Image and Vision Computing 102 (2020) 103978

Contents lists available at ScienceDirect

Image and Vision Computing

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

A two-stage real-time YOLOv2-based road marking detector with lightweight spatial transformation-invariant classification

Xing-Yu Ye [a], Dza-Shiang Hong [a], Hung-Hao Chen [a], Pei-Yung Hsiao [b,*], Li-Chen Fu [a]

[a] Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, ROC
[b] Department of Electrical Engineering, National University of Kaohsiung, Kaohsiung, Taiwan, ROC

ARTICLE INFO

Article history:
Received 28 July 2019
Received in revised form 4 May 2020
Available online 12 July 2020

Keywords:
Deep learning
Road marking
Spatial transform
Real-time object detection
Object classification

ABSTRACT

In recent years, Autonomous Driving Systems (ADS) become more and more popular and reliable. Road markings are important for drivers and advanced driver assistance systems by better understanding the road environment. While the detection of road markings may suffer a lot from various illuminations, weather conditions and angles of view, most traditional road marking detection methods use fixed threshold to detect road markings, which is not robust enough to handle various situations in the real world. To deal with this problem, some deep learning-based real-time detection frameworks such as Single Shot Detector (SSD) and You Only Look Once (YOLO) are suitable for this task. However, these deep learning-based methods are data-driven even while there is no public road marking dataset. Besides, these detection frameworks usually struggle with distorted road markings and balancing between the precision and recall. We propose a two-stage YOLOv2-based network to tackle distorted road marking detection as well as to balance precision and recall. The proposed spatial transformer layer is able to handle the distorted road markings in the second stage, so as to achieve the improvement of precision. Our network is able to run at 58 FPS in a single GTX 1070 under diverse circumstances. Furthermore, we present a dataset for the public use of road marking detection tasks, which consists of 11,800 high-resolution images captured under different weather conditions. Specifically, the images are manually annotated into 13 classes with bounding boxes. We empirically demonstrate both mean average precision (mAP) and detection speed of our system over several baseline models.

1. Introduction

Thanks to the rapid development of intelligent vehicles, Autonomous Driving Systems (ADS) and Advanced Drive Assistance Systems (ADAS) play crucial roles in autonomous driving. The most important issue for these systems is to understand the surrounding environment in real-time. By doing so, the system can make the correct decisions under various kinds of situations. The road markings (RMs) are those important symbols painted on each lane of roads. The main purposes of road markings are guiding drivers to choose the correct lane before the intersection and providing necessary information of the road for drivers. Even if the driver uses an autonomous navigation system such as google map, the road markings are still crucial for the driver making correct decisions.

Although road marking can provide some deterministic information for autonomous navigation systems, ADS and ADAS, better understanding the road environment, the detection of road markings still remains a challenging problem. There are several difficulties in road marking detection tasks. Firstly, road markings are painted on the surface of roads, that the different angles of view will cause a totally different shape of the same road marking. Different dashboard cameras and different mounting positions of the dashboard cameras may cause different perspective distortions in the image. The camera parameters are not fixed in different real situations. Secondly, various illuminations and weather conditions may also lead to large variations in the visibility of road markings. For example, the colors of road markings reflect different colors of streetlights or other lights in the evening. Last but not least, large amounts of road markings on the road are already damaged after years of use. The detection system needs to handle blurred road markings, which can be caused by the road marking itself or the rain on the windshield. In this paper, we focus on solving the first and the second challenges by a novelty architecture that alleviates the distortion and self-collected dataset consisting of a great variety of urban scenes with different illuminations and weather conditions.

Current road marking detection systems mostly use traditional low-level features to process the image, like binarization, edge detection, color segmentation, etc. These methods may perform well in specific situations or small datasets, but it is hard to deal with intense illumination changes or heavy rain. For example, imagine a case that the streetlights

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

https://doi.org/10.1016/j.imavis.2020.103978
0262-8856/

are yellow and the stop light of the front car is red, the white road marking might reflect these colors to lead to failure of color segmentation. Besides, the rain drops on the windshield may also blur the image, leading to the failure of edge detection.

Inverse perspective mapping is a method often used for eliminating perspective distortions. However, the camera parameters which are used for performing inverse perspective mapping may vary from time to time in real world cases. Considering all these issues, traditional low-level feature-based methods are not robust enough to perform accurate road marking detection under complex situations in real world.

It is noteworthy that the current deep learning method such as Convolutional Neural Network (CNN) has become a powerful solution to the object detection problem. Classification architectures such as AlexNet [25] and Inception [32] achieve great success on large-scale dataset. Detection framework like Faster R-CNN [1] and Fast R-CNN [33] show their powerful detection results on PASCAL VOC [2] dataset. Among these approaches, Faster R-CNN abandons traditional selective search [34] and adopts region proposal network (RPN) to achieve excellent performance. Although Faster R-CNN is powerful enough to tackle the road marking detection problem, the inference speed is a fatal shortcoming since the road marking detection task requires real-time processing speed. On the other hand, the current state-of-the-art detection framework like Single Shot Detector (SSD) [3] and You Only Look Once (YOLO) [4] can be inferenced in real-time and still robust for road marking detection tasks. As we all know, the key issue for those deep learning methods to perform robust detection under various environments is a large and diverse training dataset. Besides, these real-time object detectors are hard to balance the recall and precision due to a predefined threshold used for distinguishing object proposals from non-object proposals. If we set a low threshold, the recall of the detection framework may increase with the dropping precision due to the increasing false negative proposals.

In addition, the deep learning methods are extremely data-driven, which means the performance of the deep learning object detector highly relies on the quality and quantity of its corresponding training dataset. Unfortunately, the available public dataset for road marking detection and classification is limited. For instance, The Road Marking Detection dataset [5] contained 1,403 images labeled with 11 classes, in which all images were taken during sunny days with clear view. The network might encounter the problem of overfitting to the particular environment if we trained a network on the Road Marking Detection dataset [5].

In this research, we present a new road marking dataset for road markings detection since there are no proper public datasets. The dataset is collected under various weather and illumination conditions in urban scene by a dashboard camera inside the windshield. The images are manually labeled into 13 classes with bounding boxes. Besides, we also propose a two-stage network to perform real-time detection of distorted road markings on the road. In the first stage, we modify the YOLOv2 [6] detection framework to fit our road marking detection task and produce more object proposals to increase the recall. In the second stage, we propose a lightweight transformation-invariant road marking classification network (RM-Net) to reclassify those uncertain samples from the first stage to increase the precision. The concept and flowchart of our road marking detection network is illustrated in Fig. 1. The experimental results show that the proposed network outperforms other baseline detection frameworks like Faster R-CNN in the road marking detection task.

2. Related work

The road marking detection is an important part of Intelligent Transportation Systems (ITS) [7]. Many works that focus on this topic come up with their solutions. We briefly review existing works on the road marking detection from RGB images.

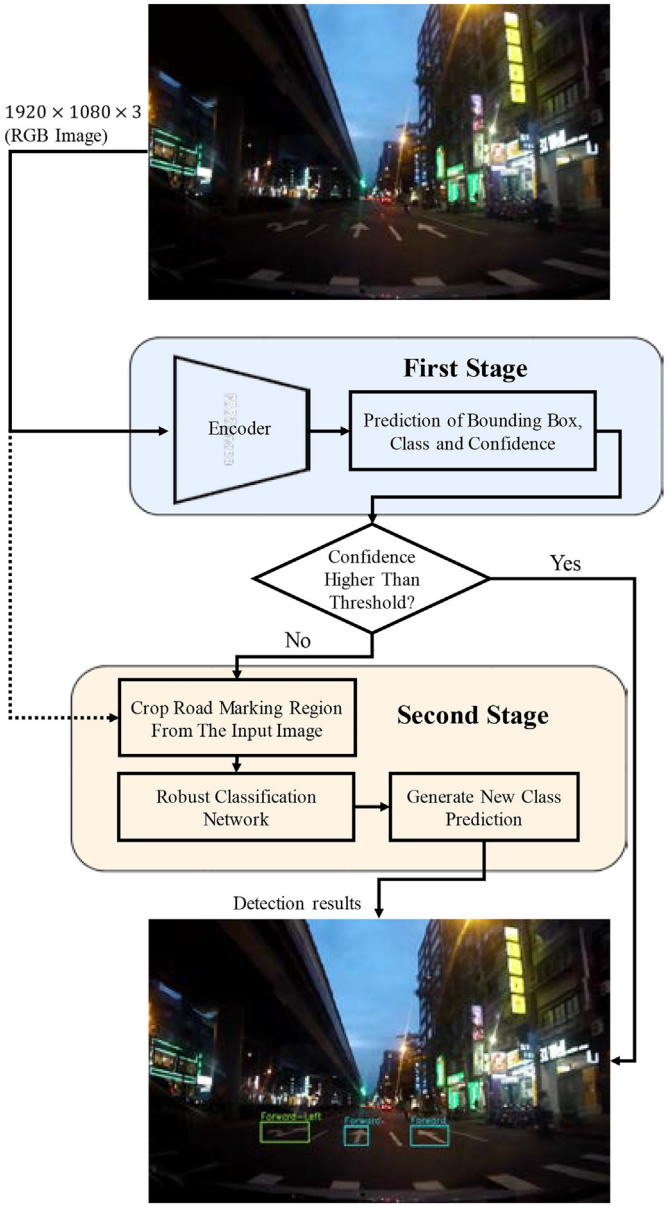

Fig. 1. Illustration of the proposed road marking detection network. The network consists of two stages. The first stage roughly predicts the bounding boxes and classes of road marking while the second stage reclassifies those bounding boxes with low confidence.

2.1. RM detection based on ROI and handcrafted features

Most existing methods rely on a pre-defined Region of Interest (ROI). Vacek et al. [8] and Suhr et al. [9] applied ROI in RM detection. In other words, these methods assumed that the RMs only appeared in some specific area, like the middle of the current driving lane. However, RMs lying on the left lane and right lane were also important for drivers. In addition to the ROI, Vacek et al. [8] took advantage of five pre-defined template RMs. Thereby, the detection could be done through matching each candidate to the five templates. Suhr et al. [9] investigated the horizontal projection for RM detection. They applied Histogram of Oriented Gradients (HOG) [10] to extract features from the road marking candidates. The authors then classified the road marking candidates by total error rate (TER) -based classifiers [11]. Nevertheless, it could only deal with the case that RMs were in front of the current driving lane and lacked the flexibility of identifying distorted RMs. In general, these methods based on handcrafted features might be

influenced by different angles of view or suffered from severe perspective distortion. In this work, we exploit convolutional neural networks to automatically define the types of features required for RMs.

2.2. RM detection based on inverse perspective mapping (IPM)

IPM is a common method in those low-level feature-based methods. Through IPM transformation, the perspective distortions can be well suppressed due to different angles of view and perspective. Besides, IPM is used for rectifying the original image and generating the bird's-eye view of a road. Liu et al. [12] performed IPM transformation to suppress the perspective distortion of input images. On top of IPM, several pre-defined road marking templates were used for performing sliding window to generate road marking candidate regions from binarized images created by bright slice extraction. To classify the road marking candidate regions, the author used ELM Classifier [13], which was a machine learning method with high training speed and suitable for multi-category classification tasks [14]. Ouerhani et al. [15] captured images from the VIAPIX acquisition module. The IPM method was applied to remove the perspective distortion of a road. In this paper, the author used color segmentation to get the object proposals under the assumption that all road markings were painted white. The HOG features and Support Vector Machine (SVM) [16] were applied to perform the final classification. Bailo et al. [17] also applied IPM to generate bird's-eye view before object proposals. In addition to IPM, the image was further transformed to gray image and the contrast was enhanced by Contrast-Limited Adaptive Histogram Equalization (CLAHE) [18] in order to remove the differences between contrasts in the image. Maximally Stable Extremal Regions (MSER) [19] was exploited to generate object proposals from enhanced image. The PCA network (PCANet) [20] was manipulated to encode each object proposal to a feature vector and the SVM classifier was utilized to get the classification results. However, the camera parameters for IPM were not fixed due to the different mounting positions of the dashboard camera despite it was an effective method to suppress perspective distortion. Our network differs from them by using the presented transformation-invariant network.

2.3. RM detection based on deep learning

The deep learning methods, such as CNN, usually outperform the traditional computer vision methods on object detection. Chen *el al.* [30] proposes a framework to carry out object classification by using binarized normed gradient (BING) [31] and PCA network (PCANet). Lee et al. [21] proposed a multi-task network for lane segmentation and road marking detection. This network took advantages of vanishing points to let the network learn more global information, which however required expensive computations. Our work shows how to achieve a robust RM detection in real time.

3. Our dataset with distorted RMs

As deep learning methods are mostly data-driven, it is important to train the deep learning network in a large-scale dataset with various circumstances. The model might over fit to the particular cases if the training data are lack of variety. Therefore, we collect a new dataset for the public use of road marking detection and classification under various roads, illuminations and weather conditions. The dataset is obtained during different time of a day with different weather conditions. The number of frames of the proposed dataset is shown in Table 1.

We mount the dashboard camera inside the vehicle to protect it from rain. Our dataset is mainly composed of urban scene. The original image resolution recorded by the dashboard camera is 2,560 × 1,440, and the frame rate is 25 frames per second. We extract 10 images every second randomly from the video and resize the image resolution to 1,920 × 1,080. Furthermore, the dataset contains all the locations of bounding boxes of every road marking with the corresponding class labels.

Table 1
Number of frames in our dataset. The dataset is composed of different combinations of weather conditions.

Time		Total frames
Daytime	Without rain	7,344
	Rain	1,456
Night	without rain	2,133
	Rain	867
Total		11,800

The presented dataset totally consists of 11,800 images collected under different time and weather conditions. These images are manually labeled into 13 classes with object bounding boxes as shown in Fig. 2. Additionally, we crop the road markings from the images based on bounding box information.

Table 2 lists the object numbers of each class in our dataset. Besides, we randomly crop 4,389 background proposals labeled with "Other". Although our dataset contains common road markings which can be found on the road, the imbalance of different classes is still existed. The road markings of class "Forward" appear the most frequently while class "Left Forward Right" appears rarely.

4. The proposed deep learning network

Traditional hand-crafted feature-based methods are generally limited to fitting some particular situations. Using a deep CNN training for a diverse large-scale dataset is a popular solution. Hence, a two-stage deep learning-based method to perform robust and real-time road marking detection on urban roads is proposed to deal with such problem. In the first stage, we present a YOLOv2-based detection framework to perform initial road marking detection of the input images with high recall in real-time. Each object proposal in this stage contains not only the coordinates of bounding boxes but also the corresponding

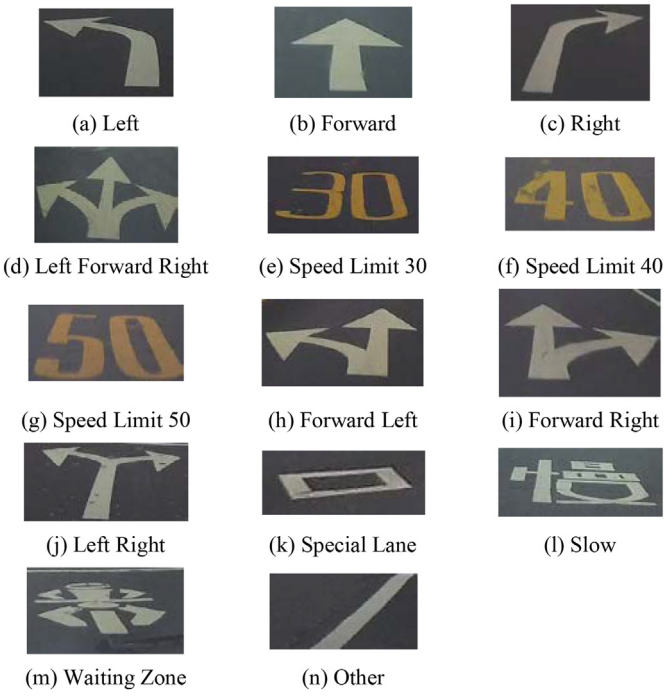

Fig. 2. Thirteen classes of the presented road marking dataset including the background. Each class is composed of various angles and suffers from distortions.

Table 2
Number of road marking objects.

Classes	Proposals
Left	1,651
Forward	7,525
Right	990
Left forward right	74
Speed limit 30	560
Speed Limit 40	408
Speed limit 50	1,155
Forward left	1,846
Forward right	2,712
Left right	108
Special lane	293
Slow	577
Motorcycle waiting zone	2,060
Other	4,389

confidences and class confidences, which are used for distinguishing uncertain samples from the other ones.

In the second stage, we present a novelty lightweight transformation-invariant road marking classification network (RM-Net) to reclassify the uncertain samples from the first stage. Through adding the RM-Net, the constraints are loosening by setting a low predefined threshold in the first stage. To the best of our knowledge, this paper is the first work that decomposes road marking detection task into two stages, in which the coarse outcomes are obtained firstly and the precise results are produced afterwards. By this way, the recall is increased from the first stage due to large amounts of proposals are generated. In case that too many false positives occur thus impacting the precision, the second stage is designed to eliminate false positives while maintaining high recall. As a result, the proposed novelty network can get high recall from the first stage and high precision from the second stage thus reaching the balance between these two metrics. The overall performance will also be benefited from the presented two-stage architecture.

4.1. Road marking detection stage

There are mainly two categories of detection frameworks. The first category of detection frameworks inherits from the famous work R-CNN [22] such as the Faster R-CNN [1]. Those detection frameworks are usually called two-stage detection frameworks, which generate region proposals firstly and then classify those region proposals properly. For example, the Faster R-CNN uses Region Proposal Network (RPN) to generate possible object regions from the input image. After the ROI pooling, those region proposals become a fixed size latent feature vector. In the end, the fully connected layer is used for generating the bounding box location and class. Although the Faster R-CNN is a state-of-the-art two-stage detection framework, the inference speed is its major disadvantage. If we want to apply deep learning-based detection framework to tasks like the ADAS or the road marking detection, the real-time inference speed is crucial.

Another category of detection framework only uses a single CNN to simultaneously predict multiple bounding boxes and the corresponding classes, which will be referred as a single shot detection framework. YOLOv2 is the state-of-the-art single shot detection framework. The most important feature of this kind of detection framework is the real-time inference speed. Moreover, the YOLOv2 detection framework outperforms the Faster R-CNN in both mean average precision and inference speed.

Inspired by the principle of the single shot detection framework YOLOv2, we construct our first stage in consideration of the importance of real-time inference speed in ADAS or self-driving system. The input image of our network is the front view of the dashboard camera. We adopt the darknet [23] as the feature extractor. The darknet is a deep CNN with 23 down-sample convolution layers designed to extract high level features from the input image. In addition, it is faster and more accurate than vanilla VGG network.

It is noteworthy that the anchor box is an important mechanism for YOLOv2 to achieve high recall. The anchor boxes are pre-defined, which can be regarded as an initial prediction on grid cells to predict the bounding boxes and their classes better. Specifically, the pre-defined scale and aspect ratios of the anchor boxes fit the object bounding boxes more, the higher performance can be achieved. In the region proposal network of Faster R-CNN, the scale and aspect ratios are manually picked, which is not accurate enough. Inspired by the anchor boxes selection method in YOLOv2, we use the dimension cluster to select the most appropriate scale and aspect ratios for the anchor boxes instead of choosing them manually. The k-means clustering method is performed on the bounding boxes of the training set in our proposed road marking detection dataset to empirically calculate the best anchor boxes. Notably, if we use standard k-means with Euclidean distance, larger bounding boxes lead to larger error than smaller boxes. Hence, the distance equation in k-means is based on the Intersection over Union (IoU) score as shown in Eq. (1).

$$d(box, centroid) = 1 - IoU(box, centroid) \tag{1}$$

We run k-means algorithm for various values of k and calculate the average IoU with closest centroid. The result is shown in Fig. 3(a). We choose k as 8 to achieve high recall in the first stage empirically while keeping a reasonable real-time inference speed of the network. We can make the IoU reach 0.79 between generated anchor boxes and the ground truth in our road marking detection dataset when k equals to 8. In Fig. 3(b), 8 anchor boxes generated by k-means cluster algorithm are illustrated.

Each anchor box takes a responsibility for predicting the potential bounding box information and the corresponding class. The bounding

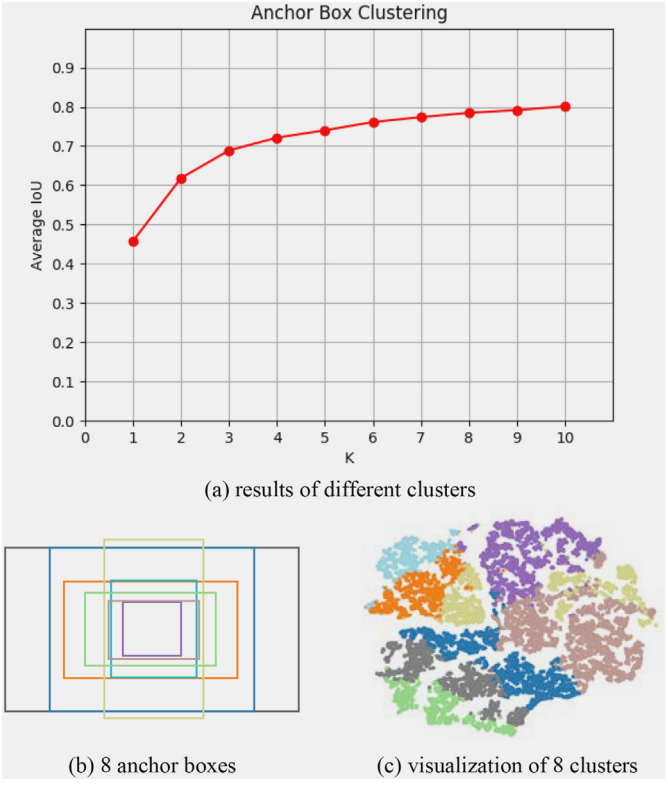

Fig. 3. Anchor box clustering using k-means. (a) relationship between number of clusters and the IOU of generated bounding boxes and the ground truth. (b) result of 8 anchor boxes and (c) the corresponding clusters which are visualized by TSNE.

box information is represented by 5 scalars of t_x, t_y, t_w, t_h and t_o, where t_x, t_y, t_w and t_h denote the bounding box coordinates, and t_o represents the bounding box confidence. The location coordinates of bounding boxes are predicted with respect to the location of the grid cell. In other words, the network predicts 5 scalars of the bounding box information based on each cell in the output feature map. If the top left corner of an anchor box A with height and width A_h, A_w located at (A_x,A_y) of the image, the prediction result of this anchor box can be calculated by Eq. (2).

$$b_x = \sigma(t_x) + A_x$$

$$b_y = \sigma(t_x) + A_y$$

$$b_w = A_w e^{t_w}$$

$$b_h = A_h e^{t_h} \tag{2}$$

where b_x, b_y, b_w and b_h are the center coordinates and the size of the bounding box, and σ is the sigmoid function which can be written as Eq. (3):

$$sigmoid(x) = \frac{e^x}{e^x + 1} \tag{3}$$

Eight anchor boxes are produced here. By combining the bounding box location prediction with the location of each anchor box, the training process of the network can be faster and more stable.

The final output dimension of the first stage is $(N_C + 5) \times N_A$, where N_C denotes the number of classes and N_A is the number of anchors. In our work, $N_C = 13$, $N_A = 8$. After investigating the detection results of the first stage, several problems are then revealed. For instance, the single shot detection framework lacks classes of background. In fact, the background is filtered by the bounding box confidence score. The confidence score can be written as following Eq. (4):

$$Pr(\text{object}) * IoU(b, \text{object}) = \sigma(t_o) \tag{4}$$

In other words, it reflects how confident the box contains an object. If the confidence score of the predicted bounding box is lower than the threshold, the predicted bounding box will be considered as the background and thus being ignored. The class prediction is generated by the softmax function, where the softmax function is defined as Eq. (5):

$$softmax_i = \frac{e^{V_i}}{\sum_j e^{V_j}} \tag{5}$$

The class corresponding to the maximum softmax value is the result of class prediction. Besides, we observed that the predicted bounding box confidence score and the class confidence score vary greatly under different road environments and angles of view. These facts often lead to a prediction with low confidence. Those predictions with low confidence usually are either false positive samples or incorrect results of class prediction. Thus, we have to design a second stage aiming to refine the detection results of our first stage.

4.2. Road marking classification stage

Object detection is more complicated than object classification, since the former takes the whole image as an input whereas the latter only takes a part of an image as its inputs. The results of detection are composed of both the object location information and the corresponding class. It means that the object detection framework firstly predicts where the possible objects are located in the image and then classifies the object region to a corresponding class. Thus, the object classification can be handled by rather a small network comparing to the detection task.

As a result, we design a lightweight transformation-invariant road marking classification network (RM-Net) in the second stage to resolve the bottlenecks of the detection framework in the first stage. Meanwhile, the proposed RM-Net in the second stage can also reclassify those uncertain samples from the first stage to increase the accuracy. The architecture of the RM-Net is shown in Fig. 4.

It is not difficult to classify road markings due to their plain color and distinguishable shape. The truly hard part for road marking classification is the severe perspective distortion caused by various angles of view. As shown in Fig. 5, although these four figures belong to two classes, road markings under different angles of view are totally with different appearance. IPM is a common method used in traditional computer vision-based works to eliminate the perspective distortion. IPM takes camera parameters to generate bird's-eye view of the road before further detection methods like template matching algorithm or horizontal projection process. However, in real-world application, camera parameters are not fixed due to different dashboard cameras, mounting positions and angles of view caused by car shaking. These factors lead to totally different camera parameters. That is the reason why IPM is not a robust method for road marking detection in real-world application.

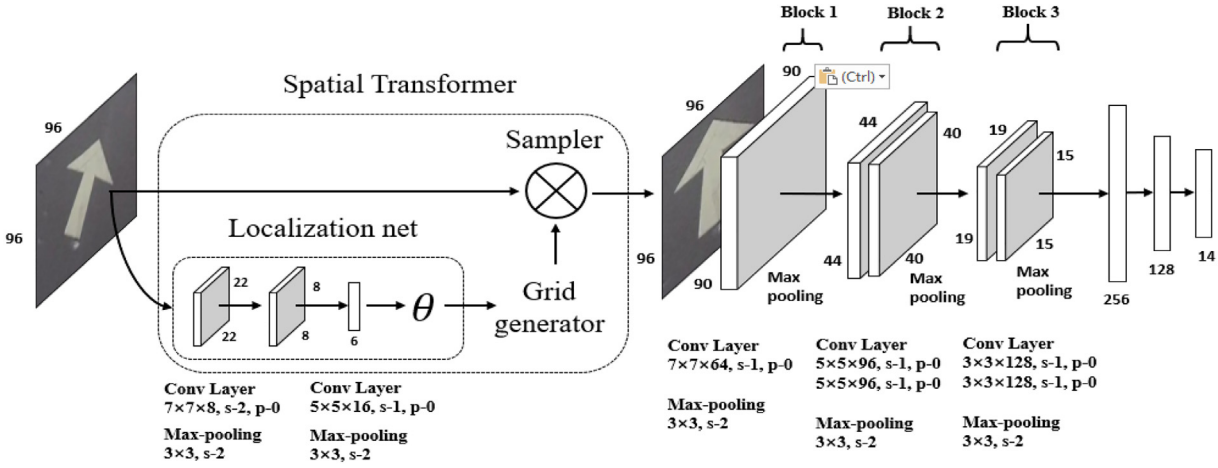

Fig. 4. Architecture of RM-Net. We crop the predicted road marking based on the first stage from the original image and use it as an input of the second stage. Inspired by [24], a spatial transformer is used for dealing with the distortion of road marking.

Fig. 5. Same road markings under different angles of view.

A general CNN is composed of convolution layers and max-pooling layers. These layers are used for extracting the latent feature from the input image. The down-sampling progress is able to get slight spatial invariance property due to the fact that the max-pooling layer only keeps the most important feature of the previous feature map. Although CNN has slight spatial invariance, still, it's not actually invariant to serious transformation of the input image.

Inspired by [24], we design a spatial transformer layer before convolutional layers in order to perform rectification of the input image to suppress the deformation caused by various angles of view. The spatial transformer layer is a differentiable module, thus the RM-Net can perform end-to-end training with back propagation. The spatial transformer layer can learn suitable transformation for each input image to make the classification more accurate.

The spatial transformer layer is composed of three main components, namely localization network, grid generator and sampler. The localization network is a fully convolutional neural network with two convolutional and max-pooling layers. The input of the localization network is the possible road marking region proposal cropped from the original high-resolution image based on the bounding box coordinates from the first stage. The reason why we crop the road marking region proposals from the original high-resolution images is that the input images will be resized to 416×416 pixels before it is fed into the detection network. If we crop the road marking patch from the 416×416 pixels image, the information of the cropped image is too poor. By cropping the road marking image from the original high-resolution image with 1920×1080 pixels, more fine-grained information can be preserved. Additionally, we also resize the input image size of the second stage to 96×96 pixels to reduce the computational cost. As shown in Fig.4, the output θ of the localization network is composed of 6 parameters for affine transformation of the grid generator.

The grid generator takes the 6 parameters to generate a gird of points. These points present where the input image pixel should be sampled to create the transformed output image. In the end, the sampler takes the input image and the sampling grid to produce the output image, which is sampled from the input image by grid points.

After the spatial transformer layer, the rectified image array is fed to the followed convolutional and max-pooling layers of the RM-Net to generate feature map which contains high level feature of the input road marking image for classification. We have chosen different filter sizes to extract features of the input road marking image and concluded that larger filter size performs better than a smaller filter size. Comparing to the smaller filter size, the larger filter size corresponds to a larger receptive field, which means more context information will be considered. In this work, we select 7×7, 5×5 and 3×3 filter sizes for the convolutional layers.

The Pooling layers in the convolutional neural network are often used for summarizing the most important feature from the feature map. The traditional max-pooling usually set the kernel size equal to the strides. For example, most of the time, the traditional max-pooling layer sets the kernel size $= 2$ and strides $= 2$. The pooling strategies we used in our RM-Net set the kernel size $= 3$ and strides $= 2$, which can consider more information between neighbor regions.

The aforementioned convolutional layers and the max-pooling layers are used for extracting high level feature from the input image. The final feature map will be flattened and then fed into the fully connected layers to generate the final class prediction.

The convolutional layers and max-pooling layers of the proposed RM-Net can be divided into three blocks. The first block contains one convolutional layer and one max-pooling layer, while the second and the third block contain two convolutional layers and one max-pooling layer.

The spatial transformer layer is a differentiable module, which not only transforms the input image but also can be applied to the mid-level feature map to perform transformation on the feature map. As we known, the spatial transformer layer adopted for road marking detection did not appear in the literature yet. To validate its effectiveness, this paper applies the spatial transformer layer to the RM-Net as well as presents six different architectures of the RM-Net as shown in Fig.7 to explore the potential novelty of the RM-Net. Instead of directly adding the spatial transformer layer to the presented RM-Net, an additional experiment was given to prove the improvement of the overall performance reflecting on the layer which applies to the different depth of the proposed network.

As demonstrated in Fig. 6, we select six different architectures by adding the spatial transformer layers into different parts of the network. The RM-Net-Zero network architecture uses normal CNN without spatial transformer layer; the RM-Net-One network architecture only performs spatial transformer layer to the input image. RM-Net-Two and RM-Net-Three network architectures apply spatial transformer layer to the input and one of the convolutional blocks. However, the RM-Net-Four network architecture applies spatial transformer layer after convolutional block1 and convolutional block 2, which means the spatial transformer layer only performs spatial transform to the mid-level feature map of the input image. The RM-Net-Five network architecture uses spatial transformer layer to transform both input images and the mid-level feature map.

We find out that the RM-Net-One achieves the highest accuracy in road marking classification task empirically. Thereby, we pick out the RM-Net-One architecture as the final version to perform road marking classification in the second stage.

4.3. Implementation of our two-stage network

In this subsection, we are going to describe the implementation issues of our two-stage detection network. The architecture of our two-stage road marking detection network is shown in Fig. 6.

We modify the output of each anchor box in the first stage based on YOLOv2 by adding the maximum class prediction confidence and predict the class label to generate a new vector. The maximum class prediction confidence is created by the softmax function. In addition, we also transform the location coordinates of the bounding box from input size 416×416 pixels to the original size 1920×1080 pixels in order to crop the high-resolution road marking proposals for the second classification stage.

The modified input vector in the second stage can be represented by 7 scalars of B_x, B_y, B_w, B_h, B_c, P_c and P_l. B_x, B_y, B_w, B_h and $B_c$denote the bounding box information and the corresponding confidence, respectively. Scalars P_c and P_l represent the class confidence and the predicted class label respectively.

The RM-Net in the second stage focuses on reclassifying the uncertain samples from the first stage, rather than reclassifying all the road marking proposals from the first stage. Only those uncertain samples are reclassified, which can save a lot of computational cost. We choose a predefined threshold α for B_c and P_c from the modified output of the first stage to distinguish certain samples and uncertain samples. Once the value B_c or P_c is lower than the threshold α in the output vector, we crop the uncertain road marking proposals from high-resolution image based on the predicted bounding box coordinates and employ

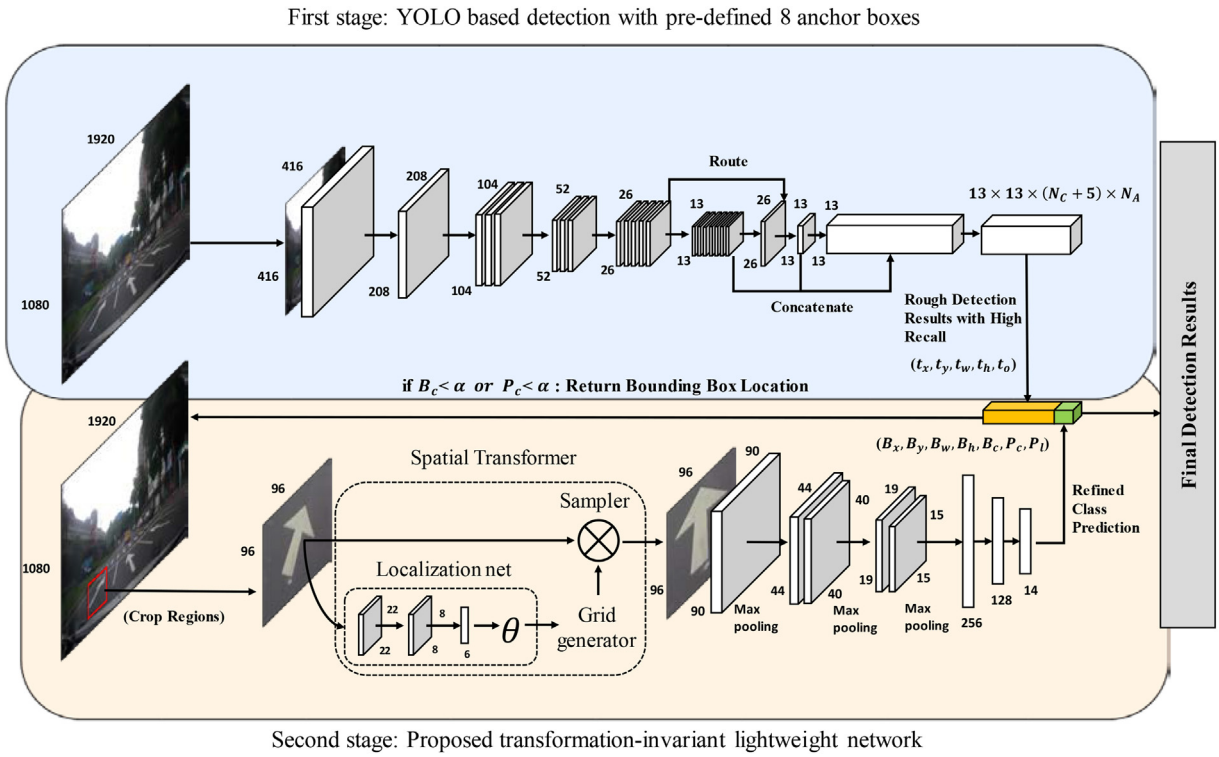

Fig. 6. Detailed architecture of our two-stage road marking detection system.

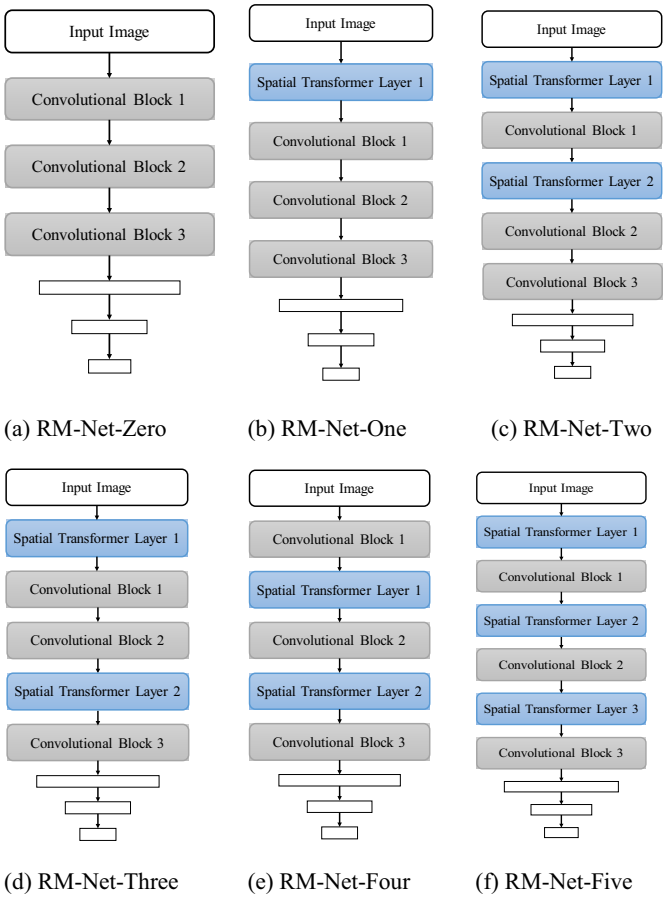

Fig. 7. Different architectures of RM-Net. We added spatial transformer into these RM-Nets in various combinations with different internal layers as well as blocks and concluded that (b) outperforms the others.

the RM-Net to reclassify the classification result. The predicted class of the second stage will replace the uncertain class prediction in the first stage. On the other hand, if values B_c and P_c from the first stage are greater than the threshold, our network directly considers it as final detection results and skips the second stage at all to save computational cost.

5. Experiments

Our network is trained and tested on personal computer with single NVIDIA GTX 1070 GPU, Intel Core i7–2600 3.4GHz CPU, 64G memory, and Ubuntu 16.04 operation system. We implement our method based on PyTorch, which is an open-source project for deep learning framework currently maintained by Facebook. PyTorch integrates NVIDIA Cuda and cudnn toolkit, which allows us to utilize stronger GPU acceleration. Our model is trained and tested by using Cuda 9.0 and cudnn 7 with PyTorch 0.4.0.

5.1. Experimental results of the RM-Net

The proposed RM-Net is a lightweight transformation-invariant road marking classification network, which is used for reclassifying uncertain samples from the first stage. The input of the RM-Net is a road marking object proposal while the output of RM-Net is one of the 13 classes of road marking or the background labeled as “Other”. Before the image is fed into the RM-Net, we resize the cropped image to 96×96 pixels and data argumentation to generate more diverse data to enhance the robustness and to prevent from overfitting. We implement data argumentation by adding random rotation to the input image with degree $-20°$ to $20°$. Despite the fact that our dataset contains various road marking samples with different angles of view, the random rotation is still a good way to increase the diversity of the data. On top of that, we randomly change the brightness, contrast and saturation of the input image to simulate the real-world condition.

Table 3
Experimental results of different RM-Net architectures.

Network	Accuracy
RM-Net-Zero	97.07%
RM-Net-One	**97.60%**
RM-Net-Two	97.35%
RM-Net-Three	97.36%
RM-Net-Four	97.22%
RM-Net-Five	97.46%

To explore the inference of spatial transformer layers in different part of the feature, we design six network architectures. We train these network architectures on our road marking classification dataset with 300 iterations until convergence and compare the overall classification accuracy. The experimental results are shown in Table 3. We find out that applying spatial transformer layer to the input image achieves the best classification performance.

Also, we train the RM-Net using Adam on our road marking classification dataset with 300 iterations and batch size 32. Adam is an optimization algorithm like stochastic gradient descent to update network weights iteratively based on the training data. For comparison, we train several classification networks like AlexNet [25], MobileNetv2 [26] and VGG16 [27] as baseline models. The AlexNet is the champion of the ILSVRC-2012 competition while the MobileNetv2 is a new architecture proposed by Google. VGG16 is a popular feature extractor which has been widely used in many detection frameworks. The input size of all three networks is 224×224 pixels, whereas the other training parameters remain the same as those of the RM-Net.

Furthermore, we implement two simple classifiers used in recent road marking detection works [28]. The former work uses the histogram of oriented gradients (HOG), and the support vector machines (SVM) to classify road marking images and the latter work uses a shallow CNN network modified from LeNet [29] called $\text{LeNet}_{96}\text{CP}_2$. The two methods are also compared with our RM-Net on the road marking classification dataset.

Table 4 shows that the proposed RM-Net achieves the best classification accuracy of 97.6% mAP in the road marking classification task in comparison with other state-of-the-art networks. Note that the overall performance is concluded in the red italic column shown in Table 4 as well as summarized from all classes in the dataset used in Table 4. Due to successful combining the spatial transformer layers, the proposed RM-Net effectively tackle the distortion problem, which is a main issue existed in the real-world scenarios. As the RM-Net is designed for solving the distortion problem in this work, we did not add the spatial transformer layer in other compared networks. However, the future work is encouraged to combine with other networks to further evaluate the potential effectiveness of the proposed RM-Net in this area of various applications.

5.2. Experimental results of the detection network

The proposed two-stage real-time road marking detection network is evaluated on our dataset. On top of that, we train two well-known real-time detection frameworks YOLOv2 and SSD as baseline models to compare with our network. Besides, the state-of-the-art of two stage detection framework Faster-RCNN is adopted to compare with our network.

The mean average precision (mAP) is used for evaluating the performance of the proposed two-stage road marking detection network. The prediction is considered as true positive if the bounding box overlapping rate between the prediction and the ground truth is greater than 0.5. The average precision for each class is also calculated. The experimental results are shown in Table 5. The best average precision of each class and mean average precision are marked with boldface. The result shows that our proposed two-stage road marking detection network achieves 86.3% mAP, which outperforms any other versions of detection frameworks. Besides, the overall performance of our proposed network achieves a significant improvement of at least 3.9% (86.3% - 82.4%) mAP in comparison with others as shown in Table 5. The contribution comes from our proposed network can deal with the distortion of the road makings, while others suffer from this annoying challenge. It is interesting to see that the one-stage YOLO-v2-based methods outperform two-stage Faster R-CNN in this case. In fact, Yolo-v2 [6] indeed announced higher mAP than the Faster R-CNN on Pascal VOC 2007 benchmark. Accordingly, it is the spatial transformer layer in the proposed two-stage network effectively contributes the performance improvement instead of the primitive two-stage architecture itself. Additionally, our network is capable of performing real-time road

Table 4
Classification results on our road marking dataset. Best results are highlighted in **bold**.

Method	*All*	Left	Forward	Right	L-F-R	F-L	F-R	L-R	30	40	50	Special Lane	Slow	Motorcycle zone	Other
Ours	***97.6***	**97.9**	**98.7**	**96.8**	**100**	96.8	98.7	**100**	94.3	**98.5**	97.4	98.4	**93.8**	**99.0**	89.1
AlexNet [25]	*96.2*	94.4	97.9	94.0	**100**	**97.8**	97.5	**100**	91.7	97.7	95.3	98.4	90.1	96.0	88.8
MobileNetv2 [26]	*97.0*	97.5	98.5	94.0	**100**	96.8	98.0	95.7	94.3	97.7	**98.6**	**99.2**	90.7	98.5	83.3
VGG16 [27]	*97.4*	97.5	**98.7**	96.1	**100**	97.6	**98.8**	**100**	**95.8**	93.9	97.7	**99.2**	90.7	97.0	**91.3**
HOG + SVM [12]	*81.2*	79.6	91.3	72.6	88.9	85.4	88.3	65.2	47.4	42.7	71.5	82.0	65.8	79.6	71.2
$\text{LeNet}_{96}\text{CP}_2$[13]	*88.8*	86.7	96.7	83.5	94.4	90.4	94.4	69.6	67.7	70.2	87.4	82.8	77.0	86.1	63.5

Table 5
Detection results on our road marking detection dataset. Best results are highlighted in **bold**.

Method	*All*	Left	Forward	Right	L-F-R	F-L	F-R	L-R	30	40	50	Special Lane	Slow	Motorcycle zone
Ours	***86.3***	**84.3**	**87.6**	**78.5**	**100**	**90.0**	88.9	**98.1**	**75.0**	**86.3**	**88.6**	87.0	**80.1**	77.0
Faster R-CNN [1]	*81.5*	77.6	87.4	71.3	95.7	88.8	**90.0**	88.0	72.8	81.8	81.0	79.4	78.0	67.6
SSD	*80.7*	79.8	87.4	63.5	94.1	83.5	85.7	85.4	64.8	79.5	82.4	**87.1**	74.1	**82.1**
YOLOv2 (Darknet)	*82.4*	79.8	81.7	70.0	99.5	88.9	88.2	94.5	74.6	80.6	86.1	80.1	73.9	71.0
YOLOV2 (Restnet50)	*79.0*	78.9	85.4	69.7	89.6	89.1	89.6	94.7	62.7	73.9	82.8	75.6	67.3	67.8
YOLOV2 (Tiny)	*75.5*	76.2	75.9	68.0	98.6	86.7	87.5	86.7	61.1	69.0	77.2	66.9	63.1	64.7

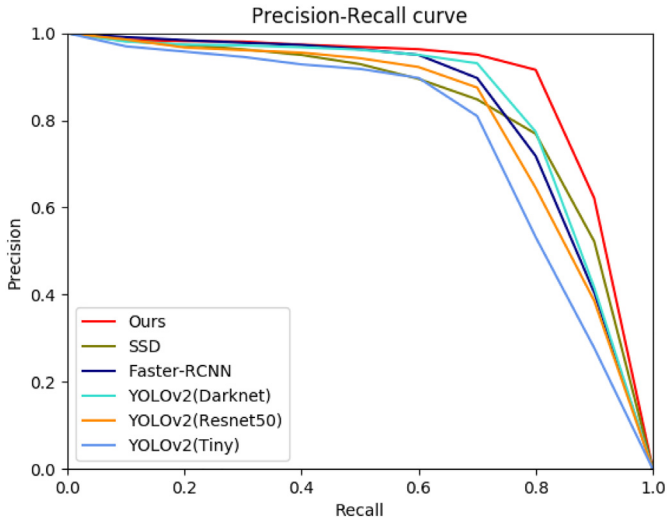

Fig. 8. The Precision-Recall curves of different road marking detection networks.

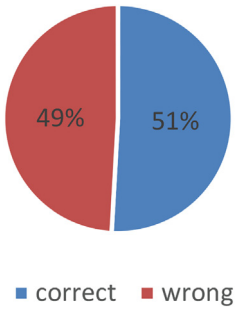

Fig. 9. The ratios of mispredictions in the first stage which can be correctly classified in the second stage.

marking detection as well as handling the second challenge of different illuminations and weather conditions based on the presented datasets.

Fig. 8 shows the precision-recall curve of our two-stage road marking detection network and other detection frameworks on our road marking detection dataset. The result verifies that the proposed two-stage road marking detection network outperforms any other detection frameworks. Through the effect of RM-Net, the proposed framework is able to eliminate the false positive proposals caused by low confidence threshold and still maintains high precision.

We compare the inference time between the proposed two-stage road marking detection network and other detection frameworks. The inference time is measured on our road marking detection dataset images with 416 × 416 pixels input image size of the network. We take the average inference time of each image in the test set of our dataset for comparison. The results are summarized in Table 6. The computing speed of our network is significantly faster than other two-stage detection frameworks and comparable with other one-stage detection frameworks. It only takes 0.017 s for two-stage network to perform accurate road marking detection of a single image. Besides, we investigate that the number of incorrect class predictions of the first stage can be correctly re-classified in the second stage. As shown in Fig. 9, more than half wrong predictions can be correctly predicted by RM-net.

5.3. Ablation study

The RM-Net is a key component of the proposed two-stage real-time road marking detection network and is employed to reclassify those samples whose confidence scores are lower than the specific threshold in the first stage. It can handle the distortion problem which might be the culprit of low performance. Since there are tremendous distorted road markings in real-world scenarios, the spatial transformer layer in the RM-Net is essential to handle such kind of task. The proposed RM-Net and the spatial transformer layer are tightly bounded together to meet with application requirements and to effectively accomplish the entire performance improvement. To estimate its contribution to performance, we train the entire network with/without RM-Net, respectively, to give an ablation study. The dataset set here we used is the same as the one for the detection task.

Table 7 shows that adding RM-Net obtains a significant improvement of 3.9% (86.3% - 82.4%) mAP. This is in our expectation because with the support of spatial transformer layers embedded in RM-Net, the objects are more likely to be detected from different angles of view.

Table 6
Comparison of inference time with other detection frameworks.

Approach	Stage	Inference Time
Ours	Two-stage	0.017 s
Faster R-CNN	Two-stage	0.072 s
SSD	One-stage	0.024 s
YOLOv2(Darknet19)	One-stage	0.015 s
YOLOv2(Resnet50)	One-stage	0.016 s
YOLOv2(Tiny)	One-stage	0.007 s

Table 7
Ablation on RM-Net.

Approach	mAP
w/o RM-Net	82.4
w/ RM-Net	86.3

6. Conclusions

We propose a YOLOv2-based two-stage real-time road marking detection network. Our two-stage road marking detection network solves several bottlenecks of the proposal-free detection frameworks by enhancing the recall of the first stage and reclassifying the predictions with low confidence. The first stage of our detection network is based on YOLOv2 with some key improvements. Each object proposal in the first stage contains bounding box confidence and class confidence, which are used for distinguishing the uncertain samples from the certain ones. In the second stage, we design a lightweight transformation-invariant road marking classification network (RM-Net) to reclassify the samples with relatively low confidence to further increase the precision. Besides, we present a dataset for distorted road marking detection and classification with more than eleven thousand high-resolution images captured under various illuminations and weather conditions.

The experiments are performed on our road marking detection and classification datasets. We evaluate RM-Net on the road marking classification dataset. The RM-Net achieves 97.5% overall accuracy, which is higher than the traditional classification methods and other CNN based classification networks. On the other hand, the presented two-stage road marking detection network achieves 86.5% mAP, which outperforms current real-time detection frameworks. Our experimental results demonstrate that the proposed network is able to perform real-time detection under extreme conditions. As a great deal of

convolutional neural networks have been developed nowadays, in our future work, the proposed RM-Net will be further integrated with different kinds of backbone networks to verify the consistency of their outperformances, from lightweight architecture such as MobileNet, to heavy structure like ResNext-101.

Declaration of Competing Interest

None.

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, 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.

References

[1] 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.
[2] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.
[3] W. Liu, et al., Ssd: Single shot multibox detector, Preceedings of the European Conference on Computer Vision, 2016.
[4] 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.
[5] T. Wu, A. Ranganathan, A practical system for road marking detection and recognition, Preceedings of the IEEE Intelligent Vehicles Symposium (IV), 2012.
[6] J. Redmon, A. Farhadi, YOLO9000: Better, Faster, Stronger, arXiv:1612.08242 2017.
[7] H. Chen, F.-Y. Wang, D. Zeng, Intelligence and security informatics for homeland security: information, communication, and transportation, IEEE Trans. Intell. Transp. Syst. 5 (4) (2004) 329–341.
[8] S. Vacek, C. Schimmel, R. Dillmann, Road-marking analysis for autonomous vehicle guidance, Preceedings of the European Conference on Mobile Robots (EMCR), 2007.
[9] J.K. Suhr, H.G. Jung, Fast symbolic road marking and stop-line detection for vehicle localization, Preceedings of the IEEE Intelligent Vehicles Symposium (IV), 2015.
[10] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, Preceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005.
[11] K.-A. Toh, H.-L. Eng, Between classification-error approximation and weighted least-squares learning, IEEE Trans. Pattern Anal. Mach. Intell. 30 (4) (2008) 658–669.
[12] W. Liu, J. Lv, B. Yu, W. Shang, H. Yuan, Multi-type road marking recognition using adaboost detection and extreme learning machine classification, Preceedings of the IEEE Intelligent Vehicles Symposium (IV), 2015.
[13] Z.-L. Sun, H. Wang, W.-S. Lau, G. Seet, D. Wang, Application of BW-ELM model on traffic sign recognition, Neurocomputing 128 (2014) 153–159.
[14] G.-B. Huang, D.H. Wang, Y. Lan, Extreme learning machines: a survey, Int. J. Mach. Learn. Cybern. 2 (2) (2011) 107–122.
[15] Y. Ouerhani, A. Alfalou, C. Brosseau, Road mark recognition using HOG-SVM and correlation, Optics and Photonics for Information Processing XI, 2017.
[16] J.A. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural. Process. Lett. 9 (3) (1999) 293–300.
[17] O. Bailo, S. Lee, F. Rameau, J.S. Yoon, I.S. Kweon, Robust road marking detection and recognition using density-based grouping and machine learning techniques, Preceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), 2017.
[18] K. Zuiderveld, Contrast limited adaptive histogram equalization, Graph. Gems (1994) 474–485.
[19] J. Matas, O. Chum, M. Urban, T. Pajdla, Robust wide-baseline stereo from maximally stable extremal regions, Image Vis. Comput. 22 (10) (2004) 761–767.
[20] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, Y. Ma, PCANet: a simple deep learning baseline for image classification, Preceed. IEEE Transac. Image Process. 24 (12) (2015) 5017–5032.
[21] S. Lee, et al., VPGNet: vanishing point guided network for lane and road marking detection and recognition, Preceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.
[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.
[23] J. Redmon, Darknet: Open Source Neural Networks in C, 2016 ([Online]. Available: https://pjreddie. com/darknet/. [Accessed 21 6 2017]).
[24] M. Jaderberg, K. Simonyan, A. Zisserman, Spatial transformer networks, Proceedings of the Advances in Neural Information Processing Systems, 2015.
[25] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Proceedings of the Advances in Neural Information Processing Systems, 2012.
[26] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.C. Chen, MobileNetV2: Inverted residuals and linear bottlenecks, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018.
[27] K. Simonyan, A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, *arXiv:1409.1556* 2014.
[28] T. Ahmad, D. Ilstrup, E. Emami, G. Bebis, Symbolic road marking recognition using convolutional neural networks, Proceedings of the IEEE Intelligent Vehicles Symposium (IV), 2017.
[29] Y. LeCun, LeNet-5, Convolutional Neural Networks, [Online]. Available http://yann.lecun.com/exdb/lenet 2015.
[30] T. Chen, Z. Chen, Q. Shi, X. Huang, Road marking detection and classification using machine learning algorithms, Preceedings of the IEEE Intelligent Vehicles Symposium (IV), 2015.
[31] M.-M. Cheng, Z. Zhang, W.-Y. Lin, P. Torr, BING: Binarized normed gradients for objectness estimation at 300fps, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014.
[32] SZEGEDY, Christian, et al., Going deeper with convolutions, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015.
[33] R. Girshick, Fast R-CNN, Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, 2015.
[34] J.R. Uijlings, K.E. Van De Sande, T. Gevers, A.W. Smeulders, Selective search for object recognition, Int. J. Comput. Vis. 104 (2) (2013) 154–171.

Xing-Yu Ye received B.S. degree in Computer and Communication Engineering, from Ming Chuan University, Taoyuan, Taiwan in 2016 and the M.S. degree in Computer Science and Information Engineering from National Taiwan University, Taipei, Taiwan in 2018. His main research interests are deep learning and computer vision.

Dza-Shiang Hong received the B.S. degree in civil engineering from National Taiwan University, Taipei, Taiwan, in 2015, where he is currently pursuing the M.S. degree with the Department of Computer Science and Engineering. 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.