Hierarchical Image Object Search Based on Deep Reinforcement Learning

Wei Zhang
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: weivanity@ gmail.com

Yuxing Tan
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: 842061340@qq.com

Hongge Yao
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, China
E-mail: 835092445@qq.com

***Abstract*—Object detection technology occupies a pivotal position in the field of modern computer vision research，its purpose is to accurately locate the object human beings are looking for in the image and classify the object. With the development of deep learning technology, convolutional neural networks are widely used because of their outstanding performance in feature extraction, which greatly improves the speed and accuracy of object detection. In recent years, reinforcement learning technology has emerged in the field of artificial intelligence, showing excellent decision-making ability to deal with problems. In order to combine the perception ability of deep learning technology with the decision-making ability of reinforcement learning technology, this paper incorporate reinforcement learning into the convolutional neural network, and propose a hierarchical deep reinforcement learning object detection model.**

Keywords-Object Detection; Deep Learning; Reinforcement Learning

I. INTRODUCTION

When observing a picture, humans can immediately know the location and category of the object in the image, and can get the information without even thinking too much. This is a breeze for us, but the computer cannot have all kinds of complicated ideas of our human brain, and it is not easy to realize it.

In computer vision, the positioning and retrieval of images will be affected by two aspects, one is the content of the image, and the other is the pros and cons of the algorithm. There are two main factors influencing the image. The first is that the background and light when taking pictures will affect the quality of the image, resulting in a decrease in the accuracy of object detection. The second is the content of the image. If there are several similar objects, or some are

DOI: 10.21307/ijanmc-2020-029

blocked by other objects, and the different angles of the object will affect the accuracy of detection. The algorithm mainly focuses on how to make the features have higher quality. Therefore, how to design an algorithm that can satisfy accurate positioning and continuously improve the object positioning speed is the key to research.

For computers, these pictures are data collections which are composed of binary digits, and the things behind the data cannot be imagined by computers. Our purpose is to let the computer simulate our human vision and simply have the ability to process the image. Human beings get a lot of information in real life every day, and most of them belongs to the information transmitted to us by vision, and only part of the information in these visual images is what human need. Therefore, by extracting the important information, positioning and identifying them accurately, human can greatly reduce the amount of data that the computer needs to process and improve the efficiency of data processing.

Reinforcement learning is an important field in machine learning. It constructs a Markov Decision Process and simulates human thinking to teach agents how to make actions that can obtain high reward values in the environment, and find the best strategy to solve the problem in such constant interaction. Based on this idea, this paper use reinforcement learning technology to simulate the human visual attention mechanism. The agent is taught to change the shape of the bounding box and focus only on a significant part of the image at a time, and then extract its features through the convolutional neural network. Finally, the object of image positioning and classification can be achieved.

II. RELATED WORK

A. Traditional object detection algorithm

Traditional object detection algorithms include primary feature extraction methods such as HOG feature extraction of objects and training SVM classifiers for recognition. Their algorithms are generally divided into three stages (see Figure 1.):

1) Select different sliding window frames according to the size of the object, and use the sliding window to select a part of the content in the figure as a candidate area.

2) Extract visual features from candidate regions.

3) Use SVM classifier for identification.

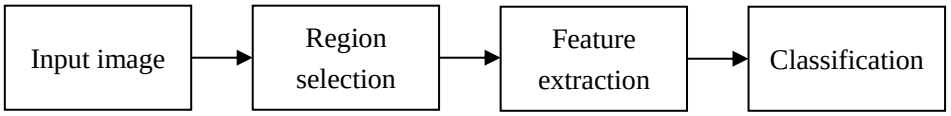

Figure 1. Traditional object detection algorithm

The traditional object search algorithm has the following disadvantages:

1) The selection strategy based on sliding windows is to slide across the entire image from beginning to end. For different object sizes, the program need windows with different size ratios to traverse. Although it can mark all the positions of the object, its brute-force enumeration search results in extremely high time complexity and a large number of windows that are not related to the object, so the speed and performance of feature extraction and classification have fallen into a bottleneck.

2) The characteristics of each object are different, which leads to the diversity of forms, and the background factors of each object will also affect the accuracy of recognition. Therefore, the features of manual design are not very robust.

B. Object detection algorithm based on deep learning

After the appearance of CNN, it has been widely used in the field of computer vision. With the continuous development of science and technology, the difficulty of obtaining a large amount of sample data has been significantly reduced, and the continuous improvement of computing capabilities has enabled CNN to have the ability to extract features from a large amount of data, which has made huge gains in computer vision.

Aiming at the shortcomings of traditional method for object detection, the object detection algorithm based on deep learning uses CPMC, Selective Search, MCG, RPN and other methods to generate candidate regions instead of window sliding strategy. These methods usually use various details of the image, such as image contrast, edge parts and color to extract higher-quality candidate regions, while reducing the number of candidate regions and time complexity.

This type of object detection method is generally divided into two types: one-stage detection algorithm and two-stage detection algorithm. The one-stage detection algorithm regards the object detection problem as a regression problem and directly obtains the category and position information of the object. The detection speed of the algorithm is fast, but the accuracy is low. The two-stage detection algorithm first generates a large number of region proposals, and then classifies these region proposals through the convolutional neural network, so the accuracy is higher, but the detection speed is slower.

C. Object detection algorithm based on deep reinforcement learning

In recent years, research on deep reinforcement learning has emerged endlessly. It has achieved excellent performance in many games than human master players, especially the success of the DeepMind team on the AlphaGO project, pushing deep reinforcement learning to a new height. In this context, many researchers try to apply deep reinforcement learning technology in the field of object detection.

In 2015, Caicedo et al. adopted a top-down search strategy, analyzed the entire scene at the beginning, and then continued to move toward the object location. That is, use a larger bounding box to frame the object, and then shrink it step by step, eventually making the object surrounded by a compact bounding box. In 2016, Mathe et al. proposed an image-based sequence search model to extract image features from a small number of pre-selected image positions in order to efficiently search for visual objects. By formulating sequential search as reinforcement learning of the search policy, their fully trainable model can explicitly balance for each class, specifically, the conflicting goals of exploration - sampling more image regions for better accuracy -, and exploitation - stopping the search efficiently when sufficiently confident about the object's location.

The above algorithm models all use reinforcement learning techniques to improve deep learning algorithms, and all have achieved good results. However, if the visual object algorithm is required to have a relatively high accuracy, it still needs to rely on a large number of candidate regions, so our research direction is to reduce the number of candidate regions while maintaining the quality of the candidate regions at a high level.

III. HIERARCHICAL OBJECT SEARCH MODEL BASED ON DRL

A. MDP formulation

This paper regard object detection as a Markov Decision Process, and find an effective object detection strategy by solving decision problems. In each process, the agent interacts with the current environment based on the current state, and decides the next search action, and gets an instant reward value. The agent continuously improves the efficiency of search in the

process of learning to obtain high cumulative reward value.

There are 6 different actions in action space A, which are composed of two different types: select action and stop action. The selection action is to frame a part of the current area as the next observation area. It consists of four borders and a center frame, which respectively reduce the current search area to different sub-regions (see Figure 2.); the stop action indicates that the object has been found, so the bounding box is no longer changed, and the search process stops.

In reinforcement learning, the state is the premise and basis for the agent to make actions. In this model, the state is composed of two aspects. One is the feature vector extracted by the convolutional neural network in the current state. The other is the historical action information performed in the process of searching for the object. This information helps to stabilize the search trajectory, so that the search process will not fall into the loop search, thereby improving the accuracy of the search.

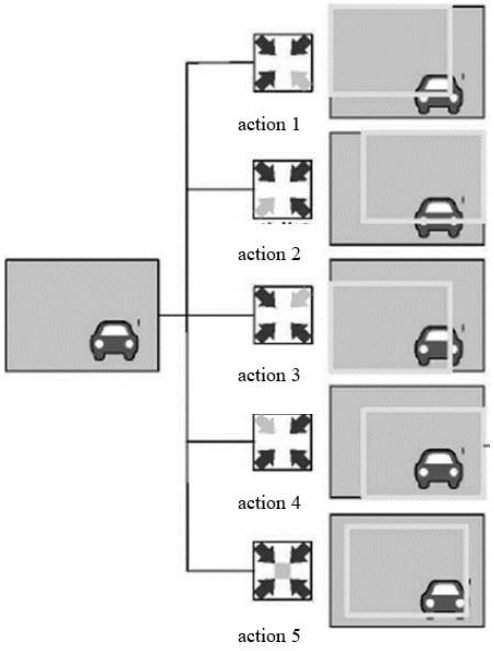

Figure 2. The selection action diagram

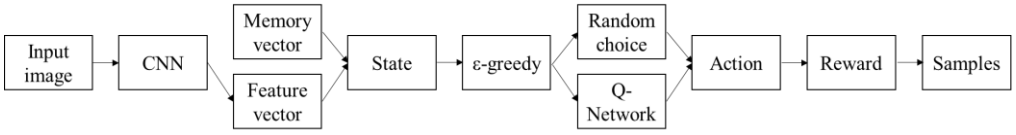

Figure 3. Sample generation process diagram based on Markov decision process

The function of the reward is to reflect the reaction obtained by the agent during the interaction with the environment. The agent judges the merits of the action according to the different rewards received, and finally learns the strategy by maximizing the cumulative reward. Since the agent have two types of actions, our calculation methods are different depending on the type of action.

The reward obtained by the agent depends on the action it takes in the current state. The model use IoU to evaluate the effect of the action, so that the accuracy of detection can be obtained. This paper take e' as the observation area after the movement, e as the observation area before the movement, and g as the area of the object area, R_m is the reward obtained after making the selection action, then our reward function can be expressed for:

$$R_m = sign(IoU(e',g) - IoU(e,g)) \quad (1)$$

If the difference of the overlap rate is positive, it means that our prediction range is closer to the object

area, if it is negative, it means that the prediction range is farther from the object area. If the decision improves the detection accuracy, the reward is positive, otherwise the reward is negative.

The model use R_n as the reward function for the stop action, and set the reward value for the stop action as δ. At the same time, the program need to add a threshold η to determine when it will end the action. When the value of IoU is greater than the threshold, indicating that object has been found, then the program can end the search and perform the stop action. At the same time as there is a reward process, there is also a punishment when IoU continues to fall below the threshold and reaches the maximum number of searches. So that the agent knows the wrong process and corrects it. The reward function for the stop action is as follows:

$$R_n = \begin{cases} +\delta, if\ IoU(e,g) \geq \eta \\ -\delta, if\ IoU(e,g) < \eta \end{cases} \quad (2)$$

B. DQN algorithm

The model use three fully connected layers to form the Q-network, its input is the information content of the image, and the activation values of the 6 neurons in the output layer represent the confidence of 6 kinds of actions, among which the highest confidence corresponding action is selected.

According to the state, action and reward function, the agent apply the Q-learning algorithm to learn the optimal strategy. Because the input image is a high-dimensional data, this paper use DQN to approximate the Q function $Q(s,a)$ in high dimensions. The Q function based on strategy π is expressed as follows:

$$Q_\pi(s,a) = E[R|s_i = s, a_i = a, \pi] \quad (3)$$

The agent selects the action with the highest Q value from the Q function, and uses the Bellman equation to continuously update the Q function:

$$Q(s,a) = r + \gamma \max Q(s',a') \quad (4)$$

Among them, s is the current state, a is the selected action in the current state, r is the immediate reward, γ is the discount factor, s' indicates the next state, and a' indicates the next action to be taken.

In order to train DQN, the program need a large number of training samples which are usually continuously sampled (see Figure 3.), but the continuity between adjacent samples will cause inefficiency and instability of Q-network learning. This paper use the experience-replay mechanism to solve this problem. When the capacity of the experience pool tends to be saturated, the program constantly replace the old samples with new samples. At the same time, in order to make most of the samples selected with nearly the same probability, the program randomly extract samples in the experience pool.

The loss function of the training process is set as follows:

$$L(w) = (r + \gamma \max Q(s',a',w) - Q(s,a,w))^2 \quad (5)$$

Among them, $Q(s,a,w)$ is the actual output of the network, $r + \gamma \max Q(s',a',w)$ is the expected output of the network, r is the current reward value, $\max Q(s',a',w)$ is the maximum expected reward value for the next decision, γ is the discount factor.

C. Hierarchical object search process

The initial candidate region of the model is the entire image. The size of the candidate region is normalized to a fixed size, and then put into a trained

CNN neural network model to extract feature values, and then rely on the greedy algorithm to use the probability ε randomly select one of all actions to search, or use the learned strategy to make action decisions with a probability of $1-\varepsilon$.

After the model made action a, it switched to a new candidate area e' which is a sub-region of the previous region, according to the reward function to give our agent the corresponding reward R_m, and at the same time normalize the new candidate area and put it into the neural network model for features extraction, combine with previous actions to get a new state s'. Repeat the above hierarchical process continuously until our action becomes a stop action, or the number of search steps reaches the upper limit. If a stop action occurs, the final reward R_n is given according to its corresponding termination reward function.

IV. EXPERIMENT

A. *Data sets and parameter settings*

Use the Pascal VOC data set to train the model, which is the most used data set for object detection. The training set uses the combination of Pascal VOC 2007 and Pascal VOC 2012, and the test set uses the Pascal VOC 2007 Test Set.

The model use three fully connected layers to form a Q-network. Its input is the information content of the image. The activation values of 6 neurons in the output layer represent the confidence of 6 actions. The parameters of the network are initialized by a standard normal distribution function. The initial value of the greedy factor ε is 1, every iteration, the ε decreases by 0.1, and stops when it decreases to 0.1. Set the size of the experience pool to 1000, the reward discount coefficient γ to 0.9, and the threshold to make the stop action is 0.5.

B. *Experimental results and analysis*

- Model training

In the process of training the model, the value of the loss function is continually declining along with the continuous iteration of the neural network, making the neural network tend to converge (see Figure 4.). When the number of training times reaches a certain level, the loss value tends to be stable, and various parameters in the network are also updated, forming a neural network model with recognition capabilities.

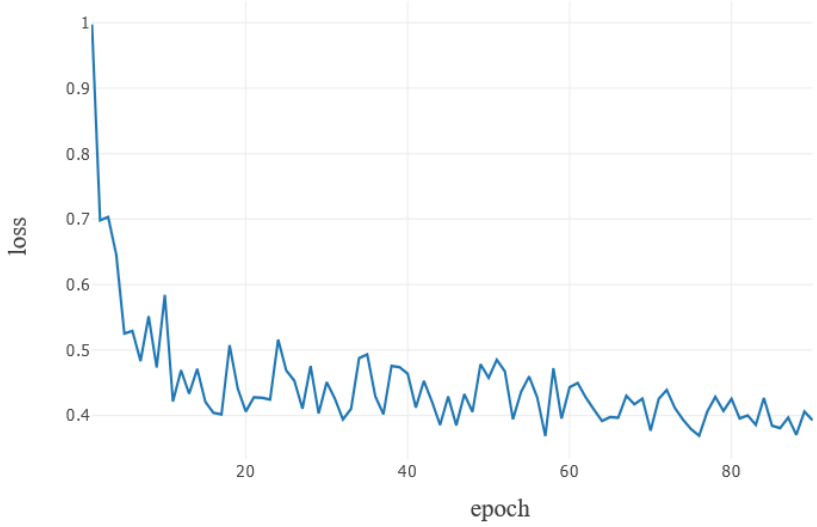

Figure 4. Schematic diagram of loss function

- Results and analysis

The model first analyzes the entire picture and finds the object through a series of frame transformation actions. Finally, the agent make the stop action indicate the end of the search. The following figure shows this hierarchical dynamic selection process in detail.

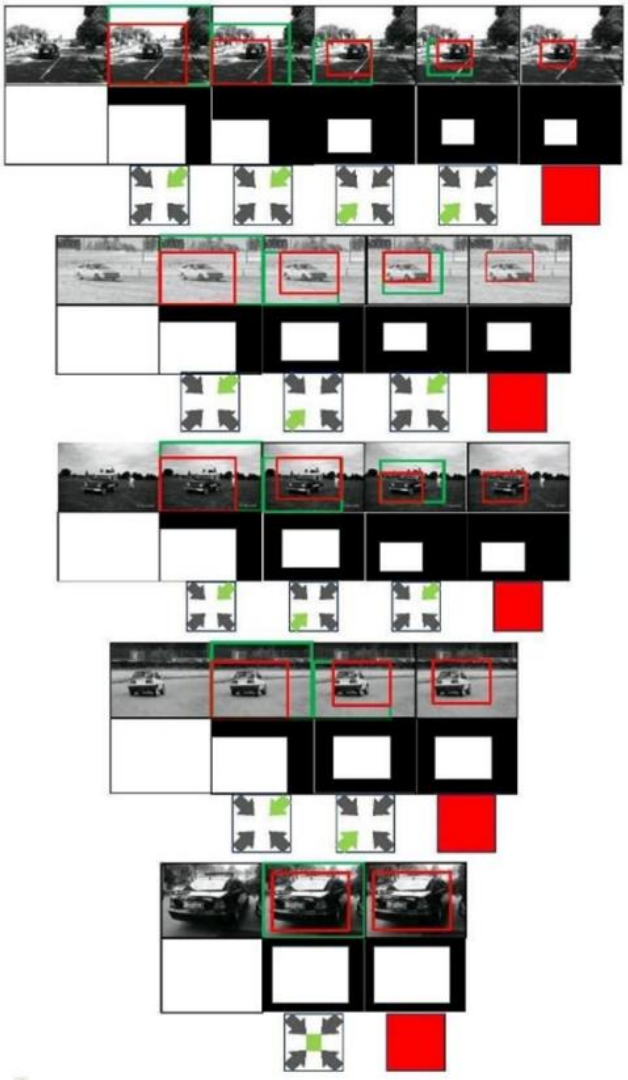

Figure 5. Hierarchical dynamic selection process

Experimental results show that the algorithm model proposed in this paper can improve the search speed and accuracy in object search. However, it can also be seen from the experiment that there may still be errors in the match between the object prediction frame and the actual bounding box of the object, because the model can only continue to select from the area selected by the previous bounding box. As a result, the predicted bounding box cannot reach other areas of the image. The model can improve the detection result by changing the appropriate proportion of the framed area.

V. CONCLUSION

This paper propose an object detection model based on deep reinforcement learning, which focuses on

different areas of the picture by performing a predefined area selection action, and iterates the process to make the bounding box tightly surround the object, Finally achieved the positioning and classification of object. Experiments show that the model can effectively detect the object in the image.

REFERENCES

[1] Sutton R S, Barto A G. Reinforcement learning: An introduction[M]. MIT press, 2018.

[2] Gagniuc P A. Markov chains: from theory to implementation and experimentation[M]. John Wiley & Sons, 2017.

[3] Hu Y, Xie X, Ma W Y, et al. Salient region detection using weighted feature maps based on the human visual attention model[C]//Pacific-Rim Conference on Multimedia. Springer, Berlin, Heidelberg, 2004: 993-1000.

[4] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.

[5] LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.

[6] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). IEEE, 2005, 1: 886-893.

[7] Boser B E, Guyon I M, Vapnik V N. A training algorithm for optimal margin classifiers[C]//Proceedings of the fifth annual workshop on Computational learning theory. 1992: 144-152.

[8] Papandreou G, Kokkinos I, Savalle P A. Modeling local and global deformations in deep learning: Epitomic convolution, multiple instance learning, and sliding window detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 390-399.

[9] Carreira J, Sminchisescu C. CPMC: Automatic object segmentation using constrained parametric min-cuts[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 34(7): 1312-1328.

[10] Uijlings J R R, Van De Sande K E A, Gevers T, et al. Selective search for object recognition[J]. International journal of computer vision, 2013, 104(2): 154-171.

[11] Pont-Tuset J, Arbelaez P, Barron J T, et al. Multiscale combinatorial grouping for image segmentation and object proposal generation[J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 39(1): 128-140.

[12] Silver D, Huang A, Maddison C J, et al. Mastering the game of Go with deep neural networks and tree search[J]. nature, 2016, 529(7587): 484-489.

[13] Caicedo J C, Lazebnik S. Active object localization with deep reinforcement learning[C]//Proceedings of the IEEE international conference on computer vision. 2015: 2488-2496.

[14] Mathe S, Pirinen A, Sminchisescu C. Reinforcement learning for visual object detection[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2894-2902.

[15] Watkins C J C H, Dayan P. Q-learning[J]. Machine learning, 1992, 8(3-4): 279-292.

[16] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[J]. arXiv preprint arXiv:1312.5602, 2013.