Risk Measurement in Driving Situations Using Real-Time Object Detection

Seong Seop Kim
Handong Global University
Department of ICT Convergence
21700109@handong.ac.kr

Dong Hyun Sin
Handong Global University
Department of ICT Convergen
s5753203@handong.ac.kr

Seok Beom Jang
Handong Global University
Department of ICT Convergence
21700625@handong.ac.kr

Yong Jun Hwang
Handong Global University
Department of ICT Convergence
21700796@handong.ac.kr

Abstract

Despite the constant advancement of autonomous driving, today's autonomous driving technology is still at Level 2. This means that only under certain conditions, a car can perform autonomous driving, which means that fully autonomous driving still has a long way to go. In order to realize safe driving in any environment, Dong-Hyun Shin, Seong-Seop Kim, Yong-Jun Hwang, and Seok-Beom Jang of the ICT Entrepreneurship Department of Handong Global University combined an object detection model using yolo v7 and a risk calculation algorithm using entropy and unpredictable indices to combine deep learning and machine learning. We designed a hybrid model that can help the autonomous driving system by calculating the risk rate on any road, and it was possible to design a system that helps safe driving based on the camera in real time.

1. Introduction

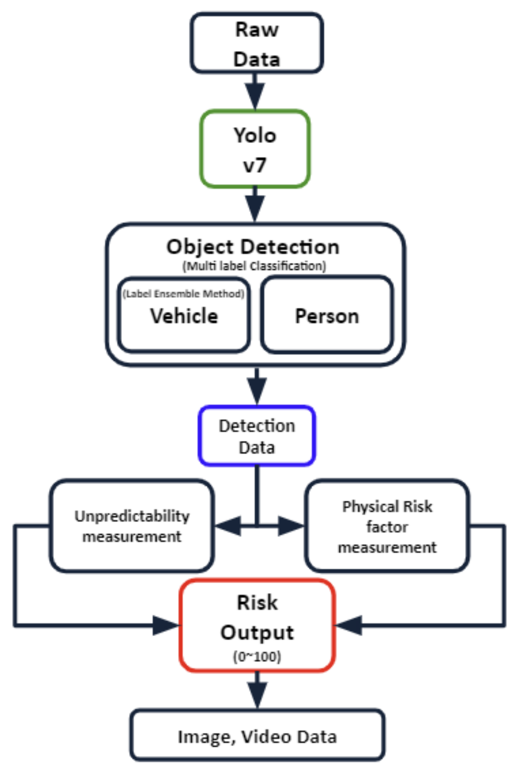
Based on AI object detection technology, we intend to determine the risk situation and its degree by developing a model that can detect risk factors by judging causal relationships to the front and periphery of road driving situations. The final result is to distinguish and detect dangerous situations based on the image file. For example, a dangerous situation is selected by autonomous judgment as an unusual situation such as jaywalking, road construction, and falling objects. Since datasets are composed of image files, it is difficult to consider the context before and after, but it can be inferred according to the results after detecting objects in the image file. The results are expressed in the form of box or segmentation, and the risk situation is notified in text and the risk level is quantified and presented.

2. Problem Statement

Currently, the proportion of Handong Global University students residing outside Yangdeok is increasing. As a result of the survey, about 35% of students live outside Yangdeok, and about 55% of those students are currently using their own car to go to school, and while driving, they complain of dangers in certain sections. We open up various possibilities for dangerous situations, define various situations, use AI to create a road risk detection model, and apply the model to actual school routes to suggest and prevent danger zones to students in advance.

3. Technical Approach

3.1 Architecture



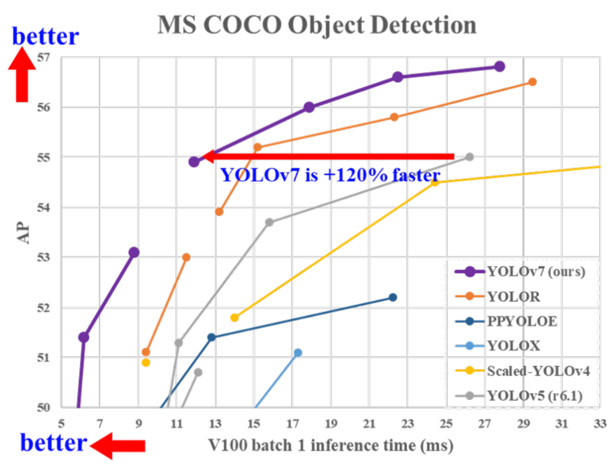
In order to apply RMOD, video data must be provided as an input. The first step of the model is to convert video data into image data using Open-cv. The transformed data is used to detect risk factors in the image using the yolov7 model trained in advance. The RMOD model is divided into a yolov7 deep learning model that performs object detection and a machine learning model that measures risk. The yolov7 model is a combination of three detection models, and is divided into a model that detects only vehicles, a model that detects only people, and a model that detects only the environment. Each model does not invade each other's domain, and the results obtained from each model are combined and calculated. The risk is measured based on the data detected by yolov7. The measured risk is expressed as a number from 0 to 100.

3.2 Methods For Object Detection

We use yolo for real-time object detection. CNN can also solve this problem, but in terms of speed, yolo is far superior. We chose yolo because we need to detect a lot of objects at once and we need to do it quickly. When an image comes in, yolo divides the image into cells of a certain size, and predicts one object for each cell. At this time, since only one object can be predicted for each cell, if several objects overlap, some objects may not be detected. For this reason, we decided to use a method that detects objects with multiple models rather than with one model.

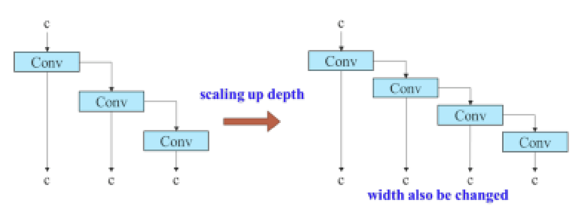
3.3. YOLO v7 Model

yolo is a 1-stage method of object detection modeling. In 2-stage, when an image is received, it finds the position of the object and proceeds sequentially with what the object is. The R-CNN series belongs to this category. yolo finds the location of an object and classifies what the object is. It has a higher speed than 2-stage but lower accuracy. However, as it was upgraded to 2~6, the accuracy gradually increased, and when we need to detect an object in real time, yolo is mostly used. The yolo v7 used in this model is a model in which the network performance is improved by applying bag-of-freebies to the existing yolo model. Model reparameterization and label assignment techniques are used here, and the above two methods are applied to show excellent performance as shown below.

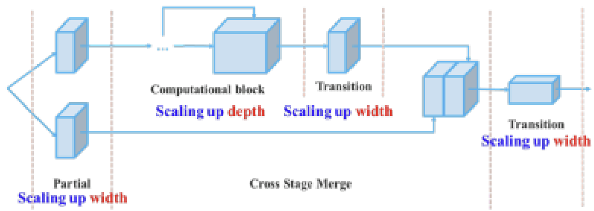


3.3.1 [7] Model scaling for concatenation-based models.

[7] Model scaling is primarily used to modify certain model characteristics and produce models at various sizes to accommodate various inference rates. For instance, the EfficientNet scaling model [1] takes resolution, depth, and width into account. The number of steps are adjusted as part of the scaling pattern for the YOLOv4 scale [2]. When doing width and depth scaling, Dollar' et al. in [3] examined the impact of vanilla convolution and group convolution on the amount of parameters and calculations, and they used this information to build an effective model scaling approach.



(a) concatenation-based model (b) scaled-up concatenation-based model



(c) compound scaling up depth and width for concatenation-based model

[7] Model scaling for concatenation-based models is seen in Figure 1. From (a) to (b), we notice that the output width of a computational block also grows when depth scaling is applied to concatenation-based models. The input width of the subsequent transmission layer will grow as a result of this event. Therefore, we suggest (c), which states that just the depth in a computational block needs to be scaled when performing model scaling on concatenation-based models, with the entire transmission layer being

[7] The aforementioned techniques are primarily employed in topologies like PlainNet or ResNet. The in-degree and out-degree of each layer will not change while these structures are scaling up or down, allowing us to independently assess the effects of each scaling factor on the number of parameters and computation. The in-degree of a translation layer that comes right after a concatenation-based computational block will, however, drop or rise if these methods are used to the concatenation-based design, as shown in Figures 1 (a) and (b). The aforementioned phenomenon suggests that, for a concatenation-based model, we cannot study various scaling parameters independently but rather that they must be taken into account jointly. Consider scaling-up depth as an example. This operation will modify the ratio between a transition layer's input channel and output channel, which could result in less hardware being used by the model. Therefore, for a concatenation-based model, we must suggest the matching compound model scaling approach. Calculating the change in the output channel of a computational block is necessary for scaling the depth factor of that block. The outcome is depicted in Figure 1 after performing width factor scaling on the transition layers with the same amount of change (c). The model's original design attributes and the ideal structure can both be preserved by our suggested compound scaling strategy.

3.4. Label Ensemble Method

When an image comes in, yolo divides the image into cells of a certain size, and predicts one object for each cell. At this time, since only one object can be predicted for each cell, if several objects overlap, some objects may not be detected. For this reason, we decided to use a method that detects objects with multiple models rather than with one model. Ensemble technology is one of the great ways to improve object detection models. For our expertise in data detection, instead of one model detecting multiple objects, one model detecting only one object and creating multiple models instead to detect objects together. Of course, if the frame of the model is different, the model cannot be ensembled, so the basic algorithm needs to be modified, but since we created the detection model using the same framework, it can work smoothly. Originally, we tried to perform image classification with different labels for truck, bus, and car, but due to the large difference in the amount of data for each label, the accuracy of the model decreased and overfitting occurred. Therefore, we ensemble car, bus, and truck into one label called vehicle to create a model that detects only the vehicle, and then combine it with a model that detects only people to complete the final detection model. To create this final model, we found that each label yielded good performance at different epoch values without overfitting.

3.5. Risk Measurement

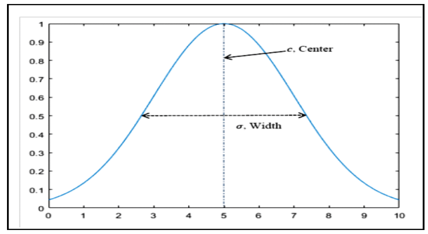
The main task after the object detection is to approximate the actual distance, angle, and speed between vehicles in the image through computer vision algorithms. Based on this, we measure the risk by predicting the distance, angle, and speed of time t based on the past time point through the time series model Prophet, and comparing how much it differs from the actual value. Risk Measurement can be divided into two ways. First one is Physical Risk factor measurement. In order to measure the 'Hazard Score' of the driving cars, we need to firstly measure the physical risk factors of each car on the road. There are three main physical factors: 1. Distance, 2. Angle , 3. Speed based on two articles(reference[13], [14]) which suggested the real-time methodology about how dangerous each car on the road is. We first measured the Real-Time Hazard Score of each car by computing those Physical Risk Factors (of each car).The methodology of measuring the Physical Risk Factors mainly uses 'Gaussian Fuzzy Membership Function' which maps the value into the range of 0 to 1 based on this function.

Table 3. Fuzzy rules table.

Combination	conclusion	Combination	conclusion	Combination	conclusion
LS SA SD	HR (high risk)	LS MA SD	MR (moderate risk)	LS LA SD	LR (low risk)
LS SA MD	MR (moderate risk)	LS MA MD	LR (low risk)	LS LA MD	S (safe)
LS SA LD	LR (low risk)	LS MA LD	S (safe)	LS LA LD	VS (very safe)
MS SA SD	D (dangerous)	MS MA SD	HR (high risk)	MS LA SD	MR (moderate risk)
MS SA MD	HR (high risk)	MS MA MD	MR (moderate risk)	MS LA MD	LR (low risk)
MS SA LD	MR (moderate risk)	MS MA LD	LR (low risk)	MS LA LD	S (safe)
HS SA SD	D (dangerous)	HS MA SD	D (dangerous)	HS LA SD	HR (high risk)
HS SA MD	D (dangerous)	HS MA MD	HR (high risk)	HS LA MD	MR (moderate risk)
HS SA LD	HR (high risk)	HS MA LD	MR (moderate risk)	HS LA LD	LR (low risk)

$$e^{-\frac{-(x-\mu)}{2\sigma^2}} \quad -\infty < x < \infty$$

To compute the Hazard Score of each Physical Risk factor, we adjusted the parameter 'sigma', which determines the location of the inflection point. We tried to compute the appropriate sigma value for each risk factor.



1) Distance, Speed: The sample mean was calculated by bootstrapping(resampling 2000 samples) the distance values of all objects from the current point to the k(k=5) time point. (In the case of Speed, we used logistic fuzzy membership function instead.) 2) Angle: Assuming that the angle at the far end of the center of one lane was a degree, the value a was used as the sigma value. (a=45)

And next one is Risk Measurement about Unpredictability. When driving there is a lot of unpredictable situations. So we append this in our risk measurement formula. To further explain the 'dangerous' event in a road driving situation, not only the physical hazard factor but also the 'unpredictability' (i.e., the degree to which the predicted value and the actual value are different when predicting the future of a car at a point in the past).

$$Unpredictability = \frac{(\bar{y} - \hat{y})^2}{(y - \hat{y})^2 + (\bar{y} - \hat{y})^2}$$

Where ybar is the sample mean of the past and current point observations of one physical risk factor of an object, and yhat is the estimated estimate of the current point in time. Therefore, considering 3 Physical Risk Factors and each of their unpredictability, we averaged these 6 factors(3 Physical Risk Factors + 3 Unpredictabilities) on each object and finally, the Hazard Score is computed.

4. Result

4.1. Testing Dataset

In order to evaluate whether the created model can be applied to actual road driving conditions in Korea, the Daejeon city road driving dataset provided by KISTI was used as a performance evaluation dataset.

4.2. Evaluation Metric

As a performance evaluation index, mAP (Mean Average Precision), a metric widely used as an evaluation measure in the field of object detection and computer vision, was used. mAP is closely related to Precision and Recall, which are a kind of measure for measuring classification performance. The formula below is a formula to calculate Precision and Recall, where TP, FP, and FN are abbreviations for ‘True Positive’, ‘False Positive’, and ‘False Negative’. Therefore, the meaning of Precision is 'the ratio of detected objects to be actually true', and the meaning of Recall is 'The ratio of objects that are actually true being detected really well'.

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

$$Recall = \frac{TP}{TP + FN}$$

AP (Average Precision) is calculated as the area of the graph when a graph is drawn to show the degree of change according to the threshold with Recall as the x-axis and Precision as the y-axis based on one label. Finally, mAP is calculated as the average of APs each label has. mAP has a value between 0 and 1, and the larger the value, the better the performance.

4.3. Testing Result

Metric	Yolov7(Single)	Yolov7(Ensembled)
mAP	0.206	0.428

As a result of evaluating the performance of the model, the mAP of the existing single Yolov7 model was 0.206, which required improvement. However, in the ensemble model, which was modeled separately for each label and the results were added up, the mAP value was 0.428, which is significantly improved compared to the previous model.

5. Conclusion & Future

We have planned a comprehensive artificial intelligence that includes environmental factors such as construction sites, potholes and cracks. However, the number of data on environmental factors was too small. Because of this, no matter how much training, AI's recognition rate did not increase. In addition, the impact of environmental factors on the risk rate in actual accidents was insignificant than expected, and as shown in the statistics introduced above, the ratio of vehicle-to-vehicle and vehicle-to-person accidents was overwhelmingly high. So we created a model with only two types of objects, a vehicle and a person. We think that if we add the help of sensors that detect physical changes to our model produced in this way, it can become much more sophisticated than it is now. If that happens, we thought that we could create a model that is more accurate than human instantaneous senses. Also we have planned to develop our model that can detect more labels. Based on the survey results, we can know more labels such as potholes on the road, animals and sign under construction etc. And finally, we can apply our model on webcam so we can found out the real-time risk.

Risk Measurement in Driving Situations Using Real-Time Object Detection

References

[1] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning (ICML), pages 6105–6114, 2019.

[2] Chien-Yao Wang, Alexey Bochkovskiy, and HongYuan Mark Liao. Scaled-YOLOv4: Scaling cross stage partial network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 13029–13038, 2021.

[3] Piotr Dollar, Mannat Singh, and Ross Girshick. Fast and ´ accurate model scaling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 924–932, 2021.

[4] GitHub - WongKinYiu/yolov7: Implementation of paper - YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors

[5] Ensemble Methods for Object Detection

[6] Park, S., Han, H., Noh, J. H., & Chi, J. (2018, October 30). MDPI.Retrieved.From Real-Time Traffic Risk Detection Model Using Smart Mobile Device

[7] Wang, C. Y., Bonchovskiy, A., Hong, Y., & Liao, M. (2018, October 30). Cornell University. Retrieved July 6, 2022, from https://arxiv.org/abs/2207.02696

[8] GitHub - WongKinYiu/yolov7: Implementation of paper - YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors

[9] Roboflow Universe

[10]대전시 도로 영상 객체 인식 데이터셋 2019

[11] The research of the hazard identification and risk assignment front of the vehicle based on machine vision. (Fangyi Yang a, Huasen Wan b, Tiezhu L i a)

[12] Real-Time Traffic Risk Detection Model Using Smart Mobile Device(Soyoung Park 1 , Homin Han 1, Byeong-Su Kim 1, Jun-Ho Noh 1, Jeonghee Chi 1 and Mi-Jung Choi 2,*)

[13]Park, S., Han, H., Noh, J. H., & Chi, J. (2018, October 30). MDPI. Retrieved from https://www.mdpi.com/1424-8220/18/11/3686 Park, S., Han, H., Noh, J. H., & Chi, J. (2018, October 30). MDPI. Retrieved from https://www.mdpi.com/1424-8220/18/11/3686

[14] Wang, C. Y., Bonchovskiy, A., Hong, Y., & Liao, M. (2018, October 30). Cornell University. Retrieved July 6, 2022, from [2207.02696] YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors

[15] Shouzhong Hu, Jianqin Gu, 1995. The Method and Application of Fuzzy Comprehensive Evaluation. Journal of China Textile University 02, 74– 80

[16] Zhao Hengfeng, Qiu Wanhua, Wang Xinzhe, 1997. Fuzzy Integrative Evaluation Method of the Risk Factor. Systems Engineering – Theory and Practice 17, 95–98.

[17] 주요 교통사고통계