Array 17 (2023) 100270

Contents lists available at ScienceDirect

Array

journal homepage: www.elsevier.com/locate/array

Salient object contour extraction based on pixel scales and hierarchical convolutional network

Xixi Yuan [a], Youqing Xiao [b], Zhanchuan Cai [b,*], Leiming Wu [c,*]

[a] College of Electronics and Information Engineering, Shenzhen University, Shenzhen 518061, China
[b] School of Computer Science and Engineering, Macau University of Science and Technology, 999078, Macao Special Administrative Region of China
[c] School of Information Engineering, Guangdong University of Technology, Guangzhou 510006, China

ARTICLE INFO

Keywords:
Salient objects
Object contours
Adaptive loss function
Hierarchical network
Pixel scales

ABSTRACT

Image salient object contours are helpful for many advanced computer vision tasks, such as object segmentation, action recognition, and scene understanding. We propose a new contour extraction method for salient objects based on pixel scale knowledge and hierarchical network structure, which improves the accuracy of object contours. First, a deep hierarchical network is designed to capture rich feature details. Then, a new loss function with adaptive weighted coefficients is developed, which can reduce the uneven distribution influence of contour pixels and non-contour pixels in training datasets. Next, the object contours are classified based on the scale information of contour pixels. By importing the prior knowledge of scale categories into the network structure, the model requires a small number of training samples. Finally, the regression task of contour scale prediction is added to the network, and the precise contour scales of foreground objects are performed as prior knowledge or an auxiliary task. The experimental results demonstrate that compared with related methods, the proposed method achieves satisfactory results from precision/recall curves and F-measure score estimation on three datasets.

1. Introduction

The salient object (or foreground object) in a natural image is always interested to human beings, and it is a research trend in the era of artificial intelligence. The contours are essential structure characteristics of objects, which provides posture, size, and category information of objects in an image [1–3]. Therefore, it is significant to study approaches of salient object contour extraction.

Object contours play an important role in semantic recognition tasks, such as object extraction, action recognition, and behavior analysis [4–6]. In fact, the contour refers to the periphery of an object or the outer frame of a figure. The contour detection aims to extract curves that can reflect the shapes of the object in an image. In addition, there are some differences among edges, boundaries and contours. The edges and boundaries generally means the discontinuities of objects in an image in terms of photometrical, geometrical, and physical features. Firstly, in [7], the boundary is defined as the contour of an image, and it reflects the changes of pixel ownership from one object to another. Other researchers tend to regard contours as boundaries of interesting regions. When the contours are not region boundaries, the closed contours cannot be generated by contour detectors, nor can the image be divided into different regions. Secondly, the image edge is the discontinuity of characteristic distribution (e.g., pixel grayscale and texture) in an image. It is a collection of pixels with step changes in the image. As one of the primary image features, image edge detection is mainly used to enhance the contour edges, details and grayscale jumps in an image.

Many researches have been conducted around image contour extraction. Firstly, the early image processing methods basically locate contours by extracting texture features [8,9]. However, due to the massive computation of traditional methods, it is difficult to deal with complex scenarios in an image. Then, with the spring of deep learning methods, image contour extraction has been further developed. For instance, the holistically-nested edge detection (HED) method [10] is based on "VGG16" (i.e., Visual Geometry Group-16) model [11] and side output strategy to extract contours from natural images, which is illustrated in Fig. 1(a). The edge detection method based on richer convolutional features (RCF) [12] concatenates adjacent convolutions at each stage of VGG16, and it effectively improves feature extraction performance, which is represented in Fig. 1(b). However, the existing end-to-end methods extract all edges in the image, and they cannot recognize the salient object [13]. The edge guidance network (EGNet) [14] adopts the step-by-step method to detect salient object: in the first

* Corresponding authors.
E-mail addresses: yuanxixi@szu.edu.cn (X. Yuan), xiaoyq11@vanke.com (Y. Xiao), zccai@must.edu.mo (Z. Cai), leiming_wu@gdut.edu.cn (L. Wu).

https://doi.org/10.1016/j.array.2022.100270
Received 10 October 2022; Received in revised form 2 December 2022; Accepted 5 December 2022
Available online 7 December 2022

step, the end-to-end method is used to obtain salient objects; in the second step, the contours are extracted based on gradient information of salient objects. However, this method with two steps is complicated in computation. At the same time, the error is expanded, including the salient object extraction error and the contour extraction error with gradient information. In the deep category-aware semantic edge detection network (CASENet) [15], each contour pixel is associated with more than one category. The pixels appear in contours or junctions belonging to two or more semantic categories. A saliency detection method based on salient contour-aware with twice learning strategy is proposed in [16], but it cannot always extract saliency contours accurately. Liu et al. [17] presented a simultaneous detection method of salient object, edge, and skeleton using the dynamic feature integration (DFI) and the task-adaptive attention module. It produces most important edges in the image, but it cannot distinguish object contours exactly.

A new salient object contour extraction method is proposed in this paper, which can recognize the foreground object as illustrated in Fig. 1(c). In this method, an improved hierarchical network is designed to integrate rich features. Then, a new Softmax loss function with adaptive coefficients is proposed to balance unevenly distributed samples, which is used to improve the accuracy of contour extraction. Finally, in order to evaluate object contour extraction performance, three new datasets are presented based on SK-SMALL [18], SK-LARGE [19], and WH-SYMMAX [20] datasets, and they have skeleton scales originally. In this paper, these datasets are regenerated for salient object contour position, and then they are used in training and testing experiments. In addition, there are four main contributions of the proposed method:

(1) Different from popular edge extraction methods of HED, RCF, and DFI, the proposed method extracts salient object contours, which is more significant than image edges for intelligent industry. Besides, this is an end-to-end method without calculating gradient information to save computing resources.

(2) The extracted contour pixels contain scale information, which can be regarded as a prior knowledge and an auxiliary task to improve the recognition accuracy.

(3) A new loss function with adaptive weighted coefficients is proposed. The new loss function is suitable for images with uneven distribution of sample categories, which helps to improve the extraction accuracy of contour pixels. While the loss function used in the HED can only be used for binary classification problems.

(4) Furthermore, a hierarchical network for contour extraction is designed to extract rich features, and new salient object contour datasets with scale information are built.

The remainders of this paper are organized as follows. Section 2 introduces related works of contour extraction and the basic network structure. The principle of the salient object contour extraction are described in Section 3. Section 4 demonstrates the effectiveness of the proposed method. At last, the conclusion and future works are stated in Section 5.

2. Related work

Different from the existing contour extraction methods, the proposed method can extract salient object contours in an image. The development of contour extraction methods and the multi-task hierarchical network are introduced below.

Contour extraction: In the past decades, many researches on image contour extraction have emerged. These are mainly classified into two categories. One is the early traditional image processing methods, such as Roberts operator, Sobel operator, and Canny operator. The other one is the machine learning methods. In recent years, machine learning has been widely used in intelligent industries [17,21,22]. In the HED method [10], the "side output" convolution is used to improve the edge extraction accuracy and computational efficiency. Afterwards, the scale-associated side-output (FSDS) is proposed in [18]. It realizes

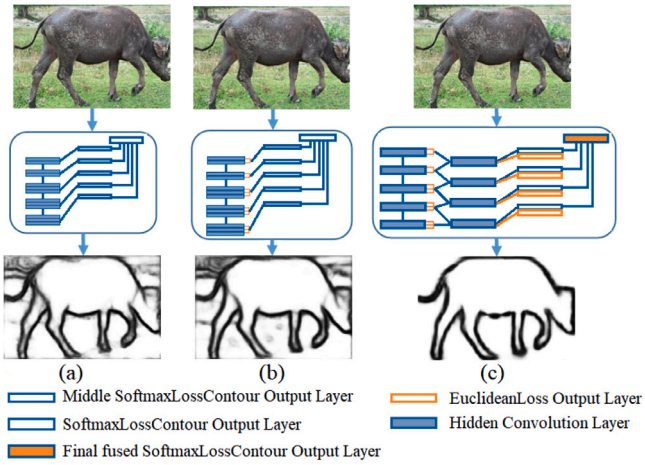

Fig. 1. Brief illustration of the proposed method and two related methods. (a) the HED method [10] fuses side-output features at different stages to extract all edges from the original image. (b) the RCF method [12] fuses convolutional layers at one stage to conduct one side output, and then fuses side outputs at all stages to generate the final edges. (c) in the proposed method, the hierarchical integration, regression, and the new Softmax function are adopted to exact salient object contours from the natural image.

different supervision according to receptive fields at different stages of the network structure. In the RCF method [12], all convolutional layers at one stage are concatenated, and then the side outputs at different stages are fused. This strategy extracts richer convolutional features than the HED. In addition, the convolutional oriented boundaries (COB) is proposed in [23]. Different from the HED that uses the side-outputs to obtain coarse contour maps, the COB uses multiple small sub-networks to predict oriented contour maps, and then the max response of the sub-network is outputted to decide the final orientation of each pixel.

However, all of these methods extract whole edge features in the image, which cannot be used to recognize salient objects. The fully convolutional encoder–decoder network (CEDN) [24] focuses on detecting higher-level object contours based on the dataset of PASCAL-VOC. But it still outputs some boundaries of background objects. The instance-level salient object segmentation network (MSRNet) [25], the aware salient object detection network (BASNet) [26], and the dynamic feature integration network [17] are salient instance segmentation methods, but they cannot generate clear object contours.

Multi-task hierarchical network: In the network of HED, the loss generated by multiple side-outputs is directly conveyed back to the corresponding convolutional layer, avoiding the gradient disappearance. At the same time, different scales of features are learned at different convolutional layers. Based on the idea of side-outputs, the RCF makes full use of multi-scale and multi-level information of the object, and completes image-to-image prediction by fusing all meaningful convolutional features. The FSDS classifies image pixels to five categories according to the size of the skeleton scale; different stages of network structure have different perception fields; and different perception fields can supervise different skeleton pixel categories. In the network of learning multi-task deep side outputs (LMDS) [19], some regression tasks are added to the FSDS network structure to predict pixel scales. The hierarchical feature integration network (Hi-Fi) [27] can further extract richer features and promote the recognition accuracy. Therefore, a new hierarchical network is designed for salient object contour extraction in this paper.

3. Methodology

In order to improve the extraction accuracy of salient object contours, a new network structure is designed in this paper, including the scale-based hierarchical structure, the weighted Softmax loss function, the contour pixel scale classification, and the contour pixel scale prediction with regression tasks.

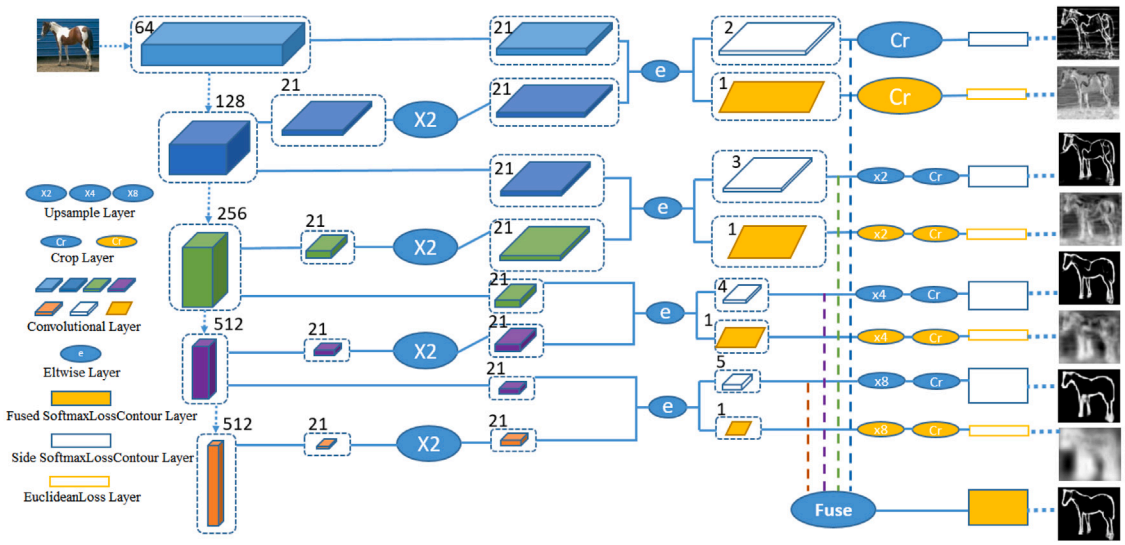

Fig. 2. The proposed network structure of salient object contour extraction.

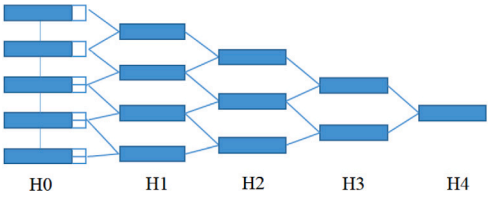

Fig. 3. The hierarchical network structure with five levels based on the VGG16 network structure.

3.1. Hierarchical network structure

The proposed network structure is shown in Fig. 2, and the new hierarchical network structure is designed based on VGG16. The VGG16 has five stages, and the pooling layer is connected at each stage to adjust the receptive field sizes. Only the last convolutional layer at each stage performs hierarchical outputs. It is known that the VGG16-based network structure loses features in pooling processes with the increase of receptive fields. The side-output strategy can reduce feature losses in feature extraction, which is adopted in the designed network structure.

The hierarchical network structure is shown in Fig. 3, which contains five levels (i.e., H0, H1, H2, H3, and H4), and the performance of them are analyzed in experiments. The method with three levels of H0, H1, and H2 are explained specifically in the following. Fig. 4(a) and (b) represents the hierarchical structure of H1 and H2, and their outputs are H1-out and H2-out. Fig. 4(c) illustrates the fused output of H1-out and H2-out, which is called H1H2-out (when the network structure only contains H0, H1, and H2 levels). It should be noted that there is no output layer at the H0 level, where each output layer corresponds to a loss function. In addition, a new Softmax loss function is customized to deal with imbalance distribution of contour and non-contour pixels. In this paper, the regression tasks are only used as auxiliary tasks to reduce losses of feature information, which are not fused in classification tasks.

The contour pixels of a salient object have scale features, which is denoted as a real number. The linear regression is used to fit a prediction model for the values of Y and X in the observed dataset. Therefore, the linear regression task is used to obtain contour scales. In order to reduce feature losses, the regression tasks are added into the network structure named as "Ours1AR" and "Ours2AR". The biggest differences between Figs. 4 and 5 lie in the linear regression tasks and loss functions.

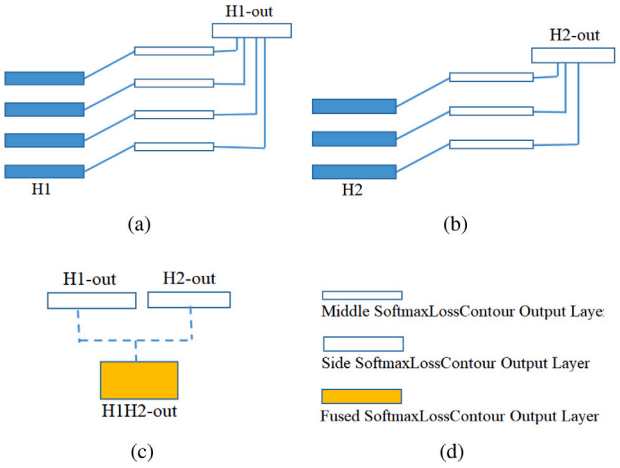

Fig. 4. The integration principle of different levels. (a) and (b) represent the hierarchical structure of H1 and H2, and their outputs are H1-out and H2-out. (c) illustrates the fused output of H1-out and H2-out. (d) the SoftmaxLossContour output layer means that the Softmax loss function has adaptive coefficients.

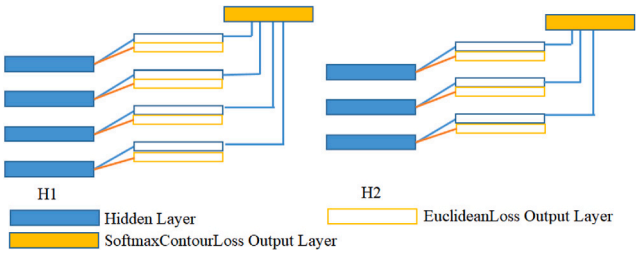

Fig. 5. The new network structures of H1 and H2 levels with regression tasks.

3.2. Contour pixel and contour map

In order to extract object contour pixels in convolutional neural network, the image pixels confront with classification problems. The pixel scales are classified by piecewise quantization. The number of categories depends on the sizes of receptive fields in the network structure. For example, from the first stage to the fifth stage of the VGG16, as the extracted features become more and more abstract, the

Table 1
The contour scale categories and the ranges.

Scales	≥ 150 ‖< 1	[1, 10)	[10, 26)	[26, 60)	[60, 150)
Category	0	1	2	3	4

sizes of receptive fields increase gradually, so the counter pixels are defined into five categories in this paper.

Contour pixel scale piecewise quantization: As shown in Fig. 6(c), the scale of a contour pixel refers to the radius of the maximum inscribed circle that is centered on the skeleton point and is tangent to the contour of the object. In other words, the scale is the distance between the skeleton point and the contour point of the object that is closest to the skeleton point.

In the classification task of contour pixels, the pixels are divided into k categories. The category 0 represents non-contour pixels. The VGG16 network structure has five stages, and the last convolution of each stage corresponds to the sizes of receptive fields with 5, 14, 40, 92, and 196, respectively [19]. According to the ranges of different receptive fields, the contour scale quantization category table is able to be obtained in Table 1. For example, the second category of contour pixels represents the scales of pixels between the range of [12, 26). In Fig. 6(d), the contours with different scales are shown in four kinds of colors, and they are red, yellow, blue, and green from big scales to small scales. Then, the corresponding skeletons are shown in four lighter colors.

Contour map generation: One image can be defined as a non-contour pixel map $\sum_{non-contour}$ and a contour pixel map $\sum_{contour}$. The final contours of the image are composed of different scales. S_0 (non-contour pixels) is defined as category 0, and S_i $(i \in \{0, 1, \ldots, k-1\})$ is the predicted category of scales. For instance, S_2 represents the pixel set with the scales belonging to category 2, i.e., the scales of those pixels are lie in $[10, 26)$. More details are shown in the contour scale piecewise quantization section. Then, it can be found that the $S_1 \bigcup S_2 \bigcup S_3 \bigcup \cdots \bigcup S_{k-1}$ is the contour map. Consequently, the following equations are obtained:

$$\sum_{contour} = S_1 \bigcup S_2 \bigcup S_3 \bigcup \cdots \bigcup S_{k-1} \tag{1}$$

$$\sum_{non-contour} = S_0 \tag{2}$$

After S_0 is obtained, $I - S_0$ is another form of object contour map with binary image, where I is the identity matrix in the trainer, as shown in Fig. 7.

3.3. Scale prediction of contour pixels

The normalization of contour scales and loss functions in the regression task: The scale of a contour pixel is a real number that can be predicted in regression tasks. In order to improve the prediction accuracy, a normalization method of contour scale is proposed in training steps. The network proposed in this paper consists of five stages, and each stage has independent regression tasks to predict the scales. Noted that there are different numbers of regression tasks at different levels. For example, the H1 level has five regression tasks and the H2 level has four regression tasks. The regression task is used to predict the contour scale at different stages, depending on the size of receptive fields. Therefore, it is necessary to establish the correspondence between the ground truth of contour scales and the values of receptive fields at different stages during the training stage, as shown in Fig. 8.

The contour scale normalized before training is in the following:

$$GT = \begin{cases} \dfrac{GT_{init}}{a_i} \times r, & GT_{init} \le a_i \\ 0, & GT_{init} > a_i \end{cases} \tag{3}$$

where a_i denotes the receptive field at stage i. GT_{norm} denotes the contour scale after normalization, which is abbreviated as GT in (3).

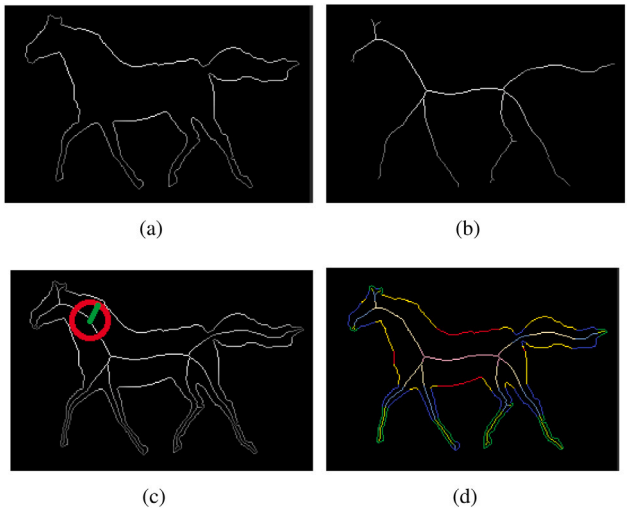

Fig. 6. The illustration of contour pixel scale. (a) is the contour of the horse. (b) is the skeleton of the horse. (c) includes the contour and skeleton. The red circle indicates the maximum inscribed circle from the skeleton pixel to the contour, and the green scale of the point of tangency on contour pixel is the radius of the inscribed circle. In (d), the contours with different scales are shown in four kinds of colors, and the corresponding skeletons are shown in lighter colors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

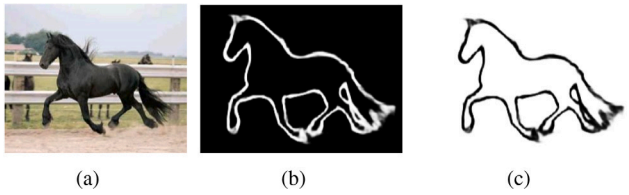

Fig. 7. The contour map generation illustration. (a) is the original image. (b) is the contour map generated by I–S_0. (c) is the contour map of S_0.

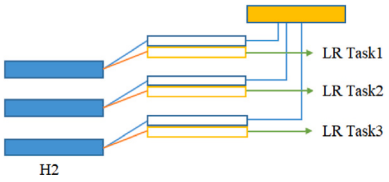

Fig. 8. The relationship between the ground truth and the sizes of receptive fields. In H2 layer, if the range of receptive fields of "LR Task1" is [0, 14), the size of the field equals to 14; if the range of receptive fields of "LR Task2" is [0, 40), the size of the field equals to 40.

GT_{init} denotes the initial contour scale. r is the magnification rate. In the standard normalization, the value of r takes "1". While r is set as "2" in the experiments of this paper. The regression loss is set as the standard Euclidean loss function.

Scale prediction of contour pixels: According to the regression task, the real numbers of contour scales are obtained. The experimental results show that the classification task is able to get a clear contour map. Because of the limitation of the VGG16-based network, only five categories can be obtained and the numerical value is inaccurate. The regression task helps to get a numerical value, but the contours are obscure, as shown in Fig. 9.

As mentioned above, the classification tasks and the regression tasks are combined to further improve the prediction accuracy of contour scales in the proposed method. The final scale values of object contours

(a) (b) (c)

Fig. 9. The contour scale prediction results. (a) is the original image. (b) is the contour map generated by classification tasks. (c) is the contour map generated by regression tasks.

can be calculated by (4):

$$LR_{scale} = \begin{cases} \frac{LR_{init}}{r} \times a_i, & LR_{init} \le a_i \\ 0, & LR_{init} > a_i \end{cases} \tag{4}$$

where LR_{init} is the initial predicted scale and LR_{scale} is the final contour scale in the regression task. In addition, r is the magnification, and it is set as "2" in this paper.

$$\begin{aligned} &\text{If} \quad LR_{scale} \in [S_{low}, S_{high}), \text{then:} \\ &\qquad scale = LR_{scale}, \\ &\text{If} \quad LR_{scale} \notin [S_{low}, S_{high}), \text{then:} \\ &\qquad scale = \frac{S_{low} + S_{high}}{2}. \end{aligned} \tag{5}$$

According to the predicted contour pixel category and the introduction of Section 3.2, the coarse range of pixel scales can be represented as $[S_{low}, S_{high})$. For example, if a pixel belongs to category 2, that means the scale of this pixel ranges at [10, 26), where the value of S_{low} is 10 and the value of S_{high} is 26. Combining the predicted values of the regression task and the classification task, the final contour pixel scale can be got according to (5).

3.4. Designed softmax loss function

It is known that the standard Softmax loss function [28] is commonly used to classify multiple classes with balanced number of samples. However, the classification accuracy of the standard Softmax loss function is bad when the classes of different samples are unbalanced. In salient object contour recognition, the contour pixels of the object are far less than the non-contour pixels, so there exists a sample imbalance problem. In order to improve the classification accuracy, a new Softmax loss function is designed, which can be specifically expressed as:

$$J(\theta) = -\frac{1}{m}\left[\sum_{i=1}^{m}\sum_{j=1}^{k}\lambda_{ij} I\{y_i = j\} log\frac{e^{Z_j}}{\sum_{i=1}^{k} e^{Z_j}}\right] \tag{6}$$

where m marks the number of samples in the training set, k is the number of classes, λ is the defined balance factor namely the weight of the loss function. For the label y, it can take on k different values, so in the training set $\{(x^1, y^1), \ldots, (x^m, y^m)\}$, $y(i) \in \{1, 2, \ldots, k\}$ is obtained. $I\{y_i = j\}$ is the indicator function that indicates whether the sample i belongs to category j. If it is, the value is 1, otherwise the value is 0. The coefficient λ_{ij} is defined as $\lambda_{ij} = 1 - \frac{Y_{ij}}{Y_i}$. i means the i-th sample in one batch of the training dataset. If the parameter of batch-size is 1 (i.e., $i = 1$), that is to say only one image for one batch is selected. $|Y_i|$ represents the total number of pixels in the i-th image. If the height of the image is h and the width of the image is w, then the value of $|Y_i|$ is $h \times w$. $|Y_{ij}|$ represents the number of pixels belonging to class j.

4. Experimental results

In order to verify the performance of the new method, some comparative experiments are performed based on different datasets, with precision/recall curves and F-measure score estimation.

Pseudo Code 1 Computation of Coefficient λ

Input: (h, w)
Output: λ

```
% h means the width of original image, w means the height of original image, and λ is the weight coefficient.
Let i represent pixel point and j be the category of pixel.
sumPixels = h × w
for i = 1 to sumPixels do
    if pixel i ∈ {j}, j = 1, 2, ··· k. then
        Compute pixelClass_j +1
    end if
end for
Compute λ_j = 1 − pixelClass_j / (h×w), and the weight coefficient λ_j for category j can be obtained.
```

4.1. Dataset and implementation details

In order to obtain suitable image datasets with object contour annotations, three different datasets are used for experiments, which are obtained from three open datasets: SK-LARGE [19], SK-SMALL [18], and WH-SYMMAX [20]. On account of the skeleton and scale information in these datasets, the contour information of objects can be located according to the scales of skeletons. Then, the training datasets with contour labels are generated. The new SK-LARGE contains 746 training images and 745 test images with corresponding contour ground-truth. The new SK-SMALL dataset contains 506 images, and the first 300 images are used for training, which is also a subset of SK-LARGE. The new WH-SYMMAX dataset contains 328 horse images, and the first 228 are used for training.

To ensure justice, the experimental results of all methods are produced under the same conditions (i.e., the same type of server and GPU). Then, the GPU configuration is Titan XP in the operation condition.

4.2. Evaluation protocol

The F-measure indicator and precision/recall (PR) curves are used as the evaluation protocol in this paper. The calculation process of F-measure is defined as follows:

$$\text{F}-\text{measure} = \frac{2PR}{P + R} \tag{7}$$

where P and R means precision and recall values, respectively, and they are calculated from the detected contour map and the ground-truth. Specifically, the calculation process of PR curve complies with the following steps:

1. The predicted contour map is transformed into a binary map according to the threshold value.
2. The predicted binary contour map is matched with the ground-truth contour map. In this process, a relatively small position offset is allowed between the detected contour pixels and the ground-truth. If a detected contour pixel matches at least one ground-truth contour pixel, this point will be marked as a matching point TP (true positive). Otherwise, the detected contour point does not match any ground-truth contour point, then this point will be marked as FP (false positive). If this predicted pixel does not belong to any contour pixel point, and it is considered as a contour point in ground-truth, then this point is called FN (false positive).
3. According to different thresholds, different $\sum TP$, $\sum FP$, and $\sum FN$ can be calculated.

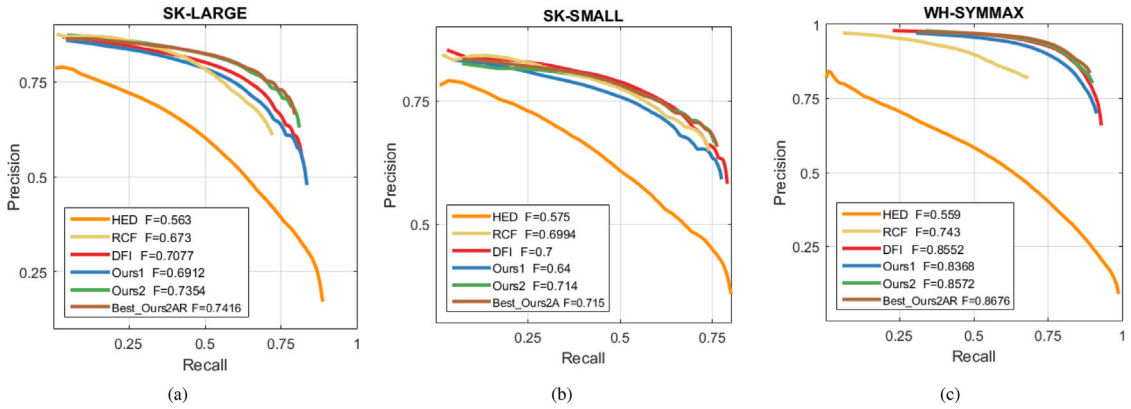

Fig. 10. The PR curves of different contour extraction methods on three datasets in experiments.

Table 2
The experiment results of different contour extraction algorithms with F-measures.

	SK-LARGE	SK-SMALL	WH-SYMMAX
HED [10]	0.5630	0.5750	0.5590
RCF [12]	0.6730	0.6994	0.7430
DFI [17]	0.7076	0.7000	0.8552
Ours1	0.6912	0.6900	0.8368
Ours1A	0.7135	0.6840	0.8412
Ours1AR	0.6525	0.7040	0.8385
Ours2	0.7354	0.7150	0.8652
Ours2A	0.7311	**0.7150**	0.8652
Ours2AR	**0.7416**	0.7110	**0.8676**
Ours3	0.7166	0.6997	0.8504
Ours3A	0.7220	0.7080	0.8637
Ours3AR	0.7130	0.7110	0.8669

The PR values are calculated according to these formulas:

$$P = \frac{\sum TP}{\sum TP + \sum FP}, \quad R = \frac{\sum TP}{\sum TP + \sum FN} \tag{8}$$

According to different thresholds, a series of precision and recall values can be obtained. Finally, these values are fitted to get a PR curve.

4.3. Results of comparative experiments

Three representative algorithms of HED, RCF, and DFI [10,12,17] are selected to compare with the method proposed in this paper. The HED originally exploits "side output" convolution, which greatly improves the edge extraction accuracy and computational efficiency. Then, the RCF further enriches edge features by fused convolutions. Next, the DFI adopts dynamic feature integration to optimize edge features, which is a state-of-the-art edge extraction algorithm. In this paper, some improvements are presented on the basis of HED and RCF network structures, and we optimize the object contours using pixel scale features.

In experiment procedures, firstly, the contour maps are generated based on the trained models. Then, the refined contours are obtained by the standard non-maximum suppression algorithm [29]. Finally, the refined contour maps are evaluated by PR curves and F-measures. In addition, the performance of the proposed algorithm is verified through comparative experiments on three datasets, and the generalization ability of the new algorithm is proved by cross validation.

The contour extraction accuracy results of several different network structures are compared in detail. Generally, the VGG16-based hierarchical network maximally has four levels. In the experiments of contour extraction, the methods of different levels are named as Ours1, Ours2, and Ours3. It is verified that for the proposed method, when the level exceeds 2, the accuracy will be reduced, so the Ours3 is adopted at most. For example, when two levels are used, they are called Ours2. Due to the limitation of the GPU memory, some higher levels of the proposed method cannot be trained. In addition, in order to verify the influence of the variable coefficient loss function and the regression task on contour extraction, relevant independent experiments are also carried out. Assuming that the Ours1A indicates the method of Ours1 with the variable coefficient Softmax loss function method. The Ours1AR indicates the method of Ours1A added with regression task. The definition of other symbols can be analogized. The Table 2 lists the F-measure values of the contour accuracy produced by different network structure experiments.

Table 3
The experimental comparison results of the Softmax function with adaptive coefficients. The values in the table are relative errors.

	SK-LARGE	SK-SMALL	WH-SYMMAX
Ours1A (Vs Ours1)	+3.23%	−0.88%	+0.53%
Ours2A (Vs Ours2)	−0.58%	+0.14%	+0.89%
Ours3A (Vs Ours3)	+0.75%	+1.18%	+1.58%

Table 4
The experimental comparison results of regression task. The sign "+" indicates that the F-measure is improved after adding the regression task, and the sign "−" indicates a decrease.

	SK-LARGE	SK-SMALL	WH-SYMMAX
Ours1AR (Vs Ours1A)	−	+	−
Ours2AR (Vs Ours2A)	+	−	+
Ours3AR (Vs Ours3A)	−	+	+

Table 5
The comparison of position accuracy between our algorithm and the HED. The values in the table are relative errors.

	SK-LARGE	SK-SMALL	WH-SYMMAX
Ours1	+22.77%	+20.00%	+49.70%
Ours2	+26.73%	+11.30%	+50.48%
Ours1A	+15.89%	+22.43%	+50.00%
Ours2A	+30.62%	+24.17%	+53.42%
Ours1AR	+29.86%	+24.35%	+54.78%
Ours2AR	+31.72%	+23.65%	+55.21%

The accuracy differences between the new Softmax loss function and the standard Softmax loss function have been tested in experiments. From Table 3, it can be found that the new Softmax loss function has a better performance than the standard Softmax loss function on three datasets, which indicates that the new Softmax loss function is valid.

In Table 4, it is shown that after adding the regression task, the contour pixel extraction accuracy is improved in most cases. The best result is obtained by using Ours2AR network structure on the SK-LARGE and WH-SYMMAX datasets, and the best F-measure value is

Fig. 11. Illustration of Object contour extraction results on SK-LARGE dataset for several selected images.

Table 6
The comparison of position accuracy between our algorithm and the RCF. The values in the table are relative errors.

	SK-LARGE	SK-SMALL	WH-SYMMAX
Ours1	+2.70%	−1.34%	+12.62%
Ours2	+6.02%	−8.49%	+13.22%
Ours1A	−3.05%	+0.66%	+12.85%
Ours2A	+9.27%	+2.09%	+15.42%
Ours1AR	+8.63%	+2.23%	+16.45%
Ours2AR	+10.19%	+1.66%	+16.77%

0.7416 and 0.8676 (see Table 2), respectively. While the results on the SK-SMALL dataset has change little by using Ours2A and Ours2AR.

Tables 5 and 6 list the F-measures of the proposed method compared with HED and RCF methods on SK-LARGE, SK-SMALL, and WH-SYMMAX datasets. It is worth noting that if the weighted loss function and the regression task are not used (i.e., Ours1 and Ours2), the performance on the SK-SMALL dataset is slightly worse than that of the RCF method. If only one level and the weighted loss function are used without regression tasks (i.e., Ours1A), the performance on the SK-LARGE dataset is 3.05 percentage points, which is lower than that of the RCF method.

The PR curves in Fig. 10 show that the proposed method is much better than the HED and RCF methods. Specifically, Ours2AR has the best results on SK-LARGE and WH-SYMMAX datasets. The Ours2A is slightly better than the Ours2AR on SK-SMALL dataset, with the F-measure values of 0.715 and 0.711, respectively.

Some typical salient object contour extraction results on SK-LARGE dataset are shown in Fig. 11. From Fig. 11, it can be found that the extracted results of HED and DFI methods contain some non-contour details and background information. In addition, the results extracted by the RCF method lose some contour information. While the proposed method avoids these problems and achieves better results.

5. Conclusion

In this paper, a salient object contour extraction method is proposed by fusing scale information on hierarchically convolutional network. In the method, a new network structure with regression task and hierarchical feature integration mechanism is established, which has good performance when extracting salient object contours from natural images; a new variable coefficient loss function is proposed to handle the unevenly distributed pixel classes in machine learning. The experimental results show that the proposed method is superior to other contour extraction methods in terms of salient object contours. Moreover, the salient object contours obtained by the proposed method contain scale information, which can be applied to the subsequent skeleton generation and salient object generation. As a prior knowledge in training, it is also helpful to improve the model performance.

In the future work, the new approach will be extendedly optimized in two possible manners. Firstly, a more effective network structure can be designed for object recognition. Secondly, multi-regressive aiding tasks like classification tasks can be merged into the network, which are not appeared in existing methods. After the fusion step, the regression data can be used to estimate more accurate inscribed circle radiuses between the contour and the skeleton. Finally, the contours are limited to five categories, which reduces the contour accuracy, so better results may be generated by increasing scale categories.

CRediT authorship contribution statement

Xixi Yuan: Investigation, Methodology, Writing – review & editing. **Youqing Xiao:** Conceptualization, Software, Methodology, Writing – original draft. **Zhanchuan Cai:** Supervision, Writing – review, Funding. **Leiming Wu:** Investigation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported in part by the Science and Technology Development Fund of Macau under Grant 0059/2020/A2, Grant 0052/2020/AFJ, and Grant 0038/2020/A, in part by the National Science Foundation of China under Grant 62101346, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2021A1515011702, and in part by the Stable Support Plan for Shenzhen Higher Education Institutions under Grant 20200812104316001.

References

[1] Gupta A, Anpalagan A, Guan L, Khwaja AS. Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. Array 2021;10:100057. http://dx.doi.org/10.1016/j.array.2021.100057.

[2] Li J, Fan J, Zhang Z. Towards noiseless object contours for weakly supervised semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. New Orleans, Louisiana; 2022, p. 16856–65.

[3] Kazuma F, Kazuhiko K. Generative and self-supervised domain adaptation for one-stage object detection. Array 2021;11:100071.

[4] Gabriel L, Nasim H, Irene C. Semi-supervised learning approach for localization and pose estimation of texture-less objects in cluttered scenes. Array 2022;100247.

[5] Cheng M, Mitra NJ, Huang X, Torr PH, Hu S. Global contrast based salient region detection. IEEE Trans Pattern Anal Mach Intell 2015;37(3):569–82.

[6] Gong XY, Su H, Xu D, Zhang ZT, Shen F, Yang HB. An overview of contour detection approaches. Int J Autom Comput 2018;15(6):656–72.

[7] Martin DR, Fowlkes CC, Malik J. Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans Pattern Anal Mach Intell 2004;26(5):530–49.

[8] Martin DR, Fowlkes CC, Malik J. Learning to detect natural image boundaries using local brightness, color, and texture cues. IEEE Trans Pattern Anal Mach Intell 2004;26(5):530–49. http://dx.doi.org/10.1109/TPAMI.2004.1273918.

[9] Rishi J, A CD. Preserving boundaries for image texture segmentation using grey level co-occurring probabilities. Pattern Recognit 2006;39(2):234–45.

[10] Xie S, Tu Z. Holistically-nested edge detection. Int J Comput Vis 2017;125(1–3):3–18.

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

[12] Liu Y, Cheng M, Hu X, Bian J, Zhang L, Bai X, et al. Richer convolutional features for edge detection. IEEE Trans Pattern Anal Mach Intell 2018;41(8):1939–46.

[13] Dollár P, Zitnick CL. Fast edge detection using structured forests. IEEE Trans Pattern Anal Mach Intell 2015;37(8):1558–70.

[14] Zhao JX, Liu JJ, Fan DP, Cao Y, Yang JF, Cheng MM. EGNet: Edge guidance network for Salient object detection. In: Proceedings of the IEEE international conference on computer vision. Seoul, Korea; 2019, p. 8778–87.

[15] Yu Z, Feng C, Liu MY, Ramalingam S. CASENet: Deep category-aware semantic edge detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Honolulu, Hawaii; 2017, p. 5964–73.

[16] Zhu C, Yan W, Liu S, Li T, Li G. Salient contour-aware based twice learning strategy for saliency detection. In: Proceedings of the IEEE/CVF international conference on computer vision workshops. Seoul, Korea; 2019, p. 2541–8.

[17] Liu JJ, Hou QB, Cheng MM. Dynamic feature integration for simultaneous detection of salient object, edge, and skeleton. IEEE Trans Image Process 2020;29:8652–67.

[18] Shen W, Zhao K, Jiang Y, Wang Y, Zhang Z, Bai X. Object skeleton extraction in natural images by fusing scale-associated deep side outputs. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Las Vegas, NV, USA; 2016, p. 222–30.

[19] Shen W, Zhao K, Jiang Y, Wang Y, Bai X, Yuille A. DeepSkeleton: Learning multi-task scale-associated deep side outputs for object skeleton extraction in natural images. IEEE Trans Image Process 2017;26(11):5298–311.

[20] Shen W, Bai X, Hu Z, Zhang Z. Multiple instance subspace learning via partial random projection tree for local reflection symmetry in natural images. Pattern Recognit 2016;52:306–16.

[21] Rampun A, López Linares K, Morrow PJ, Scotney BW, Wang H. Breast pectoral muscle segmentation in mammograms using a modified holistically-nested edge detection network. Med Image Anal 2019;57:1–17.

[22] Liu Y, Yang X, Wang Z, Lu C, Li Z, Yang F. Aquaculture area extraction and vulnerability assessment in Sanduao based on richer convolutional features network model. J Oceanol Limnol 2019;37(6):1941–54.

[23] Maninis KK, Pont-Tuset J, Arbeláez P, Gool LV. Convolutional oriented boundaries: From image segmentation to high-level tasks. IEEE Trans Pattern Anal Mach Intell 2018;40(4):819–33.

[24] Yang J, Price B, Cohen S, Lee H, Yang MH. Object contour detection with a fully convolutional encoder-decoder network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Las Vegas, Nevada, USA; 2016, p. 193–202.

[25] Li G, Xie Y, Lin L, Yu Y. Instance-level salient object segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Honolulu, HI, USA; 2017, p. 2386–95.

[26] Qin X, Zhang Z, Huang C, Gao C, Dehghan M, Jagersand M. BASNet: Boundary-aware salient object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. Long Beach, CA; 2019, p. 7479–89.

[27] Zhao K, Shen W, Gao S, Li D, Cheng M. Hi-Fi: hierarchical feature integration for skeleton detection. In: Proceedings of the 27th international joint conference on artificial intelligence. Stockholm, Sweden: AAAI Press; 2018, p. 1191–7.

[28] Liu W, Wen Y, Yu Z, Yang M. Large-margin softmax loss for convolutional neural networks. In: Proceedings of the International conference on machine learning. New York, NY, USA: Proceedings of Machine Learning Research; 2016, p. 507–16.

[29] Neubeck A, Van Gool L. Efficient non-maximum suppression. In: Proceedings of the 18th international conference on pattern recognition, Vol. 3. Hong Kong, China: IEEE; 2006, p. 850–5.