Journal of Applied Data Sciences
Vol. 3, No. 4, December 2022, pp. 158-162

Research on Saliency Detection Method Based on Depth and Width Neural Network

Guanqi He; Guo Lu *

School of Marxism Central China Normal University, Wuhan, China
hgq12354@ccnu.edu.cn
* corresponding author

(Received: August 30, 2022 Revised: October 15, 2022 Accepted: November 15, 2022, Available online: December 23, 2022)

Abstract

Image saliency detection is to segment the most important areas in the image. Solving the problem of image saliency detection usually involves knowledge in computer vision, neuroscience, cognitive psychology and other fields. In recent years, as deep learning has made great achievements in the field of computer vision, the application of deep learning has also played a good role in image saliency detection. Therefore, algorithms based on deep convolutional neural networks have become solutions to image saliency The most effective method of detection. Such a lot of information not only enriches people's lives, but also provides efficient and accurate network management platforms. The management of this image information brings difficulties. Therefore, how to understand and process these image information more intelligently and efficiently has become a hot topic for many image processing and computer vision researchers. Among them, saliency detection technology plays a key role in solving the problem of intelligent understanding and processing of images. To put it simply, saliency detection is a technology to automatically calculate or detect the most important areas in an image, and its processing results provide a basis for understanding and processing the image content. Saliency detection is a basic problem in computer vision, neuroscience and visual perception. The algorithm detects and extracts the most interesting or significant areas in the image.

Keywords: Convolutional Network; Low-Dimensional Feature Extraction; Saliency Detection, Broad Neural Network; Conditional Random Field

1. Introduction

This chapter will introduce the super pixel segmentation algorithm, manual feature extraction, deep neural network, width neural network, and post-processing algorithm conditional random field in detail. Super pixel segmentation algorithm is an algorithm that gathers features similar pixels in an image into regions [1-3]. One of the representative studies is the SLIC (simple linear iterative clustering) algorithm proposed by Achanta et al [4]. The characteristic of the SLIC algorithm is clustering based on the five-dimensional features of the color (R, G, B) and coordinate position (x, y) of the pixels in the image. This clustering method is similar to the K-means algorithm, and the final clusters are superpixels [5]. The difference is that the search space of SLIC algorithm is limited when clustering, instead of searching the entire picture for each superpixel. As shown, Figure 2-1(a) shows the search space of the K-means algorithm.

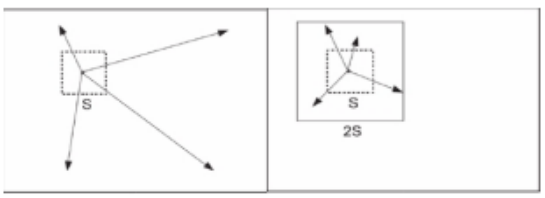

Figure. 1. (a) Standard K-means search; (b) SLIC algorithm search

Figure (b) is the search space of SLIC. Specifically, for a superpixel with a size of S×S, the search space is 2S×2S, so the SLIC algorithm runs more efficiently. The general process of the SLIC algorithm is as follows:

1. Divide the image into K patches of equal size and S×S, each patch is a cluster, and the center of the cluster is called a superpixel. Assuming that the image has N pixels, the side length of each image block is S = $S = \sqrt{\frac{N}{K}}$.
2. Initialize the cluster centers. After dividing the image blocks, select the point with the smallest gradient in the 3×3 area of the center point of each image block as the cluster center point. This prevents the sampled pixels from being noise or edge pixels.
3. After initializing the image cluster centers, for each cluster center, the pixels in the area of 2S×2S are clustered according to their distance from the cluster center. Each image pixel selects the cluster center closest to it as the superpixel to which it belongs. The formula for measuring the distance between pixels is:

$$D = \sqrt{\left(\frac{d_c}{m}\right)^2 + \left(\frac{d_c}{m}\right)^2}$$

$$d_c = \sqrt{\left(r_i - r_j\right)^2 + \left(g_i - g_j\right)^2 + \left(b_i - b_j\right)^2}$$

$$d_s = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2}$$

Before neural networks became mainstream, image researchers generally extracted hand-craft features for processing. The common method is to extract the bottom features of the image, such as color information features, spatial information features, or gradient features [6]. It is more complicated to perform histogram statistics based on the underlying features. Due to the robustness of histogram features, this method was widely used in image classification, saliency detection, and semantic segmentation in the early years [7].

Each pixel of an image stores the additive color mixture of multiple channels, usually a three dimensional vector, where each component of the pixel represents its color channel R (red), G (green), B (Blue) The intensity of the emitted light [8]. In addition, in image processing, image researchers often convert RGB images into images in other color spaces such as Lab (brightness, green, blue) color space, HSV (hue, saturation, intensity) color space and YUV (brightness, chroma, saturation) color space to make up for the lack of RGB in image processing.

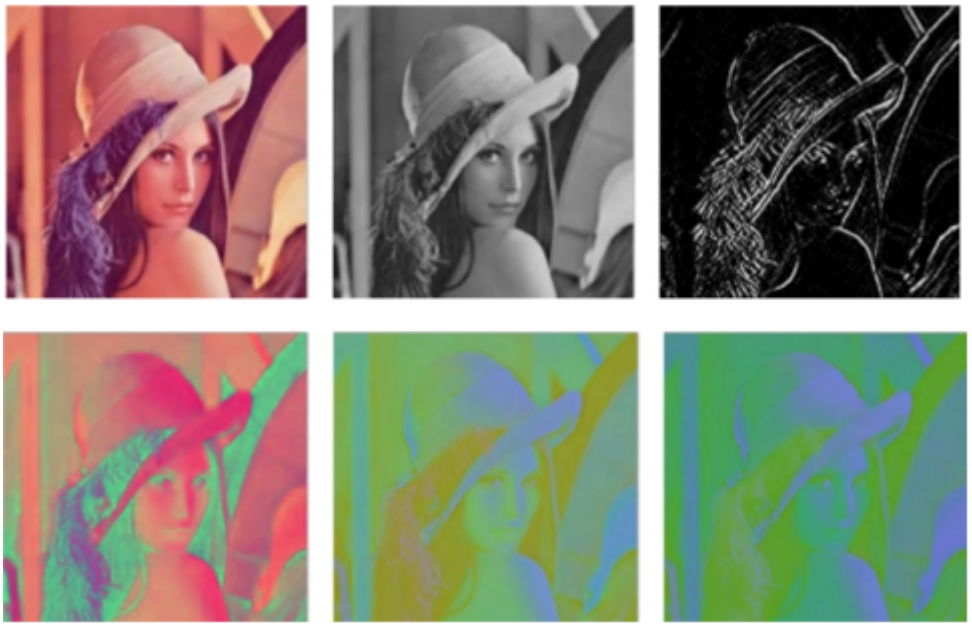

Figure. 2. YUV RGB in image processing

Image gradient is used in image processing to describe how fast the color or brightness of an area changes [9]. The smooth part of the color or brightness in the image is generally located on the surface of the object, and the sharp part

is generally the edge of the object. Therefore, the image researcher can infer the contour or other semantic information of the image based on the gradient value of the image. The figure is the RGB image, gray image, direction image, HSV image, YUV image and Lab effect image of the corresponding image. Through the different display of these 6 color spaces, it can be seen that the characteristics of its performance are more abundant.

2. Deep Convolution Algorithm Fusing Low-Dimensional Feature Extraction Layers

Saliency detection is a basic problem in computer vision, neuroscience and visual perception. In many studies, such as stereo matching, video deblurring, image and video compression, semantic segmentation or visual tracking, saliency detection is used as a preprocessing process [10]. With the rapid growth of image and video data on the Internet and the increasing demand for these media to effectively handle salient targets, it will be a very meaningful work to propose an accurate and efficient saliency detection algorithm.

The methods of saliency detection can be roughly divided into traditional saliency detection algorithms and saliency detection algorithms based on deep convolutional networks. Traditional saliency detection algorithms are divided into bottom-up saliency algorithms and top-down saliency detection algorithms [11]. In the bottom-up method, this type of method usually uses low-level cues such as color, center contrast, foreground prior and background prior to reason about saliency. Subjectively speaking, these characteristics are more directly perceivable by humans. However, not all objects can be distinguished well with these sufficient features [12]. When the feature of a salient target is not obvious enough or contains more semantic information, it is more necessary to use the high-level information of the data to detect the target. In the top-down approach, this type of research uses supervised methods to learn the extracted features to reason about saliency. For example, the literature uses random forest to learn predictive features, and then to calculate the saliency map, or establish a probability model with parameter learning for saliency detection. To some extent, these methods extend the generalization capabilities of saliency detection. However, it is also limited by the uniqueness of the extracted features and the accuracy of the learning model. In the research of saliency detection based on deep convolutional neural network (deep convolutional neural network, DCNN, early researchers used deep convolutional network to identify image local information or global information. The updated research is based on an end-to-end A full-volume deep neural network at the end to perform saliency detection. The method based on deep convolutional neural network performs well in deep feature extraction and prediction and has achieved very high accuracy in saliency detection. Inspired by this type of research, This chapter designs a convolutional deep neural network to solve the problem of visual saliency [13,14].

In order to improve computational efficiency, this paper proposes a low-dimensional feature map extraction layer from the perspective of combining deep neural networks and underlying feature extraction to decompose the image and transform it into a low-dimensional feature map. Then the calculated low-dimensional supervised features are sent to a small-sized fully convolutional network for training. During training, the algorithm calculates loss functions for the output of three scales to control the convergence of the network model and enhance the robustness of the algorithm.

It is proposed to use an end-to-end deep convolutional network to learn and calculate the saliency map. The network structure of this article is shown in the figure. The image is sent to the network and roughly undergoes three processes. First, the image is input to the image low dimensional feature extraction layer. In this layer, this paper first uses traditional methods to decompose the image into about 256 superpixels and extract the feature vector for each superpixel (the composition of the feature vector is designed, These feature vectors are then projected to a (1x84x24x24) tensor (or feature map) by linear interpolation. In this way, the image goes through the first layer of the network, and its size is reduced by more than 100 times. The contrast is only reduced 2 times the VGG network, the network proposed in this paper greatly saves computing resources. Then, after the result of the first layer is obtained, it is sent to a deep convolutional network obtained by supervised training. The network has 8 modules, each module is composed of a 3-layer convolutional network. The last three modules are designed with loss functions respectively, whose purpose is to ensure good convergence during training. After the convolutional network operation, the results obtained are sent to the combined layer In this layer, first calculate the (24x24) saliency map from the previously obtained (24x24) feature map, and then use the reverse interpolation to project the saliency map to the final effect of the original image size.

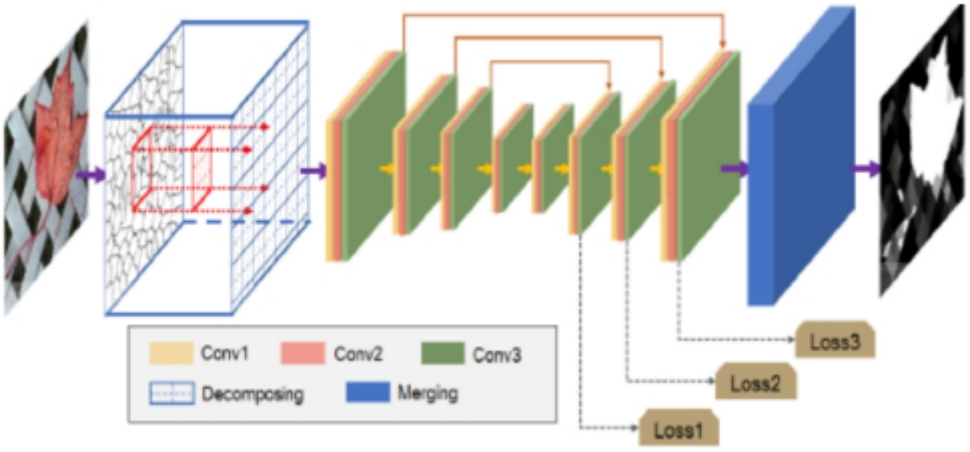

Figure. 3. End-to-end deep convolutional network

3. Research on Saliency Algorithm Based on Width Neural Network

The saliency detection model designed by fusion of low-dimensional feature extraction and deep convolutional neural network improves the calculation speed of the algorithm to a certain extent. However, in this model, there are still a lot of efficiency problems caused by convolution operations. This makes the algorithm still unable to be applied to devices that require high enough real-time and computational efficiency. If real-time saliency detection is performed on a UAV, the current saliency algorithm based on deep convolutional neural networks still cannot meet the requirements of its computational efficiency. Therefore, a more lightweight network is needed to achieve an efficient real-time saliency detection system. The width neural network, as the name implies.

It is a neural network with a shallow number of hidden layers but a horizontal extension, and this type of learning system performs well in classification and regression problems. At present, there have been many variants of the width neural network, but it is understood that there is no existing research to apply the width neural network to computer vision problems. This chapter will explore the potential of width learning systems in visual saliency tasks. In most saliency detection work, it is very common to train manually extracted features to predict the corresponding saliency value or saliency map. But in many works, the feature vector used often occupies a very high dimensionality, which not only brings more computational burden, but also adds more noise to the feature vector. In this work, we designed a condensed broad feature (Broad Feature), and combined it with color, position, prior knowledge and global contrast features to create a comprehensive and low-dimensional feature vector, the feature vector It only takes up 38 dimensions, which is very conducive to fast training. Inspired by the randomness of the width neural network and the characteristics of effective training, this chapter combines the width neural network and the boosting (Boosting Learning) algorithm to establish an effective and accurate learning model, which is called the boosting-based neural network model (Boosting Broad Neural Network, BBNN). Then use the manually extracted features to train BBNN and calculate the saliency map. Then, this chapter proposes a conditional random field (CRF) with parameter learning to optimize the saliency map obtained by the previous prediction to obtain more accurate results. Overall, this chapter proposes a supervised width neural network saliency detection system. The main idea is to train BBNN to learn the manually extracted area description features, and optimize by applying conditional random field (CRF) to obtain more accurate results. This chapter also makes experimental comparisons with a variety of saliency detection algorithms, and makes a separate accuracy and efficiency comparison with the algorithm proposed in the previous chapter. The results show that this algorithm has better calculations without losing accuracy. effectiveness.

4. Conclusion

This paper proposes to fuse a low-dimensional feature extraction layer and a deep full convolutional network to solve the saliency detection problem. The low-dimensional feature map extraction layer is used to reduce the dimensionality, and then the calculated feature map is sent to a small-sized fully convolutional network for training, which can greatly reduce the amount of calculation without loss of image information. When training the deep

convolutional network proposed in this article, this article establishes three loss functions at the output, and each loss function corresponds to a scale truth map. This allows the network to fully learn global information and enhance the robustness of the algorithm.

References

[1] Laurent Itti. Automatic foveation for video compression using a neurobiological model of visual attention[J]. IEEE Transactions on Image Processing, 2004, 13(10):1304-1318.

[2] Guo-Xin Zhang, Ming-Ming Cheng, et al. A Shape-Preserving Approach to Image Resizing[J]. Computer Graphics Forum, 2009, 28(7):1897-1906.

[3] Ueli Rutishauser, Dirk Walther, Christof Koch, et al. Is bottom-up attention useful for object recognition[C]. IEEE Conference on Computer Vision and Pattern Recognition (2) 2004: 37-44.

[4] Yunchao Wei, Xiaodan Liang, Yunpeng Chen, et al. STC: A Simple to Complex Framework for Weakly-supervised Semantic Segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(11):2314-2320.

[5] Hefeng Wu, Guan Li, Xiaonan Luo. Weighted attentional blocks for probabilistic object tracking[J]. Visual Computer, 2014, 30(2):229-243.

[6] Y.-D. Zhang, S. C. Satapathy, D. S. Guttery, J. M. Górriz, and S.-H. Wang, "Improved breast cancer classification through combining graph convolutional network and convolutional neural network," Inf. Process. Manag., vol. 58, no. 2, p. 102439, 2021.

[7] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in Proceedings of the AAAI conference on artificial intelligence, 2019, vol. 33, no. 01, pp. 7370–7377.

[8] I. Bello, B. Zoph, A. Vaswani, J. Shlens, and Q. V Le, "Attention augmented convolutional networks," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 3286–3295.

[9] W. Wu, Z. Qi, and L. Fuxin, "Pointconv: Deep convolutional networks on 3d point clouds," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 9621–9630.

[10] G. Ghiasi, T.-Y. Lin, and Q. V Le, "Dropblock: A regularization method for convolutional networks," Adv. Neural Inf. Process. Syst., vol. 31, 2018.

[11] [M. Chen, Z. Wei, Z. Huang, B. Ding, and Y. Li, "Simple and deep graph convolutional networks," in International Conference on Machine Learning, 2020, pp. 1725–1735.

[12] M. Sun, S. Zhao, C. Gilvary, O. Elemento, J. Zhou, and F. Wang, "Graph convolutional networks for computational drug development and discovery," Brief. Bioinform., vol. 21, no. 3, pp. 919–935, 2020.

[13] S. Zhang, H. Tong, J. Xu, and R. Maciejewski, "Graph convolutional networks: Algorithms, applications and open challenges," in International Conference on Computational Social Networks, 2018, pp. 79–91.

[14] F. Wu, A. Souza, T. Zhang, C. Fifty, T. Yu, and K. Weinberger, "Simplifying graph convolutional networks," in International conference on machine learning, 2019, pp. 6861–6871.