Image and Vision Computing 100 (2020) 103955

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

CrossFusion net: Deep 3D object detection based on RGB images and point clouds in autonomous driving

Dza-Shiang Hong [a], Hung-Hao Chen [a], Pei-Yung Hsiao [b,*], Li-Chen Fu [a], Siang-Min Siao [c]

[a] Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, Republic of China
[b] Department of Electrical Engineering, National University of Kaohsiung, Kaohsiung, Taiwan, Republic of China
[c] Automotive Research and Testing Center, Yunlin, Taiwan, Republic of China

ARTICLE INFO

Article history:
Received 29 December 2019
Received in revised form 20 May 2020
Accepted 21 May 2020
Available online 03 June 2020

Keywords:
Deep learning
3D object detection
Data fusion
Autonomous driving

ABSTRACT

In recent years, accurate 3D detection plays an important role in a lot of applications. Autonomous driving, for instance, is one of typical representatives. This paper aims to design an accurate 3D detector that takes both LiDAR point clouds and RGB images as inputs according to the fact that both LiDAR and camera have their own merits. A deep novel end-to-end two-stream learnable architecture, CrossFusion Net, is designed to exploit features from both LiDAR point clouds as well as RGB images through a hierarchical fusion structure. Specifically, CrossFusion Net utilizes bird's eye view (BEV) of point clouds through projection. Besides, these two feature maps of different streams are fused through the newly introduced CrossFusion(CF) layer. The proposed CF layer transforms feature maps of one stream to another based on the spatial relationship between the BEV and RGB images. Additionally, we apply attention mechanism on the transformed feature map and the original one to automatically decide the importance of the two feature maps from the two sensors. Experiments on the challenging KITTI car 3D detection benchmark and BEV detection benchmark show that the presented approach outperforms the other state-of-the-art methods in average precision(AP), specifically, as well as outperforms UberATG-ContFuse [3] of 8% AP in moderate 3D car detection. Furthermore, the proposed network learns an effective representation in perception of circumstances via RGB feature maps and BEV feature maps.

1. Introduction

Thanks to the rapid development of intelligent vehicles, autonomous driving becomes a popular issue in recent years. The most important issue for autonomous driving is to understand the surroundings of a vehicle, and one crucial key is 3D detection. By reasoning the surroundings of vehicles in 3D detection, the system can make the correct decisions under various kinds of situations. Nowadays, many of intelligent vehicles are equipped with multiple sensors at the same time, such as cameras, LiDAR and Inertial measurement unit (IMU). This motivates researchers to combine different sensors to conduct 3D object detection. In this work, the authors aim to design an accurate and stable 3D detector which is based on cameras and LiDARs. The effective fusion of RGB images and LiDAR point clouds should be capable of supplying much richer information.

Although 2D object detection has achieved great success on famous datasets, such as ImageNet [6], MS COCO [8] and KITTI [10], the 3D object detection remains an open problem because of an additional depth information. In most of the case, 2D object detectors, such as YOLO [12], Fast R-CNN [14], take only RGB images as inputs. However, the lack of depth could be a fatal flaw which leads to coarse results in 3D object detection. Hence, we propose a fusion-based network that takes advantages of mature 2D object detection methods. With the presence of LiDAR point clouds, the network is able to learn more representative information. Besides, each sensor has its own merits. Specifically, LiDAR is effective at providing depth information under various weather conditions while suffering from distant details. On the other hand, the camera preserves the detailed information of the front-view while suffering from various weather conditions. The purpose of this work is on 3D object detection that exploits both RGB images and point clouds in the on-road scene. Specifically, the study focuses on combining different sensors beneficial to each other.

Recently, some novel image-based methods explored the use of monocular [1,15–17] or stereo [2,4] images. Images usually provided detailed and dense measurements of front-view. However, these methods were limited by the loss of depth information. On the other hand, LiDAR based 3D object detection methods were rapidly developed afterwards. LiDAR brought way accurate depth information applying effective use of localization and shape description. Nevertheless, the point clouds were unordered and sparse. To deal with this problem, VoxelNet [7] and PointNet [19] grouped the points into voxel grids. Simony et al.

* Corresponding author.
E-mail address: pyhsiao@nuk.edu.tw (P.-Y. Hsiao).

https://doi.org/10.1016/j.imavis.2020.103955
0262-8856/

[9] and Yu et al. [20] projected point clouds to a ground such as bird's eye view(BEV) or front-view to avoid high computational cost of 3D convolution. In addition, the PointNet [19] directly processed point clouds through their permutation invariance. However, LiDAR suffered from distant detection due to its natural defects. As to fusion both RGB images and LiDAR point clouds methods, the ContFuse [3] successfully combined two streams of feature maps in different combinations of fusion.

The proposed CrossFusion Net is a 3D object detection network that takes RGB images and point clouds as inputs to make a valid use of both cameras and LiDARs. The presented CrossFusion Net is an end-to-end trainable architecture and capable of predicting accurate 3D bounding boxes. In addition, the novel CrossFusion layer enables the fusion between two streams of feature maps from different sensors in a cascading way. Through projecting all the points and pixels to its absolute coordinate in a 3D space, feature maps from different sensors could be passed to the other. Thus, by avoiding computational-cost 3D convolution, the 3D space relationship is kept between two kinds of feature maps during the CrossFusion layer. The presented network is evaluated by the tasks of both the 3D detection and the BEV detection benchmarks based on the popular KITTI on-road dataset. In this paper, the remaining parts are organized as follows. Section II introduces related works about RBG image based, point cloud based and fusion based methods to achieve 3D detection task. Section III mentions the formulation of the target task. Section IV proposes the overall architecture of the method. Section V elaborates the details of the proposed components. Section VI presents the experiments on the KITTI road dataset. Finally, Section VII gives a conclusion of the presented method.

2. Related works

The 3D object detection is a crucial part of intelligent transportation systems. Many works focusing on this topic come up with their solutions. After reviewing the existing works on the 3D object detection, they are basically divided into the following three categories according to the inputs.

2.1. RGB image based

RGB images provide texture and brightness information of the front-view in the form of pixel intensity; some research works directly predict 3D bounding boxes through RGB images. The 3DOP [21] inferred depth information according to stereo images and utilized mature R-CNN [22] structure to conclude the final prediction among the proposals. The Stereo R-CNN [2] extended Faster R-CNN [23] for stereo inputs to simultaneously detect images in all the views. The Pseudo-LiDAR [4] converted image-based depth maps to pseudo-LiDAR representations in order to mimic LiDAR signals and achieve impressive performance. However, it was hard to localize bounding boxes accurately due to the lack of depth information, especially in monocular images. The RGB image-based methods appeared relatively poor performance due to the pixel intensity being possible to vary under different appearances. However, it could provide distant information due to the innate advantages of cameras. As a result, by combining RGB images with point clouds, the presented method significantly improves the performance of 3D detection.

2.2. Point cloud based

LiDAR becomes an eye-catching sensor due to the rapid growth of sensing technology. Moreover, point clouds have some unique characteristics comparing to RGB images. One major difference is that point clouds are discrete and unordered. Both of the works of Yan et al. [5] and Zhou et al. [7] first grouped point clouds to voxel grids in 2018. VoxelNet [7] processed voxelwise representation via 3D convolution, which was known for its computational cost. Yan et al. [5] modified VoxelNet by applying sparse 3D convolution to achieve faster inference speed. Instead of grouping all the point clouds to voxel grids, PointNet [19] directly consumed point clouds and cleverly exploited the permutation invariance of points. Another way to deal with the preprocessing of point clouds was based on projection. VeloFCN [24] successfully projected the point clouds to front view and apply 2D fully convolutional network(FCN) [25] to reason 3D detection. The projection inevitably encountered information loss or distortion due to the data quantization along the projection axis. In this paper, the point clouds are projected to BEV and a channel of BEV maps is utilized to preserve the height information. In addition, the loss of projection is compensated by fusion with rich features of RGB images through the novel proposed CrossFusion layer.

2.3. Fusion based

As we known, only few works take both RGB images and LiDAR point clouds as inputs simultaneously. Qi et al. [18] presented the use of mature 2D objection detection in 2018. Those candidate bounding boxes proposed by 2D detector were then lifted to 3D frustum, and points inside the frustum were used to infer 3D detection results. In this case, the performance was bounded by 2D detector especially in highly occluded or truncated samples. Before then, the MV3D network [13] exploited the BEV, the front view of LiDAR point clouds and RGB images at the same time. The 3D object proposals were generated according to the BEV feature map and the corresponding feature of these inputs were appended together to infer 3D detection. As a consequence, this kind of fusion often made high-level feature being fused successfully while the low-level ones were neglected. Also in 2018, the deep fusion of the ContFuse [3] ingeniously designed a two-stream model, namely the RGB image stream and the BEV LiDAR stream. The feature maps of RGB images were fused onto the BEV feature maps in a cascading way. However, the fusion during the process was only a one-way fusion, which caused the lack of the symmetry between these two streams. In this work, we aim to design a symmetric two-stream network fusing between the RGB images and the BEV LiDAR point clouds. Additionally, the attention mechanism is applied between the two streams of feature maps to bring the two feature maps into a full play.

3. Problem formulation

The presented deep learning network simultaneously absorbs both RGB inputs of the images and the point clouds. The input RGB images can be represented as a set of integer pixel values V, where $V = \{v_{ij} | 1 \le i \le h, 1 \le j \le w\}$, h symbolizes the height and w stands for the width of images as well. Each element v_{ij} in the image is an integer within the range of [0, 255]. On the other hand, a point cloud can be parametrized as a set of points PC, where $PC = \{P_s | s = 1, 2, \ldots n\}$ and n represents the number of points in a point cloud. Furthermore, each point P is composed of a tensor of (x,y,z,r), where (x,y,z) is the coordinate with regard to the origin of coordinate system while r stands for the reflectiveness of the point P.

Given RGB images and point clouds as inputs, the goal is to predict accurate 3D detection which comprises both targets of localization and classification. Besides, the calibration matrix projecting a point cloud to the corresponding RGB image coordinate is known as another input parameter. The outputs of the network can be denoted as a set of 3D bounding boxes $BBox$, where $BBox = \{B_k | k = 1, 2, \ldots M\}$ and M symbolizes the number of 3D bounding boxes. Each 3D bounding box B, is denoted as $(x,y,z,w,h,l,\Theta,class)$, where (x,y,z) represents the center of the bounding box while (w,h,l) depicts the sizes of the bounding box. It is noteworthy that the unique assumption of the yaw rotation is measured by Θ. As for the $class$, it is a one-hot vector representing the possibility of the class and the bounding box belongs to.

Finally, the overall formula of the detection task T_{det} can be denoted as

$$T_{det}(V, PC) = BBox = \{B_k | \, k = 1, 2, \ldots, M \tag{1}$$

The goal is to propose a detection network which can generate the 3D bounding box *BBox* from the given RGB image *V* and the point cloud *PC*.

4. CrossFusion net

As more and more intelligent vehicles are equipped with both cameras and LiDARs, the CrossFusion Net is proposed to exploit the pros of these two different sensors. As shown in Fig. 1, the CrossFusion Net takes an RGB image and a point cloud to conduct the 3D object detection. Recently, Mono3D [1], Stereo R-CNN [2], Pseudo-LiDAR [4] and SECOND [5] have performed impressive results on 2D object detection based on RGB image feature maps. In contrast, Simony et al. [9] and Li et al. [24] achieved exceptional outcomes on 3D object detection based on the BEV feature maps. Due to the fact that the RGB image feature maps and the BEV feature maps are fit for 2D and 3D detection, respectively, the presented CrossFusion Net is designed to generate 3D proposals from the BEV feature maps which are more tightly fused with the RGB image feature maps within different levels.

4.1. Data preprocessing - encoding of point clouds

The raw 3D point clouds preserve the richest information in the form of a set of points. These points originally save the structure of the surroundings. However, nearly 100 k points are sparsely located over the whole 3D space. Moreover, the density across the whole 3D space varies from case to case. Taking these factors into consideration, it is inefficient in memory to directly process a raw point cloud as it usually often needs more complicated computation such like 3D convolution. Instead, another way to process a point cloud is to project it onto the BEV. Inspired by other works which also adopted point clouds as inputs, the points are removed out of the predefined region of interest(ROI). In the experiments, the ROI is set in the 3D LiDAR space with $X = (0, 80)$, $Y = (-40, 40)$, and $Z = (-2, 1.25)$. Any LiDAR point which is out of the given range will be removed; then the remaining points are projected onto the BEV with resolution limiting the BEV within size of 1024×1024. In other words, the BEV is subdivided into 1024×1024 grids. Particularly, the BEV is encoded as the form of height, reflectance and density, respectively. The coordinate and reflectance of the upmost point along with the corresponding density are recorded in each grid. In addition, the height map is normalized by dividing the height of ROI and also the density map is normalized as stated in Complex-Yolo [9].

4.2. CrossFusion layer

Fusion data between a point cloud and an RGB image is a challenging task. Eitel et al. [26] and Gupta et al. [27] exploited mature 2D detection frameworks with additional depth information by a projection manner to infer 3D detection. With a normal calibration projection matrix, the projection from a point cloud to an RGB image could be accomplished. This could be beneficial to the 2D detection. However, it still needed some modifications to successfully reason the 3D detection.

On the contrast, ContFuse [3] performed an exactly opposite fusing operation. It unprojected the BEV to RGB camera space and utilized the different levels of feature maps of the RGB images to realize the fusing operation. In this paper, we combine and modify these two methods of fusion mechanism and propose the CrossFusion layer for acting as a bridge between these two streams. As illustrated in Fig. 1, the fusion between the BEV and the image is performed in a cross way. With the aid of the presented CrossFusion layer, the spatial relationship between RGB images and point clouds can be fully connected together while avoiding time-consuming 3D convolution. The comprehensive details of the cross way fusion is described in the Section 5.

4.3. Loss function

From the aforementioned problem formulation, a 3D bounding box B can be parametrized as $(x, y, z, w, h, l, \Theta, class)$. As the anchor boxes [14] have shown the great progress on 2D and 3D detection, they are applied to the proposed CrossFusion network. Inspired by [14,28], the k-means on KITTI training dataset is performed to obtain the representative anchor boxes. Assume that there are N_{pos} positive anchor boxes and N_{neg} negative anchor boxes, while preparing anchor boxes from the ground truth, the positive samples and the negative samples can be balanced by randomly sampling from them if they are more than a predefined threshold. Note that only the positive anchor boxes are valid for regression. To fetch back the prediction from the corresponding

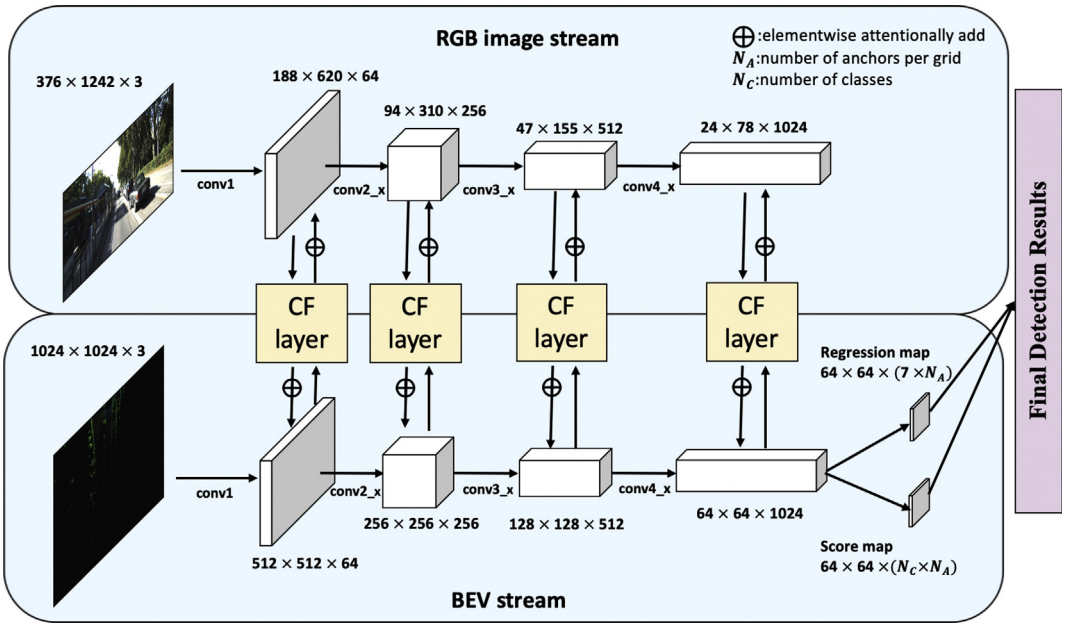


Fig. 1. Overview of the proposed CrossFusion Net.

regression tensor T_r, T_r is defined as $(\Delta x, \Delta y, \Delta z, \Delta w, \Delta h, \Delta l, \Delta\Theta)$, where Δx, Δy and Δz denote the regression for the center of a bounding box, Δw, Δh and Δl represent the regression size of a bounding box and $\Delta\Theta$ stands for the regression for yaw rotation. Here, Δx, Δy and Δz are encoded as

$$\Delta c = \frac{c_g - c_a}{e} \tag{2}$$

where subscript g denotes ground truth while subscript a stands for the anchor boxes. Besides, $c \in \{x,y,z\}$ and $e \in \{w,l,h\}$, respectively. In this situation, Δw, Δh and Δl are encoded as

$$\Delta s = \frac{s_g}{s_a} \tag{3}$$

where $s \in \{w,h,l\}$. As to $\Delta\Theta$, it can be encoded as

$$\Delta\theta = \theta_g - \theta_a \tag{4}$$

The Total loss L_t is defined as:

$$L_t = \frac{1}{N} L_{cls}\left(y_a, y_g\right) + \alpha \frac{1}{N_{pos}} L_{reg}\left(T_r^a, T_r^g\right) \tag{5}$$

where N represents the total number of samples. Namely, $N = N_{pos} + N_{neg}$, y_g denotes a binary class label while y_a stands for the prediction score output by softmax. α is a hyper parameter which controls the ratio of these two terms. L_{cls} symbolizes typical binary cross-entropy loss which can be expressed as

$$L_{cls}\left(y_a, y_g\right) = -\sum y_g \ log(y_a) + \left(1 - y_g\right) \ log(1 - y_a) \tag{6}$$

On the other hand, L_{reg} represents smooth L1 loss [14] which is defined as

$$L_{reg} = \begin{cases} 0.5 T_r^a - T_r^{g2}, \\ if \ |T_r^a - T_r^g| < 1 |T_r^a - T_r^g| - 0.5, \\ else. \end{cases} \tag{7}$$

5. Elaboration of CrossFusion layer

In order to fully exploit the potential of features of BEV images and RGB images and make them benefit each other, the proposed CrossFusion layer transforms the features from one to the other on the basis of their spatial relationship. In the following Subsections, the details of the CrossFusion layer are specified.

5.1. Mathematical formulation of CrossFusion layer

The proposed CrossFusion layer transforms the RGB image feature map and the BEV image feature map between each other at the same time as shown in Fig. 1. The task of fusion between the RGB images and the point clouds is composed of two operations, one from image to BEV(FI2B) T_{CF_FI2B} and the other from BEV to image(FB2I) T_{CF_FB2I}. Also, these two operations are performed simultaneously along with each convolutional block in the backbone network. The RGB image feature maps are denoted as F_{RGB} and the BEV image feature maps as F_{BEV}. After the transformation on the basis of their spatial relationship, BEV image feature maps and RGB image feature maps are encoded as F'_{BEV} and F'_{RGB}. Last but not the least, the transformed feature maps and the original feature maps are executed with elementwise attentionally addition denoted as $\oplus$. With the help of attention mechanism, two sources will benefit each other. As a result, the entire formula of the CrossFusion layer is listed as follows.

$$\begin{cases} T_{CF\_FI2B}(F_{RGB}) = F\prime_{BEV} \oplus F_{BEV} \\ T_{CF\_FB2I}(F_{BEV}) = F\prime_{RGB} \oplus F_{RGB} \end{cases} \tag{8}$$

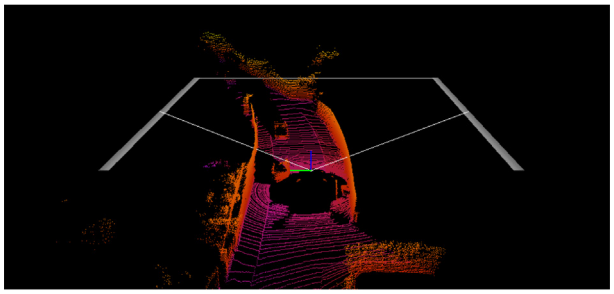
5.2. Preprocessing between RGB images and LiDAR points

As shown in Fig. 2, a map is adopted to save the LiDAR points due to mapping from RGB image to LiDAR coordinate being inevitable in the CrossFusion Net. As we known, there is no way to un-project a 2D RGB pixel to the corresponding LiDAR space without depth information. On the contrary, projecting a LiDAR point cloud onto the corresponding RGB image is a normal and practical scheme. In this case, for each image and LiDAR point cloud, the first step is to initialize an empty map with the same size as the RGB image. Secondly, a projection from the LiDAR point cloud onto the RGB image is achieved. Finally, the map preserving the coordinate of the point is projected onto the map. Considering that the projected point might not exactly match a certain integer coordinate in the RGB image in most of the cases, the projected point is directly rounded to integer for efficiency.

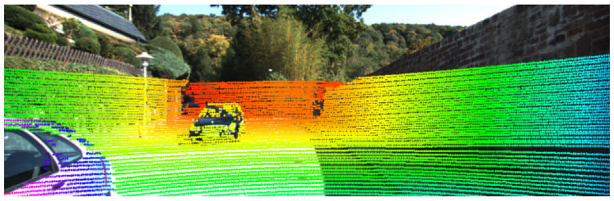
5.3. From image to BEV(FI2B)

As depicted in Fig. 3, the RGB image feature maps are transformed to the shape of the BEV image feature maps based on their spatial relationship instead of brute reshaping. Thereby, the elementwise attention is applied onto the two feature maps with a same shape but from different sources. As illustrated in Fig. 5, to decide the weights between two sources, a 1 × 1 convolution layer and a softmax layer is applied to

(a) Original image



(b) LiDAR point clouds



(c) 2D map to record LiDAR points

Fig. 2. Preprocessing of the relationship between (a) RGB image and (b) LiDAR points. (c) storing 3D LiDAR points that will be projected on left camera coordinate, as depicted in (b).

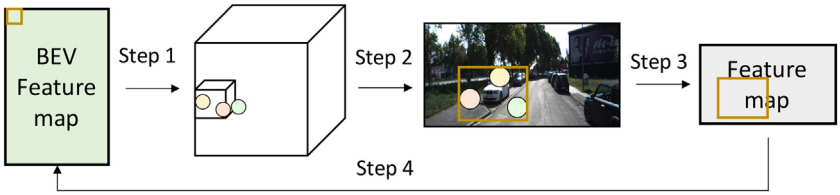


Fig. 3. Steps from image to BEV(FI2B) of the proposed CrossFusion layer: (Step 1) There are T neighbors determined from the corresponding anchor box. (Step 2) Next, these LiDAR points are projected onto the RGB image. (Step 3) Select the bounding box based on these LiDAR points. (Step 4) Take the bounding box and the RGB image feature maps as the inputs of ROI align.

produce the elementwise weights for the following addition operations. Also, the type of the adopted attention mechanism in our network is the most basic one. Because that from our experiments, different types of attention mechanisms give only less than 1% performance differences. Each pixel of the BEV feature maps is enumerated and picked out. Without loss of generality, the reception field of a pixel of the BEV feature maps is assumed to represent the compressed information inside the union of anchors. Hence, the points inside the anchors are targeted. In practice, the T points inside the multiple anchors are randomly selected out. Given that in some cases, the points inside the multiple anchors are less than T, they can be increased up to T by appending the nearest points with respect to the center of the multiple anchors. Then, these T points are projected onto the 2D image coordinate, and the bounding box is selected so that it can exactly cover these T points. However, these points might not exactly match the integer image coordinates. For this reason, according to the Mask R-CNN [29], given the bounding box and the corresponding feature maps, the ROI align is applied to deal with such a mismatch problem. The bounding box of the corresponding feature maps are further pooled into 1×1 to get the representative latent features of the RGB image feature maps. By visiting every pixel of the BEV feature maps and repeating the same procedure, the RGB image feature maps are transformed to the shape in a same way with the BEV feature maps. Now the attention mechanism can be employed on the transformed feature maps and the RGB image feature maps since they are with the same shape. With the benefit of the attention of two feature maps from two sources, each sensor is brought into full play.

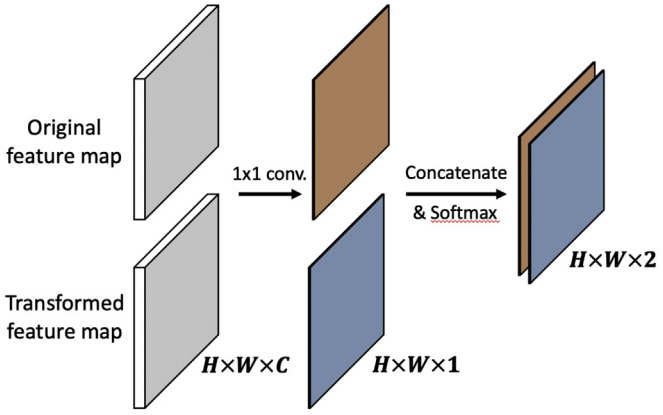


Fig. 5. Processing steps of generating elementwise weights between two feature maps. A 1×1 convolution is applied for dimension reduction and a successive softmax is utilized on each pixel location to construct the weights of two sources.

5.4. From BEV to image(FB2I)

As illustrated in Fig. 4, the BEV feature maps are transformed so that their shapes should be the same as those of the RGB image feature maps. Each pixel in the RGB image feature maps represents a certain size of grid of the original image according to the reception field. The size of the reception field varies from different stages of the feature maps and the encoders such as VGG Net [30], ResNet [31], etc. As the reception field of a pixel in the image feature maps is known, the pixel can be enumerated in the original image which belongs to the reception field in the image feature maps. The goal here is to find the 3D coordinate for each pixel. However, it is impossible to project from image coordinate to 3D space as the lack of depth information of camera. Instead, projecting a point cloud onto the camera coordinate is a feasible way. Thereby, for each pixel in the original image, the preprocessing 2D map is exploited to record the projecting relationship between the pixel in the image and the corresponding LiDAR points. Once the mapping relationship in the reception field of the RGB feature maps has been obtained, the location of each RGB image pixel in the BEV coordinate can be calculated as how the BEV is performed from point clouds. The bounding box is utilized to exactly determine an enclosed rectangle. As a result, the reception of an RGB feature map pixel could be mapped to the enclosed area in BEV via the above transformation. In addition, the ROI align is leveraged to cope with mismatch problems in the RGB image feature maps. Specifically, the whole enclosed rectangle and the RGB image feature maps are fed into ROI align together and are pooled into 1×1 to get the representative latent features of the BEV feature maps. By visiting every pixel of the RGB image feature maps and repeating the same procedure, the BEV feature maps are transformed to the shape in a same way with the RGB image feature maps. Furthermore, the attention mechanism can be exploited to the transformed feature maps and the BEV feature maps since they are with the same shape. Thereby, the RGB image feature is fused with the BEV feature in a spatially reasonable way.

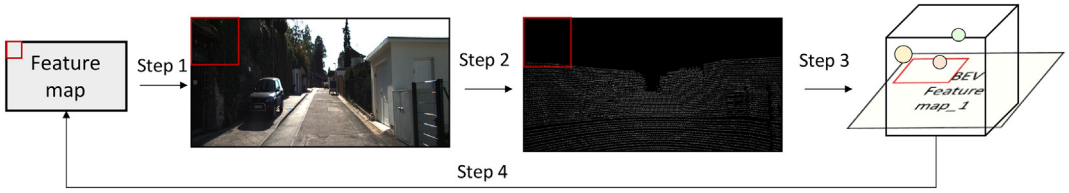


Fig. 4. Steps of from BEV to image (FB2I) of the proposed CrossFusion layer: (Step 1) The reception field of the pixel is targeted in the RGB image feature maps. (Step 2) Then the preprocessing map is utilized to map the pixel to LiDAR coordinate. (Step 3) Project the points onto BEV and select the bounding box. (Step 4) Take the bounding box and the BEV feature maps as the inputs of ROI align.

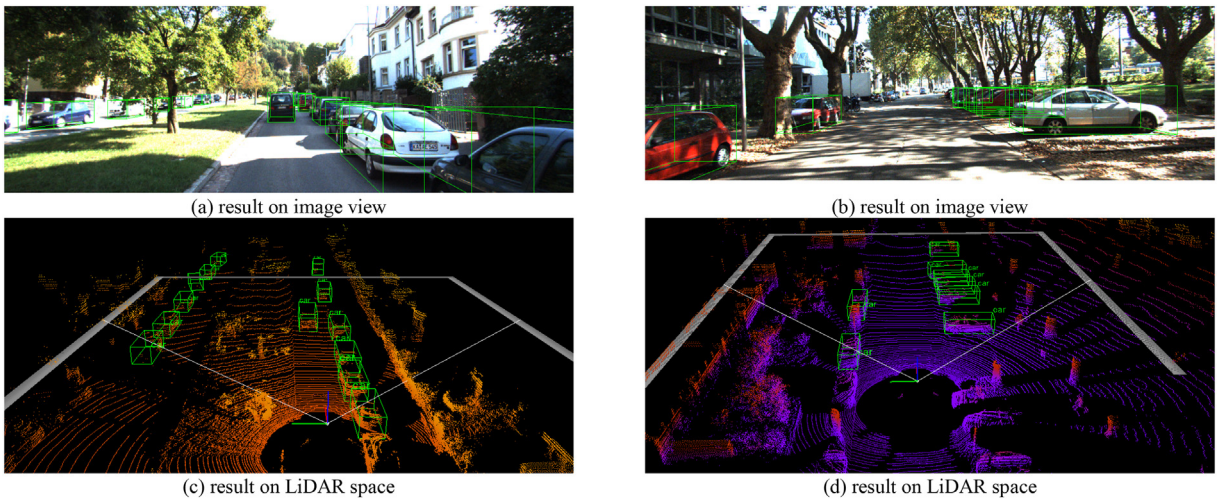


Fig. 6. Detection results of CrossFusion Net on KITTI dataset. (a)(c) are the same detection results from two sensors for a frame, while (b)(d) represent those for another frame. We project the results onto both image view and LiDAR space for visualization.

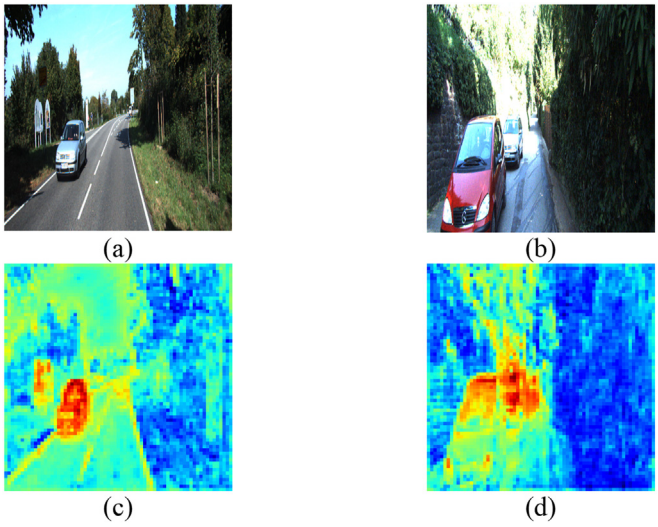


Fig. 7. Visualized attention maps of the CrossFusion Net on KITTI dataset. (a)(b) are original images, while (c)(d) represent the attention maps for the RGB stream.

5.5. Backbone network

In order to exploit mature technique of 2D detection, the ResNet50 is selected to be pre-trained on ImageNet as the backbone network. The feature maps generated by the bottleneck layers in the ResNet50 is exploited to group in 3,4 and 6. The dimensions of the feature maps are 256, 512 and 1024, respectively. Note that the width and height of the feature maps decrease to half after each grouped bottleneck layers.

5.6. Anchors setting

In 2D detectors, anchor boxes show that it is more efficient to regress boxes from the prior boxes rather than from the scratch. In this work, we modify the typical anchors with some key extensions. A set of prior 3D boxes is defined in BEV, which lies in LiDAR coordinate. Besides, the dimensions and the yaw rotation of the anchors are determined through k-means. Experimentally, it is observed that setting $k = 2$ can preserve the critical balance between the computational complexity and the performance.

6. Experimental results of the crossfusion net

The presented network is trained and tested on a personal computer with single NVIDIA GTX 1080 Ti GPU. The experiments are divided into three parts. Firstly, it begins with conducting the experiments on the challenging KITTI dataset. Secondly, an ablation study is given to evaluate the contribution of each proposed methods. Finally, the quantitative and qualitative visualization results are demonstrated by projecting the 3D bounding boxes onto 2D images. Moreover, the power and the limitations of the presented CrossFusion Net will be discussed.

6.1. KITTI object detection benchmark

The proposed CrossFusion Net has been evaluated on the KITTI object detection benchmark including the 3D object detection and the BEV detection tasks. In this benchmark, there are 7481 training data and 7518 testing data comprising images, point clouds and calibration files. The images are collected by a camera mounted on the top of a car, and the point clouds are sensed by a 3D laser scanner (Velodyne HDL-64E). In addition, each frame can be further categorized into three types of easy, medium and hard, which are classified separately, based on the distance, occlusion level and truncated level.

Moreover, the 3D detection results are verified by submitting to KITTI official test server. The Average Precision (AP) with 11 points is applied as the evaluation metric for both 3D object detection and BEV detection. Note that only the threshold of IOU is set to 0.7 in class Car. Other classes are set to 0.5. The proposed CrossFusion Net is also compared with other state-of-the-art methods in the car detection based on both 3D and BEV. These top-performing methods can be divided into three categories as mentioned in Section II of Related Works, including monocular or stereo image-based methods [1,2,4], LiDAR-based methods [5,7,9] and fusion-based methods [3,11,13,18].

6.1.1. Evaluation of 3D detection

The experimental results of the 3D detection on KITTI testing dataset is shown in Table 1. This task is more challenging than the BEV task since not only the flat localization issue but also the parameters of object height are crucial. For the 3D detection, the proposed method outperforms other methods in mAP.

6.1.2. Evaluation of BEV detection

The BEV detection results are compared with other state-of-the-art methods as shown in Table 1, which illustrates the accuracy comparison

Table 1
Resultant Average Precisions (in %) of 3D Detection on KITTI Dataset.

Method	Type of inputs	3D AP			BEV AP		
		Easy	Moderate	Hard	Easy	Moderate	Hard
Mono3D [1]	Monocular	2.53	2.31	2.31	5.22	5.19	4.13
Stereo R-CNN [2]	Stereo	49.23	34.05	28.39	61.67	43.87	36.44
Pseudo-LiDAR [4]	Stereo	55.40	37.17	31.37	66.83	47.20	40.30
SECOND [5]	LiDAR	83.13	73.66	66.20	88.13	79.40	77.95
VoxelNet [7]	LiDAR	81.97	65.46	62.85	81.97	65.46	62.85
Complex-YOLO [9]	LiDAR	55.63	49.44	44.13	76.62	67.14	65.92
AVOD [11]	Image + LiDAR	73.59	65.78	58.38	86.80	85.44	77.73
MV3D [13]	Image + LiDAR	66.77	52.73	51.31	86.02	76.90	68.49
UberATG-ContFuse [3]	Image + LiDAR	82.54	66.22	64.04	**88.81**	85.83	77.33
F-PointNet [18]	Image + LiDAR	81.20	70.39	62.19	88.70	84.00	75.33
Ours	Image + LiDAR	**83.20**	**74.50**	**67.01**	88.39	**86.17**	**78.23**

of the ability of localization. The proposed method gives best results in both case of Moderate and Hard among all of these top-performing methods.

6.1.3. Inference time

The resultant inference time of the proposed network is comparable with other methods, as depicted in Table 3, that regarding point clouds and RGB images as their inputs. From Table 3, our inference time of 100 ms is much less than that of Ref. [13,18], but only little more than those of Ref. [3,11]. Note that our framework inherently targets at providing outperformance in 3D detection in comparing with other state-of-the-art networks.

6.2. Ablation study

There are mainly three components that fuse two types of data in the presented CrossFusion Net, including FI2B, FB2I and attention mechanism. FI2B transforms the feature maps of RGB images stream to the shape of the BEV image feature maps based on the spatial correlation. Likewise, FB2I performs the same concept as FI2B with exactly opposite direction of data flow. Attention mechanism enables the network adaptively to learn the dynamically weighted features from two types of feature maps. To further explore the importance of these three components, conducting an ablation study is needed. Because of the limited submission policy of KITTI test server and the lack of the annotated ground truth of testing data, the presented ablation study is carried out exhaustively on the KITTI validation set. The protocol proposed in [13] is followed to split the training set and the validation set approximately on a fifty-fifty basis, which leads to adopt 3712 training frames and 3769 validation frames. This protocol prevents from sampling the same sequence involved in both the training set and the validation set.

Moreover, a bare BEV model without the aid of RGB images is dealt with in this study. In other words, only one stream of the proposed CrossFusion Net is activated. A second model is derived from the joined FB2I. This procedure allows the information of the BEV stream being passed to the RGB images stream. Besides, the attention mechanism is unified on the second model as the third model. Both FI2B and FB2I are employed together to become the fourth derived model. The fusion between these two streams is bidirectional in the model. Finally, the three components are combined as the last derived model. From Table 2, the model with all of the three components leads to the best results. The fourth model confirms the assumption that if two feature maps are brutally added together, it is hard for two input sources to benefit each other under strict conditions such as rainy scenes. With the favor of attention mechanism, it can help boost the performance. If attention is applied to the original feature maps without transforming them through FI2B and FB2I, the performance drops since the pixel-wise locations on two feature maps are not aligned together in the space. Therefore, the last model is concluded as the final version in this study and adopted to compare the model with other state-of-the-art methods.

6.3. Qualitative results and discussion

As shown in Fig. 6, by visualizing the results originally being set in the 3D camera coordinate through projecting the bounding boxes onto 2D image coordinate, some cases seem difficult to be predicted with only one source of data being the RGB image or the LiDAR point clouds. These cases are hard even for a human to infer the output such as parallel parked cars or severely occluded cars. Surprisingly, some of these types of data are trivial cases in the other source of data. It reminds that two sensors, cameras and LiDARs, can be placed in different positions on the car, namely, the occlusion is able to be decreased through the union of two different sensors. Most of all, through the proposed fusion mechanism in CrossFusion Net, the two types of feature maps will benefit each other. Consequently, all the models with the contribution of fusion outperform those without fusion as appeared in the results of the ablation study. In addition, attention maps of the RBG stream are visualized as depicted in Fig. 7. The attention weights are obviously dominated at the pixels which represent the cars in the images. As a result, the elementwise attentionally add of the proposed CrossFusion layer as shown in Fig. 1 can successfully fuse two feature maps specifically at foreground locations in order to make two sensors benefit with each other.

Table 2
Resultant Average Precisions (in %) of 3D Detection on KITTI Validation Set for Ablation Study.

Method	Type of inputs	3D AP		
		Easy	Moderate	Hard
Bare BEV stream (model 1)	LiDAR	78.02	66.72	61.37
FB2I (model 2)	Image + LiDAR	81.45	69.96	63.98
FB2I + attention (model 3)	Image + LiDAR	82.38	70.44	65.01
FI2B + FB2I (model 4)	Image + LiDAR	84.49	71.36	66.03
FI2B + FB2I + attention (model 5)	Image + LiDAR	**86.11**	**72.29**	**67.95**

Table 3
Resultant Inference Time of 3D Detection on KITTI Dataset.

Method	Inference Time (ms)
AVOD [11]	80
MV3D [13]	360
UberATG-ContFuse [3]	**60**
F-PointNet [18]	170
Ours	100

7. Conclusions

A novel end-to-end trainable fusion-based 3D object detection network, CrossFusion Network, is presented to take both RGB images and point clouds as inputs. Most of the existing fusion-based methods for 3D object detection do not fully take advantages of the spatial relationship between RGB images and point clouds. In this paper, the developed fusion method, CrossFusion layer, acts as a bridge between the RGB image feature maps and the BEV feature maps according to their absolute coordinate in 3D space. The CrossFusion layer plays an important role on each stage of the feature maps, which transforms one feature maps to the shape of another based on the spatial relationship. In addition, attention mechanism is applied on fusion to enable the network adaptively to choose the weights from the two sources of features. The experiments based on the KITTI dataset show the exceptional ability of the presented network in both 3D and BEV detection benchmarks over other state-of-the-art methods. Moreover, as the proposed CrossFusion layer is applied on feature-level, using other stronger feature extractor such as ResNet-101 or ResNext-101 is expected to be an alternative way to obtain better performance. Accordingly, the CrossFusion layer will continue to be further modified to fuse the raw point clouds and the RGB images based on various advanced architectures in our future research work.

Acknowledgement

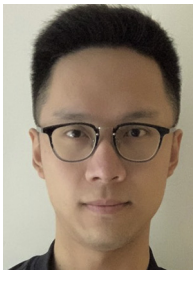
This work was partially sponsored by the Ministry of Science and Technology (MOST), Taiwan ROC, under Project 108-2634-F-002-016, 108-2634-F-002-017, 105-2221-E-390-024-MY3 and 108-2221-E-390-019-MY3. This research was also supported in part by the Center for AI & Advanced Robotics, National Taiwan University and the Joint Research Center for AI Technology and All Vista Healthcare under MOST.

Author contributions

Dza-Shiang Hong: Formal analysis, Software, Visualization, Writing.
Hung-Hao Chen: Visualization, Writing, Revising, Review & Editing.
Pei-Yung Hsiao: Investigation, Methodology, Supervision Administration, Funding Acquisition, Visualization, Revising, Review & Editing.
Li-Chen Fu: Conceptualization, Funding Acquisition, Investigation, Methodology, Resources, Supervision, Project Administration, Visualization.
Siang-Min Siao: Formal analysis, Software, Visualization, Revising.

References

[1] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, R. Urtasun, Monocular 3d object detection for autonomous driving, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2016, pp. 2147–2156.
[2] P. Li, X. Chen, S.J.a.p.a. Shen, Stereo R-CNN based 3D Object Detection for Autonomous Driving, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2019, pp. 7644–7652.
[3] M. Liang, B. Yang, S. Wang, R. Urtasun, Deep continuous fusion for multi-sensor 3d object detection, Proceedings of the European Conference on Computer Vision (ECCV) 2018, pp. 641–656.
[4] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, K.Q. Weinberger, Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2019, pp. 8445–8453.
[5] Y. Yan, Y. Mao, B.J.S. Li, Second: Sparsely embedded convolutional detection, 18 (10) (2018) 3337.
[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2009, pp. 248–255.
[7] Y. Zhou, O. Tuzel, Voxelnet: End-to-end learning for point cloud based 3d object detection, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018, pp. 4490–4499.
[8] T.-Y. Lin, et al., Microsoft COCO: Common Objects in Context, Proceedings of the European Conference on Computer Vision (ECCV) 2014, pp. 740–755 , Cham.
[9] M. Simony, S. Milzy, K. Amendey, H.-M. Gross, Complex-YOLO: an Euler-region-proposal for real-time 3D object detection on point clouds, Proceedings of the European Conference on Computer Vision (ECCV) 2018, pp. 197–209.
[10] A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? the kitti vision benchmark suite, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2012, pp. 3354–3361.
[11] J. Ku, M. Mozifian, J. Lee, A. Harakeh, S.L. Waslander, Joint 3d proposal generation and object detection from view aggregation, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2018, pp. 1–8.
[12] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2016, pp. 779–788.
[13] X. Chen, H. Ma, J. Wan, B. Li, T. Xia, Multi-view 3d object detection network for autonomous Driving, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017, pp. 1907–1915.
[14] R. Girshick, Fast r-cnn, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2015, pp. 1440–1448.
[15] F. Chabot, M. Chaouch, J. Rabarisoa, C. Teuliere, T. Chateau, Deep manta: A coarse-to-fine many-task network for joint 2d and 3d vehicle analysis from monocular image, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017, pp. 2040–2049.
[16] A. Mousavian, D. Anguelov, J. Flynn, J. Kosecka, 3d bounding box estimation using deep learning and geometry, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017, pp. 7074–7082.
[17] Y. Xiang, W. Choi, Y. Lin, S. Savarese, Data-driven 3d voxel patterns for object category recognition, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2015, pp. 1903–1911.
[18] C.R. Qi, W. Liu, C. Wu, H. Su, L.J. Guibas, Frustum pointnets for 3d object detection from rgb-d data, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2018, pp. 918–927.
[19] C.R. Qi, H. Su, K. Mo, L.J. Guibas, Pointnet: Deep learning on point sets for 3d classification and segmentation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017, pp. 652–660.
[20] S.-L. Yu, T. Westfechtel, R. Hamada, K. Ohno, S. Tadokoro, Vehicle detection and localization on bird's eye view elevation images using convolutional neural network, 2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR) 2017, pp. 102–109.
[21] X. Chen, et al., 3d object proposals for accurate object class detection, Advances in Neural Information Processing Systems 2015, pp. 424–432.
[22] R. Girshick, J. Donahue, T. Darrell, J. Malik, Rich feature hierarchies for accurate object detection and semantic segmentation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2014, pp. 580–587.
[23] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, Advances in neural information processing systems 2015, pp. 91–99.
[24] B. Li, 3d fully convolutional network for vehicle detection in point cloud, 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2017) 1513–1518 IEEE.
[25] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2015, pp. 3431–3440.
[26] A. Eitel, J.T. Springenberg, L. Spinello, M. Riedmiller, W. Burgard, Multimodal deep learning for robust RGB-D object recognition, 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2015, pp. 681–687.
[27] S. Gupta, R. Girshick, P. Arbeláez, J. Malik, Learning rich features from RGB-D images for object detection and segmentation, European Conference on Computer Vision 2014, pp. 345–360.
[28] J. Redmon, A. Farhadi, YOLO9000: better, faster, stronger, Proceedings of the IEEE conference on computer vision and pattern recognition 2017, pp. 7263–7271.
[29] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask R-CNN, IEEE International Conference on Computer Vision 2017, pp. 2980–2988.
[30] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556, 2014.
[31] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2016, pp. 770–778.



Dza-Shiang Hong received the B.S. degree in civil engineering and the M.S. degree in Computer Science and Information Engineering from National Taiwan University from National Taiwan University, Taipei, Taiwan, in 2015 and 2018, respectively. His research interests include deep learning and computer vision.

Hung-Hao Chen received the B.S. degree in Department of Computer Science and Engineering from National Chen Kung University, Tainan, Taiwan, in 2018. He is currently pursuing the M.S. degree with the Department of Computer Science and Engineering in Department of Computer Science and Engineering from National Taiwan University, Taipei, Taiwan,. His research interests include deep learning and computer vision.

Pei-Yung Hsiao(M'90) received the B.S. degree in chemical engineering from Tung Hai University, in 1980 and the M.S. and Ph.D. degrees in electrical engineering from the National Taiwan University, in 1987 and 1990, respectively. In 1990, he was an Associate Professor in the Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan. In 1998, he was the CEO of Aetex Biometric Corporation. He is currently a Professor in the Department of Electrical Engineering, National Univ. of Kaohsiung. His research interests and industrial experiences include VLSI/CAD, image processing, fingerprint recognition, visual detection, embedded systems, and FPGA rapid prototyping.

Li-Chen Fu (M'84-SM'94-F'04) received the B.S. degree from National Taiwan University in 1981, and the M.S. and Ph.D. degrees from the University of California, Berkeley, in 1985 and 1987, respectively. Since 1987, he has been on the faculty of and currently is a professor in both the Department of Electrical Engineering and Department of Computer Science & Information Engineering of National Taiwan University. He is now a senior member of both the Robotics and Automation Society and Automatic Control Society of IEEE, and he became an IEEE Fellow (F) in 2004. His areas of research interest include robotics, FMS scheduling, shop floor control, home automation, visual detection and tracking, E-commerce, and control theory & applications.

Siang-Min Siao received the M.S. and Ph.D. degrees in Electronic Engineering from National Yunlin University of Science & Technology, Yunlin, Taiwan, in 2011 and 2017 respectively. He is presently an engineer in the automotive research & testing center, Taiwan. His research interests include VLSI/CAD, digital circuit design, digital signal process, and algorithm analysis.