IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 2, 2019, accepted April 17, 2019, date of publication May 7, 2019, date of current version June 17, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2915261

Saliency Detection Using Global and Local Information Under Multilayer Cellular Automata

YIHANG LIU AND PEIYAN YUAN
College of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China
Engineering Lab of Intelligence Business and Internet of Things, Henan Normal University, Xinxiang 453007, China

Corresponding author: Peiyan Yuan (137932596@qq.com)

This work was supported in part by the National Natural Science Foundation of China under Grant U1304607 and Grant 61370169, in part by the Key Scientific Research Project of Higher School of Henan Province under Grant 15A520080 and Grant 15A520020, and in part by the Dr. Startup Project of Henan Normal University under Grant qd12138 and Grant qd14134.

ABSTRACT To detect the salient object in natural images with low contrast and complex backgrounds, a saliency detection method that fuses global and local information under multilayer cellular automata is proposed. First, a global saliency map was obtained by the iteratively trained convolutional neural network (CNN)-based encoder-decoder model. Moreover, to transmit high-level information to the lower-level layers and further reinforce the object edge, the skip connections and edge penalty term were added to the network. Second, the foreground and background codebooks were generated by the global saliency map, and sparse coding was subsequently obtained by the locality-constrained linear coding model. Thus, a local saliency map was generated. Finally, the final saliency map was obtained by fusing the global and local saliency maps under the multilayer cellular automata framework. The experimental results show that the average F-measure of our method on the MSRA 10K, ECSSD, DUT-OMRON, HKU-IS, THUR 15K, and XPIE datasets is 93.4%, 89.5%, 79.4%, 88.7%, 73.6%, and 85.2%, respectively, and the *MAE* is 0.046, 0.067, 0.054, 0.044, 0.072, and 0.049. Ultimately, these findings prove that our method has both high saliency detection accuracies and strong generalization abilities. In particular, our method can effectively detect the salient object of natural images with low contrast and complex backgrounds.

INDEX TERMS Saliency detection, global and local maps, multilayer cellular automata, CNN-based encoder-decoder model, sparse coding.

I. INTRODUCTION

In recent years, various image acquisition devices have emerged, and image resources have become increasingly more abundant, consequently resulting in the severe problem of image information redundancy. Acting as an image preprocessing method, saliency detection can make the salient objects that are of interest to human beings stand out and eliminate the background (i.e., redundant information) greatly. Thereby, saliency detection has become a hot topic in the field of computer vision [1]–[19].

For those traditional saliency detection methods that rely on global information, while they can detect the salient object to some extent, they usually cannot suppress background noise well enough [7], [16]. For those relying on local information, they often cannot find the salient objects well, or even not at all [17]. For those relying on both global and local information, they usually depend on simple contrast calculations and limited prior cues and cannot properly detect the salient object of natural images with low contrast and complex backgrounds [1]–[7].

To solve these problems, one of the mainstream methods is to mimic the mechanism of the human visual system (HVS). In the last 22 years, the research on saliency detection has made great progress. However, the mechanism of HVS has not been fully clarified. Although the deep neural network has been developed and is helpful for saliency detection [8]–[15], saliency detection failures still often occur because of the special structure of the local receptive field and the information loss after the pooling operation in the encoder part [8]–[15].

Therefore, we propose a model that combines a deep neural network and the traditional saliency detection algorithm to extract global and local information effectively, and we describe the boundary and local details of the salient

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Liu.

object sufficiently. First, a global saliency map was generated by the convolutional neural network (CNN)-based encoder-decoder model. Second, a local saliency map was obtained by the locality-constrained linear coding model. Finally, the final saliency map was generated by fusing the global and local saliency maps under the multilayer cellular automata framework.

In short, the main contributions of this work are as follows:

(1) The convolutional network have significant advantages in image processing, so we propose a CNN-based encoder-decoder model to extract the salient object global information. To reduce high-level information loss and avoid ambiguous boundaries, our network employs skip connections and an edge reinforcement term to improve the saliency detection results. In short, our network has a lightweight yet **powerful** network structure and makes the best of high-level and spatial information to achieve the best saliency detection performance.

(2) We make the utmost of deep learning and traditional methods. The global saliency maps generated by our deep network are used to generate local codebooks; then, the local information is represented in codebook reconstruction by using the locality and sparsity coding method. The coding method focuses on superpixels, which are of region-level saliency detection and can extract necessary local spatial information to make up for the lack of pixel-level saliency detection.

II. RELATED WORK

Most of the traditional saliency detection methods are used to handcraft image features and extract effective priors. Sun *et al.* [3] proposed a detection approach that highlighted salient objects by using the relationship between saliency detection and Markov absorption probability. The image is represented as a graph, which is then reconstructed by the absorbing Markov chain. The position of the salient object is determined by the absorption probability matrix, and the saliency map is obtained by foreground prior sorting. Zhu *et al.* [4] introduced a more complex and robust boundary prior to assist in saliency detection and proposed a background detection algorithm based on boundary connectivity to characterize the spatial information of image regions relative to image boundaries. Tong *et al.* [5] proposed a learning mechanism to solve the problem of the inaccuracy and restriction of feature representation in prior knowledge. Namely, the weak saliency detection model is built by merging various low-level features of the image, and the strong saliency detection model is iteratively trained by using the AdaBoost-enhanced learning method. Tong *et al.* [7] calculated the global saliency map via the low- and high-level priors, utilized the locality-constrained linear coding to generate the local saliency map based on the codebook generated from the global saliency map, and optimized them to obtain the final saliency detection results. However, for those images that do not meet a priori criteria, such as those with low contrast and complex backgrounds in our work, the feature representation ability must be stronger, and the target edge cannot be ignored. Moreover, the information of local details has to be considered.

In recent years, deep learning networks have achieved great success in saliency detection. Wang *et al.* [8] utilized the deep neural network to extract the image features of the local patch and calculated the saliency value of each pixel. Based on the local salient results, global contrast and geometric information (as the target candidate regions of the global features), the saliency values of the predicted region are calculated by the deep neural network, and the final saliency map is obtained by weighted fusion. Li and Yu [9] proposed a deep neural network saliency detection algorithm that could extract multiscale features by using the top fully connected layer of convolutional neural network (CNN) to extract image features at three different scales, and the spatial coherence was adopted in the algorithm such that the multiscale results were fused by linear combination. Lee *et al.* [10] proved that hand-crafted image features could provide complementary information to compensate for the performance of a high-level feature detection model. Therefore, a deep learning framework is proposed to fuse high- and low-level features. Among them, high-level image features are extracted by the VGG network, and low-level image features are obtained via distance maps between other parts and are encoded by CNN. Subsequently, the high- and low-level image features are connected to the fully connected classifier to obtain regional saliency results. Li *et al.* [11] proposed a multitask deep detection model based on a fully connected CNN. The intrinsic relationship between saliency detection and semantic segmentation can be explored by a multitask learning mechanism. By collaborative learning, the features of the fully connected convolution layer can be shared and can effectively reduce the redundant information. Zhang *et al.* [12] built a fully connected deep CNN model with a convolutional encoder-decoder architecture to learn deep uncertain convolutional image features via a reformulated dropout, and they proposed a novel hybrid upsampling method to reduce the checkerboard artifacts of deconvolution operators. To enable salient objects to be unambiguously annotated in complex background images, Chen *et al.* [13] put forward a two-stream fixation-semantic CNN. The fixation stream and semantic stream are utilized to obtain the fixation prediction and semantic perception, respectively; subsequently, the two predictions are fed to the inception-segmentation module to fuse the fixation density image feature and semantic segmentation image feature. Hou *et al.* [14] put forward a saliency detection model by introducing short connections into a holistically nested edge detector to provide more powerful image feature representations. The short connection structure can enrich high-level semantic image features, and high-quality regions are then detected. Zhang *et al.* [15] proposed a progressive attention-guided network via the attention mechanisms to extract attentive

image features, along with the multipath recurrent feedback to generate multilevel semantic contextual information and transmit it from the top layers to the shallower layers.

In summary, although the deep CNN can achieve significant performance in image feature extraction, traditional methods are still needed to assist the deep CNN in performing better saliency detection. Therefore, our model employs a deep CNN and local coding fusion to not only extract rich and clear global information but also address the local details. Namely, we put forward a saliency detection method based on a CNN-based encoder-decoder model and locality-constrained linear coding (LLC) model.

Zeiler *et al.* [20] extracted the mid- and low-level image features by unsupervised feature learning and iteratively trained the deconvolution network until the error between the original and reconstructed images was minimal; subsequently, Zeiler *et al.* [21], [22] proposed a deconvolution network for learning mid- and high-level features. The hierarchical network is constructed by a cross-convolution sparse coding layer and a maximum pool level. In the encoding and decoding stages, Noh *et al.* [23] introduced the autoencoding structure to integrate the convolution network with the deconvolution network for semantic image segmentation. Similarly, the global saliency detection was implemented by the CNN-based encoder-decoder model in our work, whose structure consists of two components, convolution and deconvolution.

Wang *et al.* proposed a LLC model [24], which inherited the small reconstruction error from sparse coding in the encoding process and introduced a locality constraint to represent the feature vectors. Wu *et al.* [25] utilized the LLC model, instead of the complex calculation of the posterior probability of the Fisher vector, to obtain a simpler global description for image retrieval. Wang *et al.* [26] argued that there was no salient similarity between the dictionary and the image features, and they improved the LLC model by salient similarity (the salient k-nearest neighbor search algorithm and the saliency max pooling). Therefore, we utilize the LLC model to realize locality saliency detection via minimizing its reconstruction error.

Qin *et al.* [27] proposed the hierarchical cellular automata, which included single-layer cellular automata and cuboid cellular automata, to effectively improve salience detection results. Among them, the single-layer cellular automata employs an unsupervised propagation-based approach to explore the intrinsic relevance of similar regions by using interactions with neighbors, and the cuboid cellular automata utilizes a Bayesian framework to fuse the multiple saliency maps. Zhang *et al.* [28] proposed a saliency detection algorithm via an absorbing Markov chain (AMC), which could learn a transition probability matrix from multiple-layer deep features extracted by a fully connected CNN, and the angular embedding technique was adopted to rearrange the global orderings from local orderings of AMC saliency and boundary maps. Likewise, we utilize the multilayer cellular automata structure to effectively combine the global and local saliency maps, and the final saliency map was generated.

III. OUR METHOD

The pipeline of our method is shown in Fig. 1.

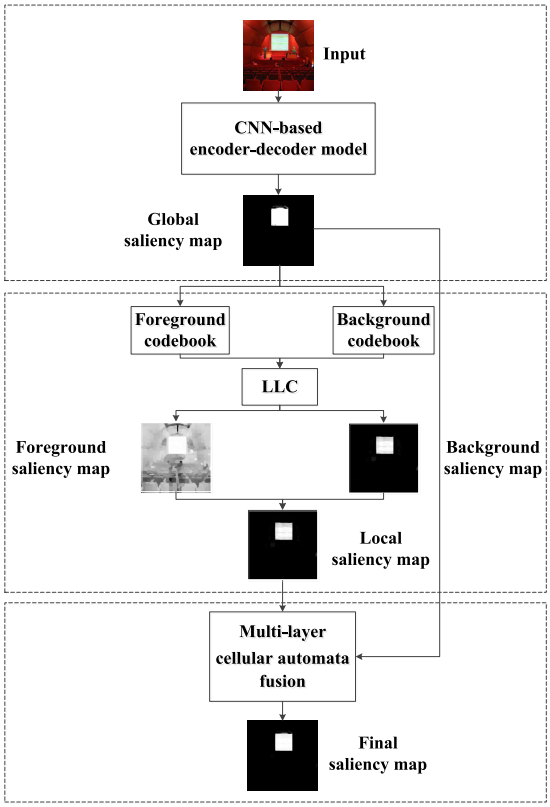


FIGURE 1. Pipeline of our method.

As seen in Fig. 1, our method involves three stages: 1) Global contour information of an image is extracted by using a CNN-based encoder-decoder model, and a global saliency map is then generated; 2) The foreground and background codebooks are obtained from the global saliency map by adaptive thresholds, the two codebooks are encoded by the locality-constrained linear coding model, and a local saliency map is subsequently acquired; and 3) The multilayer cellular automata framework is employed to combine the global and local saliency maps, and the final saliency map is thus obtained.

A. GLOBAL SALIENCY DETECTION

1) BASIC NETWORK STRUCTURE CONSTRUCTION

Traditional saliency detection methods mainly rely on simple contrast calculations and limited prior cues [1], [2], [5]–[7], which results in their inability to properly detect salient objects of images with low contrast and complex background [2]–[5]. Aiming at solving this problem, the CNN-based encoder-decoder model (see Fig. 2) is utilized to generate the global saliency map.

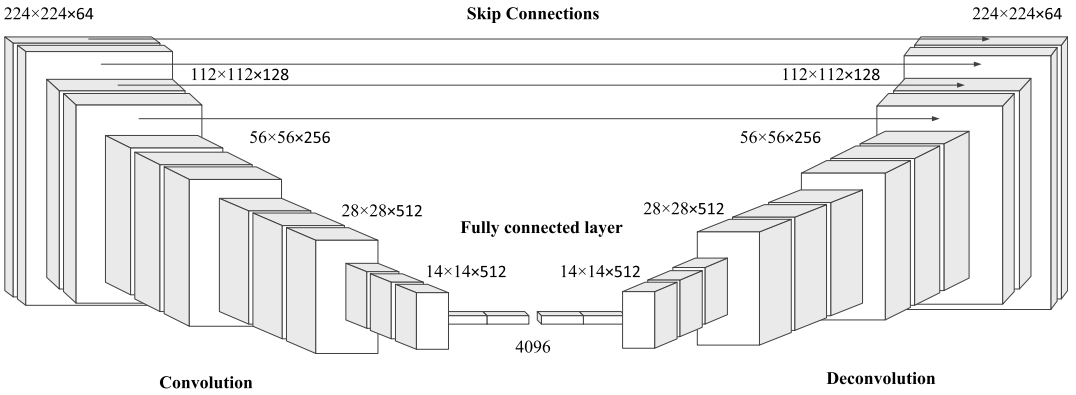


FIGURE 2. Architecture of the CNN-based encoder-decoder model.

As shown in Fig. 2, the structure of our CNN-based encoder-decoder model consists of two components: the encoder is responsible for extracting the global spatial information of the salient object via convolutional layers (see the left component of Fig. 2); the decoder is in charge of generating the global saliency map by using deconvolution layers (see the right component of Fig. 2). Obviously, the deconvolution component is strictly symmetric with the convolution one. That is, each encoder layer has its corresponding decoder layer. In this paper, the two components are of VGG-16 structure [29]. The VGG-16 model possesses five convolutional groups, followed by two fully connected layers; each group of convolutional layers is connected by a pooling layer, so the convolution component has 5 pooling layers. To transmit high-level information directly to the low-level convolution, the skip connections are added between the first four layers of the encoder part and the last four layers of the decoder part. Additionally, to reduce the ambiguous boundary of salient objects, the edge penalty term is added to the loss function of the network.

According to the literature [30], image features at the convolutional and deconvolutional layers can be defined as follows:

$$h^k = f\left(\sum_{p\in P} x^p \otimes \omega^k + b^k\right) \tag{1}$$

where h^k refers to the potential representation of the k-th feature map at the current layer; f is the ReLU activation function for nonlinear calculation; x^p denotes the p-th feature map of the feature map group P at the previous layer; $\otimes$ represents the convolution operation; and ω^k and b^k are the network weights and bias of the k-th feature map at the current layer, respectively.

We trained the CNN-based encoder-decoder model by supervised learning because the ground truth was used in calculating the loss function during the network training. Namely, using the ground truth, to fine-tune the network parameters. Hence, the decoding results of our method can be regarded as the global saliency map. Among them, the training result of each iteration needs to be compared with the ground truth to ensure that the cross-entropy loss is minimized. The cross-entropy loss can be defined as:

$$L_{cross-entropy} = \frac{1}{HW}\sum_{i=1}^{W}\sum_{j=1}^{H}\big[-G(I)_{i,j}\cdot\ln(D\circ E(I)_{i,j}) - (1-G(I)_{i,j})\cdot\ln(1-D\circ E(I)_{i,j})\big] \tag{2}$$

where H and W are the height and width of image I; $G(I)$ represents the ground truth of the image I; D refers to the decoding function; E denotes the encoding function; (i, j) represents the coordinate of a pixel; and $D \circ E(I)$ stands for the convolution output of the network.

Subsequently, the network weight ω^k and bias b^k are updated via equations (3) and (4), respectively.

$$\omega^k \to \omega^{k'} = \omega^k - \eta\frac{\partial L}{\partial \omega^k} \tag{3}$$

$$b^k \to b^{k'} = b^k - \eta\frac{\partial L}{\partial b^k} \tag{4}$$

where ω^k and b^k are the network weight and bias of the k-th image feature map at the current layer, respectively; $\omega^{k'}$ and $b^{k'}$ represent the updated network weight and bias, respectively; and η is the learning rate that we set to 10^{-4} by extensive experiments.

Due to the special structure of the CNN local receptive field, most of the deep networks will generate an ambiguous boundary of the salient object [15]. Therefore, our network employs skip connections and an edge reinforcement penalty term to improve the saliency detection results.

2) SKIP CONNECTIONS

The high-level features of CNN layers focus on the global position, while low-level features specialize on the details. The pooling operation of the encoder part often compresses and reduces the original information severely, which makes the network unable maintain enough saliency information [15]. Hence, in the deconvolution process, the high-level information cannot be effectively reconstructed, especially

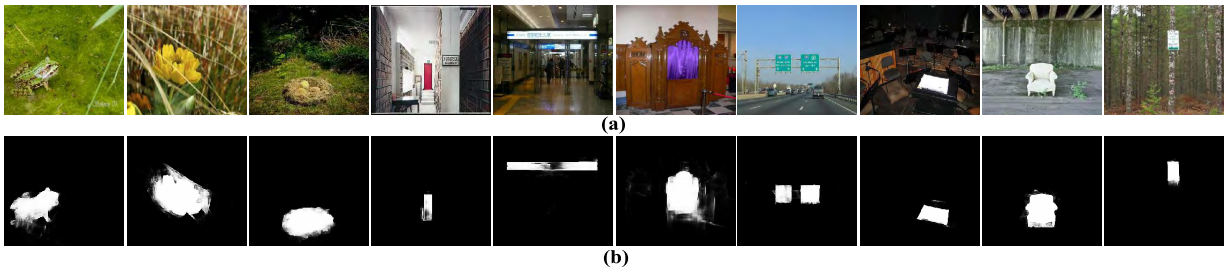


FIGURE 3. Global saliency detection results. (a) Original images; (b) Global saliency maps.

the boundary of the salient object, which cannot be reconstructed properly. To reduce the loss of high-level information, the skip connections are added to the high-level convolution layers in our network to ensure that information can be directly transmitted to the deconvolution layers, and the high-level information is also directly transmitted downward to enrich the low-level information. In reality, the skip connections lie between the first 4 layers of the convolution part and the last 4 layers of the deconvolution part.

3) EDGE REINFORCEMENT PENALTY

To further reinforce the salient object contour, we introduce the edge penalty term into the network loss function. The training of the network does not stop until the pixel-level cross-entropy loss (between the obtained result and the ground truth) is minimized. Obviously, the training process has not taken into account the spatial information of pixels, which will cause two adjacent pixels to belong to either the salient object or background; however, they indeed have very different saliency values. Alternatively, for the two adjacent pixels, one belongs to the salient object and the other belongs to the background, but in fact, they have similar saliency values. All this leads to the salient object having an ambiguous boundary. Therefore, our network employs the edge reinforcement penalty term to integrate the space information of the target boundary with the target boundary contrast.

The filter is used to separate the significant target boundary from the ground truth of the saliency map, and the difference between the boundary pixel and its direct neighbor pixels can be calculated. Therefore, the edge reinforcement penalty term is defined as follows.

$$\begin{aligned} &\forall (i,j) \in B(I), \\ &\forall (k,l) \in \{(i+1,j),(i,j+1),(i-1,j),(i,j-1)\} \\ &\begin{cases} \min \left\| I_{i,j} - I_{k,l} \right\|_2 & if \quad G(I)_{i,j} = G(I)_{k,l} \\ \max \left\| I_{i,j} - I_{k,l} \right\|_2 & otherwise \end{cases} \end{aligned} \tag{5}$$

where $B(I)$ represents the boundary of the salient object within image I; $G(I)_{i,j}$ and $G(I)_{k,l}$ denote the saliency labels of two adjacent pixels. In other words, if both the edge pixel (i,j) and its direct neighbor (k,l) belong to either the salient object or background, the difference $\left\| I_{i,j} - I_{k,l} \right\|_2$ between them is to be minimized; otherwise, the difference $\left\| I_{i,j} - I_{k,l} \right\|_2$ between them is to be maximized.

We employ these differences to measure the boundary loss, L_{edge}. Then, the total loss L is the sum of boundary loss L_{edge} and cross-entropy loss $L_{cross-entropy}$, i.e., $L = L_{cross-entropy} + L_{edge}$.

Some representative global saliency maps generated by our improved CNN-based encoder-decoder model are shown in Fig. 3.

As seen in Fig. 3, our CNN-based encoder-decoder model can achieve wonderful saliency detection performance for images with low contrast (Columns 1, 2 and 3 of Fig. 3(b)) and complex background (Columns 4, 5 and 8 of Fig. 3(b)). Obviously, the frog, flower and nest (Columns 1, 2, and 3 of Fig. 3(a)) are very similar to their surrounding environment, and the backgrounds of the library, lobby and music score (Columns 4, 5 and 8 of Fig. 3(a)) are complicated enough. Undoubtedly, it is impossible to recognize the contour of these salient objects (frog, flower, nest, door, light box, music score, etc.) immediately even for human beings, but our global network is able to detect the salient objects properly.

B. LOCAL SALIENCY DETECTION

We employ the LLC [24] model as described in section 2 to generate local saliency map by minimizing its reconstruction errors.

First, the simple linear iterative clustering (SLIC) algorithm [31] is adopted to divide the given image into N superpixels, $\{r_i\}, i = 1, 2, \ldots, N$, and the visual codebooks can be calculated from the global saliency map. Second, foreground and background codebooks are generated by two adaptive thresholds, (λ_1, λ_2). Please note that $\lambda_2 < \lambda_1 < 1$. λ_1 is the average of global saliency values in the t iterations, where $t = 1.5$ [7]. Since the codebooks are not sensitive to λ_2, we set $\lambda_2 = 0.05$ [7]. Finally, those superpixels whose average global saliency values are larger than λ_1 are regarded as the foreground codebook and otherwise as the background codebook.

For instance, the local descriptor x_i can be encoded by the LLC model, as described in equation (6).

$$\begin{aligned} &\min_{C} \sum_{i=1}^{N} \|x_i - Bc_i\|^2 + \lambda \|d_i \odot c_i\|^2 \\ &s.t.\ 1^T c_i = 1, \quad \forall i \end{aligned} \tag{6}$$

where $C = \{c_1, c_2, \ldots, c_N\}$ refers to the coding coefficient vector, which is calculated by the constraint condition;

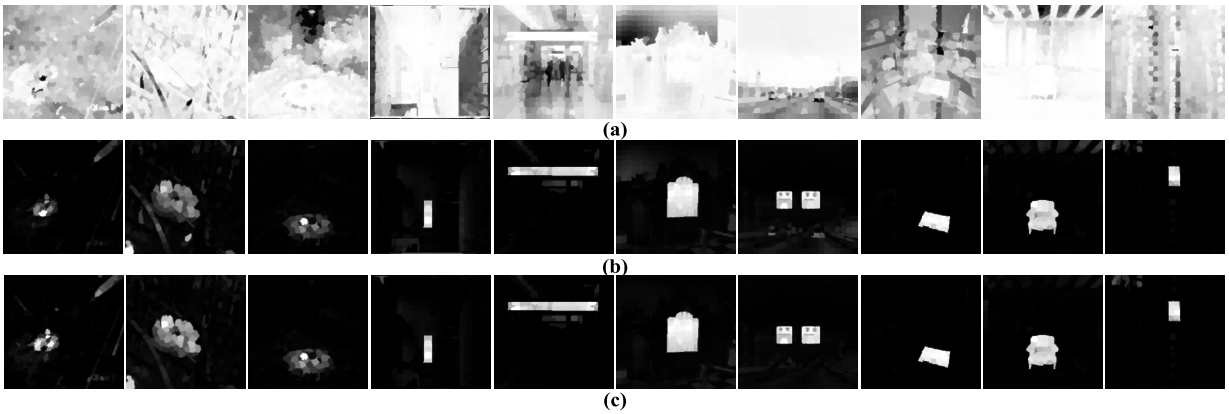


FIGURE 4. Examples of local saliency maps. (a) Saliency maps generated by the foreground codebook; (b) Saliency maps obtained by the background codebook; (c) Local saliency maps.

N stands for the number of superpixels; x_i is the local descriptor; i represents the i-th superpixel; B represents the visual codebooks; λ controls the balance between the penalty and regular terms; $\odot$ denotes the elementwise complex multiplication operation; $1^T c_i = 1$ represents that the coding is translation invariant; and d_i refers to the local adapter assigned to each basis vector, which is proportional to the similarity of x_i and can be calculated by equation (7).

$$d_i = \exp\left(\frac{dist(x_i, B)}{\sigma}\right) \tag{7}$$

$$dist(x_i, B) = [dist(x_i, b_1), \ldots, dist(x_i, b_D)]^T \tag{8}$$

where $dist(x_i, b_j)$ denotes the Euclidean distance between the local descriptor x_i and the visual codebook element b_j; σ stands for the weight of c_i, which is utilized to control the weight attenuation speed of local adapter; and D represents the dimension of the codebook.

Therefore, each local descriptor x_i can be described by the nearest code center.

In fact, equation (6) can be solved by equations (9) and (10).

$$c_i = 1/\left(C_i + \lambda \times tr(C_i)\right) \tag{9}$$

$$\widetilde{c_i} = c_i / 1^T c_i \tag{10}$$

where c_i is the coding coefficient vector of the i-th superpixel; $C_i = (B - 1x_i^T)(B - 1x_i^T)^T$ represents the covariance matrix of the input data, where B represents the visual codebooks; λ is a regularization parameter, and we set $\lambda = 0.1$ [7]; tr refers to the trace of the matrix; and $\widetilde{c_i}$ is merely an intermediate value.

By use of the reconstruction error, a local saliency map $S_l(r_i)$ can be calculated from the visual codebooks, as described in equation (11).

$$S_l(r_i) = \|x_i - B\widetilde{c_i}\|^2 \tag{11}$$

where x_i is the local descriptor; B represents the visual codebooks; and $\tilde{c}_i$ is the intermediate value.

Subsequently, the local saliency maps of the two codebooks can be generated by calculating the coded reconstruction error. In other words, the background reconstruction will be ideal when the reconstruction error equals 0; by contrast, the worse the reconstruction error is, the better the foreground reconstruction is.

Therefore, for local encoding foreground and background codebooks, their saliency values of superpixels are opposite and equal to their corresponding reconstruction error. By multiplying saliency maps generated by the two visual codebooks, the local saliency map can be obtained, as described in equation (12).

$$S_l(r_i) = (1 - S_{lf}(r_i)) \times S_{lb}(r_i) \tag{12}$$

where $S_l(r_i)$ represents the local saliency map of the superpixel region r_i; $S_{lf}(r_i)$ represents the foreground saliency map of the superpixel region r_i; and $S_{lb}(r_i)$ denotes the background saliency map of the superpixel region r_i.

The three types of intermediate saliency maps are shown in Fig. 4.

In Fig. 4, it should be noted that each of the foreground saliency maps (Fig. 4(a)) is usually opposite to its corresponding actual salient object, and the local saliency maps have laid more emphasis on the details of the salient object (Columns 4-10 of Fig. 4(c)), such as the petals of the flower, the contents of the light box and the signpost.

C. MULTILAYER CELLULAR AUTOMATA

The global saliency map S_g and the local saliency map S_l can be merged into a final saliency map by multilayer cellular automata [32]. In the structure of cellular automata, each cell represents an image pixel, and its neighbors are regarded as pixels at the same position in other saliency maps.

If the saliency value of a pixel indicates the probability of the pixel belonging to foreground F, the probability can be expressed as $P(i \in F) = S_i$. Hence, the probability of the pixel belonging to background B is expressed as $P(i \in B) = 1 - S_i$.

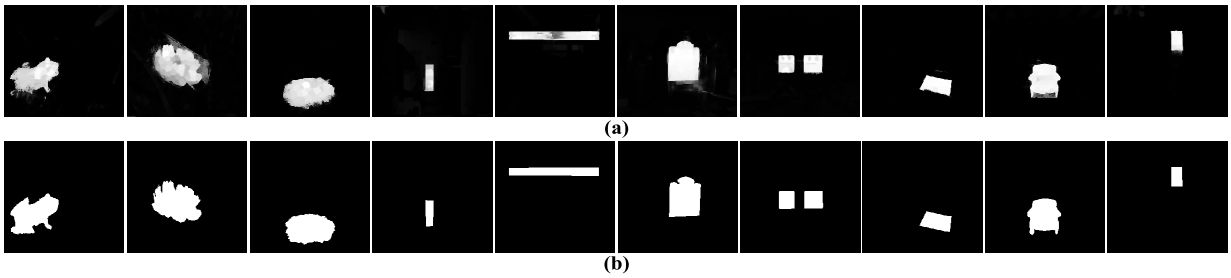


FIGURE 5. Examples of final saliency maps generated under the multilayer cellular automata framework. (a) The final saliency maps; (b) Ground truth.

Each of the global and local saliency maps is segmented by the OTSU method to obtain their foregrounds and backgrounds. If $\eta_i = +1$ indicates that the pixel i is segmented into the foreground, and $\eta_i = -1$ indicates that the pixel i is segmented into the background, the posterior probability can be calculated, as described in equation (13).

$$P\left(i \in F|\eta_j = +1\right) \propto P\left(i \in F\right) P\left(\eta_j = +1|i \in F\right) \quad (13)$$

where $\lambda = P\left(\eta_j = +1|i \in F\right)$ represents that pixel i belongs to the foreground, and its neighbor j also belongs to the foreground. Similarly, $\mu = P\left(\eta_j = -1|i \in B\right)$ denotes that pixel i belongs to the background, and its neighbor j also belongs to the background.

To avoid normalization, the proportion of the prior probability is defined as $\Delta\left(i \in F\right) = \frac{P(i\in F)}{P(i\in B)} = \frac{S_i}{1-S_i}$, and the proportion of the posterior probability is defined as $\Delta\left(i \in F|\eta_j = +1\right) = \frac{P(i\in F|\eta_j=+1)}{P(i\in B|\eta_j=+1)} = \frac{S_i}{1-S_i} \cdot \frac{\lambda}{1-\mu}$. The logarithmic operation for the proportion of posterior probability is then obtained, as described in equation (14).

$$l\left(i \in F|\eta_j = +1\right) = l\left(i \in F\right) + \ln\left(\frac{\lambda}{1-\mu}\right) \quad (14)$$

The proportions of prior and posterior probabilities are defined as:

$$\Delta\left(i \in F\right) = \frac{S_i^t}{1 - S_i^t} \quad (15)$$

$$\Delta\left(i \in F|\eta_j = +1\right) = \frac{S_i^{t+1}}{1 - S_i^{t+1}} \quad (16)$$

where S_i^t represents the saliency value of pixel i at the t-th iteration. Then, the next state is updated by the current state:

$$l\left(S_m^{t+1}\right) = l\left(S_m^t\right) + \sum_{k=1,k\neq m}^{M} sign\left(S_k^t - \gamma_k \cdot 1\right) \cdot \ln\left(\frac{\lambda}{1-\lambda}\right) \quad (17)$$

where $S_m^t = \left[S_{m1}^t, S_{m2}^t, \ldots S_{mH}^t\right]^T$ refers to the saliency values vector of all cells within the m-th saliency map at the t-th iteration, and $S_k^t = \left[S_{k1}^t, S_{k2}^t, \ldots S_{kH}^t\right]^T$ stands for the saliency values vector of all cells within the k-th saliency map at the t-th iteration, where H denotes the number of cells within an image; M denotes the number of layers of the cellular automaton; γ_k is the adaptive threshold of the OTSU algorithm for the k-th saliency map; and $\ln\left(\frac{\lambda}{1-\lambda}\right) = 0.15$ [32] represents that if the neighbor belongs to foreground, it needs to increase its own saliency value.

After N-step updates, the final saliency map is obtained:

$$S^N = \frac{1}{M}\sum_{m=1}^{M} S_m^N \quad (18)$$

where $M = 2$ because we fuse the following two maps: global and local saliency maps.

The examples of final saliency maps are shown in Fig. 5.

In fact, there are three cases of saliency detection in Fig. 5: 1) the global saliency detection performs well (Columns 1-3 of Fig. 3(b)), while the local saliency detection does not perform well (Columns 1-3 of Fig. 4(c)); 2) the global saliency detection does not perform well (Columns 4-6 of Fig. 3(b)), but the local saliency detection performs well (Columns 4-6 of Fig. 4(c)); and 3) both the global (Columns 7-10 of Fig. 3(b)) and local (Columns 7-10 of Fig. 4(c)) saliency detections perform well. In all cases, the final saliency detection under the multilayer cellular automata framework always performs well (Fig. 5(a)).

IV. EXPERIMENTAL RESULTS AND ANALYSES

A. EXPERIMENTAL SETUP

Experimental Platform. *CPU specification*: Intel Xeon E5-2650 v2 (2.6 GHz, 20 MB cache, 8 cores); *MEMORY specification*: 64 GB; *GPU specification*: NVIDIA Tesla K40M (12 GB).

SOFTWARE used: operating system, Ubuntu release 18.2; deep learning platform, TensorFlow version 1.2.0; data visualization tool, Matplotlib release 2.2.0; programming language, Python version 3.

Our method was evaluated on the MSRA 10K, ECSSD, DUT-OMRON, HKU-IS, THUR 15K and XPIE datasets [9], [33]–[37]. The **MSRA 10K** dataset [33] consists of 10,000 images: 6,000 training, 800 validation and 3,200 test images [38]. The **ECSSD** dataset [34] contains 1,000 natural images with complex structures: 600 training, 80 validation and 320 test images [38]. The **DUT-OMRON** dataset [35] includes 5,168 challenging images: 3,500 training, 468 validation and 1,200 test images [38]. It should be noted that each image has its corresponding ground truth in

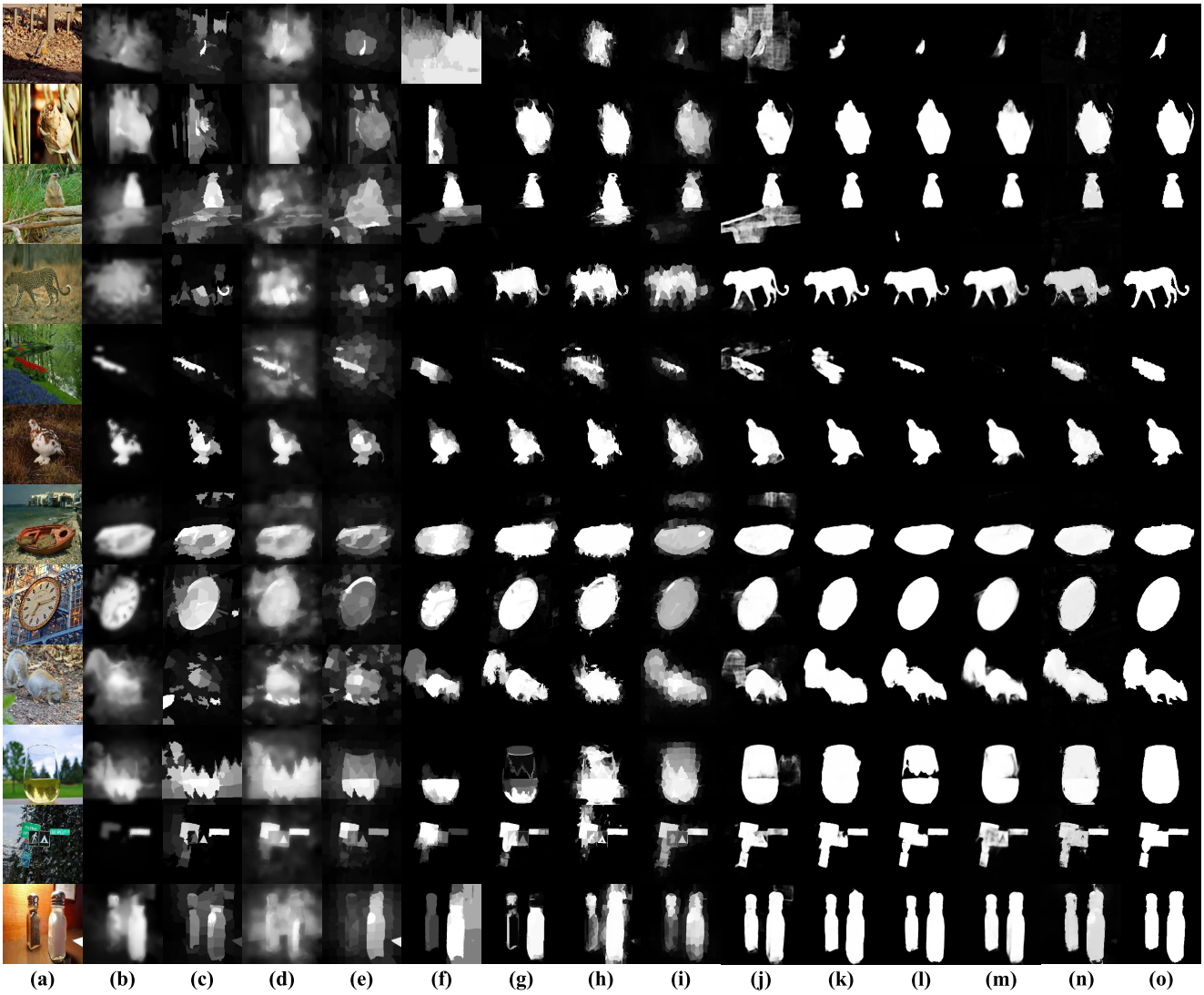


FIGURE 6. Saliency maps of related methods. (a) Original images; (b) MAP [3]; (c) RBD [4]; (d) BL [5]; (e) GL [7]; (f) LEGS [8]; (g) MDF [9]; (h) ELD [10]; (i) MTDS [11]; (j) UCF [12]; (k) FSN [13]; (l) DSS [14]; (m) PAGRN [15]; (n) Ours; (o) ground truth.

the above three datasets. To verify the generalization abilities of our method, the evaluations were further performed on the following three datasets: the **HKU-IS** dataset [9], containing 4,447 images with high-quality pixelwise annotations; the **THUR 15K** dataset [36], including 6,232 categorized images; and the **XPIE** dataset [37], consisting of 10,000 images with complex scenes.

The initial hyperparameters of our network are set the same as those of the VGG-16 network, and the number of training iterations of the basic network is set to 75,000 times. Then, our network with the skip connection and edge reinforcement penalty added is trained for 100,000 iterations. In addition, the input image was resized to 224 × 224 pixels.

B. EVALUATION METRICS

In the precision and recall (P-R) curve [39], the adaptive thresholds were set to the range [0, 255]. Each image itself has only one adaptive threshold. When the saliency value of a pixel is greater than the threshold, the value of the pixel was set to 255; otherwise, it was set to 0.

The adaptive threshold TH of an image is calculated as follows:

$$TH = \frac{2}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} S(x, y) \tag{19}$$

where W and H are the width and height of the given image, respectively, and $S(x, y)$ is the saliency value of pixel (x, y).

Therefore, *Precision*, *Recall*, and F-measure can be calculated by equations (20)-(22).

$$Precision = |S \cap T| / |S| \tag{20}$$

$$Recall = |S \cap T| / |T| \tag{21}$$

$$F_\beta = \frac{(1 + \beta^2) \times Precision \times Recall}{\beta^2 \times Precision + Recall} \tag{22}$$

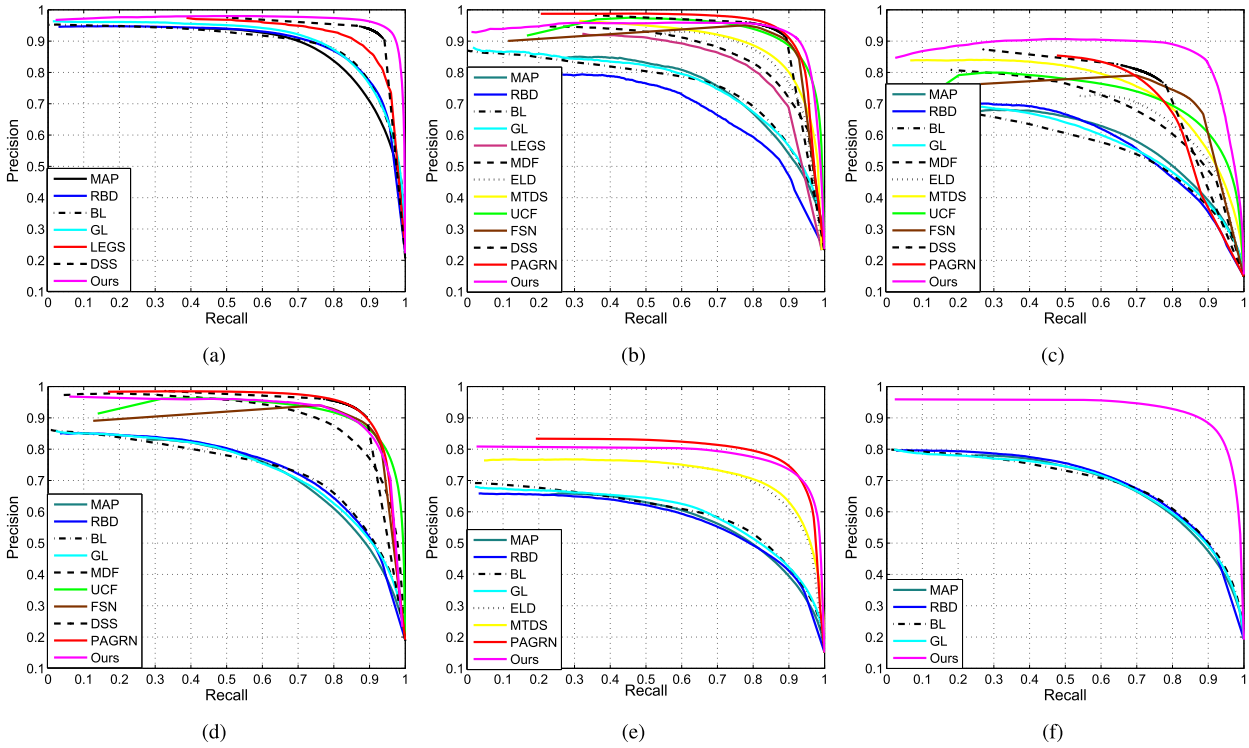


FIGURE 7. The P-R curves of related methods on six datasets. (a) MSRA 10K dataset [33]; (b) ECSSD dataset [34]; (c) DUT-OMRON dataset [35]; (d) HKU-IS dataset [9]; (e) THUR 15K dataset [36]; (f) XPIE dataset [37].

where S is the saliency map obtained from the binary mask using the adaptive threshold; T is the ground [39]; and we set $\beta^2 = 0.3$, which can adjust the balance between *Precision* and *Recall* and obtain the F-measure properly.

The above evaluation metrics are fair to those methods that can detect the salient object of the foreground well; however, they are unfair to those methods that successfully detect the salient object of background [39].

Therefore, the mean absolute error (MAE) [39] is adopted to further evaluate our method, as described in equation (23).

$$MAE = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |S(x, y) - T(x, y)| \qquad (23)$$

where W and H are the width and height of the given saliency map S, respectively; $S(x, y)$ is the saliency value of pixel (x, y); and $T(x, y)$ is the ground truth of pixel (x, y).

C. QUALITATIVE ANALYSIS

Some representative visual saliency detection results are shown in Fig. 6.

As seen in Fig. 6, saliency maps generated by our method can highlight the salient objects of challenging images well, such as images with low contrast (Rows 1-4, 6 and 9 of Fig. 6(n)), and images with complex background (Rows 5, 7 and 8 of Fig. 6(n)), and images with multiple objects (Rows 11, 12 of Fig. 6(n)). Consider the images of Row 1, for example, where most of the related methods mistake the ground as the salient object of the bird (Row 1 of Fig. 6(b)-(j)), or detect a broken bird (Row 1 of Fig. 6(k)-(m)). In reality, our method can effectively extract global image features because of the advantages of the convolutional network for image feature extraction, as well as its use of high-level information to enrich low-level information and reduce the ambiguous boundary of the salient object. To overcome the shortcomings of the deep network model (the defect, e.g., Rows 2, 5 and 10 of Fig. 6(f)-(m)), we adopt the strategy of integrating the deep model with the traditional LLC model, namely, using global results to guide local coding to extract local information of the salient object. In addition, both global and local saliency maps can be fused under the framework of multilayer cellular automata so that the fusion results contain clear and uniform contour information and local details. For more support images, please also see Rows 2, 5 and 10 of Fig. 6(n).

D. QUANTITATIVE ANALYSIS

The P-R and F-measure curves and their corresponding bar charts of related methods are shown in Figs. 7, 8 and 9, respectively.

From Figs. 7, 8 and 9 (a), (c) and (f), we can see that the *precision-recall* and F-measure curves and scores of our method are well above state-of-the-art methods, which indicate that our method is better than related methods. In Figs. 7 and 8 (b), (d) and (e), the P-R and F-measure curves of our method mainly intersect with those of the DSS and PAGRN methods, which show that the two state-of-the-art methods are comparable to our method (also see Fig. 9 (b), (d) and (e)). With respect to our method, the global

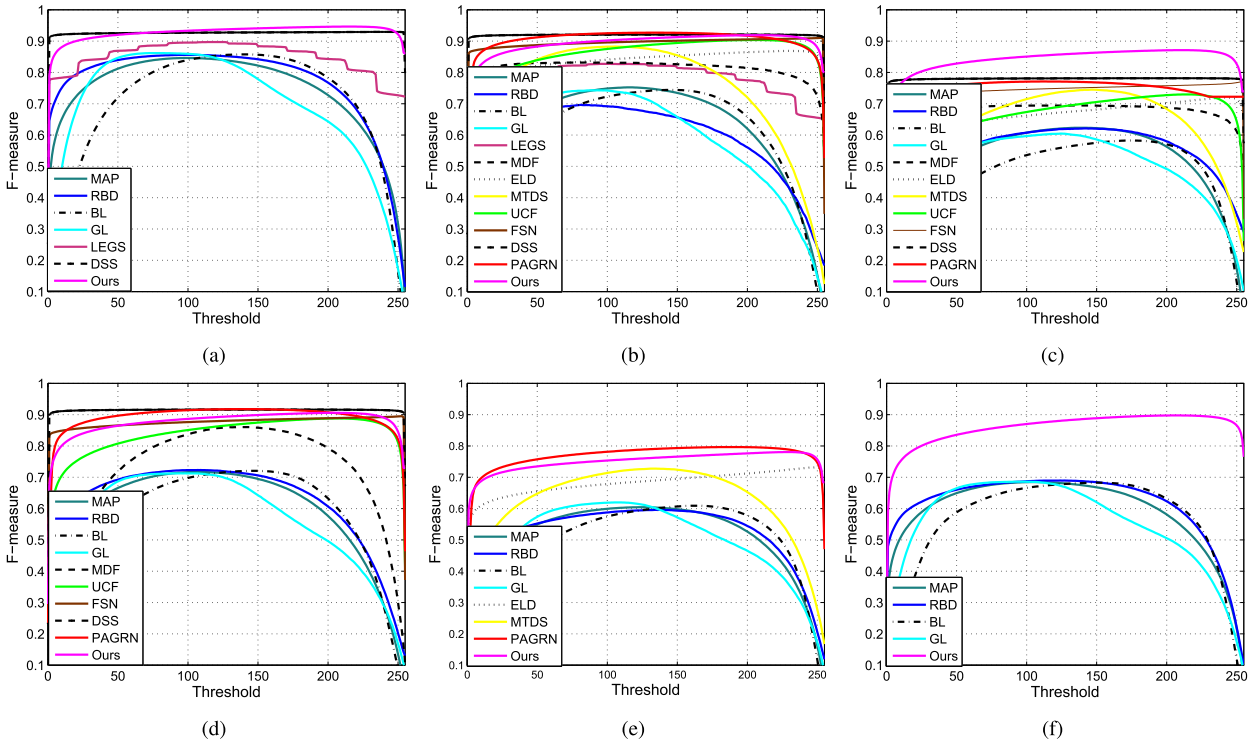


FIGURE 8. The *F*-measure curves of related methods on six datasets. (a) MSRA 10K dataset [33]; (b) ECSSD dataset [34]; (c) DUT-OMRON dataset [35]; (d) HKU-IS dataset [9]; (e) THUR 15K dataset [36]; (f) XPIE dataset [37].

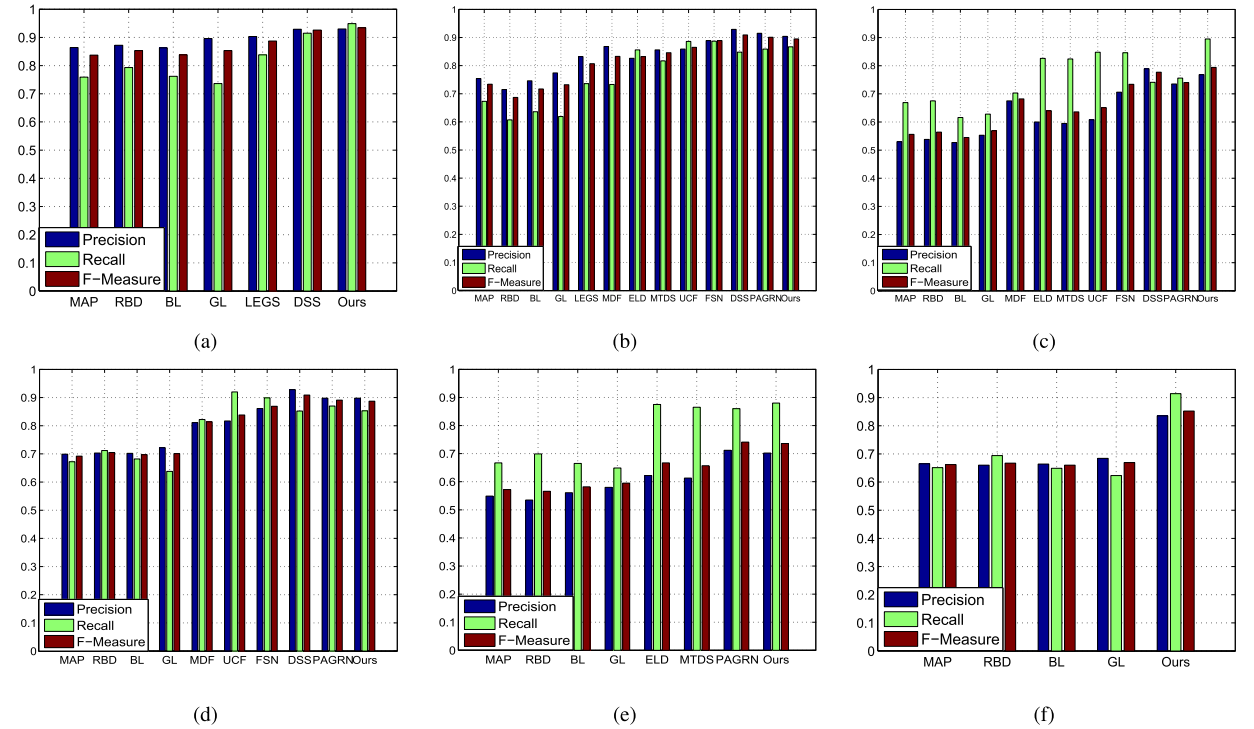


FIGURE 9. The average *precision*, *recall* and *F*-measure scores of related methods on six datasets. (a) MSRA 10K dataset [33]; (b) ECSSD dataset [34]; (c) DUT-OMRON dataset [35]; (d) HKU-IS dataset [9]; (e) THUR 15K dataset [36]; (f) XPIE dataset [37].

CNN joined with skip connections can make up for the loss of high-level information during information transmission, and the boundary penalty term can enhance the object boundary (also see Row 2 of Fig. 6 (n)). Furthermore, on the basis of global detection, locality-constrained coding performed on the superpixels can maintain the local detail

TABLE 1. Evaluation metrics of related methods on six datasets.

		MAP [3]	RBD [4]	BL [5]	GL [7]	LEGS [8]	MDF [9]	ELD [10]	MTDS [11]	UCF [12]	FSN [13]	DSS [14]	PAGRN [15]	**Ours**
MSRA 10K [33]	P	0.864	0.872	0.863	0.896	0.903	-	-	-	-	-	0.929	-	**0.930**
	R	0.759	0.793	0.762	0.736	0.838	-	-	-	-	-	0.915	-	**0.949**
	F_β	0.837	0.853	0.838	0.853	0.887	-	-	-	-	-	0.926	-	**0.934**
	MAE	0.127	0.108	0.162	0.145	0.079	-	-	-	-	-	0.030	-	**0.046**
ECSSD [34]	P	0.754	0.715	0.746	0.774	0.832	0.868	0.826	0.856	0.859	0.889	0.929	0.915	**0.904**
	R	0.673	0.607	0.636	0.619	0.736	0.733	0.856	0.817	0.886	0.887	0.848	0.859	**0.867**
	F_β	0.734	0.687	0.717	0.732	0.807	0.833	0.832	0.846	0.865	0.889	0.909	0.901	**0.895**
	MAE	0.186	0.179	0.220	0.200	0.118	0.104	0.078	0.121	0.068	0.053	0.052	0.064	**0.067**
DUT-OMRON [35]	P	0.530	0.538	0.527	0.553	-	0.675	0.600	0.595	0.608	0.706	0.789	0.735	**0.768**
	R	0.669	0.675	0.616	0.628	-	0.703	0.826	0.824	0.848	0.846	0.741	0.756	**0.895**
	F_β	0.556	0.564	0.545	0.569	-	0.682	0.640	0.636	0.651	0.734	0.777	0.740	**0.794**
	MAE	0.180	0.144	0.242	0.183	-	0.091	0.090	0.120	0.120	0.065	0.062	0.070	**0.054**
HKU-IS [9]	P	0.699	0.703	0.702	0.722	-	0.811	-	-	0.817	0.861	0.928	0.898	**0.898**
	R	0.672	0.712	0.682	0.638	-	0.822	-	-	0.920	0.899	0.852	0.870	**0.853**
	F_β	0.692	0.705	0.697	0.701	-	0.814	-	-	0.838	0.869	0.909	0.891	**0.887**
	MAE	0.170	0.142	0.207	0.177	-	0.095	-	-	0.061	0.044	0.039	0.047	**0.044**
THUR 15K [36]	P	0.549	0.535	0.561	0.580	-	-	0.622	0.613	-	-	-	0.712	**0.702**
	R	0.667	0.699	0.665	0.649	-	-	0.875	0.865	-	-	-	0.860	**0.880**
	F_β	0.572	0.566	0.582	0.595	-	-	0.667	0.657	-	-	-	0.741	**0.736**
	MAE	0.175	0.150	0.218	0.173	-	-	0.098	0.116	-	-	-	0.070	**0.072**
XPIE [37]	P	0.665	0.660	0.664	0.684	-	-	-	-	-	-	-	-	**0.835**
	R	0.651	0.694	0.649	0.623	-	-	-	-	-	-	-	-	**0.914**
	F_β	0.662	0.667	0.660	0.669	-	-	-	-	-	-	-	-	**0.852**
	MAE	0.178	0.148	0.207	0.182	-	-	-	-	-	-	-	-	**0.049**
I	P	0.677	0.671	0.677	0.702	0.868	0.785	0.683	0.688	0.761	0.819	0.894	0.815	**0.840**
	R	0.682	0.697	0.668	0.649	0.787	0.753	0.852	0.835	0.885	0.877	0.839	0.836	**0.893**
	F_β	0.676	0.674	0.673	0.687	0.847	0.776	0.713	0.713	0.785	0.831	0.880	0.818	**0.850**
	MAE	0.169	0.145	0.209	0.177	0.099	0.097	0.089	0.119	0.083	0.054	0.046	0.063	**0.055**
	III	6	6	6	6	2	3	3	3	3	3	4	4	**6**
II	P	0.682				0.789								0.854
	R	0.674				0.833								0.838
	F_β	0.678				0.795								0.849
	MAE	0.175				0.081								0.055

*Note that I denotes the total average metrics of each method; II represents the total average metrics of each category method, whose last entries consist of the total average metrics of DSS and PAGRN (the second best methods), not ours; and III refers to the number of statistical samples. In addition, each of the values within the table is the rounding number of its corresponding original data, which cannot merely be calculated by other values presented in the table.

information of the global saliency map well, and the region-level detection can maintain more spatial information as well (also see Row 10 of Fig. 6(n)). In addition, under the multilayer cellular automata framework, its propagation-based approach can explore the intrinsic connections of similar regions to improve the saliency detection results to some extent, namely, the Bayesian theory can effectively combine global and local information (also see Columns 1-6 of Figs. 3 (b), 4(c) and 5(a)). With regard to those traditional detection methods (e.g., MAP [3], RBD [4], BL [5] and GL [7]) depending on the contrast calculation and prior knowledge, for those somewhat complex images, their simple contrast calculations and limited prior cues make it difficult to achieve satisfactory performance (e.g., Figs. 7-9 (a)-(f); or Fig. 6 (b)-(e)). Regarding the deep saliency detection models based on the CNN, such as LEGS [8], MDF [9], ELD [10], MTDS [11], UCF [12], FSN [13], DSS [14] and PAGRN [15], although they have more powerful learning ability, their pooling operation of the CNN has compressed the input data severely, which can cause the loss of high-level information in some certain images (e.g., Rows 1, 2, 5 and 10 of Fig. 6 (f)-(m)). Moreover, the special structure of their local receptive field can result in the blurring of salient objects and has better performance with the assistance of traditional methods in saliency detection (also see Row 5 of Fig. 6 (m) and (n); or Rows 1, 3, 5, 7-10, 12 of Fig. 6 (j) and (n)).

To fully verify the saliency detection performance of related methods, the evaluation metrics of related methods are listed in Table 1.

From Table 1, we can see obviously that our method is better than those traditional saliency methods, MAP [3], RBD [4], BL [5] and GL [7]. On the large-scale dataset of MSRA 10K, containing 10,000 sample images, our method performs better than DSS. On the small-scale dataset of ECSSD, containing only 1,000 sample images, our method performs seemingly worse than the second-best methods of DSS and PAGRN; however, we argue here that the scale of ECSSD is so small that a fair and full comparison can hardly be made indeed. On the mid-scale dataset of DUT-OMRON, containing 5,168 sample images, our method performs better than DSS and other related methods. Nevertheless, on the mid-scale datasets of HKU-IS and THUR 15K, containing 4,447 and 6,232 sample images, respectively, the DSS and

PAGRN methods are seemingly comparable to our method. To further evaluate our model, the total average metrics of each method are calculated and listed in Row I of Table 1, from which we can see that our method performs better than PAGRN whether in terms of *precision*, *recall*, F-measure scores or *MAE*; however, DSS is still seemingly comparable to our method. In our opinion, the reason is that the evaluation metrics of DSS on the mid- and large-scale datasets of THUR 15K and XPIE are not involved in the calculation because they cannot yet be achieved temporally. Therefore, the total average metrics of the DSS and PAGRN methods are calculated and listed in the last Column of Row II in Table 1, from which we can see that our method performs better than the second best methods, although their evaluation metrics on the largest-scale dataset of XPIE are still missed. From Row II of Table 1, we may draw the conclusion that our method is better than both the traditional and CNN methods in saliency detection.

The detailed reasons are as follows: 1) The convolutional neural network, sharing a set of parameters, is invariant to translation, rotation and scale for extracting image features. Furthermore, with the increase in the number of convolutional layers, the learned image features are more global, but it also causes high-level information loss. By introducing the skip connections into our method, our network can extract rich global features. Moreover, we employ the edge penalty term to make the salient object boundary more clear. Last but not least, the supervised iterative training method of our deep network can detect the complete edge of the salient object (also see Row 10 of Fig. (6)). 2) The image is encoded locally and sparsely, and the reconstruction error is minimized in generating the local salient map, which ensures that the details of the salient object are detected as much as possible (also see Rows 1 and 2 of Fig. (6)). 3) The multilayer cellular automata, with the stability of posterior probabilities and the dynamic effect between the multilayer cellular automata, can fuse the global and local information together effectively. In short, our method can not only detect the complete edge of a salient object well but also has a powerful representation ability of the salient object details (also see Columns 1-6 of Figs. 3(b), 4(c) and 5(a)). 4) Of course, our improved deep network based on the encoder-decoder structure has a lightweight structure, and it has no more advantages than some certain CNN methods in some saliency detections (also see Rows 4, 6 and 12 of Fig. 6(m) and (n)).

To explore the contribution of each stage within our method, we carried out several experiments step by step on the MSRA 10K [33], ECSSD [34] and DUT-OMRON [35] datasets. The average F_β values of each stage are listed in Table 2.

In Table 2, the average F_β values at the 1^{st}, 2^{nd} and the 3^{rd} stage of our method are calculated from their corresponding global, local (optimized by single-layer cellular automata) and final saliency maps, respectively. From Table 2, we can see that the cellular automata, which fuses the CNN and LLC, achieves the best performance. It improves about 1% and 5% compared with the global and local saliency, respectively.

TABLE 2. The average F_β values of each stage.

	Global saliency	Local saliency	Cellular automata
MSRA 10K	0.928	0.887	0.934
ECSSD	0.890	0.825	0.895
DUT-OMRON	0.789	0.713	0.794

V. CONCLUSION

To extract the salient object of natural images with low contrast and complex backgrounds, a lightweight but powerful network method is proposed. 1) The global saliency map is generated by the CNN-based encoder-decoder model. To begin with, the VGG-16 network was utilized as the encoder, and the symmetric network was adopted as the decoder. Furthermore, the network employed the skip connections and edge enhancement term to enrich low-level information and enhance target boundaries. Finally, our network was trained iteratively via the loss function. 2) The local saliency map is obtained by the locality-constrained linear coding method. First, the foreground and background codebooks were segmented from the global saliency map. Second, the two codebooks were encoded by the locality-constrained linear coding method, and the two corresponding foreground and background saliency maps were generated. Finally, by multiplying the two saliency maps, the local saliency map was obtained. 3) The final saliency map is generated under the multilayer cellular automata framework. First, the global and local saliency maps were regarded as one-layer cellular automata. Subsequently, the multilayer cellular automata framework merged the global and local saliency maps into the final saliency map, which can maintain not only global information but also local information. The experimental results show that the average F-measure of our method on the MSRA 10K, ECSSD, DUT-OMRON, HKU-IS, THUR 15K and XPIE datasets reaches up to 93.4%, 89.5%, 79.4%, 88.7%, 73.6% and 85.2%, respectively, and the corresponding *MAE* is only 0.046, 0.067, 0.054, 0.044, 0.072 and 0.049. Ultimately, all of our findings prove that our method is good at saliency detection of natural images, especially images with low contrast and complex backgrounds. Further research is required to improve the structure of the deep CNN and enable it to more suitably extract multiscale semantic features.

REFERENCES

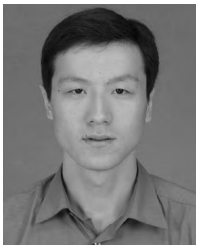
[1] R. Margolin, A. Tal, and L. Zelnik-Manor, "What makes a patch distinct?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Portland, OR, USA, Jun. 2013, pp. 1139–1146.

[2] C. Yang, L. Zhang, and H. Lu, "Graph-regularized saliency detection with convex-hull-based center prior," *IEEE Signal Process. Lett.*, vol. 20, no. 7, pp. 637–640, Jul. 2013.

[3] J. Sun, H. Lu, and X. Liu, "Saliency region detection based on Markov absorption probabilities," *IEEE Trans. Image Process.*, vol. 24, no. 5, pp. 1639–1649, May 2015.

[4] W. Zhu, S. Liang, Y. Wei, and J. Sun, "Saliency optimization from robust background detection," in *Proc. 27th IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, Jun. 2014, pp. 2814–2821.
[5] N. Tong, H. Lu, and M.-H. Yang, "Salient object detection via bootstrap learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 1884–1892.
[6] X. Shen and Y. Wu, "A unified approach to salient object detection via low rank matrix recovery," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 853–860.
[7] N. Tong, H. Lu, Y. Zhang, and X. Ruan, "Salient object detection via global and local cues," *Pattern Recognit.*, vol. 48, no. 10, pp. 3258–3267, Oct. 2015.
[8] L. Wang, H. Lu, X. Ruan, and M.-H. Yang, "Deep networks for saliency detection via local estimation and global search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 3183–3192.
[9] G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 5455–5463.
[10] G. Lee, Y.-W. Tai, and J. Kim, "Deep saliency with encoded low level distance map and high level features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 660–668.
[11] X. Li *et al.*, "DeepSaliency: Multi-task deep neural network model for salient object detection," *IEEE Trans. Image Process.*, vol. 25, no. 8, pp. 3919–3930, Aug. 2016.
[12] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin, "Learning uncertain convolutional features for accurate saliency detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, Oct. 2017, pp. 212–221.
[13] X. Chen, A. Zheng, J. Li, and F. Lu, "Look, perceive and segment: Finding the salient objects in images via two-stream fixation-semantic CNNs," in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, Oct. 2017, pp. 1050–1058.
[14] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. H. S. Torr, "Deeply supervised salient object detection with short connections," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Jul. 2017, pp. 3203–3212.
[15] X. Zhang, T. Wang, J. Qi, H. Lu, and G. Wang, "Progressive attention guided recurrent network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake, UH, USA, Jun. 2018, pp. 714–722.
[16] M. Zhang, Y. Wu, Y. Du, L. Fang, and Y. Pang, "Saliency detection integrating global and local information," *J. Vis. Commun. Image Represent.*, vol. 53, pp. 215–223, May 2018.
[17] P. Wang, G. Tian, and H. Chen, "A saliency detection model combined local and global features," in *Proc. Chin. Autom. Congr. (CAC)*, Jinan, China, Oct. 2017, pp. 2863–2870.
[18] A. Wang, M. Wang, G. Pan, and X. Yuan, "Salient object detection with high-level prior based on Bayesian fusion," *IET Comput. Vis.*, vol. 11, no. 3, pp. 199–206, 2017.
[19] N. Liu and J. Han, "A deep spatial contextual long-term recurrent convolutional network for saliency detection," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3264–3274, Jul. 2018.
[20] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Francisco, CA, USA, Jun. 2010, pp. 2528–2535.
[21] M. D. Zeiler, G. W. Taylor, and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Barcelona, Spain, Nov. 2011, pp. 2018–2025.
[22] M. D. Zeiler and F. Rob, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.* Zurich, Switzerland: Springer, Nov. 2013, pp. 818–833.
[23] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 1520–1528.
[24] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, San Francisco, CA, USA, Jun. 2010, pp. 3360–3367.
[25] Y.-H. Wu, W.-L. Ku, W.-H. Peng, and H.-C. Chou, "Global image representation using locality-constrained linear coding for large-scale image retrieval," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Melbourne, VIC, Australia, Jun. 2014, pp. 766–769.
[26] S. Wang, R. Cao, and L. Cao, "Locality-constrained linear coding combined with saliency similarity," *J. Huazhong Univ. Sci. Technol.*, vol. 45, no. 6, pp. 21–25 and 47, 2017.
[27] Y. Qin, M. Feng, H. Lu, and G. W. Cottrell, "Hierarchical cellular automata for visual saliency," *Int. J. Comput. Vis.*, vol. 126, no. 7, pp. 751–770, 2018.
[28] L. Zhang, J. Ai, B. Jiang, H. Lu, and X. Li, "Saliency detection via absorbing Markov chain with learnt transition probability," *IEEE Trans. Image Process.*, vol. 27, no. 2, pp. 987–998, Feb. 2018.
[29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556
[30] V. Turchenko and A. Luczak, "Creation of a deep convolutional auto-encoder in caffe," in *Proc. 9th IEEE Int. Conf. Intell. Data Acquisition Adv. Comput. Syst., Technol. Appl. (IDAACS)*, Bucharest, Romania, vol. 2, Sep. 2017, pp. 651–659.
[31] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.
[32] Y. Qin, H. Lu, Y. Xu, and H. Wang, "Saliency detection via cellular automata," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 110–119.
[33] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, Mar. 2015.
[34] J. Shi, Q. Yan, L. Xu, and J. Jia, "Hierarchical image saliency detection on extended CSSD," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 717–729, Apr. 2016.
[35] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Portland, OR, USA, Jun. 2013, pp. 3166–3173.
[36] M.-M. Cheng, N. J. Mitra, X. Huang, and S.-M. Hu, "SalientShape: Group saliency in image collections," *Vis. Comput.*, vol. 30, no. 4, pp. 443–453, 2014.
[37] C. Xia, J. Li, X. Chen, A. Zheng, and Y. Zhang, "What is and what is not a salient object? Learning salient object detector by ensembling linear exemplar regressors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, Jul. 2017, pp. 4142–4150.
[38] N. Liu and J. Han, "DHSnet: Deep hierarchical saliency network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, Jun. 2016, pp. 678–686.
[39] A. Borji, M.-M. Cheng, H. Jiang, and J. Li, "Salient object detection: A benchmark," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5706–5722, Dec. 2015.

YIHANG LIU was born in Henan, Xinxiang, China, in 1999. She is currently pursuing the bachelor's degree in computer science and technology with Henan Normal University.



PEIYAN YUAN received the B.S. degree, in 2001, and the M.S. and Ph.D. degrees from the Wuhan University of Technology, Wuhan, China, and the Beijing University of Posts and Telecommunications, Beijing, China, respectively, both in computer science. He is currently an Associate Professor of computer science with Henan Normal University. His research interests include future networks and distributed systems. He has authored or coauthored over 40 papers and one book in these fields. He is a Senior Member of the CCF and a member of the ACM. He was a recipient of the National Scholarship for Ph.D. students from the Ministry of Education of China, in 2012, and a recipient of the Best Paper Award of the IEEE CSE, in 2014.