Knowledge-Based Systems 223 (2021) 107033

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

A dual-stream framework guided by adaptive Gaussian maps for interactive image segmentation

Zongyuan Ding [a], Tao Wang [a,*], Quansen Sun [a,*], Qiongjie Cui [a], Fuhua Chen [b]

[a] School of Computer Science and Engineering, Nanjing University of Science and Technology, No.200, Xiao Ling Wei Street, Nanjing 210094, China
[b] Department of Physical Science & Mathematics, West Liberty University, 208 University Drive College Union Box 190, West Liberty 26074, USA

ARTICLE INFO

Article history:
Received 13 January 2021
Received in revised form 13 March 2021
Accepted 7 April 2021
Available online 9 April 2021

Keywords:
Interactive image segmentation
Dual-stream neural network
Gated convolution
Adaptive Gaussian map

ABSTRACT

Efficiently embedding user-annotation is a key issue for deep interactive image segmentation. In this paper, we propose a dual-stream framework guided by adaptive Gaussian maps for interactive image segmentation. The network architecture consists of two branches: a traditional fully convolutional neural network that produces coarse segmentation results, and an interactive shape stream that produces target boundary information by integrating user-annotations. The boundary information combined with user-intention can suppress the feature response outside the target, which boosts the performance of coarse segmentation. Additionally, we develop an adaptive Gaussian map with distinct variances to encode user-annotations, which promotes sensitivity to details by adaptively adjusting the affected region of the annotations. Specifically, when the distance between two interactions is smaller than a threshold, we shrink the Gaussian variance of these interactions to enhance the perception of details, thereby improving the segmentation performance of the details. Extensive experiments show that our algorithm effectively reduces the burden of user interaction under the restriction of clear target boundaries and excels at fine-tuning the details with the adaptive Gaussian maps.

1. Introduction

Image segmentation is a challenging task that separates and extracts the regions of interest (foreground) from other parts (background) of an image. It is the basis of many other high-level computer vision tasks. Due to the dependence of the ideal segmentation on human subjective visual perception, fully automatic image segmentation is obviously not practical for all applications. Different from fully automatic image segmentation, interactive image segmentation algorithms can segment an object in line with the user's demands by introducing a few interactions, and have a wide application in medical diagnosis and image editing.

Traditional interactive image segmentation algorithms [1–4] exploit low-level features (such as textures, and other hand-crafted features) for the segmentation, and these low-level features lack robustness to complex scenes, thereby degenerating segmentation performance. To deal with this issue, many deep interactive image segmentation algorithms [5–14] have been proposed to improve the semantic perception with limited user interactions, thanks to the powerful feature representation ability of deep learning.

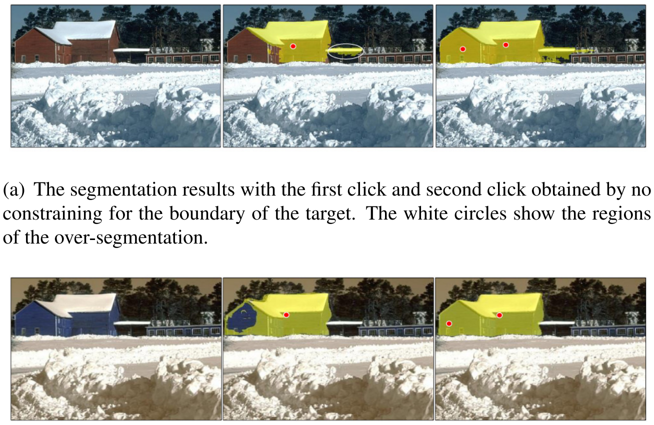
Deep interactive object selection (DOS) [5] is the first work to apply a deep learning algorithm to interactive image segmentation. DOS converts provided clicks to maps obtained by measuring the Euclidean distances between pixels and clicks with a fixed radius scale. The obtained interactive maps are then fed into a fully convolutional neural network (FCN) [15] to direct the segmentation. We denote the combination of an image and the interactive map in DOS as the early fusion strategy since these two kinds of information are concatenated at the beginning of the network. The early fusion strategy [5–7,9,11,13,14,16–18] may weaken the influence of user interactions on the final prediction results [10], which leads to more interactions need to achieve an ideal segmentation. Furthermore, the fixed affected region of each click in the interactive map also makes it difficult to distinguish the user's intention (correction or finetuning), because the affected region of close clicks may overlap with each other, making it difficult to further improve the performance of segmentation. Meanwhile, the distribution of features in the Euclidean distance based interactive map is different from that in preprocessed images, which is harmful to the training for the network. Therefore, an interactive map based on Gaussian distribution [7,8,13,19] is proposed to solve this dilemma. Nevertheless, there is still an

The code (and data) in this article has been certified as Reproducible by Code Ocean: (https://codeocean.com/). More information on the Reproducibility Badge Initiative is available at https://www.elsevier.com/physical-sciences-and-engineering/computer-science/journals.

[*] Corresponding authors.

E-mail addresses: dzyha2011@163.com (Z. Ding), wangtaoatnjust@163.com (T. Wang), sunquansen@njust.edu.cn (Q. Sun), cuiqiongjie@126.com (Q. Cui), fuhua.chen@westliberty.edu (F. Chen).

https://doi.org/10.1016/j.knosys.2021.107033
0950-7051/© 2021 Published by Elsevier B.V.



(a) The segmentation results with the first click and second click obtained by no constraining for the boundary of the target. The white circles show the regions of the over-segmentation.

(b) The segmentation results with the first click and second click obtained by constraining for the boundary of the target.

Fig. 1. The segmentation results with first two clicks by the traditional algorithm and our proposed algorithm (Red and blue dots represent the foreground and background user-clicks respectively).

issue that the segmentation is hard to be promoted further when close clicks are applied.

Most deep interactive segmentation algorithms input the color, texture, and shape information of an image into the network to extract features. These features are not conducive to improving the segmentation of the boundary as they contain very different types of information related to recognition, while improving the accuracy of boundary segmentation can also promote the segmentation result with the user's further interaction. As shown in Fig. 1(a), commonly used algorithms can easily lead to over-segmented, which is detrimental to the efficiency of the user's further interaction. By processing the shape information separately, the segmentation after the first interaction can frame the boundary of the target well, as illustrated in Fig. 1(b), so that the segmentation after the next interaction can be improved without introducing more errors. In recent years, the gated convolution commonly used in natural language processing [20] has been applied to deep image segmentation algorithms [21–23], which has effectively improved boundary segmentation. The Gated-Shape CNN (Gated-SCNN) [23] presents a dual-stream framework that explicitly utilizes a branch parallel to the regular stream (Encoder in the network) to extract shape information. The gated convolution is used to control the transmission of information between two streams, which greatly boosts the segmentation performance.

To this end, we propose a dual-stream framework guided by adaptive Gaussian maps for interactive image segmentation. First, the adaptive Gaussian maps are developed to represent user interaction. We build an interactive map by calculating the Gaussians between all pixels and clicks with adaptive variances. To be specific, we turn the Gaussian variances of two clicks to a small value when the distance between two clicks is smaller than a threshold, and thus the interactive map based on these two close clicks guides the network to concentrate on the details of the target. Then, inspired by Gated-SCNN [23], we develop an interactive shape stream to process the shape information of the desired object combined with user interaction parallel to the latest DeepLabV3+ framework (shown in Fig. 2). We utilize three sequential gated convolutions to control the information flow from the regular stream (encoder of the DeepLabV3+ [24]). Furthermore, through fusing interactive maps with the input of regular stream and interactive shape stream, the interactive shape stream can contain both interaction information and shape information, leading to a better boundary to tailor the coarse segmentation. In essence, the gated convolution is a self-attention mechanism. By applying a sigmoid function, the low-level features from the regular stream are normalized between 0 and 1, and then normalized features are multiplied by high-level features from the regular stream element by element, which is tantamount to assigning weight to the high-level features. Due to the locality of the low-level features, gated convolution strengthens the shape information of the high-level features. We name our framework Gated-iDeepLab. Finally, a new probability click loss function for training the network is built. Except for the binary cross-entropy loss, we constrain the difference between output and ground-truth with specific probabilities (supplied by interactive map), making the segmentation around the clicks more accurate.

We conduct extensive experiments on several image segmentation datasets. The ResNet50 [25] is chosen as the encoder (i.e., regular stream) of the DeepLabV3+ [24]. Compared with DOS [5], the mean number of clicks to reach the objective accuracy of our algorithm has been significantly reduced. Moreover, the segmentation after the first click using the proposed method has a clearer boundary than other interactive image segmentation algorithms. In this work, our main contributions are summarized as:

- We propose an interactive shape stream via fusing the interactive information to improve the accuracy of boundary segmentation, which relieves the burden of the user's interaction.
- We develop an adaptive Gaussian map to represent interactive information to refine the details of the desired object.
- A probability click loss function is built for training interactive image segmentation network.

2. Related work

Interactive image segmentation is the task to select objects in line with the needs of the user by supplying some cues (clicks, scribbled lines, or bounding boxes). Traditional interactive image segmentation includes intelligent scissors [26], graph cuts [27], level sets [28], and random walks [2]. The interactive image segmentation algorithms [29,30] derived from graph cut is most popular, which employ graph theory to map an image into a weighted undirected graph (pixels are treated as nodes). The image segmentation problem is regarded as a vertex division problem of the graph. Finally, the max-flow/min-cut [31] is applied to obtain the image segmentation. These conventional algorithms can obtain better segmentation results in some simply natural and medical images. Nevertheless, these algorithms are also difficult to achieve better accuracy when facing complex images.

In recent years, many deep-learning based semantic segmentation algorithms have sprung up [15,24,32–36]. These algorithms provide a better foundation for deep interactive image segmentation. The deep frameworks mainly take the form of encoder–decoder structures. Based on this structure, some tricks are designed to promote the performance of the segmentation network. For obtaining multi-scale features, the atrous convolution [36] has the ability to capture a larger receptive field of the convolution kernel without increasing the number of parameters, and the atrous spatial pyramid pooling (ASPP) stacks atrous convolution kernels of different rates in parallel to obtain multi-scale information gain [24,37]. The skip connection [15,32,33] relieves the phenomenon of gradient disappearing by transferring low-level features to the decoder blocks, and it helps the decoder capture

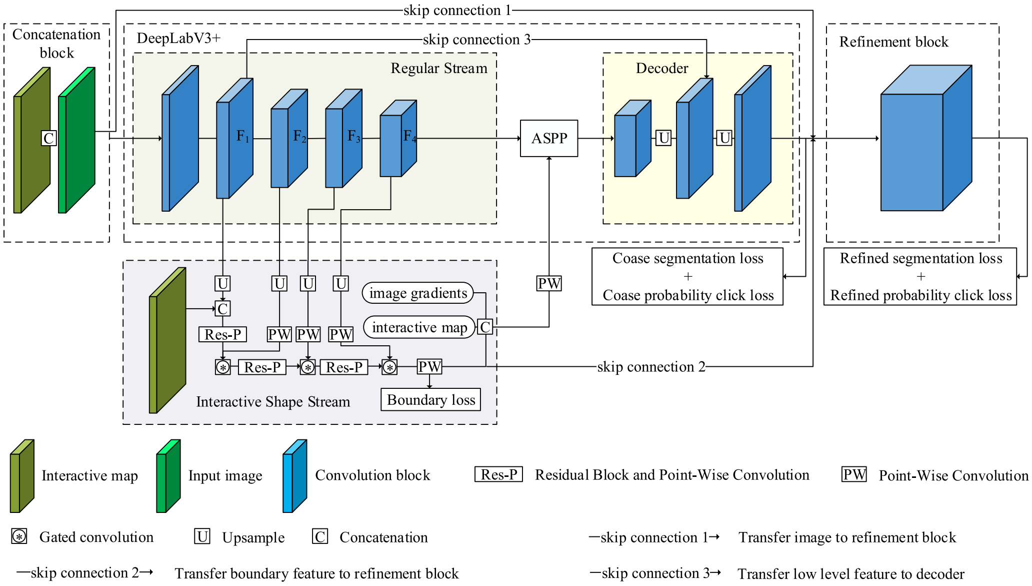


Fig. 2. Illustration of our proposed network. Our network consists of three parts: DeepLabV3+, interactive shape stream, and refinement block. DeepLabV3+ is composed of the regular stream (Encoder of DeepLabV3+), ASPP, and decoder. We select ResNet50 as the regular stream. The interactive shape stream extracts boundary information of the target object, which consists of sequential gated convolution, residual block, and point-wise convolution (Res-P). The ASPP receives features from the regular stream and interactive shape stream (concatenates image gradient and interactive map). The refinement block is composed of six consecutive convolutions. The original image, the output of the DeepLabV3+, and interactive shape stream are the input of the refinement block, and it fine-tunes the details of the segmentation result based on full resolution.

detailed information. The Gated convolution [23], often adopted in the natural language processing [20], is exploited to build the shape stream to refine boundary details. And the attention mechanism is widely used in deep image segmentation [38–40]. Dual Attention Net (DANet) [38] adaptively integrates local features and global dependencies. In the traditional dilated fully convolutional neural networks [35], channel attention and position attention are used to simulate the semantics interdependence of the position and channel dimensions, respectively. These tricks greatly improve image segmentation accuracy.

Encouraged by deep image segmentation algorithms, a large number of deep interactive image segmentation algorithms have been proposed. In general, there are three main directions for improving the performance of deep interactive image segmentation: seeking better representation of interactive information, integrating interactive cues into segmentation networks (early fusion or late fusion), and constructing a superior loss function. Wang et al. [16] proposed a geodesic distance to express interactive information, which better differentiates neighboring pixels with different appearances and improves label consistency in homogeneous regions. Majumder et al. [17] and Lin et al. [41] represented clicks or scribble lines with superpixels to make the interaction cover more semantic information. Zhang et al. [42] proposed the IOG (Inside–Outside Guidance) method to give extensive information about interaction by a bounding box, which is nevertheless more time-consuming than clicks. As for ways of fusion of interactive information, early fusion [5–7,9,11,13,14,16–18] concatenates interactive information with the RGB channel of images as the input to the segmentation networks. However, early fusion strategy may weaken the interactive information for deep networks. Late fusion strategy [10,38] fuses the outputs of two individual networks that extract features of images and interactions. It makes the interaction have a direct influence on the top layer of the decoder. Li et al. [8] exploited late fusion to convert the role of user interaction to select the desired target. Besides, Forte et al. [18] combined soft-IoU loss and the proposed click location loss to force the network segmentation result to be correct at the position of the click. BRS (Backpropagating Refinement Scheme) [11] and f-BRS (feature Backpropagating Refinement Scheme) [14] add a regularization term for click location loss to prevent the network from being over-segmented. These novel loss functions effectively boost the segmentation results at the position of a click.

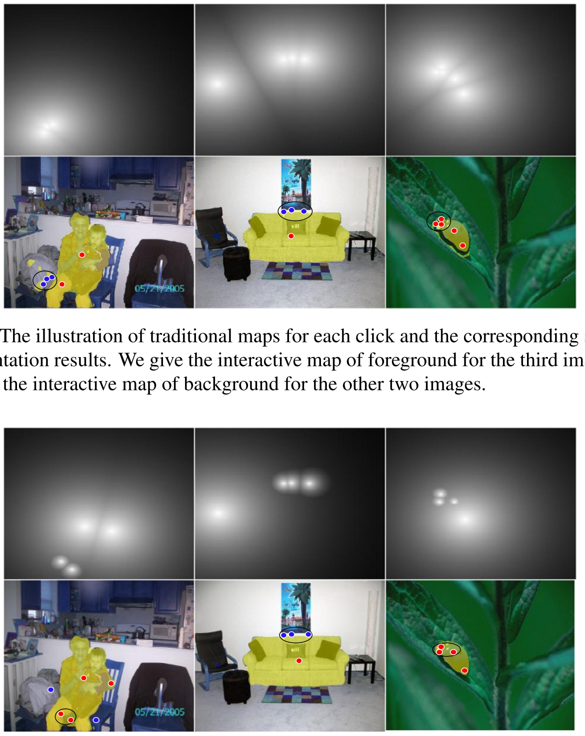
3. Method

Fig. 2 illustrates the framework of the proposed Gated-iDeepLab. The network consists of three parts: the basic feature extraction module, the interactive shape stream, and the refinement block. We choose the latest DeepLabV3+ [24] as our basic feature extraction module. Shape features obtained from the interlayer of DeepLabV3+ [24] are processed by the interactive shape stream, which consists of serialized gated convolutions [23]. In the refinement block, several convolution blocks are serially stacked to fine tune the segmentation results on full resolution. Correspondingly, the coarse segmentation of DeepLabV3+, the boundary response of the interactive shape stream, and the refined segmentation from the refinement block can be outputted from the proposed network. Moreover, the adaptive Gaussian map is proposed to represent the user interaction, making the network being able to concentrate on detailed regions based on two successive interactions. For the loss function, we designed a probability click loss function for user interaction, which makes the segmentation around the clicks more accurate.

This section includes three parts. In Section 3.1, we describe an adaptive representation for interactive information. In Section 3.2, we introduce the Gated-iDeepLab net, which is divided into three modules. In Section 3.3, we elaborate on a novel probability based click loss function to help the network better segment an image around click positions.

3.1. Adaptive representation for interactive information

Existing deep interactive image segmentation algorithms generally utilize either Euclidean distance or Gaussian distribution



(a) The illustration of traditional maps for each click and the corresponding segmentation results. We give the interactive map of foreground for the third image, and the interactive map of background for the other two images.

(b) The illustration of adaptive Gaussian maps for each click and the corresponding segmentation results. We give the interactive map of background for the second image, and the background interactive map for the other two images

Fig. 3. The illustration of the traditional maps and adaptive Gaussian maps to represent the interactive information. The segmented object masks are highlighted in yellow masks. The first row means the interactive maps of clicks (foreground or background), the second row means the segmentation results based on these two interactive maps (Red and blue dots represent the foreground and background user-clicks respectively).

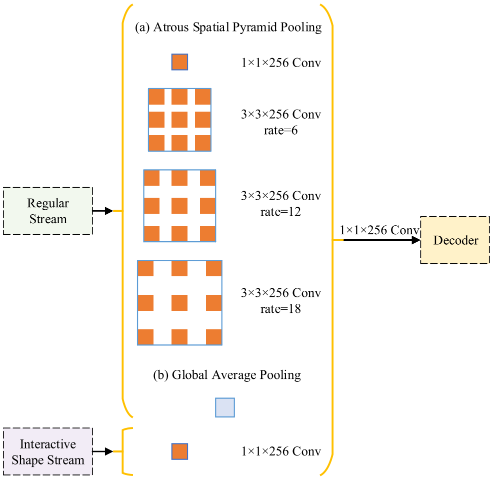


Fig. 4. Illustration of ASPP module in our proposed network. The ASPP in our network not only receives features from the regular stream but also integrates features from the Interactive shape stream. We first use one-dimensional convolution to increase the dimension of the features of the interactive shape stream, then fuse other multi-scale features and pass them to the decoder.

to represent user interactive clicks. As illustrated in the ablation experiment in [13], using Gaussian distribution to express interactions can effectively boost the segmentation performance. Hence, in this paper, we use Gaussian distribution to represent interactive information. After the user inputs the ith click c_i, the Gaussian distribution between a pixel p and the seeded pixel c_i is computed as:

$$\mathrm{I}(p) = exp(-\frac{\min d(p, c_i)}{\alpha}), \tag{1}$$

where $d(\cdot, \cdot)$ represents the Euclidean distance and α is the Gaussian variance that controls the affected region of the click. In the existing approaches, the value α is always fixed (identical for all clicks). However, in this case, the fixed Gaussian variance makes the influence of two close clicks over-redundant, instead of limiting their own characteristics. Exactly, the role of close clicks degenerates to the refinement, not highlighting the correction. As shown in Fig. 3(a), the segmentation results are guided by the interactive maps obtained by Eq. (1). It can be found that the detailed segmentation results are hardly to be further improved with close interactions. Especially in the third image, despite the clear boundary constraints, it is still difficult to improve the segmentation results in detail. To address this dilemma, we propose adaptive Gaussian maps to represent user interaction. Instead of a fixed Gaussian variance, when the distance of two clicks is less than a threshold, we will change α to a smaller value, enable the click to focus on the detailed perception of the correction:

$$\mathrm{I}(p) = \begin{cases} exp(-\frac{\min d(p,c_i)}{\alpha_l}), \; if \; \min\limits_{i \neq j} d(c_i, c_j) > T \\ exp(-\frac{\min d(p,c_i)}{\alpha_s}), \; if \; \min\limits_{i \neq j} d(c_i, c_j) \leq T, \end{cases} \tag{2}$$

where T is the threshold of the distance between any two click points. Gaussian variance equals α_l when the distance between two click points larger than the threshold, otherwise equals α_s. In our work, we set $\alpha_l = 100$, $\alpha_s = 20$ and $T = 40$. From Fig. 3(b), we find that when the adaptive Gaussian map is adopted, segmentation results based on close interactions have been further promoted in detail.

3.2. Network architecture

3.2.1. DeepLabV3+

We use the popular DeepLabV3+ [24] framework as basic network, as shown in Fig. 2, which includes three parts: Encoder (i.e., regular stream), ASPP (Atrous Spatial Pyramid Pooling) [36] and Decoder. We denote the features of the last four convolution blocks of image steam as $\{F_1, F_2, F_3, F_4\}$. The concatenation block in Fig. 2 fuses preprocessed images and interactive maps in the channel dimension. The ASPP module is composed of 3 × 3 atrous convolutions with varying rates 6, 12, and 18, pointwise convolution [43], and global average pooling. Besides, features obtained from the interactive shape stream (introduced in the next subsection) are also fed into the ASPP block (shown in Fig. 4). Pointwise convolution is first utilized to increase the dimension to 256. Then features extracted from these convolution blocks are concatenated so that we can obtain the multi-scale features from the top layer of the regular stream and interactive shape stream. The output of DeepLabV3+ is the initial coarse segmentation.

3.2.2. Interactive shape stream

To obtain the shape cue of the desired object, we use three sequential gated convolutions to extract boundary information from $\{F_1, F_2, F_3, F_4\}$. The three gated convolutions [44] are connected by residual convolution blocks [25], and a one-dimensional convolution for dimension reduction follows each residual convolution block. Different from [23], we reduce the number of the channel of features from F_1 to 64, and others from F_2, F_3, F_4 are

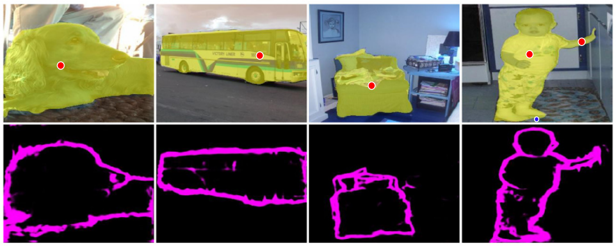


Fig. 5. Segmentation results and boundary map. Red and blue dots denote the foreground and background user-clicks respectively. The segmented object masks are highlighted in yellow. The boundary map is the visualization of the output of interactive shape stream.

reduced to 2 instead of 1. These features are first upsampled to the full resolution. We then concatenate feature F_1 with the interactive map to introduce the user intention into the interactive shape stream. Besides, we also combine the interactive map with the image gradients at the top of the interactive shape stream and input them to ASPP. The boundary loss (introduced in Section 3.3) is calculated by the boundary groundtruth and the shape output. From Fig. 5, it shows that the output boundary from the proposed interactive shape stream can well match the user's desired object.

The gated convolution is the key component of the interactive shape stream. It processes task-related information from the regular stream by suppressing irrelevant information. These gated convolutions control the flow of information between the two streams. Although the interactive shape stream processes contour information at full resolution, the powerful contour semantic perception makes it a shallow architecture, which is quite efficient [23].

The foreground and background of the target contour in the interactive shape stream should be distinguished, because of two different dimensions of interactive maps are supplied to direct the shape information processing. We denote the features from the tth regular stream and the interactive shape stream by r_t^o and i_t, where $o \in \{f, b\}$, and f, b represent the foreground and the background, respectively. Through concatenating r_t^o and i_t followed by point-wise convolution PW and sigmoid function α, the attention map is obtained:

$$\alpha_t = \delta(PW(r_t^o \oplus i_t)), \tag{3}$$

where $\oplus$ means the feature concatenation operation. After obtaining the attention map α_t, the gated convolution is performed on i_t by element-wise product $\odot$ with α_t, then the residual connection is attached and finally multiplies them with the channel-wise weight w_t. The feature of the next layer of interactive shape stream $\hat{i}_t$ is calculated as:

$$\hat{i}_t = ((i_t \odot \alpha_t) + i_t)^T w_t. \tag{4}$$

Note that the dimension of the last layer in the interactive shape stream is two, meaning that the interactive shape stream contains both the boundary response of the foreground and that of the background. Since the interactive map is fused with features from the first convolution block, the interactive shape stream only processes the boundary information related to the user's interest. Since Eqs. (3) and (4) are differentiable, the end-to-end training can be achieved by backpropagation.

3.2.3. Refinement block

The segmentation result obtained from the DeepLabV3+ is directly up-sampled by interpolation based on low-resolution feature maps, which easily leads to inaccurate segmentation of details. Note that the boundary response outputted from the interactive shape stream can be utilized to refine the coarse segmentation. We therefore attach a refinement block based on the coarse segmentation result, original image, and boundary feature (i.e., skip connection 1 and skip connection 2 in Fig. 2), which is composed of six convolution layers. All operations are based on full resolution. The convolution kernel size is set as {7, 5, 3, 3, 3, 1}, gradually converted from larger receptive fields to smaller ones, in order to fine tune details of the segmentation.

3.3. Loss function

The output of the segmentation network consists of three parts: the output of the interactive shape stream (boundary response), the coarse segmentation of DeepLabV3+ [24] and the sophisticated segmentation of the refinement block. We calculate the binary cross-entropy (BCE) loss, denoted by $l^e(p)$, for the boundary response using the boundary groundtruth:

$$l^e(p) = -(y_p^e log(x_p^e) + (1 - y_p^e)log(1 - x_p^e)), \tag{5}$$

where x_p^e represents the probability of point p in boundary response, and y_p^e denotes the label (0 or 1) of the point p in the boundary groundtruth (which is obtained by employing the Canny edge detector on mask groundtruth).

For the coarse segmentation, we propose a novel probability click loss, denoted by $l^c(p)$ based on the BCE:

$$\begin{aligned} l^c(p) = &- (y_p^m log(x_p^c) + (1 - y_p^m)log(1 - x_p^c)) \\ &+ \lambda_c abs(x_p^c - y_p^m)I(p), \end{aligned} \tag{6}$$

where x_p^c denotes the probability of point p in the coarse segmentation result, and y_p^m denotes the label of point p in the mask, and $abs(\cdot)$ represents the absolute value, and λ_c is used to adjust the weight of the coarse BCE loss and coarse probability click loss. The second term $abs(x_p^c - y_p^m)I(p)$ is the probability click loss we proposed. Similarly, we can get the loss function of the refined segmentation result:

$$\begin{aligned} l^r(p) = &- (y_p^m log(x_p^r) + (1 - y_p^m)log(1 - x_p^r)) \\ &+ \lambda_r abs(x_p^r - y_p^m)I(p), \end{aligned} \tag{7}$$

where x_p^r denotes the probability of the point p in the refined segmentation result, and λ_r is utilized to control the weight of the refined BCE loss and the refined probability click loss.

The interaction map constructed by Gaussian distribution provides the probability of each pixel belonging to the foreground in the image. The closer the value of the point p is to 1, the greater the probability of the point p being foreground is, and vice versa. By introducing the probability information to our proposed novel click loss function, it makes the optimization range larger than the loss function in [18], i.e., the closer the area near the click, the greater the probability that it will be optimized. Combined these loss functions together, the overall loss function to train the network is calculated as:

$$l(p) = l^e(p) + \beta_c l^c(p) + \beta_r l^r(p), \tag{8}$$

where β_c and β_r control the importance of the coarse segmentation and refined segmentation, respectively. In our experiments, we set $\lambda_r = \lambda_c = 2$, $\beta_c = 1$ and $\beta_r = 0.5$.

4. Experiments

4.1. Training settings

Similar to [18], we train the segmentation network sequentially. We click the center of the largest connected error region in the segmentation result each time. The Gated-iDeepLab is trained on the augmented dataset (PASCAL VOC [47]+SBD [48]), which

Table 1
Several details of previous deep interactive image segmentation. Click embedding: the way to calculate the interactive map, Fusion strategy: the strategy of fusion interactive map, Encoder: the backbone of main segmentation network, Training schedule: the way to train the network in terms of interaction.

Name	Click embedding	Fusion strategy	Encoder	Training schedule
DOS [5]	Euclidean Distance	Early	VGG16 [45]	All clicks at once
ITIS [13]	Gaussian Map	Early	Xception [43]	Adding one click per epoch
FCTSFN [10]	Gaussian Map	late	VGG16	All clicks at once
RIS [6]	Euclidean Distance	Early	VGG16	All clicks at once
LD [8]	Gaussian Map	late	VGG16	All clicks at once
BRS [11]	Euclidean Distance	Early	DenseNet [46]	All clicks at once
Ours	Adaptive Gaussian Map	Both	ResNet50 [25]	Adding one click per iteration

excluded the validation set of these two datasets. For multi-object images, we randomly select one of the objects for training each iteration. Random horizontal flipping, fixed-scale cropping, and random Gaussian blur are applied to augment the training images. All images are cropped to the size of 320 × 320. We utilize ResNet50 [25] as regular stream, and initialize it with the parameters pre-trained on ImageNet [49]. The network is trained with the SGD optimizer with momentum (momentum is 0.99, weight decay is 10^{-4}). Besides, the learning rate for the regular stream is 10^{-5} mainly for finetuning, and for the other modules are 10^{-4}. We employ the "poly" policy [35] as the learning rate schedule, and the output stride of the DeepLabV3+ is 16. All experiments are performed on a single NVIDIA RTX 2080Ti using the PyTorch framework.

4.2. Evaluation details

4.2.1. DataSets

We apply several commonly used datasets to evaluate the performance of our algorithm: GrabCut [29], Berkeley [50], SBD [48] and DAVIS [51]. GrabCut dataset is a popular dataset used to verify interactive segmentation algorithms. It consists of 50 single target images. Most of these images have a relatively clean background. The Berkeley dataset consists of 96 images with 100 targets. Some of these images have similar background and foreground colors. The SBD dataset is an enhanced version of the PASCAL VOC dataset. It provides the segmentation masks that are not supplied in the PASCAL VOC dataset. The validation set has 2857 images with 20 categories. The DAVIS is a video sequence segmentation dataset. It consists of 50 high-definition sequences (480p and 1080p). We randomly choose 10% frames (total 345 images) as [11] for validation. Unlike the interactive video object segmentation algorithm [52,53] only interacts with a specific frame and then predicts all frames (round-based interaction scheme), we need to interact with all frames to verify our interactive image segmentation algorithm. This dataset also provides four kinds of pixel-level annotations for different frames: people, animals, vehicles, and objects.

4.2.2. Metrics

In this paper, we utilize the commonly used mean Intersection over Union (mIoU) [5,15,35] to evaluate the performance of the compared approaches. The curve of the mIoU versus the number of clicks is given to evaluate the performance of different algorithms. Furthermore, we employ a metric commonly used in interactive image segmentation: mean Number of Clicks (mNoC) [5], which is used to evaluate the number of clicks required to achieve the specified mIoU. It means less user interaction if mNoC is small. Same as [5], we set the maximum number of clicks to 20. To examine the promotion of the interactive shape stream on interaction, the more specific over-segmentation rate (OSR) and under-segmentation rate (USR) are proposed (see Section 4.6 in detail).

4.3. Comparison with the other methods

We give a thorough qualitative and qualitative experiments to compare Gated-iDeepLab with the conventional approaches: GraphCut (GC) [1], Geodesic Matting (GM) [54], Random Walks (RW) [2], Growcut (GRC) [3], Euclidean Star Convexity (ESC) [4], Geodesic Star Convexity (GSC) [4] and the deep interactive approaches: deep object selection (DOS) [5], iteratively trained interactive segmentation (ITIS) [13], fully convolutional two-stream fusion network (FCTSFN) [10], regional image segmentation (RIS) [6], latent diversity based segmentation (LD) [8] and back-propagating refinement scheme (BRS) [11]. The difference in architecture, the interaction encoding, and training of these remarkable deep interactive image segmentation are listed in Table 1. Note that the fusion strategy of our algorithm includes three parts: the input of DeepLabV3+, the input, and output of interactive shape stream, where fusing with the output in interactive shape stream plays the role of late fusion.

4.3.1. Quantitative benchmark results

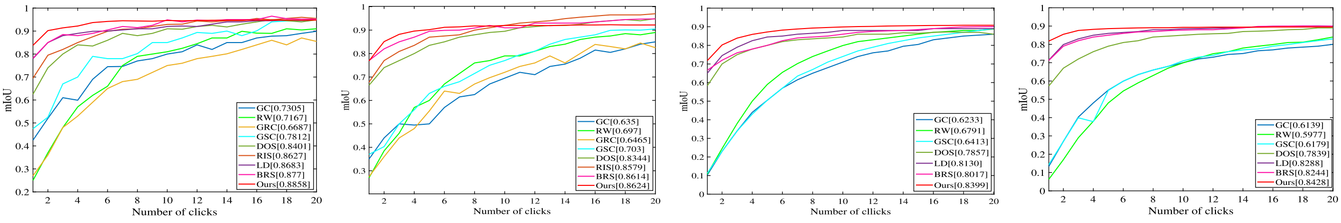
Fig. 6 shows the number of clicks versus the mean IoU curves for the compared methods on four datasets, and we also report the area under curve (AuC) values across the full range of clicks (1–20), shown in the legend of Fig. 6. From the AuC values, we find our algorithm is superior to all others on four datasets. Although our algorithm is comparable to BRS [11] in terms of AuC on the Berkeley dataset, more precise segmentation results of Gated-iDeepLab on all datasets can be achieved by the first few clicks. This is due to the fact that the interactive shape stream adopted for our method can obtain the approximate boundary to frame the desired object, which relieves the burden of further interaction by the user. Table 2 lists the mNoC@85 and mNoC@90 of the compared methods on different datasets, which is the mean numbers of clicks needed to reach the 85% and 90% mIoU values, respectively. "DOS with GC" and "DOS w/o GC" represent the deep object selection [5] with and without the GraphCut [1] post-processing, respectively. As illustrated in Table 2, the performance of the proposed algorithm has been improved significantly on four datasets. Compared with conventional algorithms, our proposed algorithm needs much fewer interactions, which indicates the strong expressive ability of deep features. And for deep-learning-based algorithms, our algorithm is slightly second to the BRS on the Berkeley dataset but outperforms other algorithms. Note that the largest improvement of our algorithm in terms of mNoC@90 has achieved on the SBD dataset, reducing 2.59 clicks. This means more precise segmentation can be achieved with fewer efforts.

4.3.2. Qualitative benchmark results

Fig. 7 demonstrates the example of qualitative comparisons of GraphCut, LD, BRS, and Gated-iDeepLab. We display the corresponding segmentations by gradually adding 5 clicks. From Fig. 7(a), we intuitively see that the defects of the conventional algorithms are obvious, and the improvement of the segmentation

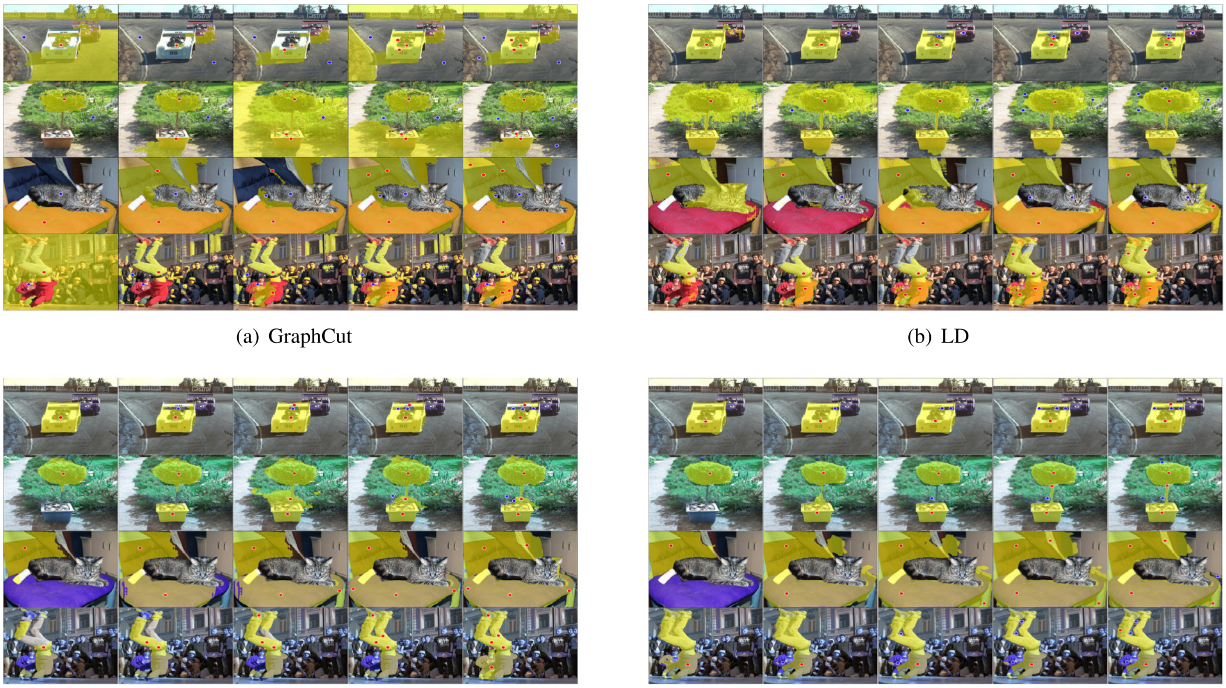
Table 2
Comparison of mNoC@85 and mNoC@90 metrics on GrabCut, Berkeley, SBD, and DAVIS datasets. Best performance in bold.

Methods	GrabCut		Berkeley	SBD		DAVIS	
	85%	90%	90%	85%	90%	85%	90%
GC [1]	7.98	11.1	14.33	13.6	15.96	15.13	17.41
GM [54]	13.32	14.57	15.96	15.36	17.6	18.59	19.5
RW [2]	11.36	13.77	14.02	12.22	15.04	16.71	18.31
ESC [4]	7.24	9.2	12.11	12.21	14.86	15.41	17.7
GSC [4]	7.1	9.12	12.57	12.69	15.31	15.35	17.52
GRC [3]	–	16.74	18.25	–	–	–	–
DOS with GC [5]	5.08	6.08	8.65	9.22	12.8	9.03	12.58
DOS w/o GC [5]	8.02	12.59	–	14.3	16.79	12.52	17.11
ITIS [13]	–	5.6	–	–	–	–	–
FCTSFN [10]	–	3.76	6.49	–	–	–	–
RIS [6]	–	5	6.03	6.03	–	–	–
LD [8]	3.2	4.79	–	7.41	10.78	5.95	9.57
BRS-DenseNet [11]	2.6	3.6	**5.08**	6.59	9.78	5.58	8.24
Ours	**1.88**	**3.4**	5.69	**4.83**	**7.19**	**4.71**	**7.29**



(a) GrabCut (b) Berkeley (c) SBD (d) DAVIS

Fig. 6. The number of clicks vs. mean IoU (NoC-mIoU) curves of our Gated-iDeepLab and other methods on GrabCut, Berkeley, SBD and DAVIS datasets.



(a) GraphCut (b) LD

(c) BRS (d) Ours

Fig. 7. The segmentation results of sequential click corrections of GraphCut, LD, BRS, and Gated-iDeepLab algorithms on four images. We list the first five corrective segmentation results. Note that two clicks (foreground and background annotations) should be provided for the GraphCut algorithm first. The segmented object masks are highlighted in yellow masks. Foreground and background user-clicks are denoted in red and blue dots respectively.

results with user interaction is minimal. However, we find that even if the second and fourth images with complex backgrounds, or the third non-closed targets, Gated-iDeepLab only requires few interactions to predict the better boundary of the target. This is due to the joint estimation between the boundary output of the interactive shape stream and coarse segmentation. The boundary response combined with interactive information alleviates the possibility of over-segmentation, thus reducing the efforts of the user's interaction. Benefit from adaptive Gaussian maps, Gated-iDeepLab has better correction capabilities in details, such as the background between the body and the tail of the car in the first image, the elongated branches of the bushes in the second image,

Table 3
Mean NoC for ablation experiments by removing proposed modules. AGM means adaptive Gaussian maps for interactions, RF means refinement block, PCL means probability click loss function, ISS means interactive shape stream. The numbers with signs in parentheses represent the changes of value in mNoC under these ablation settings, and "+" means the increase in mNoC value. Best performance in bold.

Settings	Berkeley mNoC@90	SBD mNoC@85
w/o AGM	6.43 (+0.74)	5.41 (+0.58)
w/o AGM+RF	8.24 (+2.55)	7.35 (+2.52)
w/o AGM+RF+PCL	9.14 (+3.45)	7.98 (+3.15)
w/o AGM+RF+PCL+ISS	12.92 (+7.23)	12.2 (+6.37)
Gated-iDeepLab	**5.69**	**4.83**

and the background between the dancer's legs in the fourth image.

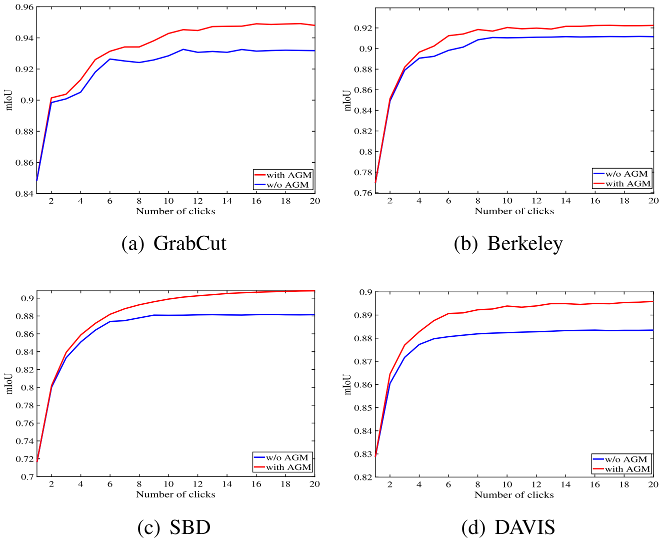
4.4. Ablation study

4.4.1. Ablation study for each module

To verify the effectiveness of the proposed modules, several ablation experiments on the Berkeley and SBD validation set are performed. Based on Gated-iDeepLab, we sequentially remove adaptive Gaussian maps for interactions (AGM) and set $\alpha_l = \alpha_s = 100$ in Eq. (2), refinement block (RF), probability click loss function (PCL), and interactive shape stream (ISS). Table 3 and Table 4 list the mNoC and mIoU at certain number of clicks for the ablation settings by removing the proposed modules, respectively. From Table 3, we derive that the influence of each module of Gated-iDeepLab in descending order is interactive shape stream, refinement block, probability click loss function, and adaptive representation for interactive information. It is simple for the idea of the adaptive representation of interactive information, and the segmentation results are not benefited by the first click or first few clicks, so its impact on the number of click reaches to certain mIoU is relatively small. However, the influence on mIoU with respect to subsequent clicks, please refer Section 4.5 for more analysis. It is markedly seen from Table 4 that after removing the AGM module, the mIoUs are stay still, while the superiority of the AGM gradually manifest as the number of clicks increases. The probability click loss function makes the network focus larger affected region of click, which can promote the performance of the segmentation network. The refinement block takes full advantages of the coarse segmentation and the boundary constraints obtained by the interactive shape stream, and it makes the network reach the target accuracy earlier. The improvement of the segmentation performance by the first few clicks is obviously insufficient after removing the refinement block, as shown in Table 4. The interactive shape stream can better make the network frame the target's boundary information, and removing it will greatly increase the number of clicks, which directly reflects that the interactive shape stream is an important part of Gated-iDeepLab.

4.4.2. Ablation study for fusion strategy

We also conduct ablation experiments for the fusion strategy of interactive maps and the network. As is shown in Fig. 2, we integrate interactive maps with the input of the regular stream, the input and the output of the interactive shape stream. Therefore, we removed the early fusion for the regular stream (EF4IS), the early fusion for the interactive shape stream (EF4ISS), and late fusion (LF). Table 5 shows the mNoC on the Berkeley and SBD datasets corresponding to these settings of ablation experiments. It can be seen that when we remove the early fusion for the regular stream, the performance of the network drops drastically. This is because the regular stream lacks user interactive information, so it is hard to obtain the feature of the target from the encoder accurately. Late fusion affects next to the early fusion with the regular stream on the segmentation results. It indicates that there will be little loss of the interactive information at the top layer of the encoder when only the early fusion strategy is applied. The late fusion can strengthen user interaction at the top layer of the encoder, thereby improving the segmentation accuracy. Early fusion for the interactive shape stream has the least impact on the segmentation results. The early fusion with the interactive shape stream is equivalent to providing a boundary restriction for the target object, and this restriction reflects more changes in details, which has little impact on the overall accuracy. Hence, the fusion strategy, both fusion, plays a crucial role in the improvement of segmentation results.



(a) GrabCut

(b) Berkeley

(c) SBD

(d) DAVIS

Fig. 8. The number of clicks vs. mean IoU (NoC-mIoU) curves of Gated-iDeepLab with adaptive Gaussian maps (with AGM) and without adaptive Gaussian maps (w/o AGM) on GrabCut, Berkeley, SBD and DAVIS datasets.

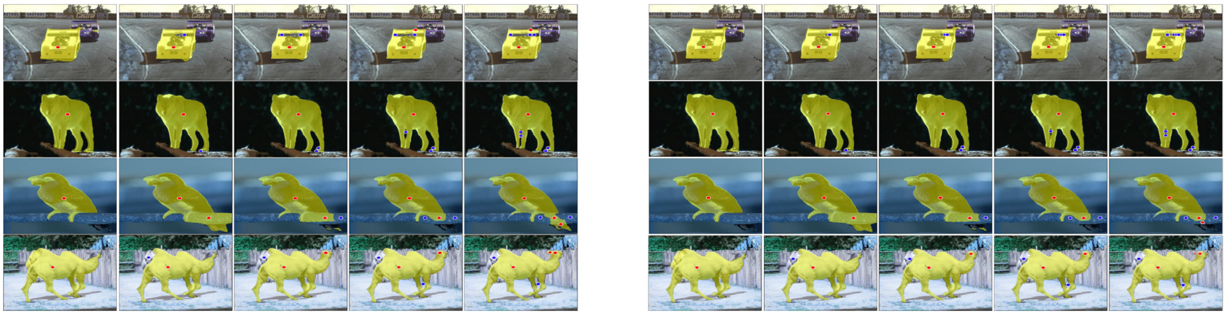
4.5. Analysis of the effectiveness of adaptive Gaussian maps

Adaptive Gaussian map (AGM) is a vital part of our proposed method. Thus it is necessary to conduct more experiments to verify the efficiency of the adaptive Gaussian map. Fig. 8 shows the curves of mIoU-Number of clicks for Gated-iDeepLab with and without adaptive Gaussian maps (with AGM and w/o AGM) on four datasets. We can see that Gated-iDeepLab with these two settings obtains the same mIoU after the first click, yet as the number of clicks increases, our method without adaptive Gaussian maps converges prematurely. This is because subsequent interactions focus more on fine-tuning, and the affected regions of two close clicks overlap each other, which reduces the segmentation performance.

Fig. 9 illustrates the qualitative experiments to validate the superiority of the adaptive Gaussian maps. We list the segmentation results of the first five interactions for each image. From these representative images, we can conclude that when two clicks are far apart, the Gated-iDeepLab can both segment the target effectively with these two settings. Otherwise, the segmentation performance in detail is different. Specifically, in the blank part of the car's tail wing, the wolf's front legs, the tail of the bird, and the head of the camel, Gated-iDeepLab with adaptive Gaussian maps can boost the segmentation performance significantly when the user places two close clicks. On the contrary, no effective segmentations are achieved by clicking multiple times in these regions, which confirms the advantage of adaptive Gaussian maps.

Table 4
Mean IoU at certain number of clicks for the ablation experiments by removing each proposed modules. AGM means adaptive Gaussian maps for interactions, RF means refinement block, PCL means probability click loss function, ISS means interactive shape stream. (in percentage, best performance in bold.)

Dataset	w/o AGM	w/o RF	w/o PCL	w/o ISS	Gated-iDeepLab
Berkeley (1 click)	0.769	0.744	0.752	0.686	**0.769**
SBD (1 click)	0.716	0.702	0.692	0.626	**0.716**
Berkeley (3 clicks)	0.856	0.816	0.86	0.795	**0.882**
SBD (3 clicks)	0.815	0.783	0.822	0.76	**0.839**
Berkeley (10 clicks)	0.912	0.909	0.895	0.872	**0.92**
SBD (10 clicks)	0.875	0.861	0.886	0.843	**0.901**



(a) Gated-iDeepLab with AGM

(b) Gated-iDeepLab without AGM

Fig. 9. The segmentation results of sequential click corrections Gated-iDeepLab with and without adaptive Gaussian maps (AGM) on different images. We list the first five corrective segmentation results. The segmented object masks are highlighted in yellow masks. Foreground and background user-clicks are denoted in red and blue dots respectively.

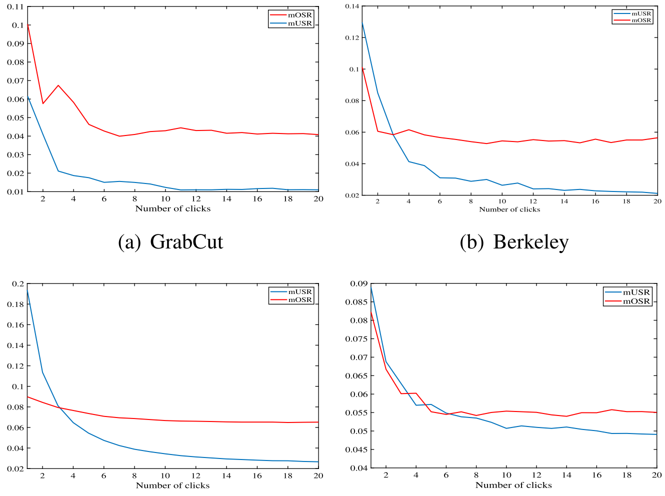
4.6. Analysis of the impact of boundary information on the effectiveness of interaction

In order to testify the promotion of the interactive shape stream on interaction, we propose an over-segmentation rate (OSR) and under-segmentation rate (USR) to analyze the influence of boundary information on the effectiveness of interaction. The part where the foreground of the groundtruth overlaps with the foreground of the prediction is the true positive region (TP); the part where the foreground of the prediction coincides with the background of the groundtruth is the false positive region (FP), i.e., the over-segmented region; the part where the background of the prediction coincides with the foreground of the groundtruth is the false negative region (FN), that is, the under-segmented region. In order to measure the over-segmentation and under-segmentation, we propose the over-segmentation rate (OSR) and under-segmentation rate (USR) to validate the impact of interaction with boundary information on the segmentation results:

$$OSR = \frac{FP}{FN + FP + TP}, USR = \frac{FN}{FN + FP + TP}, \tag{9}$$

where the addition operation is to count the number of all pixels in these areas. The commonly used error rate cannot distinguish the error caused by over-segmentation and under-segmentation, therefore, these two proposed indicators are more suitable for verifying the ability of the algorithm to control over-segmentation.

Fig. 10 shows the USR/OSR-Number of clicks curves of our algorithm on four datasets. We notice that the error rate of our algorithm on the four datasets decreases with the increase of the number of clicks, indicating the interaction has a significant improvement in segmentation results. Moreover, we conclude that the under-segmentation rates on the four datasets decrease faster than the over-segmentation rates. This is because, under the constraint of a clear boundary, the over-segmentation phenomenon tends to be stable only with a few numbers of interactions, and the phenomenon of under-segmentation quickly relieves by filling the under-segmented region under boundary constraints, thereby reducing the burden of user interaction.



(a) GrabCut

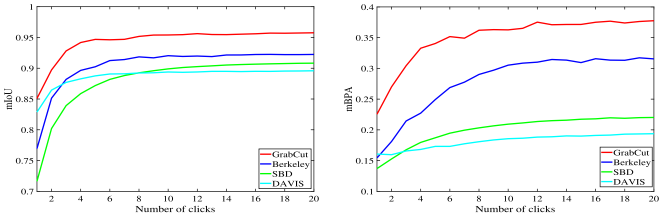
(b) Berkeley

(c) SBD

(d) DAVIS

Fig. 10. The number of clicks vs. mUSR/mOSR curves of the Gated-iDeepLab on GrabCut, Berkeley, SBD and DAVIS datasets.

To illustrate the influence of boundary prediction on segmentation, we calculate the mean boundary pixel accuracy (mBPA) for the specified number of clicks. Fig. 11 shows the curves of the mBPA/mIoU-number of clicks of our algorithm on four datasets. Our method only utilizes the coarse boundary response to constraint the segmentation results (as demonstrated in Fig. 5). Hence the boundary prediction is relatively poor, yet it has a positive correlation with mIoU. As the number of clicks increases, the mBPA and the mIoU of Gated-iDeepLab are increasing simultaneously. However, after placing multiple clicks, mIoU converges earlier than mBPA, reflecting that better segmentation results



(a) mIoU (b) mBPA

Fig. 11. The number of clicks vs. mIoU/mBPA curves of the Gated-iDeepLab on GrabCut, Berkeley, SBD and DAVIS datasets.

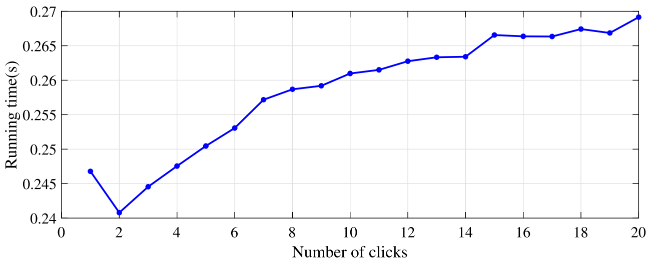


Fig. 12. Running time according to the number of clicks.

Table 5
Mean NoC for the ablation experiments of different fusion strategies. EF4IS means early fusion for regular stream, EF4ISS means early fusion for interactive shape stream, LF means late fusion. The numbers with signs in parentheses represent the changes of value in mNoC under these ablation settings, and "+" means the increase in mNoC value. Best performance in bold.

Settings	Berkeley mNoC@90	SBD mNoC@85
w/o EF4IS	18.51 (+12.82)	18 (+13.17)
w/o EF4ISS	6.11 (+0.42)	5.17 (+0.34)
w/o LF	9.11 (+3.42)	6.20 (+1.37)
w/o EF4IS+EF4ISS	18.79 (+13.1)	18.08 (+13.25)
w/o EF4ISS+LF	9.81 (+4.12)	6.5 (+1.67)
Gated-iDeepLab	**5.69**	**4.83**

are maintained under the boundary constraint. Therefore, the boundary information has played a crucial role in promoting the segmentation performance.

4.7. Time complexity analysis

To examine the time complexity of our proposed algorithm, we evaluate the running time of our algorithm on the DAVIS dataset in seconds per click (SPC). The running time of our algorithm is 0.26 SPC, which meets the real-time applications. Fig. 12 shows the running time (s)-Number of clicks curve. Although the time complexity increases with more and more interactions, it is still in the tolerance interval.

5. Discussion

The two essential components of our proposed method are adaptive Gaussian maps and interactive shape stream. We argue that these two modules have improved the segmentation performance locally and globally by performing extensive experiments. The idea of the adaptive Gaussian map is simple, and excellent segmentation can be obtained in most cases. Nevertheless, this method of changing the affected region of the annotations via simply setting a threshold may fail for tiny targets. Therefore, we will design an efficient function that adaptively changes the affected region of the annotations according to the target scale in future work. Hence the prior information provided by the user's click is more accurate.

6. Conclusion

In this paper, we propose a novel dual-stream framework via adaptive Gaussian maps for interactive image segmentation. An interactive shape stream is utilized to extract the boundary information of the user's desired target. The clear boundary excels at suppressing the phenomenon of over-segmentation, alleviating the burden of user interaction. To better represent user input, we design an adaptive Gaussian map to model user clicks. In this way, the mapping range for details can be automatically adjusted. Besides, we also develop a probability click loss function to expand the affected region of each click. Four public datasets show the superiority of our algorithm in interactive image segmentation.

CRediT authorship contribution statement

Zongyuan Ding: Conceptualization, Methodology, Software, Formal analysis, Writing - original draft. **Tao Wang:** Supervision, Funding acquisition, Resources, Writing - review & editing. **Quansen Sun:** Funding acquisition, Resources. **Qiongjie Cui:** Investigation, Writing review & editing. **Fuhua Chen:** Investigation, Writing - review.

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.

Acknowledgments

The authors would like to thank the editor and the anonymous reviewers for their critical and constructive comments and suggestions. This work was supported in part by the Natural Science Foundation of Jiangsu Province, China, under Grant BK20180458, in part by the National Science Foundation of China under Grants 61673220, and 61976028, in part by Project funded by China Postdoctoral Science Foundation under Grant 202M681530.

References

[1] Y.Y. Boykov, M.-P. Jolly, Interactive graph cuts for optimal boundary & region segmentation of objects in ND images, in: Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, Vol. 1, IEEE, 2001, pp. 105–112.

[2] L. Grady, Random walks for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 28 (11) (2006) 1768–1783.

[3] V. Vezhnevets, V. Konouchine, GrowCut: Interactive multi-label ND image segmentation by cellular automata, in: Proc. of Graphicon, Vol. 1, 2005, pp. 150–156.

[4] V. Gulshan, C. Rother, A. Criminisi, A. Blake, A. Zisserman, Geodesic star convexity for interactive image segmentation, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 3129–3136.

[5] N. Xu, B. Price, S. Cohen, J. Yang, T.S. Huang, Deep interactive object selection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 373–381.

[6] J. Liew, Y. Wei, W. Xiong, S.-H. Ong, J. Feng, Regional interactive image segmentation networks, in: 2017 IEEE International Conference on Computer Vision (ICCV), IEEE, 2017, pp. 2746–2754.

[7] A. Benard, M. Gygli, Interactive video object segmentation in the wild, 2017, arXiv preprint arXiv:1801.00269.

[8] Z. Li, Q. Chen, V. Koltun, Interactive image segmentation with latent diversity, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 577–585.

[9] R. Benenson, S. Popov, V. Ferrari, Large-scale interactive object segmentation with human annotators, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11700–11709.

[10] Y. Hu, A. Soltoggio, R. Lock, S. Carter, A fully convolutional two-stream fusion network for interactive image segmentation, Neural Netw. 109 (2019) 31–42.

[11] W.-D. Jang, C.-S. Kim, Interactive image segmentation via backpropagating refinement scheme, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5297–5306.

[12] Z. Lin, Z. Zhang, L.-Z. Chen, M.-M. Cheng, S.-P. Lu, Interactive image segmentation with first click attention, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13339–13348.

[13] S. Mahadevan, P. Voigtlaender, B. Leibe, Iteratively trained interactive segmentation, 2018, arXiv preprint arXiv:1805.04398.

[14] K. Sofiiuk, I. Petrov, O. Barinova, A. Konushin, f-BRS: Rethinking backpropagating refinement for interactive segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 8623–8632.

[15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440.

[16] G. Wang, M.A. Zuluaga, W. Li, R. Pratt, P.A. Patel, M. Aertsen, T. Doel, A.L. David, J. Deprest, S. Ourselin, et al., DeepIGeoS: a deep interactive geodesic framework for medical image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 41 (7) (2018) 1559–1572.

[17] S. Majumder, A. Yao, Content-aware multi-level guidance for interactive instance segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11602–11611.

[18] M. Forte, B. Price, S. Cohen, N. Xu, F. Pitié, Getting to 99% accuracy in interactive segmentation, 2020, arXiv preprint arXiv:2003.07932.

[19] K.-K. Maninis, S. Caelles, J. Pont-Tuset, L. Van Gool, Deep extreme cut: From extreme points to object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 616–625.

[20] Y.N. Dauphin, A. Fan, M. Auli, D. Grangier, Language modeling with gated convolutional networks, in: International Conference on Machine Learning, PMLR, 2017, pp. 933–941.

[21] H. Wang, Y. Wang, Q. Zhang, S. Xiang, C. Pan, Gated convolutional neural network for semantic segmentation in high-resolution images, Remote Sens. 9 (5) (2017) 446.

[22] M. Siam, S. Valipour, M. Jagersand, N. Ray, Convolutional gated recurrent networks for video segmentation, in: 2017 IEEE International Conference on Image Processing (ICIP), IEEE, 2017, pp. 3090–3094.

[23] T. Takikawa, D. Acuna, V. Jampani, S. Fidler, Gated-scnn: Gated shape cnns for semantic segmentation, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 5229–5238.

[24] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 801–818.

[25] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[26] E.N. Mortensen, W.A. Barrett, Interactive segmentation with intelligent scissors, Graph. Models Image Process. 60 (5) (1998) 349–384.

[27] D. Freedman, T. Zhang, Interactive graph cut based segmentation with shape priors, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), Vol. 1, IEEE, 2005, pp. 755–762.

[28] S. Zheng, C.-C. Yang, S.-L. Xiang, J. Ye, Makers based level set method for image segmentation, in: 2008 International Conference on Machine Learning and Cybernetics, Vol. 2, IEEE, 2008, pp. 947–952.

[29] C. Rother, V. Kolmogorov, A. Blake, Grabcut interactive foreground extraction using iterated graph cuts, ACM Trans. Graph. 23 (3) (2004) 309–314.

[30] T. Wang, J. Yang, Z. Ji, Q. Sun, Probabilistic diffusion for interactive image segmentation, IEEE Trans. Image Process. 28 (1) (2018) 330–342.

[31] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, IEEE Trans. Pattern Anal. Mach. Intell. 26 (9) (2004) 1124–1137.

[32] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2015, pp. 234–241.

[33] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (12) (2017) 2481–2495.

[34] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Semantic image segmentation with deep convolutional nets and fully connected crfs, 2014, arXiv preprint arXiv:1412.7062.

[35] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2017) 834–848.

[36] L.-C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, 2017, arXiv preprint arXiv:1706.05587.

[37] M. Yang, K. Yu, C. Zhang, Z. Li, K. Yang, Denseaspp for semantic segmentation in street scenes, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3684–3692.

[38] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, H. Lu, Dual attention network for scene segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3146–3154.

[39] A. Tao, K. Sapra, B. Catanzaro, Hierarchical multi-scale attention for semantic segmentation, 2020, arXiv preprint arXiv:2005.10821.

[40] Z. Zhong, Z.Q. Lin, R. Bidart, X. Hu, I.B. Daya, Z. Li, W.-S. Zheng, J. Li, A. Wong, Squeeze-and-attention networks for semantic segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13065–13074.

[41] D. Lin, J. Dai, J. Jia, K. He, J. Sun, Scribblesup: Scribble-supervised convolutional networks for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3159–3167.

[42] S. Zhang, J.H. Liew, Y. Wei, S. Wei, Y. Zhao, Interactive object segmentation with inside-outside guidance, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 12234–12244.

[43] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.

[44] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T.S. Huang, Free-form image inpainting with gated convolution, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 4471–4480.

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

[46] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.

[47] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (voc) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.

[48] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, J. Malik, Semantic contours from inverse detectors, in: 2011 International Conference on Computer Vision, IEEE, 2011, pp. 991–998.

[49] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: A large-scale hierarchical image database, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 248–255.

[50] K. McGuinness, N.E. O'connor, A comparative evaluation of interactive segmentation algorithms, Pattern Recognit. 43 (2) (2010) 434–444.

[51] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, A. Sorkine-Hornung, A benchmark dataset and evaluation methodology for video object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 724–732.

[52] J. Miao, Y. Wei, Y. Yang, Memory aggregation networks for efficient interactive video object segmentation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 10366–10375.

[53] S. Oh, J. Lee, N. Xu, S. Kim, Space-time memory networks for video object segmentation with user guidance, IEEE Trans. Pattern Anal. Mach. Intell. (2020) 1–14.

[54] X. Bai, G. Sapiro, Geodesic matting: A framework for fast interactive image and video segmentation and matting, Int. J. Comput. Vis. 82 (2) (2009) 113–132.