RoadNet: An 80-mW Hardware Accelerator for Road Detection

Yuteng Zhou, Yecheng Lyu, and Xinming Huang, *Senior Member, IEEE*

***Abstract*—As a fundamental feature of intelligent vehicles, vision-based road detection must be executed on a real-time embedded platform with high accuracy. Road detection is often applied in conjunction with lane detection to determine the drivable regions. Although some existing research based on large deep learning models achieved high accuracy using the road detection dataset, they often did not consider the low power requirement of a typical embedded system. In this letter, an ultralow-power hardware accelerator for road detection is proposed. By adopting a top-down convolutional neural network (CNN) structure, a small CNN, namely RoadNet, is trained that can achieve near state-of-the-art detection accuracy. Furthermore, each CNN layer is trimmed to be computationally identical and every processing element in the architecture is fully utilized. When implemented using 32-nm process technology, the proposed hardware accelerator requires the chip area of 0.45 mm^2 and the power consumption of only 80 mW, which results in an equivalent power efficiency of about 300 GOP/s/W. The RoadNet chip is capable of processing 241 frames/s at 1080P image resolution. It stands out as an ultralow-power hardware accelerator in an embedded system for road detection.**

***Index Terms*—Computer vision, convolutional neural network (CNN), intelligent vehicle, low power, road detection, very large-scale integration.**

I. Introduction

ADVANCED driver assistant system (ADAS) is the dominant technology of intelligent vehicles that can improve road safety and reduce accidents. In recent years, many ADAS systems had been developed, such as vehicle detection, pedestrian detection [1], and road marking detection. In general, there are two major tasks in ADAS: 1) road/lane detection; and 2) obstacle detection [2]. The road detection is usually the initial step of ADAS, hence is more important than obstacle detection. As shown in Fig. 1, detecting lanes requires the road to be well structured, well-painted, and confusion free. In cases like unstructured road, unclear painted lane marker, intersection, construction zone, and detour, where lane detection cannot be applied very well, road detection is often employed to detect the drivable area [3].

Manuscript received April 5, 2018; accepted May 13, 2018. Date of publication May 27, 2018; date of current version February 26, 2019. This work was supported in part by the U.S. National Science Foundation under Grant 1626236, and in part by MathWorks. This manuscript was recommended for publication by Y. Chen. *(Corresponding author: Xinming Huang.)*

The authors are with the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA 01609 USA (e-mail: xhuang@wpi.edu).

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

Digital Object Identifier 10.1109/LES.2018.2841199

Vision-based road detection is a challenging task due to road appearance diversity, image clarity issues, and poor visibility conditions [2]. Some unpaved roads in rural areas make the task even harder. In order to solve the road detection problem, several sensing modalities are used, including monocular vision [4], stereo vision [5], and light detection and ranging [6]. Among these sensors, cameras are most widely used due to the fact that lane markings are made for human vision [2].

The real-time requirement and system power consumption limitation also pose challenges to road detection [7], making most of the road detection solutions less appealing when coming to real-world applications. In this letter, we aim to provide robustness to various driving conditions, real-time performance, and low power consumption at the same time. We propose a convolutional neural network (CNN)-based chip "RoadNet" for the road detection problem. Our contributions in this letter are as follows.

1) Proposing a top-down CNN structure for road detection. A regular CNN typically has a bottom-up structure in which only later CNN layers carry spatial information of the whole input image. In this letter, we adopted the top-down structure by introducing two additional channels with position information along with the RGB channels. As a result, the proposed CNN achieves near state-of-the-art performance through a mere four-layer network model.
2) Designing an ultralow-power hardware accelerator for road detection. We deliberately trim the CNN model such that each layer contains exactly the same number of neurons. Hence each processing element on the chip is fully utilized, achieving the power consumption of only 80 mW and power efficiency of about 300 GOP/s/W.
3) Proposing a highly extendable chip architecture for end-to-end image processing. Our proposed chip architecture is designed for road detection but is not restricted to it. With an elastic structure, the proposed architecture can cope with the computations of any number of CNN layers.

This letter is organized as follows. Section II introduces related work of road detection. Algorithms used in this letter are presented in Section III. Section IV presents the hardware architecture of our customized chip. Section V presents the software test results on KITTI road benchmark [8] and Cityscapes dataset [9]. Implementation results are shown in Section VI. Finally, Section VII concludes this letter.

Fig. 1. Different road/lane conditions in KITTI dataset. (a) Well structured lanes. (b) Unclear structured lanes. (c) Unstructured road. (d) Confusing lanes at intersection.

Fig. 2. Images with extreme light condition affecting clarity and visibility. (a) Overexposure. (b) Underexposure.

II. Related Work

Research on road detection has been around for decades. Previously, researchers adopted hand-crafted features as the main feature extraction technique [10]. In their work, roads were often modeled as a polygon with certain texture or color with lane markers [11]. Among those features, lane marker was the most important feature because of its unique color and shape. However, when lane markers were unclear or invisible, lane marker-based algorithms failed easily. In recent years, researchers switch to deep learning to solve vision-based tasks when hand-crafted features do not work well.

In the road detection/segmentation task, the likelihood of a pixel in the image being a road pixel is highly related to its spatial position. Computer vision approaches typically leverage region of interest [12], vanishing point and road contour [11] to process spatial information. Common deep learning approaches are resizing input [4], gridding map [13], and dilating convolutions [14]. However, these approaches still retain the bottom-up CNN structure. One major drawback of such structure is that a tensor can only "see" limited number of pixels due to the nature of convolution. In order to obtain the spatial information of the entire image, dozens of convolutional layers are cascaded until tensors on later layers get an eyesight over all pixels in the input image. The depth of a CNN model grows, which results a lot more weights and computations. Therefore, we opt to a top-down CNN structure for road detection.

III. Algorithms Design

The road detection/segmentation problem faces challenges arising from differed weather conditions, exposure conditions and shadows. As shown in Fig. 2, when over-exposure or under-exposure occurs, little texture can be captured for road detection. Simply relying on feature extraction cannot resolve the problem very effectively.

Owing to this issue, we seek for additional information that can help on detecting road. Spatial information is considered to be useful in the road detection task. By analyzing the heatmap of road samples from KITTI dataset, we develop a sense on the probability of a pixel to be a road pixel. As indicated in Fig. 3, pixels at bottom center are more likely to be road pixels. Detecting roads in scenes like shown in Fig. 2 could be amended by leveraging spatial information. This section mainly presents our approach on adopting spatial information into the CNN model as well as the procedure to find the best configuration for the CNN model.

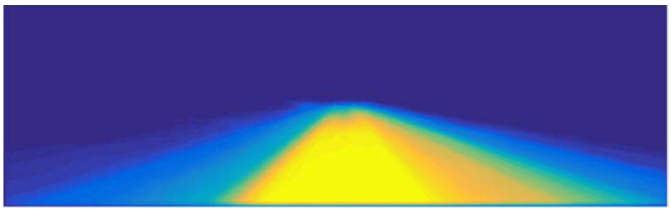

Fig. 3. Heatmap of a road example in KITTI dataset.

A. Fusion of Spatial Information

Inspired by Brust *et al.*'s [15] work, we use the top-down CNN structure instead of a traditional bottom-up structure for road detection. A tensor in the bottom-up structure gets to know its relative position on the entire image only after several convolutional layers, while tensors in a top-down structure knows their relative positions from the very beginning. Hence, the top-down structure fits the road detection task very well by fusing spatial features at the input.

In this letter, each pixel's position is labeled and concatenated to the input image. Input to the first convolutional layer contains five feature channels: 1) red; 2) green; 3) blue; 4) row coordinate; and 5) column coordinate. We test different configurations of CNN to validate the proposed method. The CNN architecture consists of multiple cascaded 5-by-5 convolutional layers followed by a 2-by-1-by-1 mapping layer. The reason we chose 5-by-5 convolution is a tradeoff between breadth and depth of convolutional layers in this task. We set cross entropy as the loss function, and adopt Adam optimizer with hyperparameters $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $lr = 10^{-4}$. Through the experiments with differed number of layers and differed feature sizes, we find that adding spatial information provides approximately 6% improvement on accuracy.

Furthermore, fusion with spatial information improves throughput on the hardware. The proposed hardware accelerator processes a single convolutional layer each time, which will be explained in Section IV. With the introduction of two positional channels, the CNN model requires less number of convolutional layers which effectively reduces the processing time. Meanwhile, processing two additional channels in hardware circuit design only increases the resource usage slightly.

B. RoadNet: Combined-Kernel Network

Our experiments show that four convolutional layers produce the best tradeoff between performance and complexity. Each convolutional layer is a group of parallel convolutional kernels. Further experiments show that a mixture of different

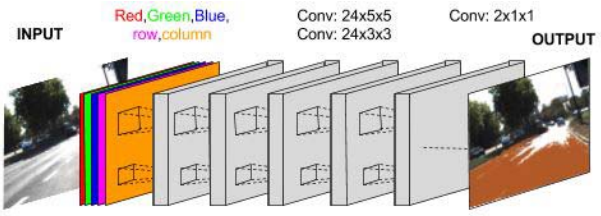

Fig. 4. RoadNet contains four convolutional layers and one fully connected layer.

kernel sizes improves the network performance. We choose to use 24 5-by-5 and 3-by-3 convolutional kernels. Dilated convolution with dilation factor of 2 is applied to each 5-by-5 convolution [14]. It is the best tradeoff between accuracy and number of parameters among multiple experiments with different kernel types. Fig. 4 shows the overall architecture of the RoadNet.

IV. Hardware Architecture

This section presents the hardware architecture of the RoadNet that is optimized to achieve both high throughput and low power for chip implementation.

A. AMBA Bus Support

Since the proposed chip is targeted for embedded environment and ARM microprocessor is the most widely used processor in the embedded ecosystem, the RoadNet chip is AMBA bus compatible. The AMBA AXI4 stream bus provides a very effective way for point-to-point high speed digital data transmission while still retains simplicity.

B. Position Generation

As presented in Section III, one novelty of our proposed CNN model is the introduction of two additional channels with spatial information, which significantly improves the performance of road detection. These two channels are the X channel and the Y channel, and they record corresponding pixel positions on x- and y-axis. Two counters are used to generate corresponding positions of each pixel in the input image. The X position counter works by increment at each valid signaling and by resetting itself at each end of line signaling. Similarly, the Y position counter works by increment at each end of line signaling and by resetting at end of frame signaling.

C. Elastic Architecture

Many existing works have been done on circuit design for CNNs, such as NeuFlow [16] and Origami [17]. These general neural network chips usually contain two parts: 1) execution; and 2) memory. The execution part typically contains an array of processing units, with each processing unit handling a few lines of pixel data. The memory part usually contains buffers, which store filter weights, pixel data, and intermediate results. The merit of such an architecture is that the chip area can be made very small with low power consumption, but it is still able to fit a very large CNN model such as AlexNet.

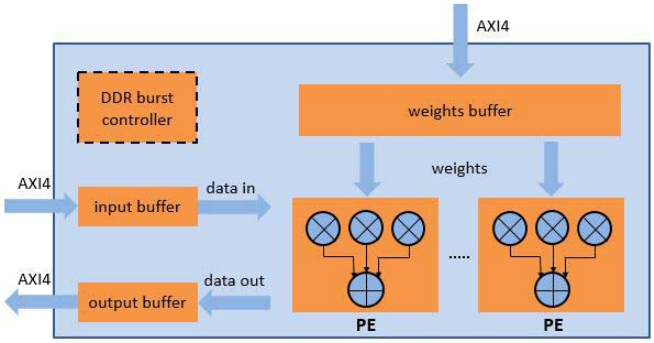

Fig. 5. RoadNet supports AMBA AXI4 interface, and contains buffers and processing elements.

Applying similar strategy, we propose a reusable architecture similar to the Origami CNN accelerator [17]. As stated in Section III, we intentionally train our neural network to have exactly the same number of convolutional modules at each layer. Such a structure is very hardware friendly and largely improves power efficiency. We only need to implement one convolutional layer on chip and reuse this structure for all the convolution layers repetitively. Such a neural network also simplifies the control logic compared to other modern neural network chips. Extra buffers are needed to store image pixels, filter weights, and intermediate results as shown in Fig. 5.

A total of 24 5-by-5 convolutional modules and 24 3-by-3 convolutional modules are used in the structure. A basic multiply-and-accumulate (MAC) unit contains three multipliers and one adder, and this is the critical path of the entire system, running at a maximum frequency of 500 MHz. Reducing the number of multipliers could improve the operating frequency of the MAC unit, but it would result a much larger adder tree that brings in extra latency and makes the overall throughput performance worse. Hence, we choose the MAC unit to contain three multipliers and one three-input adder.

D. Efficient Buffer Structure

In our proposed design, the latency for each processor element in Fig. 5 is four clock cycles. Latency of the following adder tree is five clock cycles. With a total of nine clock latency in the execution part, we can avoid a large size buffer since results can be generated in nine clock cycles. An efficient buffer structure is deigned to provide valid data to the execution part continuously. Taking a 3-by-3 kernel as an example, the entire feature map is split into nonoverlapping 3-by-3 sections, and each section is stored into nine discrete memories with the same addresses. A sliding window starts from the top left 3-by-3 window, and fetches nine values from nine discrete memories at address zero. As shown in Fig. 6, the sliding window shifts to its next position, and then the addresses of those three rightmost memories are incremented by one to fetch three newly encountered data in current sliding window. In this way, buffer is used intelligently and sliding window does not need to wait for additional cycles for incoming data.

The same methodology can be applied to 5-by-5 kernel as well. The only difference is that 25 discrete memories are

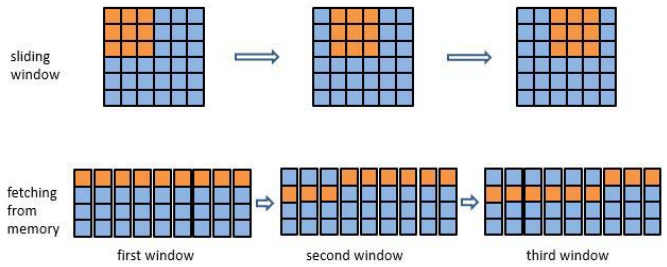

Fig. 6. Each 3-by-3 nonoverlapping window is stored into nine discrete memories. When the current windows slides, the addresses of three memories increment to fetch new data.

TABLE I
COMPARISON WITH OTHERS ON KITTI ROAD BENCHMARK

	Number of Convolution Layers	Number of Parameters	Average Precision
RoadNet	4	0.12M	91.60%
FCN-LC [18]	5	20M	85.83%
MAP [19]	16	20.5M	89.96%
SegNet [20]	26	1.4M	78.76%

needed. The proposed buffer structure is general and can be applied to any kernel sizes and any striding sizes.

V. PERFORMANCE EVALUATION

We evaluate the RoadNet on Cityscapes dataset [9] and KITTI road benchmark [8]. On Cityscapes dataset, our network is trained on 3475 samples from 21 video clips through 12 epochs, and tested on 1525 samples from six other video clips. As a result, the precision rate is 96.76% and the recall rate is 89.58%.

On KITTI dataset, we first augment 289 training samples to 20 500 samples and utilize the pretrained weights from Cityscapes. After 20 training epochs, we evaluate the CNN performance on 290 testing samples. As described in [8], the average precision (AP) is the key measurement to evaluate the performance of road detection algorithms. We compare RoadNet performance with some latest results in the literature. As presented in Table I, the RoadNet significantly reduces parameter usage yet still obtains near state-of-the-art AP.

VI. VLSI RESULTS

The RoadNet chip design is implemented with 32-nm process technology. The maximum operating frequency of the chip is about 500 MHz. When processing 1080P images, the chip achieves a constant throughput of about 241 frames/s. The estimated die size is 0.45 mm^2 and power consumption is 80 mW. At each clock cycle, the chip is capable of processing 48 MAC operations, which is equivalent to a power efficiency of about 300 GOP/s/W. Power consumption is an important factor for an embedded system design. The RoadNet accelerator is suitable for embedded system because of its ultralow-power consumption and elastic architecture.

VII. CONCLUSION

In this letter, an ultralow-power CNN-based hardware accelerator is designed for road detection in intelligent vehicles. By introducing two additional positional channels in a top-down architecture, the total number of layers and parameters of the CNN is largely reduced while it still retains good accuracy. Since all convolutional layers are in identical structure, the hardware-friendly architecture improves the power efficiency significantly. The proposed RoadNet chip is capable of processing real-time video with the resolution of 1080P at 241 frames/s. The chip design is targeted on 32-nm process technology and results the power consumption of only 80 mW. The RoadNet hardware architecture can be extended for many other CNN applications on low-power embedded platforms.

REFERENCES

[1] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 7, pp. 1239–1258, Jul. 2010.
[2] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: A survey," *Mach. Vis. Appl.*, vol. 25, no. 3, pp. 727–745, 2014.
[3] S. Yenikaya, G. Yenikaya, and E. Düven, "Keeping the vehicle on the road: A survey on on-road lane detection systems," *ACM Comput. Surveys*, vol. 46, no. 1, p. 2, 2013.
[4] G. L. Oliveira, C. Bollen, W. Burgard, and T. Brox, "Efficient and robust deep networks for semantic segmentation," *Int. J. Robot. Res.*, vol. 37, nos. 4–5, pp. 472–491, 2018, doi: 10.1177/0278364917710542.
[5] N. Einecke and J. Eggert, "Block-matching stereo with relaxed fronto-parallel assumption," in *Proc. IEEE Intell. Veh. Symp.*, IEEE, 2014, pp. 700–705.
[6] L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde, "Fast LIDAR-based road detection using fully convolutional neural networks," in *Proc. IEEE Intell. Veh. Symp.*, Los Angeles, CA, USA, 2017, pp. 1019–1024.
[7] R. Okuda, Y. Kajiwara, and K. Terashima, "A survey of technical trend of ADAS and autonomous driving," in *Proc. Techn. Program Int. Symp. VLSI Technol. Syst. Appl. (VLSI-TSA)*, IEEE, 2014, pp. 1–4.
[8] 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)*, IEEE, 2013, pp. 1693–1700.
[9] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 3213–3223.
[10] D. B. Lewis, J. M. Keller, M. Popescu, and K. Stone, "Dirt road segmentation using color and texture features in color imagery," in *Proc. IEEE Symp. Comput. Intell. Secur. Defence Appl. (CISDA)*, IEEE, 2012, pp. 1–6.
[11] J. Son, H. Yoo, S. Kim, and K. Sohn, "Real-time illumination invariant lane detection for lane departure warning system," *Expert Syst. Appl.*, vol. 42, no. 4, pp. 1816–1824, 2015.
[12] J. Zhao, X. Huang, and Y. Massoud, "An efficient real-time FPGA implementation for object detection," in *Proc. IEEE 12th Int. New Circuits Syst. Conf. (NEWCAS)*, IEEE, 2014, pp. 313–316.
[13] M. Teichmann, M. Weber, M. Zoellner, R. Cipolla, and R. Urtasun, "MultiNet: Real-time joint semantic reasoning for autonomous driving," *arXiv: 1612.07695 [cs.CV]*, 2016.
[14] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv:1511.07122[cs.CV]*, 2015.
[15] C.-A. Brust, S. Sickert, M. Simon, E. Rodner, and J. Denzler, "Convolutional patch networks with spatial prior for road detection and urban scene understanding," in *Proc. 10th Int. Conf. Comput. Vis. Theory Appl. (VISAPP)*, vol. 2, Berlin, Germany, 2015, pp. 510–517.
[16] P.-H. Pham *et al.*, "NeuFlow: Dataflow vision processing system-on-a-chip," in *Proc. IEEE 55th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, IEEE, 2012, pp. 1044–1047.
[17] L. Cavigelli *et al.*, "Origami: A convolutional network accelerator," in *Proc. 25th Ed. Great Lakes Symp. VLSI*, ACM, 2015, pp. 199–204.
[18] C. C. T. Mendes, V. Frémont, and D. F. Wolf, "Exploiting fully convolutional neural networks for fast road detection," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, IEEE, 2016, pp. 3174–3179.
[19] A. Laddha, M. K. Kocamaz, L. E. Navarro-Serment, and M. Hebert, "Map-supervised road detection," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2016, pp. 118–123.
[20] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *arXiv:1511.00561[cs.CV]*, 2015.