segmentation accuracy (in %) using DL architectures.

Method	KITTI [57]				Cityscapes [61]			
	Pixel Accuracy	Mean Accuracy	mean IoU)	f.w. IoU	Pixel Accuracy	Mean Accuracy	mean IoU	f.w. IoU
UNet [33]	96.9	88.7	78.9	95.7	94.8	88.1	78.3	94.9
ResNet [34]	97.1	89.9	79.5	96.4	95.4	88.5	78.9	95.8
SegNet [35]	97.8	90.8	80.0	97.1	95.2	89.1	79.1	96.5
MVS-CNN	98.3	91.1	81.2	98.3	96.7	90.4	79.3	97.1
SEANet [55]	-				85.9	84.3	74.2	75.5

close to the ground-truth labels. Whereas, UNet [33], ResNet [34] and SEANet [55] have misclassified some of the pixels in either road regions or background regions.

The experimental results exhibit that our proposed MVS-CNN model shows supremacy in terms of model accuracy, processing time, and segmentation accuracy when compared to other state-of-the-art DL networks.

V. CONCLUSION

In this study, A MVS-CNN model has been proposed, which uses additional features such as gradient information along with the RGB channels of road images to train the network. The lesser number of convolutional layers in the proposed MVS-CNN architecture makes it faster and less complex, while additional features help the network to learn quickly without going deeper. The validation accuracy of the MVS-CNN comprising of six channels achieved an improvement of 2.7% as compared to the baseline three-channel CNN (RGB). The proposed MVS-CNN is evaluated in term of model accuracy, computational complexities and segmentation accuracy result. In comparison with ResNet [34], UNet [33] and SEANet [55], the proposed MVS-CNN outperforms while achieving comparable results to SegNet [35]. The proposed shallow architecture consisting of only seven convolutional layers, require prediction time of 0.6-0.8 millisecond (depending on the size of input image), which is less complex and faster as compared to other evaluated state-of-the-art DL architectures. In this study, only road labels are considered for evaluation and experimentation. In future work, the lane labels, as well as the multi-class segmentation approach will be considered to expand the scope of our proposed MVS-CNN. Similarly, other than gradient information of input images, additional different features representation (i.e., curvatures, blob, ridges, etc) will be also considered to enhance the performance of the proposed multi-feature view-based network architecture. This study deals with the driving image databases only, the performance of the proposed MVS-CNN architecture can be further evaluated on the driving video repository.

REFERENCES

[1] A. Daniel, A. Paul, A. Ahmad, and S. Rho, "Cooperative intelligence of vehicles for intelligent transportation systems (ITS)," *Wireless Pers. Commun.*, vol. 87, no. 2, pp. 461–484, Mar. 2016.

[2] L. Janušová and S. Čičmancová, "Improving safety of transportation by using intelligent transport systems," *Procedia Eng.*, vol. 134, pp. 14–22, Jan. 2016.

[3] G. Ros, S. Ramos, M. Granados, A. Bakhtiary, D. Vazquez, and A. M. Lopez, "Vision-based offline-online perception paradigm for autonomous driving," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 231–238.

[4] J. Brinkley, J. E. Gilbert, and S. B. Daily, "A survey of visually impaired consumers about self-driving vehicles," *J. Technol. Persons Disabilities*, vol. 6, pp. 274–283, Mar. 2018.

[5] Y. Yang, H. Luo, H. Xu, and F. Wu, "Towards real-time traffic sign detection and classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 2022–2031, Jul. 2016.

[6] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2110–2118.

[7] G. Mu, Z. Xinyu, L. Deyi, Z. Tianlei, and A. Lifeng, "Traffic light detection and recognition for autonomous vehicles," *J. China Univ. Posts Telecommun.*, vol. 22, no. 1, pp. 50–56, 2015.

[8] M. P. Philipsen, M. B. Jensen, A. Mogelmose, T. B. Moeslund, and M. M. Trivedi, "Traffic light detection: A learning algorithm and evaluations on challenging dataset," in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, Sep. 2015, pp. 2341–2345.

[9] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Towards reaching human performance in pedestrian detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 973–986, Apr. 2018.

[10] U. Ozgunalp, R. Fan, X. Ai, and N. Dahnoun, "Multiple lane detection algorithm based on novel dense vanishing point estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 621–632, Mar. 2017.

[11] B. Li, T. Zhang, and T. Xia, "Vehicle detection from 3D Lidar using fully convolutional network," 2016, *arXiv:1608.07916*. [Online]. Available: http://arxiv.org/abs/1608.07916

[12] A. C. Mersky and C. Samaras, "Fuel economy testing of autonomous vehicles," *Transp. Res. C, Emerg. Technol.*, vol. 65, pp. 31–48, Apr. 2016.

[13] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 354–370.

[14] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[15] H. A. Alhaija, S. K. Mustikovela, L. Mescheder, A. Geiger, and C. Rother, "Augmented reality meets deep learning for car instance segmentation in urban scenes," in *Proc. Brit. Mach. Vis. Conf.*, vol. 1, 2017, p. 2.

[16] J. M. Alvarez, A. M. Lopez, T. Gevers, and F. Lumbreras, "Combining priors, appearance, and context for road detection," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 3, pp. 1168–1178, Jun. 2014.

[17] M. Aly, "Real time detection of lane markers in urban streets," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2008, pp. 7–12.

[18] L. Deng, M. Yang, Y. Qian, C. Wang, and B. Wang, "CNN based semantic segmentation for urban traffic scenes using fisheye camera," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 231–236.

[19] Z. Kim, "Robust lane detection and tracking in challenging scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 16–26, Mar. 2008.

[20] X. Liu and Z. Deng, "Segmentation of drivable road using deep fully convolutional residual network with pyramid pooling," *Cogn. Comput.*, vol. 10, no. 2, pp. 272–281, Apr. 2018.

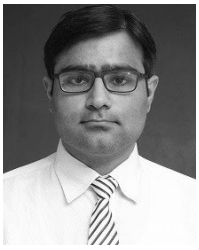
[21] C. C. T. Mendes, V. Fremont, and D. F. Wolf, "Exploiting fully convolutional neural networks for fast road detection," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 3174–3179.

[22] Y. Wang, E. K. Teoh, and D. Shen, "Lane detection and tracking using B-Snake," *Image Vis. Computing*, vol. 22, no. 4, pp. 269–280, Apr. 2004.

[23] L. Fang and X. Wang, "Lane boundary detection algorithm based on vector fuzzy connectedness," *Cogn. Comput.*, vol. 9, no. 5, pp. 634–645, Oct. 2017.
[24] W. Zhu, J. Miao, J. Hu, and L. Qing, "Vehicle detection in driving simulation using extreme learning machine," *Neurocomputing*, vol. 128, pp. 160–165, Mar. 2014.
[25] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 2241–2248.
[26] T. Malisiewicz, A. Gupta, and A. Efros, *Ensemble of Exemplar-SVMs for Object Detection and Beyond*. Figshare, 2011.
[27] L. Yuan, X. Wei, H. Shen, L.-L. Zeng, and D. Hu, "Multi-center brain imaging classification using a novel 3D CNN approach," *IEEE Access*, vol. 6, pp. 49925–49934, 2018.
[28] T. Zia, M. Ghafoor, S. A. Tariq, and I. A. Taj, "Robust fingerprint classification with Bayesian convolutional networks," *IET Image Process.*, vol. 13, no. 8, pp. 1280–1288, Jun. 2019.
[29] M. Ghafoor, S. A. Tariq, M. A. Bakr, Jibran, W. Ahmad, and T. Zia, "Perceptually lossless surgical telementoring system based on non-parametric segmentation," *J. Med. Imag. Health Inf.*, vol. 9, no. 3, pp. 464–473, Mar. 2019.
[30] A. Imran, J. Li, Y. Pei, J.-J. Yang, and Q. Wang, "Comparative analysis of vessel segmentation techniques in retinal images," *IEEE Access*, vol. 7, pp. 114862–114887, 2019.
[31] Z. Yi, T. Chang, S. Li, R. Liu, J. Zhang, and A. Hao, "Scene-aware deep networks for semantic segmentation of images," *IEEE Access*, vol. 7, pp. 69184–69193, 2019.
[32] A. Hassan, M. Ghafoor, S. A. Tariq, T. Zia, and W. Ahmad, "High efficiency video coding (HEVC)–based surgical telementoring system using shallow convolutional neural network," *J.s Digit. Imag.*, vol. 32, no. 6, pp. 1027–1043, 2019.
[33] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.
[34] 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.
[35] 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.
[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556
[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[38] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "A new deep convolutional neural network for fast hyperspectral image classification," *ISPRS J. Photogram. Remote Sens.*, vol. 145, pp. 120–147, Nov. 2018.
[39] B. Pan, Z. Shi, and X. Xu, "MugNet: Deep learning for hyperspectral image classification using limited samples," *ISPRS J. Photogram. Remote Sens.*, vol. 145, pp. 108–119, Nov. 2018.
[40] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.
[41] H. Chen, Q. Dou, L. Yu, J. Qin, and P.-A. Heng, "VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images," *NeuroImage*, vol. 170, pp. 446–455, Apr. 2018.
[42] B. Du, L. Ru, C. Wu, and L. Zhang, "Unsupervised deep slow feature analysis for change detection in multi-temporal remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 12, pp. 9976–9992, Dec. 2019.
[43] P. F. Alcantarilla, S. Stent, G. Ros, R. Arroyo, and R. Gherardi, "Street-view change detection with deconvolutional networks," *Auton. Robots*, vol. 42, no. 7, pp. 1301–1322, Oct. 2018.
[44] J. Shao, B. Du, C. Wu, and L. Zhang, "Can we track targets from space? A hybrid kernel correlation filter tracker for satellite video," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 8719–8731, Nov. 2019.
[45] J. Shao, B. Du, C. Wu, and L. Zhang, "Tracking objects from satellite videos: A velocity feature based correlation filter," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7860–7871, Oct. 2019.
[46] 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.
[47] M. Anthimopoulos, S. Christodoulidis, L. Ebner, T. Geiser, A. Christe, and S. Mougiakakou, "Semantic segmentation of pathological lung tissue with dilated fully convolutional networks," *IEEE J. Biomed. Health Inform.*, vol. 23, no. 2, pp. 714–722, Mar. 2019.
[48] M. Rezaei, K. Harmuth, W. Gierke, T. Kellermeier, M. Fischer, H. Yang, and C. Meinel, "A conditional adversarial network for semantic segmentation of brain tumor," in *Int. MICCAI Brainlesion Workshop*. Springer, 2017, pp. 241–252.
[49] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, pp. 1097–1105.
[50] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3150–3158.
[51] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[52] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[53] E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo, "Efficient ConvNet for real-time semantic segmentation," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 1789–1794.
[54] 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.
[55] Y. Xu, B. Du, L. Zhang, Q. Zhang, G. Wang, and L. Zhang, "Self-ensembling attention networks: Addressing domain shift for semantic segmentation," in *Proc. AAAI*, vol. 33, Aug. 2019., pp. 5581–5588.
[56] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3640–3649.
[57] J. Fritsch, T. Kuhnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC 2013)*, Oct. 2013, pp. 1693–1700.
[58] J. Han and C. Moraga, "The influence of the sigmoid function parameters on the speed of backpropagation learning," in *Proc. Int. Workshop Artif. Neural Netw.* Berlin, Germany: Springer, 1995, pp. 195–201.
[59] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.
[60] Y. LeCun, Y. Bengio, and G. J. N. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
[61] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.
[62] T. Tieleman and G. Hinton, "Lecture 6.5-RMSPROP: Divide the gradient by a running average of its recent magnitude," *COURSERA, Neural Netw. Mach. Learn.*, vol. 4, no. 2, pp. 26–31, 2012.
[63] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, Jun. 2013, vol. 30, no. 1, p. 3.
[64] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*. [Online]. Available: http://arxiv.org/abs/1502.03167
[65] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[66] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, and M. Isard, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Operating Syst. Design Implement. (OSDI)*, 2016, pp. 265–283.
[67] *Google Colaboratory*. Accessed: May 2, 2019. [Online]. Available: https://colab.research.google.com/

MUHAMMAD JUNAID received the MIT degree from Quaid-e-Azam University, Islamabad. He is currently pursuing the M.S. degree in CS with COMSATS University, Islamabad, Pakistan. His research interests are in data sciences, image processing, machine vision systems, and deep learning.

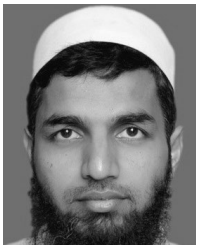
MUBEEN GHAFOOR received the Ph.D. degree in image processing from Mohammad Ali Jinnah University, Pakistan. Earlier, he was an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad. He is currently working as a Research Fellow in data sciences at the University of the West of England (UWE), Bristol, U.K. He has vast research and industrial experience in the fields of data sciences, image processing, machine vision systems, biometrics systems, signal analysis, GPU-based hardware design, and software system designing.



ALI HASSAN received the B.S. and M.S. degrees in computer science from COMSATS University, Islamabad. He is currently working as a Researcher with the Medical Imaging and Diagnostics Laboratory, COMSATS University. His areas of interests include image processing, light field compression, HEVC/H.265, artificial intelligence, computer vision, and deep learning.

SHEHZAD KHALID received the graduation degree from the Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Pakistan, in 2000, the M.Sc. degree from the National University of Science and Technology, Pakistan, in 2003, and the Ph.D. degree from the University of Manchester, U.K., in 2009. He has also authored various books and book chapters. He is currently a Professor and the Head of the Department of Computer Engineering. He is a qualified Academician and a Researcher with over 70 International publications in conferences and journals.

SYED ALI TARIQ received the B.S. degree in computer and information sciences from PIEAS, Pakistan, and the M.S. degree in computer science from Abasyn University, Islamabad. He is currently pursuing the Ph.D. degree with COMSATS University, Islamabad. He is also working at the Medical Imaging and Diagnostics Lab, COMSATS. His areas of interests include image processing, deep learning, bio-metric systems, and GPU-based parallel computing.



GHUFRAN AHMED received the Ph.D. degree from the Department of Computer Science, Mohammad Ali Jinnah University (renamed to Capital University of Science and Technology), Islamabad, in 2013.

During his Ph.D., he worked as a Visiting Research Scholar at the CReWMaN Laboratory, Department of Computer Science and Engineering, University of Texas at Arlington, from 2008 to 2009. He has been serving as an Associate Professor with the Department of Computer Science, FAST—National University of Computer and Emerging Sciences (NUCES), since January 2020. Before joining FAST, he has served as an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad, Pakistan. In 2016, he completed his postdoc from the Department of Computer Science and Digital Technology, Faculty of Engineering and Environment, Northumbria University, Newcastle Upon Tyne, U.K. His areas of research are the IoT, wireless sensor networks, and wireless body area networks. He has chaired a number of IEEE conferences and workshops such as iThing2019, UIC2019, IOP2019, and SCI2019. He is also serving as an Editorial Board Member of the Ad Hoc & Sensor Wireless Networks (AHSWN). He has served as a Lead Guest Editor for special sections in the *International Journal of Distributed Sensor Networks* (IJDSN), the *Journal of Sensors* (Hindawi), and the *Journal of Wireless Communication and Mobile Computing* (Hindawi). He is working as an Associate Editor of IEEE Access, an Academic Editor of the *Wireless Communications and Mobile Computing* (Hindawi), and the *Journal of Sensors* (Hindawi).

TEHSEEN ZIA received the Ph.D. degree, under the supervision of Prof. Dr. D. Dietrich at ICT, from the Vienna University of Technology, in 2010. He was an Assistant Professor with the Department of Computer Science, University of Sargodha, a Researcher at the Vienna University of Technology Austria, and a Lecturer at the University of Sargodha. He is currently an Assistant Professor with the Department of Computer Science, COMSATS University Islamabad, Pakistan. He was selected for Ph.D. scholarship by the Higher Education Commission Pakistan, in 2007. He is an approved Ph.D. supervisor from the Higher Education Commission of Pakistan and a Technical Reviewer of the ICT Research and Development projects. His research interests are in machine learning, particularly neural networks and probabilistic models with applications in computer vision, text, and sequential data processing.