electronics

MDPI

Article

Traffic Landmark Matching Framework for HD-Map Update: Dataset Training Case Study

Young-Kook Park [1], Hyunhee Park [1], Young-Su Woo [2], In-Gu Choi [3] and Seung-Soo Han [1,*]

[1] Department of Information and Communication Engineering, Myongji University, Yongin-si 17058, Korea; ypk2001@mju.ac.kr (Y.-K.P.); hhpark@mju.ac.kr (H.P.)
[2] R&D Center of Daitron Co., Ltd., 705, ENC Venture Dream Tower 6cha, Guro-dong, 41 Digital-ro 31gil, Seoul 08375, Korea; yswoo@daitron.co.kr
[3] SMART Highway R&D Division, Korea Expressway Corp., 24, Dongtansunhwan-daero 17-gil, Hwaseong-si 18489, Korea; guguci@ex.co.kr
* Correspondence: shan@mju.ac.kr

Abstract: High-definition (HD) maps determine the location of the vehicle under limited visibility based on the location information of safety signs detected by sensors. If a safety sign disappears or changes, incorrect information may be obtained. Thus, map data must be updated daily to prevent accidents. This study proposes a map update system (MUS) framework that maps objects detected by a road map detection system and the object present in the HD map. Based on traffic safety signs notified by the Korean National Police Agency, 151 types of objects, including traffic signs, traffic lights, and road markings, were annotated manually and semi-automatically. Approximately 3,000,000 annotations were trained based on the you only look once (YOLO) model, suitable for real-time detection by grouping safety signs with similar properties. The object coordinates were then extracted from the mobile mapping system point cloud, and the detection location accuracy was verified by comparing and evaluating the center point of the object detected in the MUS. The performance of the groups with and without specified properties was compared and their effectiveness was verified based on the dataset configuration. A model trained with a Korean road traffic dataset on our testbed achieved a group model of 95% mAP and no group model of 70.9% mAP.

Keywords: autonomous driving; YOLOv3; traffic sign; traffic dataset; HD map; object detection

Citation: Park, Y.-K.; Park, H.; Woo, Y.-S.; Choi, I.-G.; Han, S.-S. Traffic Landmark Matching Framework for HD-Map Update: Dataset Training Case Study. *Electronics* **2022**, *11*, 863. https://doi.org/10.3390/electronics11060863

Academic Editor: Nikolay Hinov

Received: 6 December 2021
Accepted: 7 March 2022
Published: 9 March 2022

1. Introduction

One of the important elements in an automated driving system (ADS) is the high-definition (HD) map embedded in a vehicle. When sensors installed in autonomous vehicles do not detect the surrounding situation, high-definition maps that include spatial information on roads and road facilities should be used. A method for maintaining and rapidly updating changes in object data contained within the HD map is a major challenge. The equipment required to update HD maps usually involves an MMS equipped with light detection and ranging (LiDAR), a global navigation satellite system (GNSS), an inertial navigation system (INS), and vision sensors [1,2]. Unfortunately, equipment related to map renewal is very expensive, costing up to hundreds of millions of dollars. Furthermore, much of the work, such as processing and matching the information acquired from the MMS equipment, and extracting and converting spatial objects, is performed manually. Therefore, an update system that can acquire road images and quickly update them by installing industrial cameras on many vehicles is required. Figure 1 shows the proposed update scenario. For example, in order to utilize vehicle resources such as public route buses, taxis, and express buses, the cost of the equipment has to be considered [3]. An increasing number of domestic and foreign companies have started acquiring road image information from vehicles and automating HD map updates using camera-based mobile mapping technology to provide rapid updates.

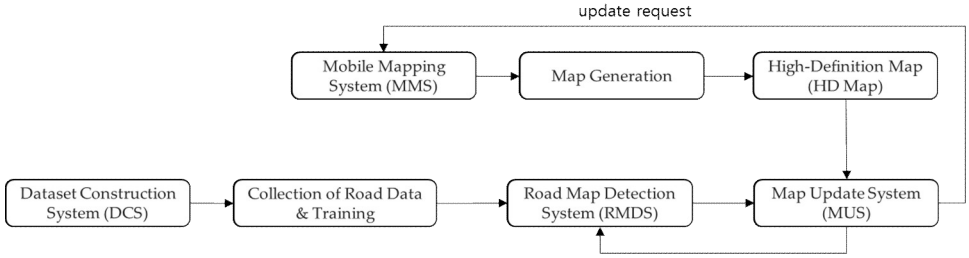

Figure 1. HD map generation and update scenario.

High detection rates and precision localization are essential for the automation of HD map updates. Commercial map providers, such as Mobileye [4], HERE [5], CARMERA [6], and TomTom [7], have already produced HD maps annotated with 3D geographic and semantic information on traffic landmarks with an accuracy of 10 cm [8].

In South Korea, the National Geographic Information Institute (NGII) built HD maps with an accuracy comparable to that of the HD maps provided by HERE and TomTom [9]. It is necessary to develop a detector that is as robust as possible to the environment for detecting road facilities in order to update the HD map generated with MMS equipment. In previous studies, the detection of traffic signs, road markings, and traffic lights have traditionally been developed based on computer vision.

Typically, traffic signs, lights, and road markings are designed to be easily distinguished from their surroundings [10].

The computer vision algorithms used to detect traffic signs, lights (signals), and road markings are commonly divided into three types. As shown in Figure 2, extensive research has been conducted using color-based methods (e.g., color thresholding, region growing, color indexing, dynamic pixel aggregation, and CIECAM97 model), shape-based methods (e.g., Hough transformation, similarity detection, distance transform matching, edge detection features, and Haar-like features), and hybrid methods (i.e., color- and shape-based features) [11]. However, these algorithms lack robustness because their detection performance depends on the climatic conditions, such as weather, sunsets, and sunrises.

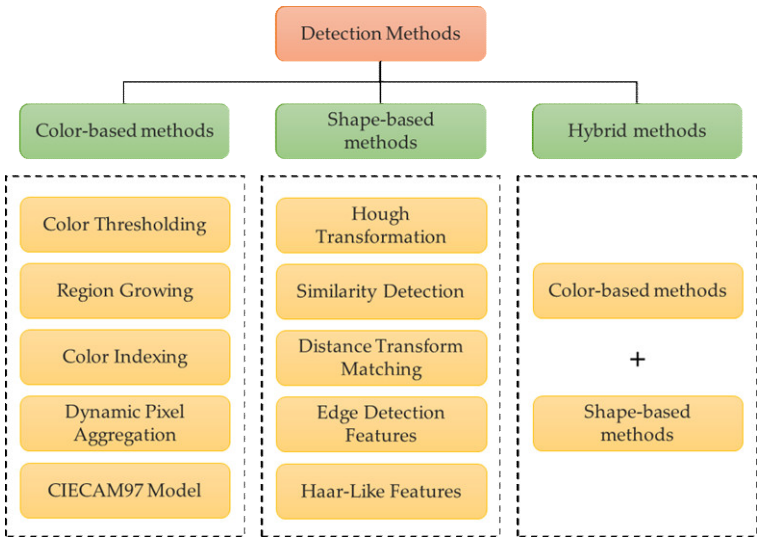

Figure 2. Most popular methods applied for traffic sign detection [11].

To overcome this limitation, a detector based on deep learning was applied rather than a computer vision algorithm in order to increase the robustness of the detector to environmental conditions.

Extensive research has been conducted to improve the detection performance by applying various deep learning algorithms to object detection systems, owing to the rapid development of deep learning techniques [12].

He et al. [13] applied a dual-view convolution neutral network framework for lane detection and road markings. The dataset was composed of 47 batch images, where 20,000 images were included in each batch. Eight batches with a total of 10,000 images were randomly chosen for the experiments.

Additionally, Zhang et al. [14] detected small traffic signs in real scenes using YOLOv3. They proposed a new detection scheme to improve the detection efficiency. Zhou et al. [15] proposed an ice environment traffic sign detection benchmark, ITSRB detection benchmark (ITSDB), and an attention network based on high-resolution traffic sign classification. The benchmarks included 5806 images with 43,290 traffic sign instances with different climates, lights, times, and occlusion conditions. They tested the robustness of the Libra-RCNN and HRNetv2p on the ITSDB compared with Faster-RCNN. The German traffic sign detection dataset is the newest and most effective network.

Li et al. [16] proposed a multi-scale MobileNeck module and an algorithm to improve the performance of an object detection model by outputting a series of Gaussian parameters. Based on the above two methods, a new confidence aware mobile detection (MobileDet) model was proposed. They tested MobileDet on the KITTI and VOC datasets.

Moreover, Lee et al. [8] proposed a semi-automatic method that speeds up the annotation by a factor of 3.19 in comparison to manual annotation. The dataset consists of approximately 150,000 images and includes approximately 470,000 annotated traffic landmarks. They trained a deep neural network on their dataset to detect traffic landmarks, and its performance was evaluated using a novel evaluation metric.

Zhu et al. [17] also applied a robust end-to-end convolutional neural network (CNN) and created a large traffic sign benchmark from 100,000 Tencent Street View panoramas, which provided 100,000 images containing 30,000 traffic sign instances. These images covered large variations in the illuminance and weather conditions. They called this benchmark Tsinghua Tencent 100K and confirmed its effectiveness. Because HD maps require a high positional precision, the positioning performance of the detector plays an important role.

Deep learning detectors are usually divided into CNN-based one- and two-stage detectors. The two-stage family proceeds sequentially with the region proposals and classification. The earliest object detection method based on an RCNN [18] is a two-stage detector.

The backbone of RCNN uses AlexNet [19], the winner of the ILSVRC2012 competition. Although the two-stage detectors achieve a good detection performance, the training and testing speeds are extremely slow. Fast R-CNN [20] and Faster R-CNN [21] have been proposed, along with the two-stage families SPPNet [22], FPN [23], and Cascade R-CNN [24], to solve these problems.

Real-time detection requires a minimum of 15 to 30 fps, and Faster-RCNN is usually approximately 5fps. Therefore, they are unsuitable for real-time applications.

To solve this speed problem, a one-stage method has been proposed, which simultaneously proceeds with region proposal and classification. A typical detector is the you only look once (YOLO) detector proposed in 2015 [25]. YOLO divides the input image into a grid and synchronously predicts the confidence score and probability values for each region of the bounding boxes. YOLO has shown a fast detection speed, although its accuracy is reduced. Joseph et al. has since made some incremental improvements and proposed YOLOv2 [26] and YOLOv3 [27]. In addition to the YOLO family, SSD [28], Retina-Net [29], EfficientDet [30], and RefineDet [31] are representative state-of-the-art models.

In this study, a device equipped with Jetson AGX Xavia, vision sensor, global positioning system (GPS), and an inertial measurement unit (IMU) was developed at a low cost to quickly update facilities, such as traffic signs, in HD maps. This device is termed the road map detection system (RMDS). Additionally, the YOLOv3 model was adopted for fast

detection, and a study was conducted to determine the method of configuring the dataset to achieve an optimal detection performance.

Furthermore, the detection performance, which was varied based on the dataset configuration, was evaluated. The construction of an accurate training dataset was the most crucial aspect of the research and development of deep learning-based detection systems.

The remainder of this paper is organized as follows. Section 2 describes the system and the contents of the annotation data. Section 3 describes the experimental method used, while Section 4 discusses the experimental results. Lastly, Section 5 presents the conclusions of the study.

2. Materials and Methods

The majority of the research conducted in this field in Korea is based on the open datasets KITTI [32], BBD100K [33], GTSDB [34], LISA [35], PASCAL VOC [36], and MS COCO [37], rather than Korean-specific datasets.

This research has mainly focused on the evaluation of the improved algorithm performance based on open datasets, or the dataset has been constructed by selecting only portions of the traffic signs. As the road transportation environments vary from country to country, the composition of the datasets should also vary.

In our dataset, 151 types of objects were applied as road objects for the HD-map updates based on the list of road traffic safety signs notified by the Korean National Police Agency (4 July 2014).

The image datasets were collected at 4K (3840 × 2160) and 30 fps for more than 40,000 km, including Seoul, Busan, and highways with more than four lanes.

Additionally, the data were constructed using the most suitable method for a road environment in order to analyze the degree of improvement in the detection performance based on data balancing and augmentation.

As there are traffic signs that do not exist in the datasets, several types of signs have been made in this study with the support of the Korea Automobile Testing and Research Institute. The traffic signs were installed in the K-City facility and additional data were collected (a test facility for autonomous vehicle experiments) [38].

2.1. Dataset Construction System (DCS) Description and Data Collection

The dataset construction system (DCS) hardware was developed to build the road traffic datasets. The vision sensor is the most important part of a DCS. A vision sensor manufactured by the CIS company that can quickly control environmental changes was installed, as the illuminance varied according to the time and place in the outdoor environment and a lens with a wide horizontal angle was selected. Additionally, the exposure and sensitivity could be adjusted within a maximum period of 1 s after automatic metering during shooting. The specifications for the vision sensor, lens, and installation method are listed in Table 1 and Figures 3 and 4.

Table 1. DCS camera and lens specifications.

Parameters	Specification
Image sensor	SONY IMX255 CMOS Sensor
Sensor size	1″
Aspect ratio	16:9
Resolution	4K, 3840(H) × 2160(V)
Shutter	Global shutter
Frame rate	60 fps
Lens Focal length	8 mm
Angle of view	85.7 × 67.5
Lens resolution	2.5 μm

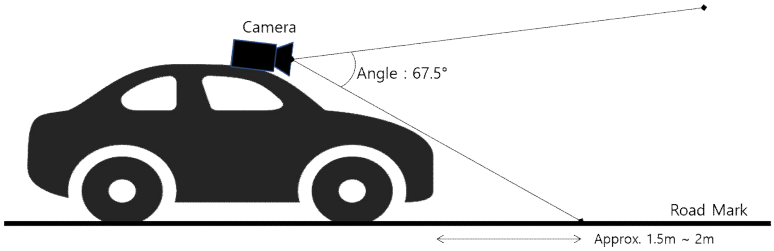

Figure 3. Illustration of the camera setup.

Figure 4. (**a**) DCS vehicle and (**b**) 3D model of the camera lens housing.

Our image dataset included day, night, sunset, sunrise, snow, and rain images of Seoul, Busan, and highways with more than four lanes (Figure 5). Table 2 presents the number of annotations for each class. The routes were set to areas with as many traffic signs as possible. The Pangyo District in Gyeonggi Province was selected as the testbed road and ground truth for the annotation.

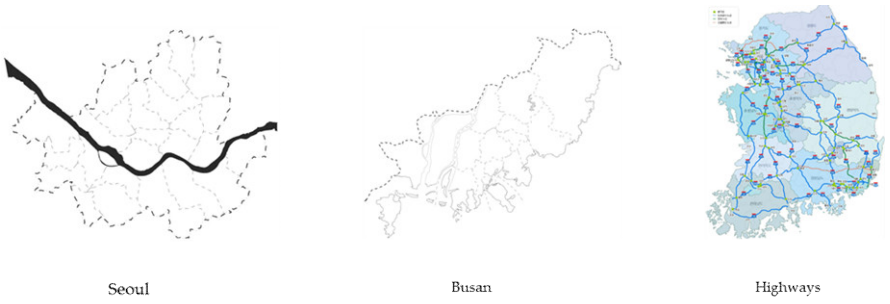

Figure 5. Dataset of urban road (Seoul and Busan) and inter-urban highways [39,40] (Copyright 2019 National Geographic Information Institute, Copyright 2014 Korea Expressway Corporation).

Table 2. Number of annotations for each class in the dataset.

Objects	Number of Annotations	Percentage
Warning sign	177,069	5.2%
Prohibition sign	316,690	9.2%
Mandatory sign	339,347	9.9%
Road marking	1,847,541	53.8%
Traffic light	494,311	14.4%
Rubber cone	60,480	1.8%
Manhole Cover	195,506	5.7%
Total	3,430,944	100%

Additionally, performance tests were conducted in the testbed region to verify the detection performance. Table 3 presents the total number of annotated data for the Pangyo District ground truth annotation configuration.

Table 3. Testbed ground truth.

Objects	Number of Annotations	Percentage
Warning sign	634	3.2%
Prohibition sign	1842	9.4%
Mandatory sign	1919	9.8%
Road marking	12,349	63.1%
Traffic light	1783	9.1%
Rubber cone	61	0.3%
Manhole Cover	998	5.1%
Total	19,586	100%

2.2. Data Annotation and Classification Structure

A total of 151 types of annotation data objects were selected based on the Traffic Safety Sign List of the Korea National Police Agency (Figure 6). However, it was difficult to balance the number of objects.

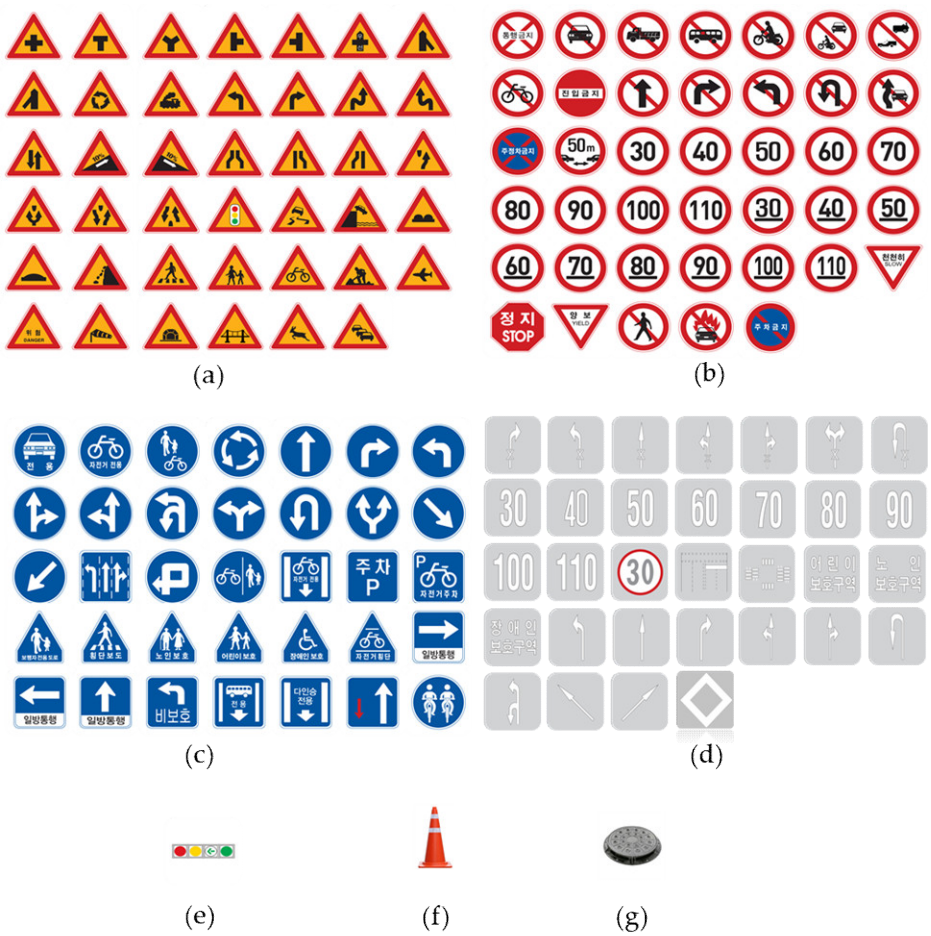

Figure 6. Total of 151 types were selected, including: (**a**) warning signs, (**b**) prohibition signs, (**c**) mandatory signs, (**d**) road markings, (**e**) traffic lights, (**f**) rubber cones, and (**g**) manholes.

For example, imbalance occurred mainly because other objects such as crosswalks and stop lines existed together where the traffic lights were present. This imbalance resulted in a poor detection performance. Additionally, there were frequent signs for approximately

40,000 km, while there was only one specific sign or none on the actual road. An assumption was made to address the problem of data imbalance.

Assumption:

If the shapes and colors are similar, the algorithm will not be biased during training.

As shown in Figure 7, the dataset was divided into seven groups based on this assumption. The dataset was distinguished by the warning signs of group 1, prohibition signs of group 2, mandatory signs of group 3, road markings of group 4, traffic lights of group 5, rubber cones of group 6, and manholes of group 7.

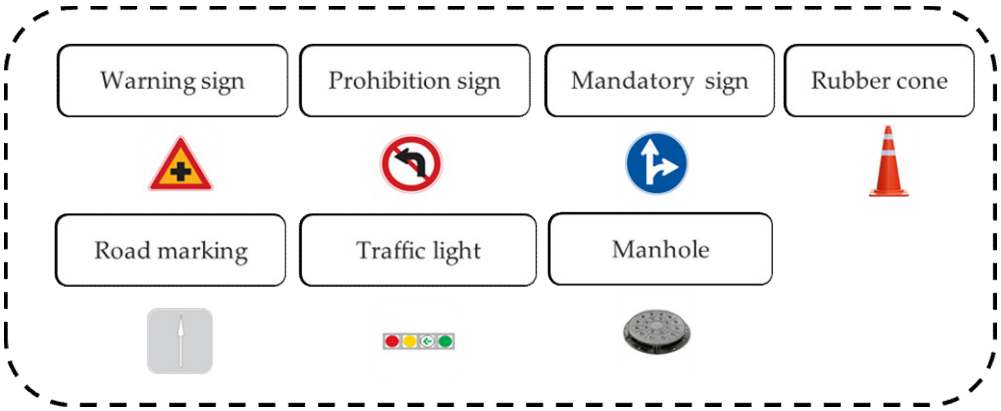

Figure 7. Training dataset group.

The rubber cone of group 6 was selected to check the dynamic data (to update the data when construction or critical issues suddenly occur on the road), and the manholes of group 7 were added for use as future ground control points in the HD map. The detection results were provided, and groups were created with at least 40,000 objects in constant quantities. However, all the imbalances in the datasets with groups could not be resolved. Thus, HD maps could not be easily achieved, even if only seven groups were detected well. A CNN network was added to the final classification map update server to classify the properties of each group specific detected object.

A CNN is added to improve the performance of the classifier. Therefore, a structure was designed to classify the object properties using a CNN classifier by extracting the detected region of interest (ROI) from the group-trained detectors (Figure 8). A dataset was further trained to classify the 151 types of objects by using the Darknet-53 convolution network. However, in the collected image data, 93 types (Figure 9) of data with no or insufficient target object data were created for less than 1000 and additionally collected by the K-CITY facility (Figure 10) for each time period [38]. K-City is a facility for conducting autonomous driving experiments.

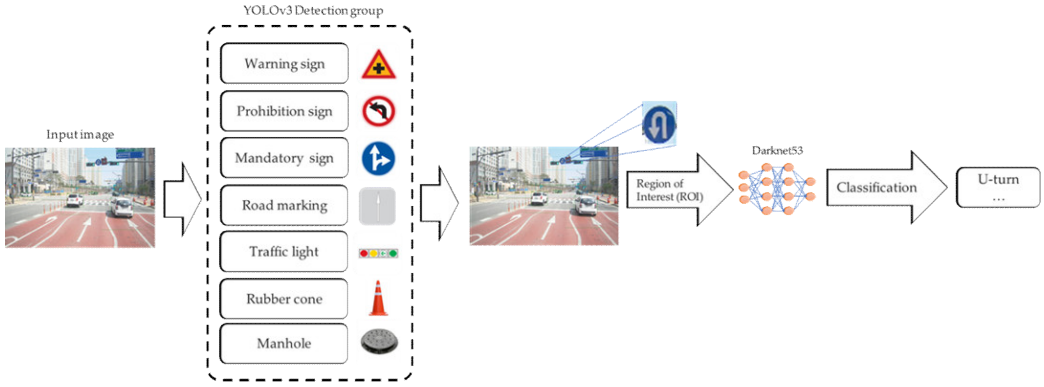

Figure 8. Designed structure to be classified.

Figure 9. Illustration of 93 traffic signs with no or less than 1000.

Figure 10. Image data collection field photography in K-CITY facility.

Despite the imbalance of data for each object in each group, it was judged to be sufficient for machine-learning-based applications because there were more than 10,000 objects for each group.

Furthermore, for classification using a CNN, insufficient class data were balanced to approximately 10,000 per object through augmentation and up-sampling for data balancing and performance improvements (Figure 11).

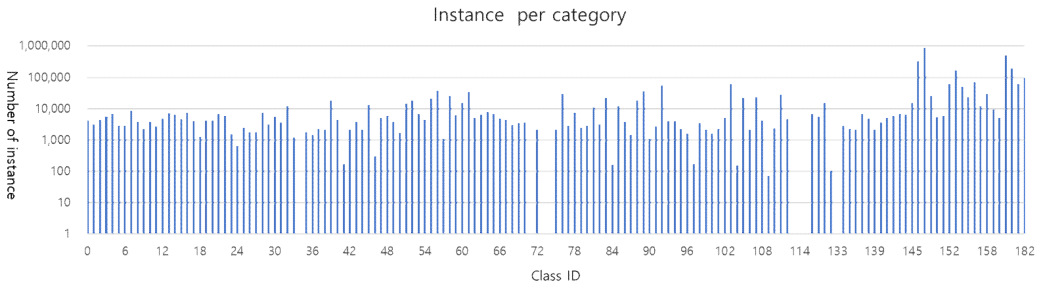

Figure 11. Number of instances in each class (151 classes).

YOLO is limited by its poor detection performance for small objects. Although YOLOv3 has relatively improved with the use of multiple scales, a higher accuracy is required for HD maps [14]. Essentially, when the object detected in the original image is reclassified by a CNN, the detection structure is changed to improve the performance when compared to the classification by using only YOLOv3.

2.3. Augmentation and Balancing

Augmentation is widely used in CNN structures. It is used in this study to reinforce the balance of our datasets and the relatively insufficient number of object datasets. There is inevitably an imbalance in the object when annotating the road environment data. It is also

difficult to build all the objects using real data in every situation. The amount of data are augmented through brightness, contrast, shifting, noise, distortion, and random erasures to add an insufficient amount of data. Additionally, in order to set the minimum maximum limit of parameters for image processing, the parameters were set by testing whether each processing-specific object detection was available. A data augmentation combination scenario was also created to perform the CNN performance tests in parallel [41]. As shown in Table 4 and Figures 12 and 13, augmentation was performed by randomly mixing the functions from A to G.

Table 4. Augmentation type and parameter range.

No.	Function	Random Param-1	Random Param-2
A	Brightness	−30	+30
B	Contrast	0.8	1.4
C	Translation	−15	+15
D	Rotation	−0.5	+0.5
E	Affine	−10	+10
F	Gaussian blur	1.6	3.8
G	Random erasing	18	27

Figure 12. Augmentation brightness factor control example: (**a**) original image, (**b**) −30% brightness, (**c**) −15% brightness, (**d**) +30% brightness, and (**e**) +15% brightness.

2.4. Road Map Detection System (RMDS) Based on YOLOv3

The RDMS algorithm for HD map updates was based on YOLOv3. This is because a fast-paced deep-learning algorithm was required, and a CNN was further applied to reinforce the poor classification performance.

The processing speed inevitably decreased if multiple algorithms were simultaneously applied to the RDMS equipment. Therefore, the classification- and judgment-related algorithms, such as a CNN, were configured to be handled by the map update server. Figure 14 shows the concept of the RMDS. When a traffic sign is detected by loading the trained weights into the RMDS, the result of the center point and the recognition area of the image is transmitted to the map update server.

Figure 13. Augmentation test (first column, original image): (**a**) contrast (daytime), (**b**) contrast (night), (**c**) translation, (**d**) affine, and (**e**) random erase.

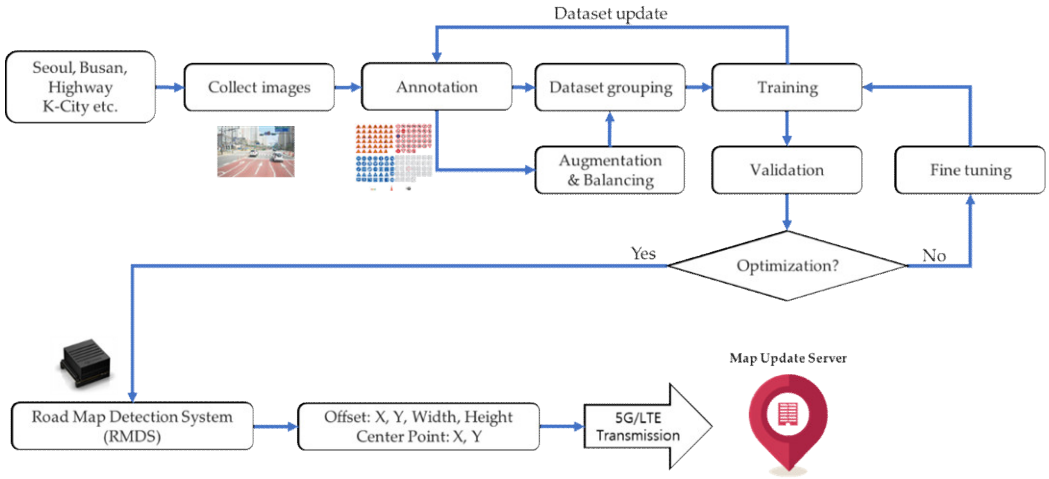

Figure 14. Flowchart of the process of sending the RMDS object detection result value (center point, detection area) to the update map server.

In YOLOv3 detection, all the bounding boxes and category probabilities from the entire image are simultaneously generated by a single convolutional network, as shown in Figure 15. Firstly, the network divides each image in the training set into S × S grids, where each grid is given candidate boxes of three different sizes. If the center of the object ground truth falls in a grid, that particular grid is responsible for detecting the object. Subsequently, the features are extracted through the convolutional layer, Darknet-53. Lastly, the yolk layer is used for multi-scale prediction. Each grid predicts the bounding boxes and their confidence scores, as well as the class conditional probabilities [14,27]. Table 5 presents the YOLOv3 network structure.

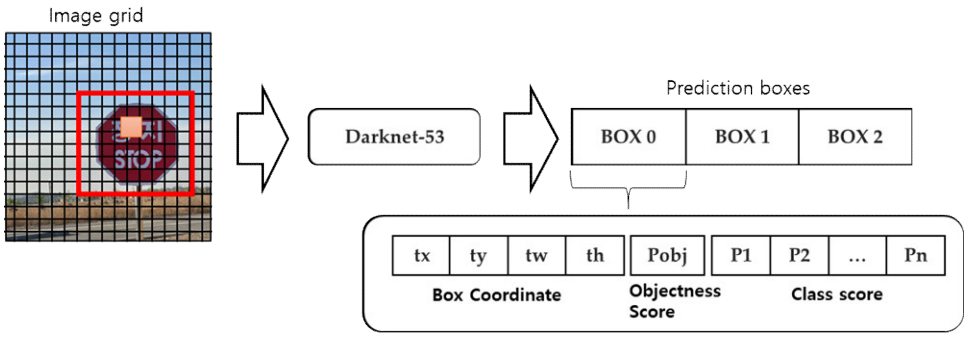

Figure 15. YOLOv3 detection and prediction feature map.

Table 5. Feature extraction network of YOLOv3 (Darknet-53).

	Type	Filter	Size	Output
	Convolutional	32	3 × 3	480 × 480
	Convolutional	64	3 × 3/2	240 × 240
	Convolutional	32	1 × 1	
1×	Convolutional	64	3 × 3	
	Residual			240 × 240
	Convolutional	128	3 × 3/2	120 × 120
	Convolutional	64	1 × 1	
2×	Convolutional	128	3 × 3	
	Residual			120 × 120
	Convolutional	256	3 × 3/2	60 × 60
	Convolutional	128	1 × 1	
8×	Convolutional	256	3 × 3	
	Residual			60 × 60
	Convolutional	512	3 × 3/2	30 × 30
	Convolutional	256	1 × 1	
8×	Convolutional	512	3 × 3	
	Residual			
	Convolutional	1024	3 × 3/2	15 × 15
	Convolutional	512	1 × 1	
4×	Convolutional	1024	3 × 3	
	Residual			15 × 15

2.5. Map Update System (MUS) Framework

This framework study analyzed the logic of matching the detected object and the object existing in the HD map for map updates. As shown in Figure 16, the error between the center point of the three-dimensional (3D) spatial coordinates of the sign object existing in the HD map and the center point of the object detected by our RMDS was minimized [42,43]. The accuracy of the HD map in matching the detected 2D coordinates increased. If the error was large, there was a problem in matching between the objects, and it was impossible to know which object was updated on the map. MUS calculated the errors and updated the changes.

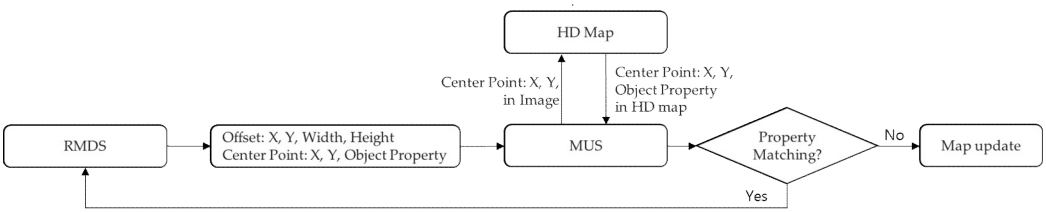

Figure 16. Flowchart of MUS that determines whether to update by comparing the RMDS detection result and the data in the HD Map.

3. Experiments

The evaluation metrics that are most widely used for evaluating the performance in object detection are the intersection over union (IOU) and mean average precision (mAP). However, the evaluation in this study employs the evaluation indicators as they are used for map updates, and a new part of the detection rate is defined. The detection rate is determined to be successful if, for example, an object is detected at least once until it disappears from the image.

As the detection object is reclassified into a logic judgment algorithm in an actual update system, it can be easily updated after detection. Table 6 presents the experimental conditions.

Table 6. Experimental environment.

Parameters	Specifications
CPU	Intel$^{(R)}$ Xeon$^{(R)}$ E5-1650 v3 3.5 GHz
RAM	64 GB
GPU	Nvidia GeForce RTX 3090 (24 GB)
Accelerated environment	CUDA 11.1, cuDNN v8.1.0
Operating system	Window 10 Pro x64
Training framework	Darknet

3.1. Definition of Detection Rate

In this study, the detection rates, which are the most widely used parameters in the binary classification model, are defined as follows: true positives (TPs), true negatives (TNs), false positives (FPs), and false negatives (FNs). The mAP metric evaluates the overall performance of the object detector. The precision at each recall level must be obtained and then averaged for each class in order to calculate the mAP. Precision is defined as the ratio of the total number of true predictions to the total number of predictions. The IoU threshold determines whether the prediction is true or false [8]. Precision and recall are defined in Equations (1) and (2), and Equations (3) and (4) define the accuracy and F1 score, as given below.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + TN} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

$$F1score = \frac{2 * Recall \times Precision}{Recall \times Precision} \tag{4}$$

The detection rate definition expression for the map update is defined as follows.

$$Detection\ Rate\ (\%) = \frac{Object\ Detection\ Count}{Total\ Object\ Count} \times 100 \tag{5}$$

3.2. Training Method for YOLO v3

The networks used in this study are trained based on the official YOLOv3 code. The K-fold cross-validation method is used for the dataset to prevent overfitting beforehand, as shown in Figure 17, and the hyperparameters are adjusted and fine-tuned.

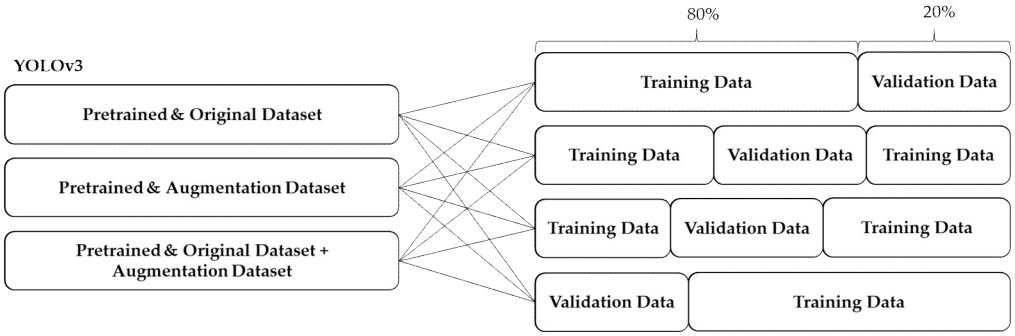

Figure 17. K-fold cross validation methods.

The currently annotated bounding box and anchor box are recalculated and corrected, and the initial value is set as shown in Table 7.

Table 7. Experimental parameters of YOLOv3.

Input Size	Batch Size	Subdivisions	Momentum	Decay
480 × 480	64	16	0.9	0.0005
Angle	Saturation	Exposure	Hue	Flip
0	1.5	1.2	0.1	0
Learning Rate	Iterations	Anchor		
0.001–0.0001	31,000	10, 13, 16, 30, 33, 23, 30, 61, 62, 45, 59, 119, 116, 90, 156, 198, 383, and 326		

Table 7 presents the initial training parameters are presented. The learning rate applies 0.001 to 0.0001 and disables the flip function because the flip causes classification problems with data related to the left turn, right turn, etc. Tables 8 and 9 show the experimental results for the testbed. The results of the original pretrained dataset are compared with the purely augmented dataset, and the results of both the original pretrained and augmented datasets in order to compare the differences in the results trained with our dataset. The detection rate of this study is determined by the number of detections for the actual objects present on the road. As the map update server determines whether the map is to be updated with a logical filter, only one is detected even if it is detected multiple times.

Table 8. Testbed dataset experiment with no group (the input size of the image is 1920 × 1080).

Pretrained	Original Datasets	Aug. Datasets	mAP	Recall	F1-Score
√ (√ Dataset used for training)	√		69.0%	73.7%	81.7%
√		√	68.7%	72.1%	81.4%
√	√	√	70.9%	76.1%	83.0%

Table 9. Testbed dataset experiment with group (the input size of the image is 1920 × 1080).

Pretrained	Original Datasets	Aug. Datasets	mAP	Recall	F1-Score
√	√		84%	85%	91%
√		√	76%	77%	86%
√	√	√	95.0%	96.0%	97.5%

4. Results

Tables 8 and 9, and Figure 18 present the results of counting the detected results while driving along the testbed route every two hours between 08:00 a.m. and 18:00 p.m. The number of detected data points is manually determined during real-time testing.

Figure 18. Experiment results of detecting traffic signs, road signs, and traffic lights in RMDS.

Accurate detection of true positives is the most crucial aspect of updating the HD map. False positives (FP) must be extremely low in order to be evaluated as a good model. Therefore, it is determined that the higher the recall value of 90% [1,44], the better the model. The change in the model performance is determined depending on the data combination. As shown in Tables 8 and 9, the changes in the mAP and recall and in the F1-score values can be observed based on the combination of pre-trained data, original data, and augmentation data. It is a matter of principle, but it can be observed from the experiment that optimal performance is obtained only when three types of data are trained simultaneously.

Additionally, although it may be limited to the HD map updates, the difference between the individually trained results and the group training is approximately 20% of the performance improvement in recall. The difference in the performance is determined by the data imbalance problem. Table 10 shows the results of the detection rate defined in Section 3.2.

Table 10. Detection rate experiment.

No.	Object Detection Count	Total Object Count	Detection Rate
1	2092	2247	93.1%
2	2102	2247	93.5%
3	1983	2247	88.2%
4	2142	2247	85.6%
5	1932	2247	95.3%
		Average	91.14%

The detection rate evaluated while driving on an actual road shows an average of 91%. Figure 18 shows the results of detecting traffic signs, road signs, and traffic lights.

In conclusion, the detection rate is lowered when experimenting with the testbed at sunrise or sunset, and under shadows and conditions of partially occluded road marking objects, owing to traffic congestion. However, after grouping the data into seven groups and balancing by attribute, very high detection rates are obtained.

5. Conclusion and Future Work

The contributions of this paper are as follows:

(1) A low-cost image road map detection system (RMDS) is developed with a map update system (MUS) framework to quickly update the HD map. It can be installed in many vehicles to increase the update time for road changes, owing to the low cost of the system.
(2) More than 3,000,000 Korean road traffic annotation data have been built for MUS.
(3) The performance differences are evaluated depending on how the training set is built into a YOLOv3 model.

The most important aspects of this study are the accurate training dataset, number of datasets, and appropriate modeling. It is impossible to achieve a good detector even with a good training model, if the annotation data are inconsistent or if the data classification are unclear. In this study, an accurate training dataset was constructed by collecting millions of high-resolution 4 K images suitable for Korean terrain using the DCS. RMDS and MUS, to which the YOLOv3 model was applied, were developed, and the detection performance of road traffic signs, road signs, and traffic lights was completed.

Unfortunately, the annotation was inappropriate for application as a bounding box for road markings. For example, the annotation of straight marks and lane changes is difficult owing to the lens distortion. In the future, the HD maps will be updated through the detection of signs, owing to the detection of dotted lanes, solid lanes, sidewalks, and roadway divisions; median strip separation algorithms must also be added. The classifier performance will also be improved and a segmentation algorithm will be added to further apply the dotted lane start and end points.

In order to reduce the time of dataset annotation, an algorithm will be developed in combination with a deep learning algorithm and computer vision to improve the auto-annotation function and to increase the accuracy of detection and dotted-lane start and end point extraction.

Author Contributions: Conceptualization, Y.-K.P. and I.-G.C.; writing—original draft preparation, Y.-K.P.; writing—review and editing, H.P., S.-S.H. and Y.-K.P.; funding acquisition, Y.-K.P. and I.-G.C.; data curation, Y.-S.W.; project administration, Y.-K.P. and I.-G.C.; software, Y.-K.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Korea Ministry of Land/Korea Agency for Infrastructure Technology Advancement, under grant number 21NSIP-B145070-04.

Acknowledgments: The authors would like to thank the reviewers for their helpful comments and criticism.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ham, S.; Im, J.; Kim, M.; Cho, K. Construction and Verification of a High-Precision Base Map for an Autonomous Vehicle Monitoring System. *ISPRS Int. J. Geo Inf.* **2019**, *8*, 501. [CrossRef]
2. Ilci, V.; Toth, C. High Definition 3D Map Creation Using GNSS/IMU/LiDAR Sensor Integration to Support Autonomous Vehicle Navigation. *Sensors* **2020**, *20*, 899. [CrossRef]
3. Seman Tov Bus Company Lowers Collision Rate with Mobileye. Available online: https://www.mobileye.com/us/fleets/resources/case-studies/ (accessed on 1 April 2021).
4. Mobileye. Available online: https://www.mobileye.com/ (accessed on 1 April 2021).
5. HERE. Available online: https://www.here.com/platform/automotive-services/hd-maps (accessed on 1 April 2021).

6. CARMERA. Available online: https://www.carmera.com/ (accessed on 1 April 2021).
7. TomTom. Available online: https://www.tomtom.com/products/hd-map/ (accessed on 1 April 2021).
8. Lee, W.H.; Jung, K.; Kang, C.; Chang, H.S. Semi-Automatic Framework for Traffic Landmark Annotation. *IEEE Open J. Intell. Transp. Syst.* **2021**, *2*, 1–12. [CrossRef]
9. National Geographical Institutes Precision Map. Available online: http://map.ngii.go.kr/ms/pblictn/preciseRoadMap.do (accessed on 1 April 2021).
10. Saturnino, M.B.; Sergio, L.A.; Pedro, G.J.; Hilario, G.M.; Francisco, L.F. Road-Sign Detection and Recognition Based on Support Vector Machines. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 264–278.
11. Wali, S.B.; Abdullah, M.A.; Hannan, M.A.; Hussain, A.; Samad, S.A.; Ker, P.J.; Mansor, M.B. Vision-Based Traffic Sign Detection and Recognition Systems: Current Trends and Challenges. *Sensors* **2019**, *19*, 2093. [CrossRef] [PubMed]
12. Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object Detection in 20 Years: A Survey. *arXiv* **2019**, arXiv:1905.05055.
13. He, B.; Ai, R.; Yan, Y.; Lang, X. Accurate and Robust Lane Detection Based on Dual-View Convolutional Neutral Network. In Proceedings of the 2016 IEEE Intelligent Vehicles Symposium (IV), Gothenburg, Sweden, 19–22 June 2016; Volume IV.
14. Zhang, H.; Qin, L.; Li, J.; Guo, Y.; Zhou, Y.; Zhang, J.; Xu, Z. Real-Time Detection Method for Small Traffic Signs Based on Yolov3. *IEEE Access* **2020**, *8*, 64145–64156. [CrossRef]
15. Zhou, K.; Zhan, Y.; Fu, D. Learning Region-Based Attention Network for Traffic Sign Recognition. *Sensors* **2021**, *21*, 686. [CrossRef] [PubMed]
16. Li, W.; Liu, K. Confidence-Aware Object Detection Based on MobileNetv2 for Autonomous Driving. *Sensors* **2021**, *21*, 2380. [CrossRef] [PubMed]
17. Zhu, Z.; Liang, D.; Zhang, S.; Huang, X.; Li, B.; Hu, S. Traffic-Sign Detection and Classification in Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
18. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
19. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *25*, 1097–1105. [CrossRef]
20. Girshick, R. Fast, R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015.
21. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster, R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *arXiv* **2016**, arXiv:1506.01497. [CrossRef] [PubMed]
22. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
23. Lin, T.; Dollár, P.; Girshick, R.B.; He, K.; Hariharan, B.; Belongie, S.J. Feature Pyramid Networks for Object Detection. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 936–944.
24. Cai, Z.; Vasconcelos, N. Cascade R-CNN: Delving into High Quality Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
25. Redmon, J.; Divvala, S.; Girshick, R.B.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
26. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525.
27. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
28. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. In *European Conference on Computer Vision*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; pp. 21–37.
29. Lin, T.Y.; Goyal, P.; Girshick, R.B.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327. [CrossRef] [PubMed]
30. Tan, M.; Pang, R.; Le, Q.V. EfficientDet: Scalable and Efficient Object Detection. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Online, 14–19 June 2020; pp. 10778–10787.
31. Zhang, S.; Wen, L.; Bian, X.; Lei, Z.; Li, S. Single-Shot Refinement Neural Network for Object Detection. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4203–4212.
32. Geiger, A.; Lenz, P.; Urtasun, R. Are We Ready for Autonomous Driving? the KITTI Vision Benchmark Suite. In Proceedings of the IEEE Conference Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012.
33. Yu, F.; Chen, H.; Wang, X.; Xian, W.; Chen, Y.; Liu, F.; Madhavan, V.; Darrell, T. BDD100K: A Diverse Driving Dataset for Heterogeneous Multitask Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020.
34. Houben, S.; Stallkamp, J.; Salmen, J.; Schlipsing, M.; Igel, C. Detection of Traffic Signs in Real-World Images: The German Traffic Sign Detection Benchmark. In Proceedings of the 2013 International Joint Conference on Neural Networks (IJCNN), Dallas, TX, USA, 4–9 August 2013; pp. 1–8.

35. Mogelmose, A.; Trivedi, M.M.; Moeslund, T.B. Vision-Based Traffic Sign Detection and Analysis for Intelligent Driver Assistance Systems: Perspectives and Survey. *IEEE Trans. Intell. Transport. Syst.* **2012**, *13*, 1484–1497. [CrossRef]
36. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. (Pascal VOC) Development Kit. The Pascal Visual Object Classes. 2006. Available online: http://host.robots.ox.ac.uk/pascal/VOC/ (accessed on 28 March 2021).
37. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In (MSCOCO). In Proceedings of the IEEE International Conference on European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
38. Korea Transportation Safety Authority. Available online: http://www.kotsa.or.kr/eng/main.do (accessed on 1 April 2021).
39. National Geographic Information Institute (NGII). Available online: https://www.ngii.go.kr/ (accessed on 5 July 2021).
40. Korea Expressway Corporation (EX). Available online: https://www.ex.co.kr/site/com/pageProcess.do (accessed on 5 July 2021).
41. Fábio, P.; Cristina, V.; Sandra, A.; Eduardo, V. Data Augmentation for Skin Lesion Analysis. In *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*; Springer: Cham, Switzerland, 2018; pp. 303–311.
42. Baumker, M.; Heimes, F.J. New Calibration and Computing Method for Direct Georeferencing of Image and Scanner Data Using the Position and Angular Data of an Hybrid Inertial Navigation System. *Integr. Sens. Orientat.* **2002**, *43*, 197–212.
43. Pix4d. Available Online. Available online: https://support.pix4d.com/hc/en-us/articles/202559089-How-are-the-Internal-and-External-Camera-Parameters-defined (accessed on 5 July 2021).
44. Zhang, P.; Zhang, M.; Liu, J. Real-Time HD Map Change Detection for Crowdsourcing Update Based on Mid-to-High-End Sensors. *Sensors* **2021**, *21*, 2477. [CrossRef] [PubMed]