IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 38, No. 3, June, 2021, pp. 719-730
Journal homepage: http://iieta.org/journals/ts

An Object Detection Framework Based on Deep Features and High-Quality Object Locations

Yurong Guan[1], Muhammad Aamir[1*], Zhihua Hu[1], Zaheer Ahmed Dayo[1], Ziaur Rahman[1], Waheed Ahmed Abro[1], Permanand Soothar[2]

[1] Department of Computer Science, Huanggang Normal University, Huanggang 438000, China
[2] School of Electronic and Information Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

Corresponding Author Email: aamirshaikh86@hotmail.com

https://doi.org/10.18280/ts.380319

Received: 28 December 2020
Accepted: 5 May 2021

Keywords:
object detection, high-quality proposals, convolutional neural network (CNN), deep features

ABSTRACT

Objection detection has long been a fundamental issue in computer vision. Despite being widely studied, it remains a challenging task in the current body of knowledge. Many researchers are eager to develop a more robust and efficient mechanism for object detection. In the extant literature, promising results are achieved by many novel approaches of object detection and classification. However, there is ample room to further enhance the detection efficiency. Therefore, this paper proposes an image object detection and classification, using a deep neural network (DNN) for based on high-quality object locations. The proposed method firstly derives high-quality class-independent object proposals (locations) through computing multiple hierarchical segments with super pixels. Next, the proposals were ranked by region score, i.e., several contours wholly enclosed in the proposed region. After that, the top-ranking object proposal was adopted for post-classification by the DNN. During the post-classification, the network extracts the eigenvectors from the proposals, and then maps the features with the softmax classifier, thereby determining the class of each object. The proposed method was found superior to traditional approaches through an evaluation on Pascal VOC 2007 Dataset.

1. INTRODUCTION

The recognition of objects in images has been the focus of many academicians and practitioners. Over the years, more appropriate contemporary algorithms have emerged to enhance the performance of object detection. However, most object detection algorithms perform poorly, because different viewpoints vary in size, angle, perspective, deformation, occlusion, illumination, and background clutter. In recent years, much efforts have been made to improve object detection methods.

Based on modern technologies like deep learning, numerous cutting-edge object detection approaches are now available for detecting and classifying diverse objects in images. Most of them rely on the deep learning tool of convolutional neural network (CNN) to classify the objects, such as to achieve remarkable detection performance. This technological advancement opens new horizons for researchers, owing to its superior performance in many practical disciplines, including object detection [1], object localization [2], object tracking [3], image generation [4], human pose estimation [5], text recognition and detection [6], visual question answering [7], action recognition [8], visual saliency detection [9], and scene labeling [10]. However, more novel technologies are necessary to further enhance the overall detection efficiency.

In the last decade, many have strived to make detection models more efficient. Object detection is a two-step process: object proposal (location) generation and post-classification. Therefore, the performance of object detection hinges on both object proposal algorithms and post-classification networks. The location of objects in the image has drawn significant attention from the academia. To realize efficient and accurate classification, it is necessary to cover as many image objects as possible in a few object proposals. This can be achieved by reducing the search space for an object's location, and minimizing the number of false alarms in resulting locations.

Previously, many approaches have been introduced to generate a few proposals that cover as many objects as possible, namely, constrained parametric min-cuts (CPMC) [11], Rantalankila's method [12], Rahtu's method [13], Objectness [14], binarized normed gradients (BING) [15], selective search [16], edge box [17], and Endres' method [18]. Unfortunately, none of these approaches could generate high-quality proposals. All of them create too many proposals per image, resulting a high detection recall at the expense of computing cost. Besides, lots of false alarms will appear, and the search space will remain large, which undermine the performance of subsequent tasks like classification. The other prominent defects of the said proposal generation approaches include inaccuracy, redundancy, class dependence, and excessive locations. What is worse, there is no proper mechanism to refine the generated proposals, so as to improve classification accuracy at the low cost. Hence, there is ample room to improve the proposal generation mechanism to produce a few high-quality proposals with a high recall, a high computing efficiency, and a high detection performance. For this purpose, our research deduces a mechanism that generates fewer yet high-quality proposals based on scoring and ranking mechanism.

For object detection, the improvement of proposal

generation is beneficial, but not adequate. Any advancement in proposal generation needs to be backed up with a robust classification network. In recent years, the classification accuracy is rising due to the development of deep learning, especially the progress of CNN. A variety of powerful CNN architectures have been designed to realize desired accuracy in classification tasks, such as AlexNet [19], VGGNet [20], GoogleNet [21], ResNet [22], DenseNet [23], to name but a few. In practice, these networks are pretrained on the opensource dataset of ImageNet, and adopted as a backbone network for various detection tasks. Nevertheless, it takes a long time and a heavy computing load to retrain them for classification tasks. The CNN-based based object detection methods can be roughly divided into proposal-based techniques and non-proposal-based techniques. Each proposal-based technique consists of two steps: proposal generation, and post-classification, while each non-proposal-based technique completes detection in one step only. The representative proposal-based techniques include region-based CNN (R-CNN) [24], Fast R-CNN [25], Faster R-CNN [26], region-based fully convolutional network (R-FCN) [27], and Mask R-CNN [28].

The existing literature provides insights into CNN-based object detectors. For example, R-CNN first acquires high-quality class-independent object proposals from an image through selective search, and extracts the regional features by forward computing to predict the class of each region. However, the high computing cost makes the network unfit for actual applications. One of the main impediments of R-CNN lies in the slow speed. For a single image, the detection may require thousand rounds of forwarding computing. Later, Microsoft researchers improved R-CNN into Fast R-CNN. This network is more efficient than R-CNN, which shares the computing load of the backbone network. Unlike R-CNN, Fast R-CNN obtains all the proposals in an image simultaneously. This improved region extraction process leads to better efficiency. Furthermore, Fast R-CNN adopts a region of interest (ROI) pooling layer on the feature map for classification and regression tasks. Despite being faster than R-CNN, Fast R-CNN faces a main drawback: the dependence on selective search for proposal extraction. To acquire more robust outcomes, Ren et al. [26] introduced Faster R-CNN, which overcomes the computing problems of both R-CNN and Fast R-CNN. In Faster R-CNN, selective search is replaced with a region proposal network (RPN) to prevent the generation of redundant proposals. Thus, the detection becomes faster and less costly. In addition, Dai et al. [27] proposed another improved CNN-based object detector called R-FCN. This network combines the two stages of detection in previous approaches into a single stage, using only convolutional layers. Besides, fewer convolutional layers are adopted to accelerate the detection. He et al. [28] extended Faster R-CNN into Mask R-CNN for object detection, which achieves a high performance by segmenting the target image pixel by pixel. Furthermore, you only look once (YOLO) [29], and single shot detector (SSD) [30] were developed for regression-based object detection. The two single-stage, proposal-free methods are widely used to detect objects in real time. In addition, many regularization techniques emerged to enhance the classification accuracy of the above networks [31, 32].

This paper puts forward a two-stage object detection technique. In the first stage, high-quality object proposals were generated, and ranked by score to improve classification accuracy and reduce computing cost. In the second stage, the ranked proposals were combined into an eigenvector by modified VGGNet, and the softmax classifier was called to classify the objects. The VGGNet was modified to extract dense pixel-level features. The modification, i.e., the removal of the last layer, substantially improves the overall performance of our technique. There are two advantages of this modification: First, the time and memory costs of the fully connected layer are reduced during training and testing; Second, full-size prediction can be realized more accurately, thanks to the removal of the last pooling layer, whose output is much smaller than the input image. The most significant contributions of this research are outlined as follows:

(1) Proposing a suitable image processing technique to generate a few high-quality proposals that may contain the objects in the image, and prevent the generation of redundant proposals.

(2) Proposing and validating a robust image processing technique that recognizes the objects in the generated proposals, and improves the classification performance of the overall detection task.

(3) Proposing and assessing a high-confidence, efficient, and precise technique for object detection based on distinct class proposals in images.

The remainder of this paper is organized as follows: Section 2 presents a thorough literature review; Section 3 introduces the proposed method and the dataset used in this research; Section 4 evaluates the experimental results; Section 5 draws the conclusions and predicts future research directions.

2. LITERATURE REVIEW

Object detection approaches generally cover two stages, i.e., proposal generation, and object classification. In most cases, object proposals are created by proposal generation algorithms, and categorized with object classifiers. There are generally two kinds of proposal generation algorithms: grouping and window scoring.

The grouping algorithms decompose each image into multiple hierarchical segments that are expected to contain an object, and merge them based on similarities in color, texture, size, shape, etc. The performance of grouping algorithms exclusively depends on the technique that determines the initial segment. Felzenszwalb and Huttenlocher [33] presented an efficient and fast approach for segment initialization. More suitable than any other public solutions, Felzenszwalb's approach produces a small set of initial segments rapidly, and treats the segmentation as a graph problem, where the vertices and edges are the components to be segmented [34]. Carerira and Sminchisescu [11] and Endres and Hoiem [18] developed several models to generate class-independent object proposals. In their models, different seeds and parameters are adopted to solve multiple graph-cuts for generating two-fold segments (foreground and background), and the resulting segments are taken as object proposals. These models can efficiently forecast the segment that encloses the whole object, according to the ranking of the proposals, and generate high-quality segmentation masks. However, their speed is dragged down by the dependence on gPb edge detector. Similarly, selective search [16] prepares high-quality class-independent proposals using super pixels and clustering technique, and becomes a common approach for object detection. During selective search, the initial segments are grouped through hierarchical

clustering, in the light of color, shape, size, or texture, and various proposals are covered by different color spaces. Compared to other methods, selective search generates high-quality proposals at a high recall and a fast speed. However, the proposal generation is not well controlled. The lack of scoring and ranking mechanism could bring unwanted proposals, which undermine the classification performance.

Window scoring determines the presence of an object by computing the score of each generated window, i.e., the likelihood of the object being contained in the window. Window scoring algorithms are much faster than grouping algorithms, but less accurate in object localization. As a popular way to generate object proposals, Alexe et al. [14] calculates the probability for an object to present in an image. Different image cues, including edge density, saliency, super pixel straddling, and color contrast, are used to derive image features, and are merged with the Bayesian model to calculate the score of each candidate window. Capable of achieving a few high-quality proposals, Objectness operates fast but has an overall low accuracy of localization. Likewise, edge box [17] determines the edges in an image with an edge detector, combines the eight neighboring edges into an edge set (object proposal), computes the score of each box by sliding a window over an image scale, and relies on the score to refine the proposals. Similarly, Rahtu et al. [13] generated many random proposals (candidate boxes) by Objectness, obtained even more proposals through threefold super pixel segmentation (i.e., single, double, and triple), and multiplied these proposals to obtain candidate proposals. Rahtu's method improves the scoring mechanism of Objectness, and rates each proposal with additional low-level features.

Every object detection task needs to be completed in two stages. Apart from improving proposal generation, it is important to enhance the overall efficiency with a robust proposal classifier. Deep learning neural networks like CNN are extensively adopted to improve classification accuracy. CNN has achieved exceptional results in machine learning problems, notably involving image classification datasets. The advancement of CNN and other deep learning techniques has greatly promoted image classification precision. Many researchers and developers are energized to develop advanced models to solve complex problems, which are beyond the capacity of standard artificial neural networks (ANNs). In this process, CNN has been tailored to various machine learning problems (object recognition, object classification, and speech recognition), particularly those related to massive image data. The earliest CNN was designed by LeCun et al. [35] in 1990, and was enhanced in 1998 [36]. LeNet-5 is a multi-layer CNN for handwritten digit classification. The network can be trained by the backpropagation algorithm [37]. By converting the original image into useful representations, LeNet-5 could identify visual patterns explicitly from raw pixels without laborious preprocessing [38]. Nonetheless, the network performance is often dampened by the lack of sufficient training data and computing capacity for complex problems, such as largescale image and video classification.

Since the inception of LeNet-5, many other methods have been created to overcome the issues through the training of deep neural network (DNN). In 2012, Krizhevshy et al. [19] proposed a novel deep CNN called AlexNet, an ultradeep network of 60,650,000 neurons. The incorporation of high-performance computing machines like graphics processing unit (GPU) enables AlexNet to outperform all the previous networks by reducing the error rate by 15.3%. The performance gap between AlexNet and the second-best performer, which is not a CNN variant, is as much as 10%, a huge edge in feature-based object detection. Marveled at the prominence of AlexNet, Zeiler and Fergus [39] established a model to visualize and comprehend CNN, endeavoring to yield better outcomes than AlexNet. After visualizing AlexNet, they noted that the minor modification in the network perspective could improve classification effect, and discovered that AlexNet contains too many parameters, which brings a high computing cost. To improve AlexNet, Fan et al. [10] offered an architecture of much fewer neurons, namely, network in network (NIN) (7.5 million vs. 60 million). Google's team extended AlexNet into a deep CNN called Inception [40, 41], which curtails the neurons further to 4 million. For object detection, Inception utilizes the same method as R-CNN, except that the proposals are generated by multi-box technique and selective search: half of the proposals are produced through selective search, while 200 are selected by multi-box technique [42]. Dai et al. [43] improved Inception into ResNet, which enhances the efficiency of CNN with two new models, namely, deformable convolution and deformable ROI pooling. The VGGNet team also built a network much deeper than CNN, and indicated that image object detection hinges on the depth of the convolutional mechanism. The team designed 19 weighted layers, along with a convolutional stride of 1 and a small 3×3 convolutional filter, and proved that no piece of information is squandered during the detection of object features.

Much efforts have been made to improve the robustness and efficiency of CNN in various computer vision tasks, and to solve the varied problems in adopting CNN and its components, such as network optimization, regularization, layer design, activation and loss functions, computing speed, etc. For instance, Iandola et al. [44] introduced a small DNN architecture with $50x$ fewer parameters, and attained comparable accuracy on ImageNet as AlexNet. The architecture can be compressed to less than 0.5 MB, $510x$ smaller than the latter. Redmon et al. [29] designed YOLO based on CNN architecture for unified and real-time object detection. The network contains 24 convolutional layers, followed by two fully connected layers. To reduce the feature map from previous layers, each convolutional layer can be adjusted to 1×1. Besides, the convolutional layers are pre-trained on the ImageNet database by setting the resolution of an input image to half (224×224). Furthermore, the resolution is doubled for detection purposes. To realize efficient feature-based object detection, Gehring et al. [45] presented a CNN-based architecture for sequence-to-sequence learning, which outshines the existing approaches, which are unable to discover compositional structure in the sequences. Their architecture can parallelize all the elements during the training phase for better computations. Besides, nonlinearities are made constant and independent of the input length, making the optimization more natural. In addition, Bansal et al. [46] put forward PixelNet to enhance the overall pixel-based detection performance for representations. However, the network is not sufficient to achieve state-of-the-art detection performance. The proposal-based object detection methods with CNN as a classifier include R-CNN [24], Fast R-CNN [25], Faster R-CNN [26], R-FCN [27], and Mask R-CNN [28].

This paper offers a method to detect and classify objects in images using object proposals and deep learning neural network, with the aim to eliminate the problems with the above techniques in object detection. Firstly, high-quality class-

independent object proposals were generated from each image. Next, all the pixels of these proposals were warped to the input size of the deep neural feature detector, producing a fixed-length input eigenvector for each proposal. After that, these features were imported to the softmax classifier to categorize objects. The research provides a brand-new paradigm compared to the earlier studies.

3. METHODOLOGY

Figure 1 explains the proposed method for object detection, which uses the deep learning network based on class-independent proposals, stage by stage. There are three major steps of our method: proposal generation, feature computing, and object classification.

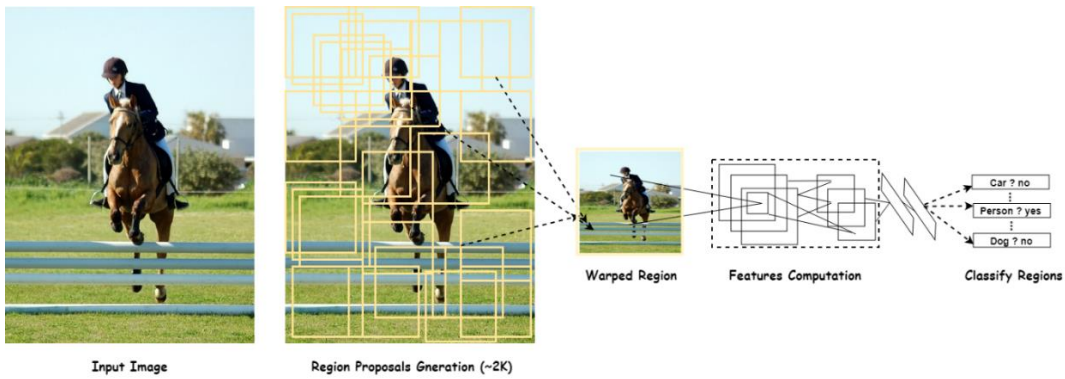

Figure 1. Roadmap of our method

3.1 Proposal generation

Figure 2. Proposal ranking

The first step is to obtain a small set of high-quality proposals for an object, which are independent of the class (Figure 2). The traditional window scoring technique was integrated with hierarchical segmentation to obtain high-quality class-independent object proposals efficiently. First, agglomerative clustering was implemented to acquire the proposals (bounding boxes). Second, each proposal (box) was rated by subtracting the sum of the strengths of all the edges in each edge set with the strength of the edges that overlap the box. Third, the proposals were ranked by the score. Fourth, the proposal with the highest score (top-ranking proposal) was chosen for object classification. This strategy can generate object proposals in a very short time, lower the false positive rate, and dramatically improve the classification performance. The proposal generation is summarized as follows:

Step 1. Segmentation

Initialize a set of proposals with Felzenszwalb and Huttenlocher's graph-based segmentation methods.

Step 2. Hierarchical clustering

Group the obtained proposals by the similarity ratio amongst the neighboring proposals.

Step 3. Edge detection and edge grouping

Plot image edges with the structured edge detector, and group neighboring edges in the edge map by their similarity in orientation.

Step 4. Proposal scoring

Rate each proposal group by subtracting the edge strength in the proposal with that of the edges that overlap the box. For each edge set, compute the value of $w_b(s_i)$ to see if the set is wholly enclosed in the proposal. If $w_b(s_i) = 0$, the edge set is not entirely enclosed in the box. If the edge set is wholly enclosed in the box b, (s_i) can be calculated by:

$$w_b(s_i) = 1 - max_t \prod_j^{|T|-1} a(t_j - t_{j+1}) \quad (1)$$

where, a is the affinity; t is the order path with length T. Thus, formula (1) yields the order path with the maximum affinity within each edge set. Then, the score of each proposal can be computed by:

$$h(b) = \frac{\sum_i w_b(s_i) m_i}{2(b_w + b_h)^k} \quad (2)$$

where, b_w and b_h are box width and box height, respectively; k is the bias of large boxes.

Step 5. Ranking

Rank the proposals by score.

3.2 Feature computing

So far, the proposals have been obtained. The next step is to extract the features from the obtained proposals. First, the proposals were wrapped into a fixed-length input required by the feature detector. Then, a fixed size eigenvector was generated by the detector through each forward computing. Finally, the eigenvectors were mapped to classify each object. The feature computing takes place in two stages.

3.2.1 Proposal warping

The feature detector in our model requires the input to be $224 \times 224 \times 3$ in size. However, the proposals obtained in step 1 are not necessarily of that size. To generate fixed-length eigenvectors, the proposals were converted regardless of their sizes or aspect ratios into the input size required by the feature detector. In our experiments, 2,000 proposals were extracted from each image, and then wrapped to the said input size (Figure 3).

Figure 3. Obtained proposals

3.2.2 Feature detection

As shown in Figure 4, our object detection model relies on VGGNet to extract features from the obtained proposals. Developed by Simonyan and Zisserman [20], VGGNet is an ultradeep network of 13 convolutional layers and 3 fully connected layers. The network is very attractive for its uniform architecture. It has been widely adopted to extract features from images in the field of computer vision. Karen's team discovered that the depth of CNN has a considerable impact on object detection, and eliminated information loss by adopting 19 weighted layers, 3×3 small convolutional filters, and a convolutional stride of one. Figure 5 presents the standard architecture of VGGNet.

In our approach, the first model extracts the features, and the second predicts the class of each object, using softmax classifier. Tables 1 and 2 describe the layers of the first and second models, respectively. For comparison, the baseline VGGNet was trained on an extensive dataset (ImageNet) that contains 1,000 classes of objects. The trained network could extract universal features in images. To verify the robustness of our system, most experiments were conducted on images from Pascal VOC 2007 database. The dataset contains 9,963

images about 20 classes of objects: Person, Animal, Vehicle, and Indoor. Unlike the baseline VGGNet, our model does not contain the final prediction layer to improve the performance. Besides, the eigenvectors obtained by the feature extractor model are passed to the prediction model for object classification by Softmax classifier.

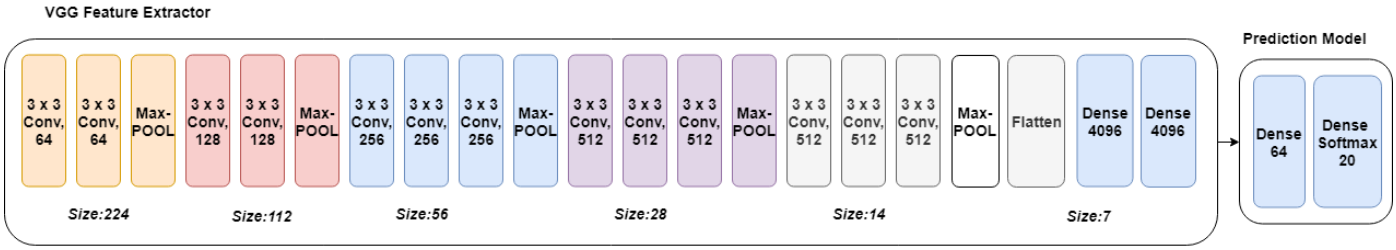

Figure 4. Workflow of feature detector in our system

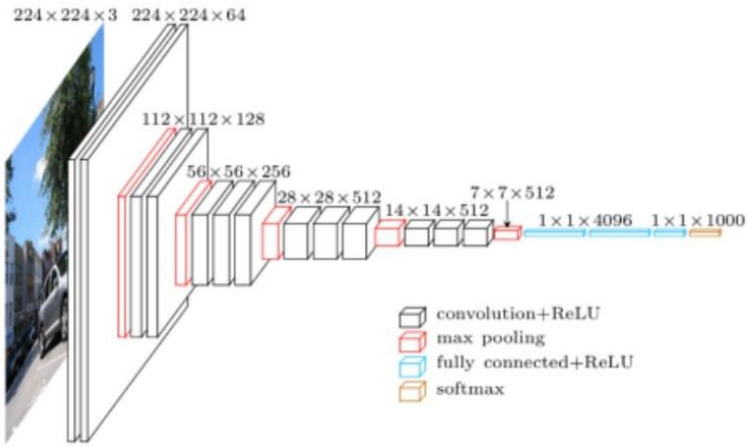

Figure 5. Standard architecture of VGGNet [20]

Table 1. Layers of feature detector

Layer (type)	Output size	Number of parameters
block1_2Dconv1 (Conv2D)	(None, 64, 224, 224)	1,792
block1_2Dconv2 (Conv2D)	(None, 64, 224, 224)	36,928
pool1 (MaxPooling2D)	(None, 64, 112, 112)	0
block2_2Dconv1 (Conv2D)	(None, 128, 112, 112)	73,856
block2_2Dconv2 (Conv2D)	(None, 128, 112, 112)	147,584
pool2 (MaxPooling2D)	(None, 128, 56, 56)	0
block3_2Dconv1 (Conv2D)	(None, 256, 56, 56)	295,168
block3_2Dconv2 (Conv2D)	(None, 256, 56, 56)	590,080
block3_2Dconv3 (Conv2D)	(None, 256, 56, 56)	590,080
pool3 (MaxPooling2D)	(None, 256, 28, 28)	0
block4_2Dconv1 (Conv2D)	(None, 512, 28, 28)	1,180,160
block4_2Dconv2 (Conv2D)	(None, 512, 28, 28)	2,359,808
block4_2Dconv3 (Conv2D)	(None, 512, 28, 28)	2,359,808
pool4 (MaxPooling2D)	(None, 512, 14, 14)	0
block5_2Dconv1 (Conv2D)	(None, 512, 14, 14)	2,359,808
block5_2Dconv2 (Conv2D)	(None, 512, 14, 14)	2,359,808
block5_2Dconv3 (Conv2D)	(None, 512, 14, 14)	2,359,808
pool5 (MaxPooling2D)	(None, 512, 7, 7)	0
flatten_6 (Flatten)	(None, 25,088)	0
dense_22 (Dense)	(None, 4,096)	102,764,544
dense_23 (Dense)	(None, 4,096)	16,781,312
Total params: 134,260,544 Trainable params: 134,260,544 Non-trainable parameters: 0		

Table 2. Layers of prediction model

Layer (type)	Output size	Number of parameters
dense_1 (Dense)	(None, 64)	262,208
dense_2 (Dense)	(None, 20)	1,300
Total params: 263,508 Trainable params: 263,508 Non-trainable parameters: 0		

3.3 Object classification

The next task is to create an efficient classifier to differentiate between objects based on the features obtained from Step 2, with the focus on all classes of VOG. Hence, a softmax classifier was created by:

$$Y = Softmax(W(ft)) \quad (3)$$

where, W is the weight matrix of dense layers in the prediction model; ft is the features obtained by the feature extractor. The weight matrix was trained by minimizing the cross-entropy loss between predicted Y' and actual labels of Y.

To train the classifier with the obtained eigenvectors, it is necessary to have two kinds of labeled data, namely, images with ground-truth labels and the coordinates of the corresponding bounding boxes. Therefore, any proposal tightly enclosing an object was considered a positive sample, while any proposal containing no part of an object was considered a negative sample. However, it is hard to label the proposals that overlap an object. To solve the problem, the intersection over union (IOU) overlap threshold value was introduced. It is the similarity between predicted and ground-truth boxes. If the threshold is greater than 0.5IoU, a region overlapping an object will be considered positive; otherwise, it will be considered negative.

4. EVALUATION AND RESULTS

The performance of our technique was evaluated on a popular and challenging dataset: Pascal VOC 2007 [47]. This database was selected because our technique intends to detect objects on both large scale and small scale. The 9,963 images in the dataset were divided into a training set of 2,501 images, a validation set of 2,510 images, and a test set of 4,952 images. Every image contains at least 20 objects, which meet the detection purpose of our technique.

The proposal quality was measured by the average best overlap (ABO) and the mean ABO (MABO). The latter is an intersection of the obtained bounding box area and the ground-truth bounding box area of the corresponding object class over their union:

$$IoU\ (box, gtruth) = \frac{area(box) \cap area\ (gtruth)}{area(box)\ \cup area\ (gtruth)} \quad (4)$$

The detection efficiency of our technique was evaluated by mean average precision (mAP), the primary quality criterion in object detection [48]. The mAP refers to the mean value of the average precision over all classes. In contrast, average precision (AP) stands for the precision over a class depending on the intersection of a proposal with the ground-truth over an area of their union (IoU) [49, 50].

In our experiments, the IoU was set to 0.5, the learning rate to 0.01, the batch size to 64 to control the size of training set, and the maximum number of iterations to 150 to regulate the number of complete passes through the training set. The iterative learning algorithm of stochastic gradient descent (SGD) was selected as the optimizer to update our model.

In addition, domain-specific finetuning was adopted to train our network. The ImageNet-specific 1,000-channel classification layer of VGGNet was replaced with a 64-channel random dense layer, followed by a 20-channel classification layer. The rest of the VGGNet architecture was kept unchanged. Then, the weights were learned for the two alternative layers. During the training, only the proposals with IoU overlap > 0.5 with ground-truth were selected from the 2,000 proposals. Further, an upper limit was imposed on the top 50 training samples, i.e., $IoU > 0.5$. During the training, the domain-specific finetuning lasted only 5h per session.

Finally, the testing took a total of 83s, including 15s for proposal generation, 66s for feature extraction, and 2s for object classification. Figures 6-9 display the qualitative results of our experiments.

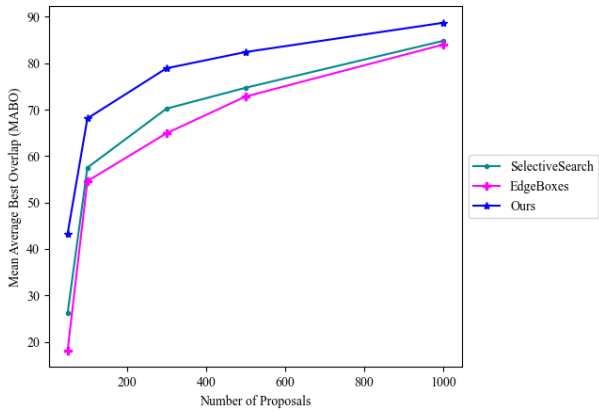

Figure 6. Variation of MABO with the number of proposals

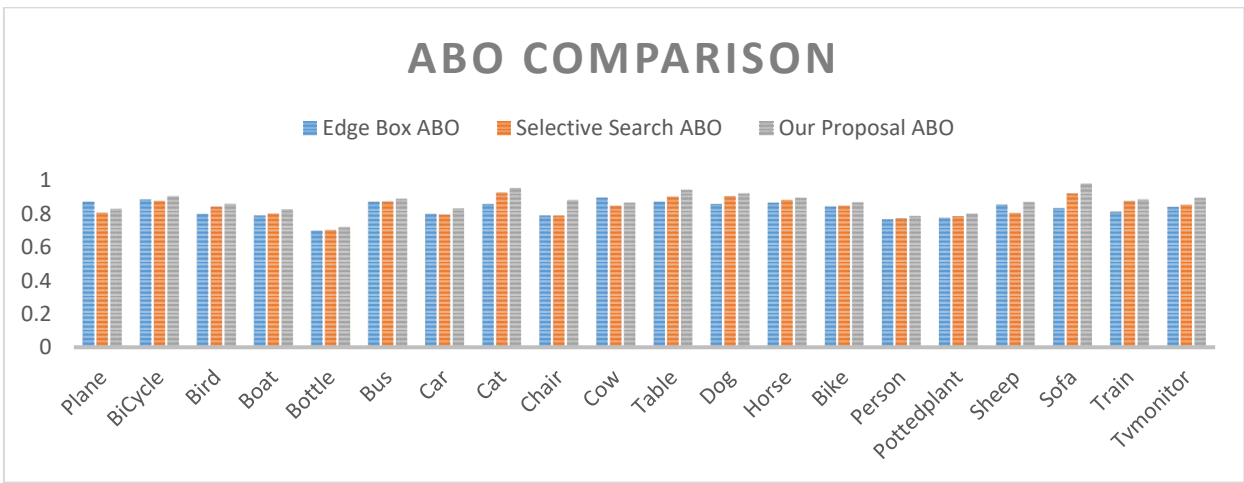

Figure 7. Comparison of ABOs for all Pascal VOC classes on top 1,000 proposals

Table 3. Comparison of proposal generation ability

Proposals Methods	Type	Segmentation	Scoring mechanism	Computational Time (sec)
Selective search [16]	Grouping	✓	×	10
CPMC [11]	Grouping	✓	✓	250
Endres' method [37]	Grouping	✓	✓	100
Rantalankila's method [12]	Grouping	✓	×	10
Objectness [14]	Window scoring	×	✓	3
Rahtu's method [13]	Window scoring	×	✓	3
Edge box [17]	Window scoring	×	✓	0.3
BING [15]	Window scoring	×	✓	0.2
Our technique	Grouping and window scoring	✓	✓	5

Table 3 compares the proposal generation abilities of our technique with multiple competitors. Unlike most competitors, our technique adopts both segmentation and scoring mechanism. The former provides insights into the image structure, guides the sampling process, and improves object localization, while the latter ensures that high-quality proposals are selected for post-classification. Our technique only consumed 5s for proposal generation, which is acceptable for most real-time applications. As shown in Table 3, most window scoring methods, namely, Objectness [40], Rahtu's method [41], and edge box [42] took less time to generate proposals than our technique, because they do not need to segment the images. However, the lack of segmentation results in low-quality proposals, and suppresses the overall detection performance. By combining segmentation and scoring mechanism, our technique can generate a few high-quality proposals, and consume a limited time.

Table 4. MABOs on Pascal VOC 2007

Methods	Test set	Number of proposals	MABO
Edge box	4,952	1,000	0.828
Selective search	4,952	1,000	0.840
Our technique	4,952	1,000	0.867

Table 5. ABOs for 20 classes of VOC on top 1,000 proposals

VOC Classes	Edge box's ABO	Selective search's ABO	Our technique's ABO
Plane	0.871	0.806	0.827
Bicycle	0.884	0.877	0.901
Bird	0.798	0.842	0.856
Boat	0.789	0.801	0.824
Bottle	0.699	0.703	0.718
Bus	0.871	0.874	0.886
Car	0.798	0.795	0.828
Cat	0.857	0.926	0.949
Chair	0.788	0.789	0.878
Cow	0.897	0.849	0.864
Table	0.871	0.901	0.941
Dog	0.857	0.905	0.92
Horse	0.865	0.883	0.893
Bike	0.842	0.849	0.866
Person	0.766	0.774	0.784
Potted plant	0.776	0.784	0.799
Sheep	0.854	0.804	0.867
Sofa	0.833	0.923	0.977
Train	0.811	0.876	0.883
TV monitor	0.841	0.853	0.892

Tables 4 and 5 compare the ABO and MABO of our technique with other baselines, respectively. As shown in Figure 7, our technique achieved a higher ABO for all Pascal VOC classes than the contrastive methods. The variation of MABO with the number of proposals (Figure 6) indicates that our technique had an MABO of 0.867 for 1,000 proposals, indicating that the proposals are good enough for post-classification task. Compared to other approaches, our technique can generate a few high-quality, class-independent proposals with a high recall.

Tables 6 and 7 present the overall classification performance of our technique on Pascal VOC 2007 dataset. The object detection effect of our technique was contrasted with that of existing techniques, which generate object proposals before classifying them with VGGNet. It can be observed that our technique achieved superior performance in object classification, with an mAP of 69.21% compared to other approaches. The results also illustrate that our technique can detect objects much more accurately than the other methods. By our technique, the high-quality proposals were generated with a higher ABO than that of any other traditional approach. The best performance in proposal generation undoubtedly contributes to classification accuracy. In addition, our technique performed the best on non-rigid classes like cat (86.09%), horse (83.15%), and dog (82.60%). The superiority of our method is also demonstrated by the variation of detection recall with IoU overlaps (Figure 8). In summary, our technique is both practical in proposal generation, and efficient in object classification.

Table 6. Overall detection results

Method	Number of boxes	mAP
Selective search (VGGNet)	2,000	68.19
Edge box (VGGNet)	2,000	67.01
Our technique	2,000	69.20

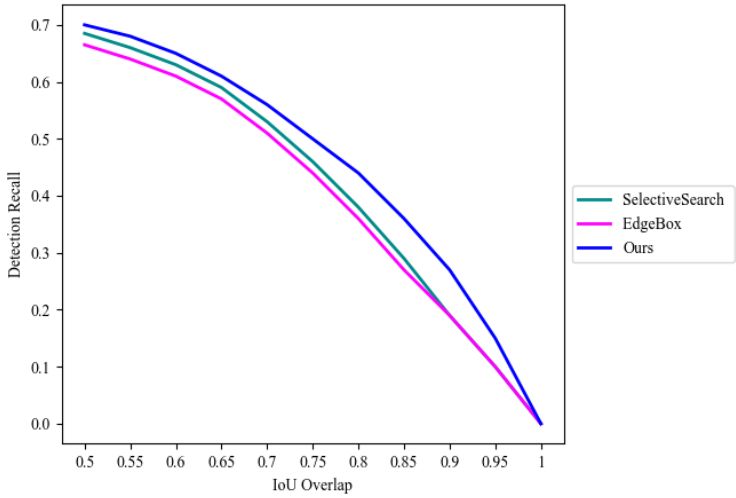

Figure 8. Variation in detection recall with IoU overlaps

Table 7. Comparison of detection performance on Pascal VOC 2007

VOC classes	Edge box's mAP (2,000)	Selective search's mAP (2,000)	Our technique's mAP (2,000)
Plane	74.31	74.79	77.07
Bicycle	78.54	78.68	79.61
Bird	69.32	69.82	69.91
Boat	53.28	51.96	53.78
Bottle	36.69	35.66	37.67
Bus	77.41	79.86	81.68
Car	78.78	79.63	79.08
Cat	82.17	85.06	86.09
Chair	40.83	42.79	42.80
Cow	72.27	75.29	75.30
Table	67.81	68.59	69.89
Dog	79.39	82.35	82.60
Horse	79.15	81.28	83.15
Bike	73.81	74.79	75.46
Person	69.55	69.17	69.96
Potted plant	30.46	30.74	31.75
Sheep	65.14	64.79	68.28
Sofa	70.28	74.60	74.90
Train	75.80	77.56	77.63
TV monitor	65.21	66.42	67.68

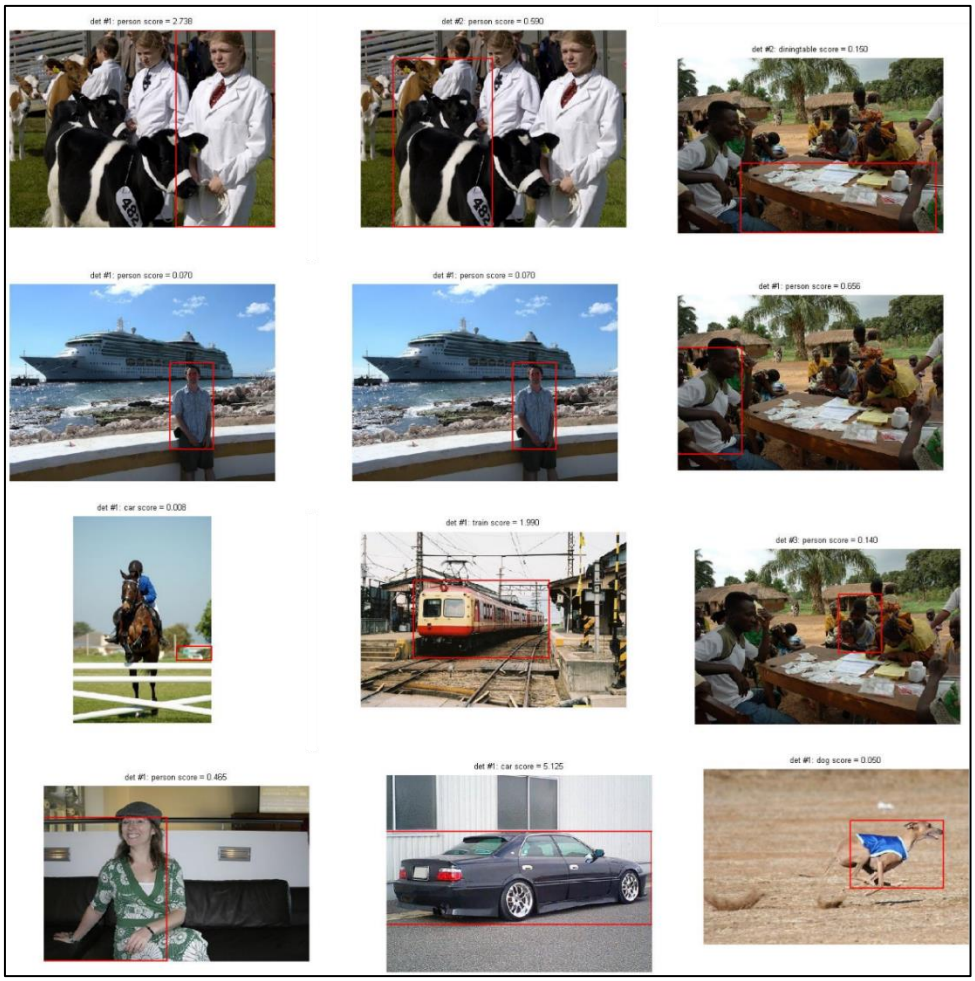

Figure 9. Object detection results of our technique

5. CONCLUSIONS

This paper mainly proposes an efficient object detection method. Firstly, high-quality class-independent object proposals were generated. Then, the eigenvector of each proposal was extracted by a DNN, and used to classify each object. Experimental results show that our technique generated high-quality class-independent proposals with the highest ABO, which promotes the performance of post-classification. This means the adoption of DNN in post-classification can boost classification effect. Our technique also achieved the best mAP of 69.21%, indicating that it is slightly better than conventional methods.

Of course, our technique still has several defects. First, the number of proposals was reduced to 2,000 locations per image, but an enormous amount of time is needed to classify these locations. Hence, our technique is expensive to implement for real-time applications. It takes about 83s to classify an object for a new test. Second, the proposal generation stage adopts a fixed algorithm. The absence of learning might create lousy candidate proposals.

Despite these defects, our technique remains an efficient tool of object detection. It provides a useful solution to a broad range of applications, where object detection is the top priority. The possible fields of application include transport, security, robotics, retrieval, consumer electronics, and human-computer interaction.

In the future, deep learning and fractional calculus will be introduced to improve the robustness in both stages. Proposal generation can be improved with deep learning features and fractional calculus. The two techniques help to remove unnecessary proposals, and further increase the efficiency of post-classification, making object classification more robust.

ACKNOWLEDGMENT

The work is supported by "Hundreds of Schools Unite with Hundreds of Counties-University Serving Rural Revitalization Science and Technology Support Action Plan" (Grant No.: BXLBX0847), and “Hubei Self Science Fund Project (Grant Name. Brain tumor diagnosis based on capsule neural network)”, and “National Statistical Science Research Project in 2020, China (Grant No.: 2020LY023)”.

REFERENCES

[1] Fink, M., Liu, Y., Engstle, A., Schneider, S.A. (2019). Deep learning-based multi-scale multi-object detection and classification for autonomous driving. In Fahrerassistenzsysteme, 233-242. https://doi.org/10.1007/978-3-658-23751-6_20

[2] Levine, M., De Silva, T., Ketcha, M.D., Vijayan, R., Doerr, S., Uneri, A., Siewerdsen, J.H. (2019). Automatic vertebrae localization in spine CT: A deep-learning

approach for image guidance and surgical data science. In Medical Imaging 2019: Image-Guided Procedures, Robotic Interventions, and Modeling, 10951: 109510S. https://doi.org/10.1117/12.2513915

[3] Espinosa, J.E., Velastin, S.A., Branch, J.W. (2019). Detection and tracking of motorcycles in congested urban environments using deep learning and Markov decision processes. In Mexican Conference on Pattern Recognition, 11524: 139-148. https://doi.org/10.1007/978-3-030-21077-9_13

[4] Yao, H., Zhang, S., Hong, R., Zhang, Y., Xu, C., Tian, Q. (2019). Deep representation learning with part loss for person re-identification. IEEE Transactions on Image Processing, 28(6): 2860-2871. https://doi.org/10.1109/TIP.2019.2891888

[5] Pereira, T.D., Aldarondo, D.E., Willmore, L., Kislin, M., Wang, S.S.H., Murthy, M., Shaevitz, J.W. (2019). Fast animal pose estimation using deep neural networks. Nature Methods, 16(1): 117-125. https://doi.org/10.1038/s41592-018-0234-5

[6] Deng, L., Gong, Y., Lin, Y., Shuai, J., Tu, X., Zhang, Y., Xie, M. (2019). Detecting multi-oriented text with corner-based region proposals. Neurocomputing, 334, 134-142. https://doi.org/10.1016/j.neucom.2019.01.013

[7] Diaconu, C., Freedman, C.S., Larson, P.A., Zwilling, M.J. (2016). U.S. Patent No. US9,251,214. Washington, DC: U.S. Patent and Trademark Office.

[8] Gite, S., Agrawal, H. (2019). Early prediction of driver's action using deep neural networks. International Journal of Information Retrieval Research (IJIRR), 9(2): 11-27. https://doi.org/10.4018/IJIRR.2019040102

[9] Rahman, Z., Pu, Y.F., Aamir, M., Ullah, F. (2019). A framework for fast automatic image cropping based on deep saliency map detection and gaussian filter. International Journal of Computers and Applications, 41(3): 207-217. https://doi.org/10.1080/1206212X.2017.1422358

[10] Fan, H., Chu, P., Latecki, L.J., Ling, H. (2019). Scene parsing via dense recurrent neural networks with attentional selection. In 2019 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 1816-1825). https://doi.org/10.1109/WACV.2019.00198

[11] Carreira, J., Sminchisescu, C. (2011). CPMC: Automatic object segmentation using constrained parametric min-cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(7): 1312-1328. https://doi.org/10.1109/TPAMI.2011.231

[12] Rantalankila, P., Kannala, J., Rahtu, E. (2014). Generating object segmentation proposals using global and local search. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2417-2424.

[13] Rahtu, E., Kannala, J., Blaschko, M. (2011). Learning a category independent object detection cascade. In 2011 International Conference on Computer Vision, pp. 1052-1059. https://doi.org/10.1109/ICCV.2011.6126351

[14] Alexe, B., Deselaers, T., Ferrari, V. (2012). Measuring the objectness of image windows. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(11): 2189-2202. https://doi.org/10.1109/TPAMI.2012.28

[15] Cheng, M.M., Liu, Y., Lin, W.Y., Zhang, Z., Rosin, P.L., Torr, P.H. (2019). BING: Binarized normed gradients for objectness estimation at 300fps. Computational Visual Media, 5(1): 3-20. https://doi.org/10.1007/s41095-018-0120-1

[16] Uijlings, J.R., Van De Sande, K.E., Gevers, T., Smeulders, A.W. (2013). Selective search for object recognition. International Journal of Computer Vision, 104(2): 154-171. https://doi.org/10.1007/s11263-013-0620-5

[17] Zitnick, C.L., Dollár, P. (2014). Edge boxes: Locating object proposals from edges. In European Conference on Computer Vision, 8693: 391-405. https://doi.org/10.1007/978-3-319-10602-1_26

[18] Endres, I., Hoiem, D. (2013). Category-independent object proposals with diverse ranking. IEEE Transactions on Pattern Analysis and Machine Intelligence, 36(2): 222-234. https://doi.org/10.1109/TPAMI.2013.122

[19] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25: 1097-1105.

[20] Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[21] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-9.

[22] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778.

[23] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q. (2017). Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700-4708.

[24] Girshick, R., Donahue, J., Darrell, T., Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 580-587.

[25] Girshick, R. (2015). Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, pp. 1440-1448.

[26] Ren, S., He, K., Girshick, R., Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in neural Information Processing Systems, 28: 91-99.

[27] Dai, J., Li, Y., He, K., Sun, J. (2016). R-FCN: Object detection via region-based fully convolutional networks. In Advances in Neural Information Processing Systems, pp. 379-387.

[28] He, K., Gkioxari, G., Dollár, P., Girshick, R. (2017). Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, pp. 2961-2969.

[29] Redmon, J., Divvala, S., Girshick, R., Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779-788.

[30] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C. (2016). Ssd: Single shot multibox detector. In European Conference on Computer Vision, pp. 21-37. https://doi.org/10.1007/978-3-319-46448-0_2

[31] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The Journal of

Machine Learning Research, 15(1): 1929-1958.
[32] Ioffe, S., Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International Conference on Machine Learning, pp. 448-456.
[33] Felzenszwalb, P.F., Huttenlocher, D.P. (2004). Efficient graph-based image segmentation. International Journal of Computer Vision, 59(2): 167-181. https://doi.org/10.1023/B:VISI.0000022288.19776.77
[34] Albawi, S., Mohammed, T.A., Al-Zawi, S. (2017). Understanding of a convolutional neural network. In 2017 International Conference on Engineering and Technology (ICET), pp. 1-6. https://doi.org/10.1109/ICEngTechnol.2017.8308186
[35] LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., Jackel, L. (1989). Handwritten digit recognition with a back-propagation network. Proceedings of the 2nd International Conference on Neural Information Processing Systems, pp. 396-404.
[36] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11): 2278-2324. https://doi.org/10.1109/5.726791
[37] Gu, J., Wang, Z., Kuen, J et al. (2018). Recent advances in convolutional neural networks. Pattern Recognition, 77: 354-377. https://doi.org/10.1016/j.patcog.2017.10.013
[38] Hecht-Nielsen, R. (1992). Theory of the backpropagation neural network. In Neural Networks for Perception, pp. 65-93. https://doi.org/10.1016/B978-0-12-741252-8.50010-8
[39] Zeiler, M.D., Fergus, R. (2014). Visualizing and understanding convolutional networks. In European Conference on Computer Vision, pp. 818-833. https://doi.org/10.1007/978-3-319-10590-1_53
[40] Lin, M., Chen, Q., Yan, S. (2013). Network in network. arXiv preprint arXiv:1312.4400.
[41] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-9.
[42] Erhan, D., Szegedy, C., Toshev, A., Anguelov, D. (2014). Scalable object detection using deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2147-2154.
[43] Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y. (2017). Deformable convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, pp. 764-773.
[44] Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K. (2016). SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. arXiv preprint arXiv:1602.07360.
[45] Gehring, J., Auli, M., Grangier, D., Yarats, D., Dauphin, Y.N. (2017). Convolutional sequence to sequence learning. In International Conference on Machine Learning, 70: 1243-1252.
[46] Bansal, A., Chen, X., Russell, B., Gupta, A., Ramanan, D. (2017). Pixelnet: Representation of the pixels, by the pixels, and for the pixels. arXiv preprint arXiv:1702.06506.
[47] Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A. (2015). The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision, 111(1): 98-136. https://doi.org/10.1007/s11263-014-0733-5
[48] Aamir, M., Pu, Y.F., Rahman, Z., Abro, W.A., Naeem, H., Ullah, F., Badr, A.M. (2018). A hybrid proposed framework for object detection and classification. Journal of Information Processing Systems, 14(5): 1176-1194. https://doi.org/10.3745/JIPS.02.0095
[49] Aamir, M., Pu, Y.F., Abro, W.A., Naeem, H., Rahman, Z. (2017). A hybrid approach for object proposal generation. In International Conference on Sensing and Imaging, 506: 251-259. https://doi.org/10.1007/978-3-319-91659-0_18.
[50] Guan, Y., Aamir, M., Hu, Z., Abro, W.A., Rahman, Z., Dayo, Z.A., Akram, S. (2021). A region-based efficient network for accurate object detection. Traitement du Signal, 38(2): 481-494. https://doi.org/10.18280/ts.380228