REVIEW OF METHODS FOR DETERMINING OBSTACLES IN THE WAY OF VEHICLES

Togzhanova K.O., Nokerova S.O.

Almaty Technological University, Almaty, Kazakhstan.

Abstract. As regions develop, it becomes important to adapt the transport system to the changing socio-economic dynamics of municipal districts, industrial centers, urban areas, and transportation networks. This includes improving cargo flow management and traffic control systems. Today, vision systems are widely used, especially in road safety and logistics, to address various challenges.

Keywords: Information system, method, transport, object, sensor, algorithm, recognition.

Introduction. The automatic vehicle detection system helps estimate vehicle size and classify it based on size. To determine the size, it is important to know the vehicle's distance, which can be measured using sensors like wave radars, laser radars, or stereo cameras. Among these sensors, monocular cameras (single-lens cameras) are expected to be the most widely used due to their low cost and ability to perform multiple tasks. However, using these cameras for vehicle recognition in real-world conditions still presents challenges.

Vehicle Detection and Recognition. To improve vehicle recognition, a large collection of images from various environments needs to be gathered, and a suitable recognition algorithm must be developed. Recent methods have focused on using features like edge information (the outline of objects) and machine learning algorithms. One commonly used feature set is Histograms of Oriented Gradients (HOGs), which helps detect edges in images.

Many algorithms have been created to improve upon the basic HOG method, but they still face some limitations. For example, these algorithms only use edge information and ignore other important visual elements, such as brightness or color, which humans use to recognize objects. A different method, called the bag-of-functions algorithm, tries to classify objects by combining different feature sets. However, this approach can lead to false positives (incorrectly identifying

something as an object) because it doesn't consider how these features are related to each other.

These methods rely on creating classifiers—models trained to recognize objects based on sample images. However, classifiers can’t be directly corrected when they make mistakes. If an error occurs, the only way to improve the system is to add more examples of incorrect models, which makes fixing specific errors difficult.

Advanced Detection Methods. The first generation of models works by searching the image and then performing classification. One advanced method, R-CNN, uses a technique called selective search, developed by J.R.R. Wylings et al. in 2012, to improve object location. Instead of searching the entire image, the selective search method focuses on smaller regions of the image and groups them hierarchically. These smaller regions are then clustered based on color and similarity. The final result is multiple proposals for regions that can be combined to detect the full object.

Conclusion. Automatic vehicle detection and classification are key to improving road safety and traffic management. While there have been many advancements in image recognition technologies, challenges remain in real-world applications. Future improvements will focus on refining algorithms to use more relevant information, such as color and brightness, and developing systems that can be easily corrected when mistakes occur.

Figure 1 – Custom Search Application

Top: Visualization of the algorithm’s segmentation results.

Bottom: Visualization of the algorithm’s domain recommendations.

R-CNN (Region-based Convolutional Neural Network) Model

The R-CNN model combines two important methods:

1. Selective search: This method helps find possible regions in an image where an object might be located.

2. Deep learning: This technique is used to detect objects in the regions identified by the selective search.

Here's how it works:

- The image is divided into regions, and each region is resized to fit the input size of a CNN (Convolutional Neural Network).
- The CNN extracts a feature vector (a numerical description of the region) with 4096 dimensions.
- This feature vector is then passed to SVM (Support Vector Machine) classifiers that predict the probability of the object being in that region.
- A linear regressor is used to adjust the bounding box (the rectangular box around the object) to improve its accuracy and reduce errors in locating the object.

R-CNN Model Details

- The CNN model was originally trained to classify images using the 2012 ImageNet dataset.
- The model is configured with regional recommendations (called IoUs or Intersection over Union) that are higher than 0.5, which means it only accepts proposals that cover at least 50% of the ground truth area.
- There are two main versions of the model: one trained with the PASCAL VOC 2012 dataset and the other with the 2013 ImageNet dataset.

Performance

- On the PASCAL VOC 2012 test dataset, R-CNN achieved a mAP (mean Average Precision) score of 62.4%, which was 22.0 points higher than the second-best score.

- On the ImageNet 2013 dataset, R-CNN scored 31.4%, which was 7.1 points higher than the second-best score.

Disadvantages of R-CNN:

1. Slow Training: R-CNN requires classifying 2000 region proposals for each image, which takes a lot of time.
2. Not Real-Time: It's too slow for real-time applications, as it takes about 47 seconds per image.
3. Inefficient Region Proposals: The selective search method used for finding regions doesn't adapt during processing, which can lead to poor region proposals.

Fast R-CNN

The Fast R-CNN model, developed by R. Girshick, aims to solve the time issues by improving how region proposals are handled:

- Instead of running a CNN on each region proposal (like R-CNN does), Fast R-CNN processes the entire image at once.
- It uses a selective search method to identify Regions of Interest (RoIs) directly on the generated feature maps (the features extracted by the CNN).
- The regions are then resized using the RoI pooling layer to fit a fixed size.
- The resized regions are passed through fully connected layers to generate a feature vector.
- This feature vector is used to classify the object and adjust the bounding box using a linear regressor.

YOLO (You Only Look Once)

YOLO (You Only Look Once) is a simpler and faster model for object detection developed by J. Redmon et al. in 2016. It works as follows:

- YOLO divides the image into a grid (SxS).
- Each grid cell predicts bounding boxes (areas where objects might be) and provides a confidence score, which is the probability that an object is present in that box.

- The confidence score is calculated by measuring how well the predicted box overlaps with the ground truth box (IoU).

The YOLO model is much faster than R-CNN and Fast R-CNN because it performs object detection in a single pass through the network, making it suitable for real-time predictions.

Summary:

- R-CNN is accurate but slow due to the need to classify many regions and the time it takes to process each image.
- Fast R-CNN speeds things up by processing the whole image at once and using more efficient methods for region proposals.
- YOLO is even faster, as it makes predictions in a single step, and is ideal for real-time applications.

Non-Maximum Suppression (NMS) is a technique used to remove extra boxes (or predictions) that overlap too much, keeping only the best ones.

In YOLOv2, batch normalization was added along with convolutional layers to improve accuracy and reduce the chance of overfitting (when the model learns too much from the training data and performs poorly on new data).

For YOLOv3, the original Darknet19 backbone was replaced by Darknet53, a better feature extraction network, because Darknet19 wasn’t great at detecting small objects.

YOLOv4 made another improvement by replacing the backbone with CSPDarknet53, which helped the model run faster and more accurately.

YOLOv5 is the lightest version, using the PyTorch framework instead of Darknet, making it easier to work with. It also introduced a new focal layer that replaced the first three layers of YOLOv3’s backbone, making the model faster with little loss in accuracy.

In conclusion, YOLO (You Only Look Once) offers a new approach to solve the object detection problem by turning it into a simpler task called regression. YOLO is a one-step deep learning algorithm that uses powerful neural networks to detect objects in images.

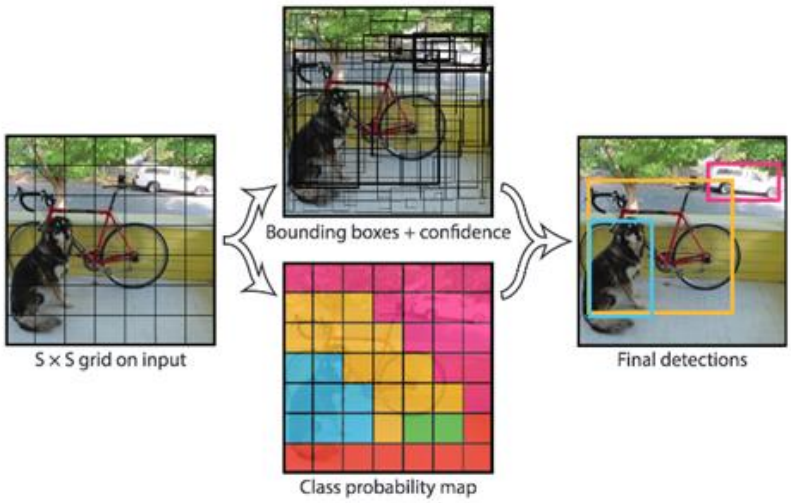

Figure 2 - the working principle of the YOLO V5 algorithm

There are different versions of YOLO. In YOLOv1, the image is split into several smaller, equal-sized grid cells. Each grid cell is responsible for finding the center of an object within it. Each cell can predict a set number of boxes and each box has a confidence score. The box prediction includes five values: the x and y coordinates of the center, the width and height of the box, and the confidence score that the box contains an object.

After predicting the bounding boxes, YOLO uses a method called Intersection over Union (IoU) to compare the boxes and choose the most accurate ones.