Journal on Artificial Intelligence JAI, vol.2, no.3, pp.125-135, 2020

Vehicle Target Detection Method Based on Improved SSD Model

Guanghui Yu[1], Honghui Fan[1], Hongyan Zhou[1], Tao Wu[1] and Hongjin Zhu[1,*]

Abstract: When we use traditional computer vision Inspection technology to locate the vehicles, we find that the results were unsatisfactory, because of the existence of diversified scenes and uncertainty. So, we present a new method based on improved SSD model. We adopt ResNet101 to enhance the feature extraction ability of algorithm model instead of the VGG16 used by the classic model. Meanwhile, the new method optimizes the loss function, such as the loss function of predicted offset, and makes the loss function drop more smoothly near zero points. In addition, the new method improves cross entropy loss function of category prediction, decreases the loss when the probability of positive prediction is high effectively, and increases the speed of training. In this paper, VOC2012 data set is used for experiment. The results show that this method improves average accuracy of detection and reduces the training time of the model.

Keywords: Improved SSD, object detection, vehicles, ResNet101.

1 Introduction

With the development of national economy, more and more people can afford cars. This situation leads to increasing pressure on road traffic. Road traffic management departments are also facing new challenges. The method of using computer to detect and locate vehicles can greatly release the labor force and improve the efficiency of supervision. Target detection of vehicles is a specific application of computer vision, which has been the focus of research workers.

When we use traditional computer vision to detect targets, we usually learn from experience. We extract image feature to object detection by some Maths methods, such as SHIFT [Lowe (2004)], HOG [Dalal and Triggs (2005)], Haar [Panning, Alhamadi and Niese (2008)]. Of course, there are more mature methods, such as deformable part model [Felzenszwalb, Girshick and Allesterd (2010)], background cancellation [Lee (2005)], SVM [Viola and Jones (2003)], sliding window algorithm. Because these methods are basically based on artificial methods to extract image features, resulting in incomplete feature extraction, omission of important features, and it is not enough to deal with the application of scene with changeable background, the scene complexity of vehicle detection is high, and the detection effect of traditional machine vision methods cannot

[1] College of Computer Engineering, Jiangsu University of Technology, Changzhou, 213001, China.

[*] Corresponding Author: Hongjin Zhu. Email: 1240606314@qq.com.

Received: 07 March 2020; Accepted: 06 June 2020.

achieve ideal purpose of practical application due to the influence of lighting, weather, angle of view and other factors.

With the popularity of artificial intelligence, deep learning is sought after by many researchers. When classifying pictures, the method of deep learning achieves higher accuracy than traditional machine vision methods. At the same time, more and more improved convolutional neural networks (CNNs) are being applied to real life, such as LeNet, AlexNet, VGG, NiN, RestNet. In the field of target detection, there are more and more methods combined with deep learning, such as R-CNN [Girdhick, Donahue and Darrell (2014)], Fast-RCNN [Girshick(2015)], Faster-RCNN [Ren, He and Girshick (2017)], SSD [Liu, Anguelov, Erhan et al. (2016)], YOLO [Redmon, Divvala and Girshick (2016)]. Compared with traditional detection methods, the deep learning methods have higher accuracy of detection and faster speed of detection. The deep learning methods are more suitable for complex and changeable scenes.

R-CNN selects multiple proposed areas from the image, and labels them with categories and bounding boxes, then extracts the features of each proposed area by forward calculation of CNNs, and finally, uses these features to predict categories and bounding boxes. However, the obvious disadvantage of R-CNN is that there are too many proposed areas that result in slower speeds. A major improvement of Fast-RCNN is that the forward calculation of the CNN is performed only on the entire image. Then, Faster R-CNN searches selectively bounding box to generate a number of proposed regions, which are respectively labeled on the output of the convolutional neural network. Different areas of interest increase the speed of detection. Faster-RCNN proposes to replace the selective search with the region proposal network, thereby reducing the number of generated regions and ensuring the accuracy of target detection. YOLO detects targets through a single convolutional neural network, which simplifies the process and reduces the accuracy of detection, especially for smaller target recognition. SSD is mainly composed of a basic network block and several multi-scale feature blocks. The basic network block is used to extract the features of the original image. Generally, the deep convolutional neural network is selected, which has better ability to extract feature. Different number and size of bounding boxes are generated based on the basic network block and each multi-scale feature block, and different size targets are detected by predicting the category and offset of the bounding box. In this article, we have improved the SSD model and trained it with a large number of vehicle data sets. The optimization effect of this algorithm is illustrated by comparing with the training results of the classic SSD model.

2 Classic SSD target detection algorithm

At ECCV2016, Liu et al. [Liu, Anguelov, Erhan et al. (2016)] proposed SSD, which is a single-shot multi-frame detection model and a multi-scale target detection model. It has been used widely because of simple and fast characteristics. SSD is mainly composed of a basic network block and many multi-scale feature blocks of different sizes that generated on the basis of the series. In general, the algorithm model presents a pyramid type, and the multi-scales feature block near the bottom layer detect small targets, and the multi-scales feature block near the top of the pyramid are used to detect larger targets because the receptive field of each unit is larger.

2.1 Classic SSD algorithm model

The basic network block of SSD is composed of VGG model which is truncated before the classification layer and two convolutions, and then four multi-scale feature blocks of different sizes are connected in series. The shape of convolution layer decreases gradually, and it is pyramid shaped, which can realize the detection of different size targets. the input is a color RGB image with size of 300×300. The image features are extracted from image by the VGG16 model and then connected to two convolutional layers that size is 19×19 and the number of channels is 1024. The basic network is generated. Then, the basic network block is connected with four convolutional layers of different sizes, that is, multi-scale feature blocks, and each unit of each feature block generates a bounding box with different sizes and shapes, and the number of bounding frames generated by each multi-scale feature block is sequentially decreased. But the receptive field of each unit is getting wider, and the target that can be detected is getting bigger.

2.2 Loss function model

Each prediction bounding box is labeled with two types of labels before training the model. One of the labels represents the category of the target contained in the predicted bounding box, and the other label represents the offset of the real bounding box relative to the predicted bounding box. When the model is trained, first, the model generates some prediction bounding boxes and predicts the category and offset for each predicted bounding box. Then, the model uses the intersection-over-union to measure the similarity between the real bounding box and the predicted bounding box. As shown in Eq. (1), the predicted bounding box position is adjusted according to the predicted offset to obtain the prediction bounding box with the highest similarity. The offset is defined as shown in Eq. (2), and the non-maximum suppression method is used to filter the excess. The prediction bounding box ultimately produces the most similar prediction bounding box.

$$O(a,b) = |\frac{a \cap b}{a \cup b}| \tag{1}$$

In the above formula, a and b represent different bounding boxes respectively. $a \cap b$ indicates intersections, $a \cup b$ indicates merges.

$$\mathrm{R} = (\frac{\frac{\mathrm{x_b}-\mathrm{x_a}}{\mathrm{w_a}}-\mu_\mathrm{x}}{\sigma_\mathrm{x}}, \frac{\frac{\mathrm{y_b}-\mathrm{y_a}}{\mathrm{h_a}}-\mu_\mathrm{y}}{\sigma_\mathrm{y}}, \frac{\log\frac{\mathrm{w_b}}{\mathrm{w_a}}-\mu_\mathrm{w}}{\sigma_\mathrm{w}}, \frac{\log\frac{\mathrm{h_b}}{\mathrm{h_a}}-\mu_\mathrm{h}}{\sigma_\mathrm{h}}) \tag{2}$$

In the above formula, The central coordinates of bounding box a and its assigned real bounding box b are respectively (x_a, y_a), (x_b, y_b).The width and height of the bounding box are $\mathrm{w_a}, \mathrm{h_a}$ and $\mathrm{w_b}, \mathrm{h_b}$. The default value of the constant are $\mu_x = \mu_y = \mu_w = \mu_h = 0$, $\sigma_x = \sigma_y = 0.1, \sigma_w = \sigma_h = 0.2$.

When we train the model, we use two loss functions, one for predicting the loss of the bounding box category. We use the cross-entropy loss function. Another loss function is used to test the loss of the offset of the positive class prediction bounding box, which is a regression problem. The overall loss function is represented by a weighted sum of position *loss* (L_{loc}) and confidence *loss* (L_{conf}), as shown in Eq. (3):

$$L(x,c,l,g) = \frac{1}{N}(L_{conf}(x,c) + \alpha L_{loc}(x,l,g)) \tag{3}$$

N represents the number that matches the real bounding box,l represents the predicted bounding box, g represents the true bounding box, c represents the confidence level for each category, α is a weight parameter that is generally set to 1.

The position loss is as shown in Eq. (4):

$$L_{loc}(\mathrm{x},\mathrm{l},\mathrm{g}) = \sum_{i\in Pos}^{N}\sum_{m\in\{cx,cy,w,\mathrm{h}\}} x_{ij}^{k} smooth_{L1}(l_i^m - G_j^m) \tag{4}$$

G_j^m represents the regression prediction bounding box of the matched real category, and the calculation method of $\mathrm{G_j^m}$is as shown in Eq. (5)-(8).

$$G_j^{cx} = (g_j^{cx} - d_i^{cx})/d_i^w \tag{5}$$

$$G_j^{cy} = (g_j^{cy} - d_i^{cy})/d_i^h \tag{6}$$

$$G_j^w = \log\left(\frac{g_j^w}{d_i^w}\right) \tag{7}$$

$$G_j^h = \log\left(\frac{g_j^h}{d_i^h}\right) \tag{8}$$

cx and *cy*, w and h respectively represent the coordinates and size of the prediction bounding box. d_i^w, d_i^h represent the size scaling factor. d_i^{cx}, d_i^{cy}represents the offset size.

The L1 norm loss used for predicting the offset is as shown in Eq. (9):

$$\mathrm{smooth_{L1}(x)} = \begin{cases} 0.5\mathrm{x}^2, & |\mathrm{x}| < 1 \\ |\mathrm{x}| - 0.5, & \text{otherwise} \end{cases} \tag{9}$$

Confidence level is shown in Eq. (10):

$$\mathrm{L_{conf}(x,c)} = -\sum_{\mathrm{i\in Pos}}^{\mathrm{N}} \mathrm{x_{ij}^p \log\left(C_i^p\right)} - \sum_{\mathrm{i\in Neg}} \log\left(\mathrm{C_i^0}\right) \tag{10}$$

C_i^p indicates the confidence of the *i*-th bounding box of category p, $\mathrm{C_i^p}$ is calculated as $\frac{exp\,(c_i^p)}{\sum_p exp\,(c_i^p)}$.

3 Improved SSD model

3.1 ResNet101 extracts image features

In the traffic image with resolution of 300×300, when the vehicle is in a far position, the vehicle has a small proportion in the picture and the resolution is low. When we use the classic VGG16 to extract image features, misdetection and missed detection often occur. The reason is that the convolutional layer of the VGG model is shallow, and it is impossible to extract higher semantic image features. So, the detection effect on smaller targets is not good. As the number of network layers increases, more features are extracted and the accuracy of target detection is improved. However, when the number of neural network layers reaches a certain level, the detection accuracy will no longer increase or even decrease, and even the gradient will disappear. This is because the training model cannot find the optimal solution due to the increase of the number of neural network layers. To solve this problem, He et al. [He, Zhang and Ren (2016)] proposed a residual network. The residual network structure can largely avoid the situation that the gradient disappears with the increase of the number of neural network layers. Therefore, this network structure can

be used to increase the number of network layers to extract higher semantic image features, thereby improving the accuracy of target detection. In the residual block, the input can propagate forward faster through the data lines across the layers. The implementation of this process is called identity mapping, that is, the input can be connected to the output of the network layer with skipping several neural network layers, as shown in Fig. 1.

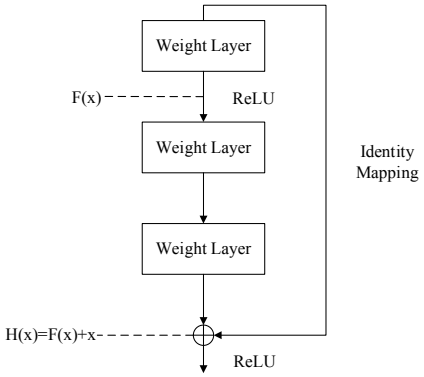

Figure 1: Residual Block

In the residual block, the residual function *F(x)* is added to the input feature x across the layer by normal forward calculation, thereby generating a new feature function *H(x)*. As shown in Eq. (11). This paper uses ResNet101 instead of the classic VGG16, which has a good effect on the detection of relatively distant vehicle targets on the road.

$$H(x) = F(x) + x \tag{11}$$

3.2 Improve the loss function of the predicted offset

The loss function of the predicted offset in the classic SSD model is L_1norm. In this paper, a Hyper Parameter is added to the loss function to control the smooth region. The loss function is shown in Eq. (12).

$$f(x) = \begin{cases} (\sigma x)^2/2\ , & |x| < 1/\sigma^2 \\ |x| - 0.5\ /\sigma^2\ , & \text{otherwise} \end{cases} \tag{12}$$

When σ is large, the loss function is similar to the L_1 norm loss. When σ is small, the loss function is smoother, as shown in Fig. 2.

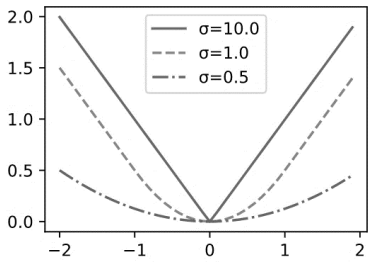

Figure 2: Improved L1 norm comparison

When the bounding box offset in the vehicle target detection is back to the prediction, the improved loss function can reduce the loss more quickly at the same time, which is convenient for the target training to improve the target detection prediction accuracy and save the training time.

3.3 Improve the loss function of class prediction

In the category prediction of target detection, the classical SSD model uses a cross entropy loss function. Let the prediction probability of the real category j be p_j, and the cross entropy loss be $-logp_j$, the improvement of this paper is to add the given positive hyperparameter γ and α.The loss function is defined as shown in Eq. (13).

$$L_{p_j} = -\alpha(1-p_j)^{\gamma}\log p_j \tag{13}$$

It can be seen from the formula that the loss of the positive class prediction probability can be significantly reduced by increasing the value of γ, as shown in Fig. 3.

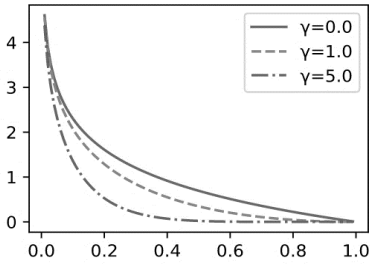

Figure 3: Comparison of improved cross entropy loss

When the model calculates the category prediction in the vehicle target detection, the improved cross entropy loss function can reduce the loss more quickly at the same time, which is convenient for the target training to improve the target detection prediction accuracy and save the training time.

4 Experimental analysis and discussion

4.1 Experimental data and settings

The experimental data of this paper is voc2012 data set. We choose the image including the vehicle from the data set, and then expand the data set through the image augmentation technology. The expanded data set is used as the data set of this experiment. Some vehicle sample images are shown in the Fig. 4. In order to increase the generalization ability of the training model, a random sampling method is adopted for the data set. We use the ten-fold cross-validation method to verify the trained model. The data set is randomly divided into 10 parts. Each time the model is trained, nine data sets are selected as the training set, and the remaining data sets are used as verification set. The experiment was carried out ten times, and finally the average accuracy rate MAP was taken as an index for evaluating vehicle target detection. There are two types of annotations for data sets, one for the category and the other for the bounding box position information. The sample labeled for the image data set is shown in the Fig. 5.

Figure 4: Sample of vehicle image

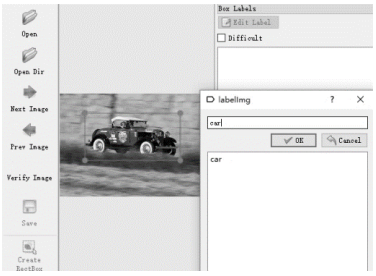

Figure 5: Example of vehicle image annotation

4.2 Experimental platform and configuration parameters

The operating environment for this article is Windows 10, and the required software and hardware are as follows: the GPU is NVIDIA GTX2070, the CPU is Intel Core i7-8700 3.2 GHz, the memory is 32 GB, the deep learning framework is Tensorflow, the programming language is Python 3.6, and the deep learning network acceleration library is CUDA 8.0 combined with CUDNN 5.1.

Because of the high complexity of the training model, the number of iterations set 60000, the learning rate set 10^{-4}, the image batch size set 8, the weight attenuation parameter set 5×10^{-4}, the learning rate attenuation factor set 0.94, and the proportion of GPU set 0.7.

4.3 Process of the experiment

When comparing the results of the classical SSD model with the results of the improved SSD model in this paper, we choose Mean average precision (*MAP*) as the indicator, as shown in Eq. (14).

$$\mathrm{MAP} = \int_0^1 \mathrm{P(R)dR} \tag{14}$$

P represents the accuracy rate; R represents the recall rate.

Because the model trained in the experiment contains many convolutional neural networks, a large number of prediction bounding boxes of different sizes are generated. To eliminate useless bounding boxes, we discard the bounding box below the threshold. The specific method is: assume that the predicted bounding boxes in the image are respectively $A_1, A_2, \ldots, A_{n_a}$,the real bounding boxes are respectively $B_1, B_2, \ldots, B_{n_b}$ and $n_a \geq n_b$.The matrix is defined as $X \in R^{n_a \times n_b}$, and the element x_{ij} located in the j-th column of the i-th row is the intersection ratio of the predicted bounding box A_i to the real bounding box B_j.

First, we find the largest element in the matrix *X* and record the row index and column index of the element as i_1, j_1. We assign a real bounding box B_{j_1} to the predicted bounding box A_{i_1}. Obviously, the paired similarity between the predicted bounding box A_{i_1} and the real bounding box B_{j_1} in this experiment is the highest in all pairs of "predicted bounding box-real bounding box". Next, we discard all the elements in row i_1 and column j_1 of matrix *X*. We find the largest element remaining in matrix *X* and record the row index and column index of the element as i_2, j_2 respectively. The real bounding box B_{j_2} is assigned to the predicted bounding box A_{i_2}. Finally, all elements on the i_2-th row and the j_2-th column in the matrix *X* are discarded. At this time, the elements of two rows and two columns in the matrix *X* are discarded. And so on, the cycle ends until all n_b column elements in the matrix *X* are discarded. At this time, a real bounding box has been assigned to each of the n_b prediction bounding boxes. Then, traverse the remaining $n_a - n_b$ predict boundary boxes. However, there may still be many prediction bounding boxes in the same detection target. In order to preserve the most ideal unique prediction bounding box, this paper uses non-maximum suppression to remove redundant prediction bounding boxes.

In the training target detection algorithm model, the specific algorithm detection process is shown in the Fig. 6.

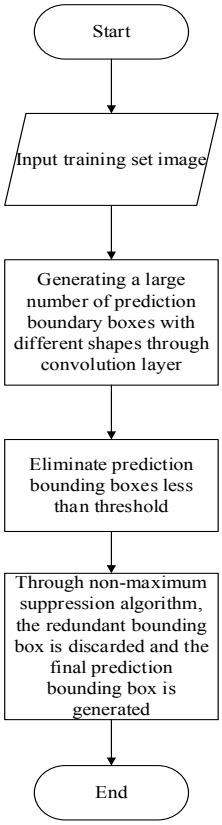

Figure 6: Algorithm flow of target detection

4.4 Experimental results and analysis

In this experiment, Fast-RCNN, Faster-RCNN, classic SSD and improved SSD algorithm are compared in average accuracy. Experiments are carried out on the VOC2012 data set. It can be seen from Tab. 1 that the average accuracy of the Fast-RCNN algorithm is 67.4%, the average accuracy of the Faster-RCNN algorithm is 74.2%, the average accuracy of the classic SSD algorithm is 73.9%, Due to the stronger feature extraction capability of the multi-layer residual network, the average accuracy of the improved SSD algorithm is 76.8%.

Table 1: MAP Comparison

Method	Data set	(mAP)/%
Fast-RCNN	VOC2012	67.4
Faster-RCNN	VOC2012	74.2
classical SSD	VOC2012	73.9
Improved SSD	VOC2012	76.8

The improved SSD model optimizes the loss function of the predicted offset and the loss function of the class prediction, respectively, so that the loss function is smoother near the zero point and the rate of decline is faster at larger values. By comparing the algorithm in this experiment with the classic SSD algorithm model, the results show that, the average accuracy of the classic SSD model is 53.1%, 68.1% and 73.9% when the number of iterations is 20000, 40000 and 60000 respectively, and when the Hyper Parameter σ is set to 1.0, γ is set to 1.0, the average accuracy of the improved SSD model is 56.9%, 71.4% and 75.8% when the number of iterations is 20000, 40000 and 60000 respectively. When the Hyper Parameter σ is set to 0.5, γ is set to 5.0, the average accuracy of the improved SSD model is 59.1%, 73.1% and 76.8% when the number of iterations is 20000, 40000 and 60000 respectively. As shown in Tab. 2

Table 2: MAP comparison of the same number of iterations

Method	Value of σ	Value of γ	mAP of 20000 iterations /%	mAP of 40000 iterations /%	mAP of 60000 iterations /%
classical SSD	—	—	53.1	68.1	73.9
Improved SSD	1.0	1.0	56.9	71.4	75.8
Improved SSD	0.5	5.0	59.1	73.1	76.8

In the target detection algorithm model adopted in this paper, some samples of the renderings of vehicle target detection are shown in Fig. 7. The results of experiments based on the VOC2012 data set show that the improved SSD model proposed in this paper has better vehicle detection results. Similarly, the algorithm is also applicable to the detection of other objects such as cats and dogs. The algorithm only needs to modify the input of the model and use the corresponding data set image for training. Therefore, the improved SSD model algorithm proposed in this paper has wide applicability.

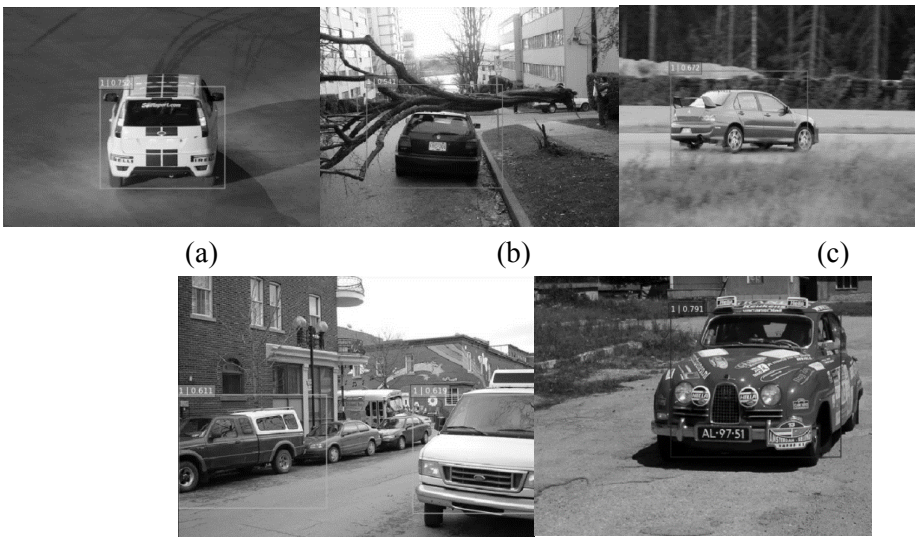

Figure 7: Renderings of vehicle target detection

5 Conclusion

This paper proposes a vehicle detection method based on the improved SSD model. This method uses the ResNet101 neural network layer instead of the VGG16 feature extraction neural network layer, because the ResNet101 neural network layer has stronger feature extraction capabilities. At the same time, this method optimizes the predicted offset loss function and the class prediction loss function to make the loss drop faster and more stable. The results show that the improved SSD model proposed in this paper has higher average accuracy and can effectively improve the accuracy and robustness of vehicle detection compared with the traditional SSD model. At the same time, the improved SSD model speeds up the training of the model. But the model also has drawbacks. When there are overlapping objects in an image, the accuracy of the detection is not ideal. We will begin to solve this problem in the next step to further improve the detection accuracy of the model.

Funding Statement: This work was supported in part by National Natural Science Fund of China (61806088, 61902160), Qing Lan Project of Jiangsu Province and Natural Science Foundation of Jiangsu Province (BK20160293), Changzhou Science and Technology Support Plan (CE20185044).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Dalal, N.; Triggs, B. (2005): Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, CA, USA, pp. 886-893.

Felzenszwalb, P. F.; Girshick, R. B.; Allesterd, M. C. (2010): Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627-1645.

Girdhick, R.; Donahue, J.; Darrell, T. (2014): Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587.

Girshick, R. (2015): Fast R-CNN. *Proceedings of IEEE International Conference on Computer Vision.*

He, K.; Zhang, X.; Ren, S. (2016): Deep residual learning for image recognition. *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA. pp. 770-778.

Lee, D. S. (2005): Effective gaussian mixture learning for video background subtraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 827-832.

Liu, W.; Anguelov, D.; Erhan, D.; Lou, X. P.; Parker, G. A. et al. (2016): SSD: single shot multibox detector. *European Conference on Computer Vision*, pp. 21-37.

Lowe, D. G. (2004): Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91-110.

Panning, A.; Alhamadi, A. K.; Niese, R. (2008): Facial expression recognition based on Haar-like feature detection. *Pattern Recognition & Image Analysis*, vol. 18, no. 3, pp. 447-452.

Redmon, J.; Divvala, S.; Girshick, R. (2016): You only look once: Unified, real time object detection. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition.*

Ren, S.; He, K.; Girshick, R. (2017): Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149.

Viola, P.; Jones, M. (2003): Rapid object detection using a boosted cascade of simple features. *Proceedings of 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 511-518.