Journal of Biomedical Semantics

RESEARCH Open Access

Residual refinement for interactive skin lesion segmentation

Dalei Jiang[1†], Yin Wang[2†], Feng Zhou[3†], Hongtao Ma[2], Wenting Zhang[1], Weijia Fang[4], Peng Zhao[4*] and Zhou Tong[4*]

Abstract

Background: Image segmentation is a difficult and classic problem. It has a wide range of applications, one of which is skin lesion segmentation. Numerous researchers have made great efforts to tackle the problem, yet there is still no universal method in various application domains.

Results: We propose a novel approach that combines a deep convolutional neural network with a grabcut-like user interaction to tackle the interactive skin lesion segmentation problem. Slightly deviating from grabcut user interaction, our method uses boxes and clicks. In addition, contrary to existing interactive segmentation algorithms that combine the initial segmentation task with the following refinement task, we explicitly separate these tasks by designing individual sub-networks. One network is SBox-Net, and the other is Click-Net. SBox-Net is a full-fledged segmentation network that is built upon a pre-trained, state-of-the-art segmentation model, while Click-Net is a simple yet powerful network that combines feature maps extracted from SBox-Net and user clicks to residually refine the mistakes made by SBox-Net. Extensive experiments on two public datasets, PH2 and ISIC, confirm the effectiveness of our approach.

Conclusions: We present an interactive two-stage pipeline method for skin lesion segmentation, which was demonstrated to be effective in comprehensive experiments.

Keywords: DCNN, Skin lesion, Interactive segmentation, Residual refinement, Two-stage pipeline

Background

After years of rapid growth, the DIKW (data, information, knowledge, and wisdom) hierarchy [1] has been closely related to the development of artificial intelligence and, more precisely, deep learning. As a paradigm of deep learning, AI algorithms hierarchically transform data into information, then knowledge, and finally wisdom by building deep layers of the network to represent different levels of abstraction. Artificial intelligence has contributed to the resolution of a variety of biomedical problems, including cancer and have the potential to deliver better management services to deal with chronic diseases. Nowadays, artificial intelligence methods have been progressively established as suitable tools for use in clinical daily practice. Deep learning is a subfield of artificial intelligence, which is highly flexible and have been applied in various areas of both basic and clinical research. One of the applications of deep learning that greatly benefits from this paradigm is image segmentation.

Image segmentation has an important role in medical diagnosis and research. Its results can help professionals to obtain accurate pathological regions, thus reducing the possibility of artificial empirical misjudgment. In its early days, experienced professionals worked assiduously to delineate diseased areas for better diagnosis. This kind of complete manual segmentation approach requires a

* Correspondence: zhaop@zju.edu.cn; zju_tz@zju.edu.cn
†Dalei Jiang, Yin Wang and Feng Zhou contributed equally to this work.
[4]Department of Medical Oncology, The First Affiliated Hospital, Zhejiang University School of Medicine, Hangzhou, CN, China
Full list of author information is available at the end of the article

significant amount of domain knowledge. Besides, it is time-consuming and laborious and could be severely affected by inter- and intra-observer variability [2]. In order to reduce the burden of manual segmentation, researchers have developed many automatic approaches [3–5] for medical image segmentation. These approaches range from early stage low-level-feature based methods such as Otsu [6], region growth [7], and grab-cut [8] to deep convolutional neural network-based methods such as FCNs (fully convolutional networks) [9] and U-Net [10]. In medical image segmentation, U-Net is one of the most popular network architectures and has been commonly used in various medical imaging modalities [10, 11]. Fully automatic segmentation methods do not require user interaction, which greatly decreases the costs. However, this is a double-edged sword since there is no means for them to refine the segmentation result when it is not satisfactory. Thus, even the more sophisticated DCNN (deep convolutional neural network)-based methods could not achieve accurate and robust results that are clinically useful.

To address the limitations of automatic segmentation approaches, a trade-off was made between complete manual and fully automatic approaches. The interactive segmentation approach, which uses user interactions as input signals to guide segmentation, could alleviate the clinician's burden and, at the same time, achieve satisfactory segmentation results by incremental refinement. ITK-SNAP [12] provides an interactive segmentation mechanism that employs an active contour model for segmentation that accepts user-provided seeds or blobs as a starting point. Although it is 'Interactive', it lacks the ability to refine the segmentation result, and its underpinning model is not powerful enough to model the variability of our target medical images. Similar approaches such as random walks [13], graph cuts [14] and grabcut [8] provide mechanisms to incrementally refine the segmentation result, but the performances of those methods are limited by the representativeness of the underpinning model since they only incorporate primitive low-level features for inference. Li [15] proposed a stacked adversarial learning (SAL) method based on an FCN to improve the dermoscopic image segmentation method. The authors build upon generative adversarial networks with a novel SAL architecture such that skin lesion features can be learned iteratively in a class-specific manner. However, the stability of the generated samples is not satisfactory; and complex parameter adjustment is required, which increases the time costs of the training model. Liu [16] proposed efficient skin lesion segmentation based on an improved U-net model, which mainly includes batch normalization and dilated convolution. However, the model regards dark regions as regions of interest, and the segmentation performance of the model was poor in a few cases where the region of interest was brighter than the surrounding skin region. Qin [17] proposed an asymmetric encode-decode network with two decoding paths for skin lesion segmentation. A skip pathway was designed to transfer the more representative features from the encoder to the decoder. However, in clinical work, the images are much more complex than the experimental data. Without the interaction of medical personnel, it is difficult to obtain a satisfactory segmentation result.

Due to the recent explosive growth of deep neural networks [18, 19] and their application in semantic segmentation problems [9, 10, 20, 21], the method for interactive object segmentation has experienced a swift change from traditional low-level-feature-based algorithms [13, 14, 22, 23] to deep convolutional neural networks [24–31]. Consequently, the results, in terms of accuracy and the intensity of user interaction, have improved tremendously.

In this article, in order to compensate for the defects of traditional methods that cannot effectively extract the deep information of images and because the depth model is not flexible enough for clinical use, we proposed a novel algorithm that combines the representational power of deep convolutional neural networks (DCNNs) and the flexibility of grab-cut (8)-like user interaction. While previous methods normally combine the initial segmentation task with the following refinement task in a single network, we explicitly separate them by designing individual sub-networks. One sub-network is SBox-Net, and the other sub-network Click-Net. Compared to the current skin lesion segmentation methods, our model has the characteristics of flexibility and precision. Clinicians can judge whether the segmentation results are satisfactory according to the segmentation model of the first-stage SBox-Net. If the segmentations are satisfactory, further refinement is not required. Otherwise, we can click on the pathological pictures. Click-Net will automatically process clinicians' clicks and refine the segmentation results. Comprehensive experiments are then conducted to demonstrate the effectiveness of our approach. The workflow of our method is shown in Fig. 1.

Methods

The network structure consists of three main parts: 1. The network encoder, which was used to encode the features of different levels of abstraction. 2. SBox-Net. In addition to the feature encoder, we obtain features of two abstract levels, namely, low-level features and high-level features. SBox-Net highlights high-level features by reducing the number of channels of low-level feature mapping and obtains rough prediction segmentation. 3. Click-Net, whose main goal is to restore details

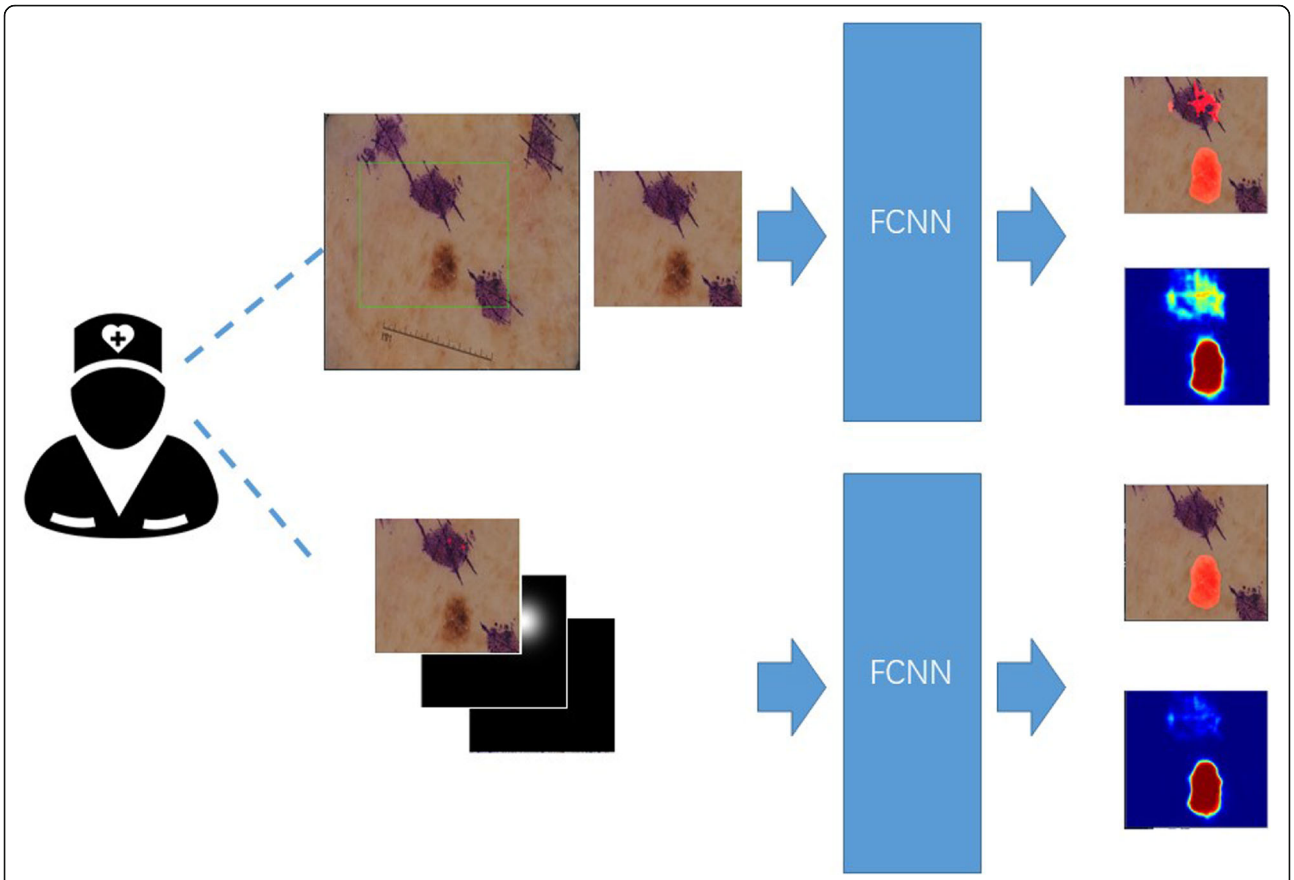

Fig. 1 The workflow of our method. First, a box is drawn by the user who possesses domain knowledge to delineate the region of interest (ROI). The ROI is then cropped out to exclude most of the useless background and fed into a fully convolutional neural network to generate a segmentation. Most of the time, this segmentation will be very satisfactory. In addition, no further process is required. However, not all situations are ideal. Take the one in the figure as an example. There is a blot that we would like to remove. To address this type of situation, the user can provide another FCN with positive and negative clicks, which are transformed into Gaussian centred distance maps, to further refine the segmentation

according to the user's clicks. We generated a Gaussian distance map with the same size as the input image by using the user's clicks and used the Gaussian distance map as the weight of the segmentation results of the final upsampling layer of Click-Net. Finally, the segmentation results of Click-Net are refined according to the weight. The architecture of our proposed method is illustrated in Fig. 2.

SBox-Net

Our SBox-Net was designed to be a binary segmentation network. Except for the last inference layer, there is no difference between our SBox-Net and a normal semantic segmentation network. Thus, SBox-Net could smoothly utilize a pre-trained state-of-the-art semantic segmentation network by simply replacing the top segmentation layer of an existing state-of-the-art model with our binary segmentation layer. We can then fine-tune the network to fit our goal. This strategy saves us considerable training time and computational resources. As for the simulation of a user drawing a surrounding box, we take the bounding box of the ground truth mask jittered randomly by up to 30 px in each direction. In this way, the randomness of user behaviour is well modelled.

As shown in Supplementary Fig. 1, in SBox-Net, in order to concatenate the shallow and deep features in the encoder, the features extracted from the encoder should be 'concatenated' first. Then, the 3×3 convolution is used to refine the features, and the refined features have deeper semantic features. During the upsampling process, factor = 4 bilinear interpolation is used to recover the pixel-wise prediction of the image resolution entered in the encoder. We define this prediction as a rough prediction; and in clinical segmentation, if the physician is satisfied with this result, he can obtain a satisfactory result without any manipulation. Otherwise, he can use Click-Net for refinement. Section "User interaction simulation" and "User interaction transformation" introduce some preliminary information about Click-Net, and section "Click-Net" describes Click-Net in detail.

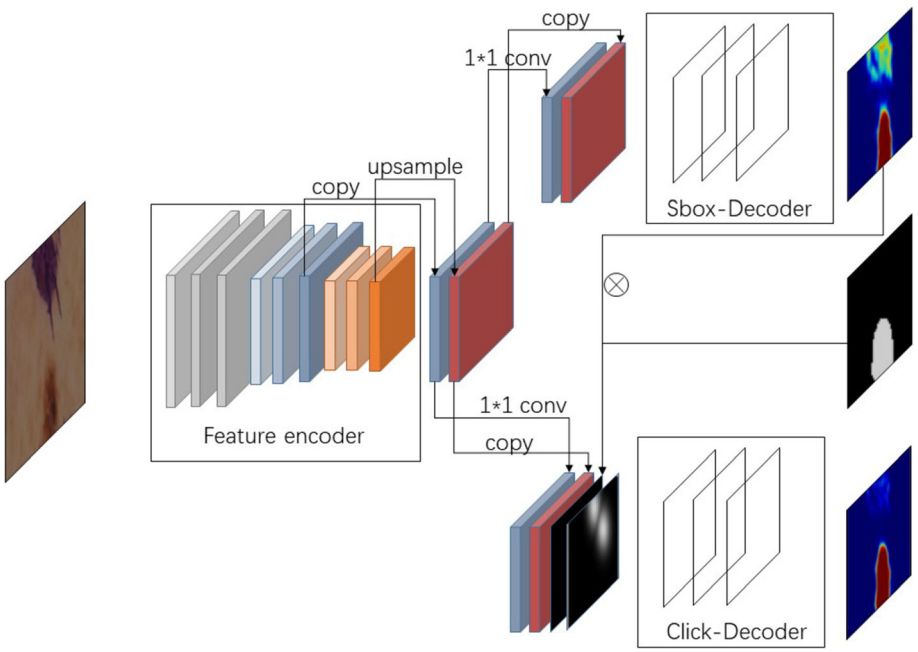

Fig. 2 The overall architecture of our proposed method. It is composed of three major parts: a feature encoder for encoding features at different abstract levels, an SBox-Net for initial segmentation and a Click-Net for refinement. Using the feature encoder, we obtain feature maps at two levels of abstraction, namely, low-level features and high-level features. Our SBox-Net is used to predict segmentation at a coarse level; thus, we highlight our high-level features by reducing the number of channels of low-level feature maps. In addition, in our Click-Net, we reduce the channels of high-level features since our goal is to recover details according to user clicks. All previously mentioned channel reduction operations are performed by 1*1 convolution. Finally, we simulate user clicks by sampling from differences of SBox-Net segmentation and the ground truth (denoted by ⊗ in the figure)

User interaction simulation

Surrounding box simulation is quite straight-forward, as previously stated in section "SBox-Net". Click simulation requires slightly more caution.

Before delving into the details of click simulations, we need to go through the workflow of a typical interactive object segmentation process. First, a user draws a surrounding box around the target object. Based on the surrounding box, the SBox-Net will perform one pass of inference on the patch of the image cropped by the surrounding box. If the result needs to be refined, typically, it would contain two types of mistakes, namely, extra pixels and left-behind pixels (from a user's perspective). In these two types of mistakes, a user adds clicks to refine the segmentation result.

By separating our architecture into SBox-Net and Click-Net, we can perfectly simulate those two types of mistakes during training time. After a forward pass of SBox-Net, we obtain a preliminary result. We then can calculate the differences between the preliminary result and the ground truth mask, obtaining the false positives and false negatives of the preliminary result, which are a close simulation of the two types of mistakes previously mentioned. Thus, we can directly sample clicks on false positives and false negatives (see Fig. 2). Our strategy for simulating user clicks is simpler, more straightforward and more effective than that introduced by [24].

User interaction transformation

At the inference time of our Click-Net, a user can provide positive and negative clicks to refine the results of SBox-Net. All user interactions can be grouped into two sets: a positive click set S^1, which contains all user-proved positive clicks; and a negative click set S^2, which contains all user-provided negative clicks. A Gaussian distance transformation was used to transform those two sets into two separate channels G^1 and G^2, respectively. Both G^1 and G^2 were initialized to zero. Let $G^1_{(m,n)}$ and $G^2_{(m,n)}$ be the elements at location (m, n) in matrices G^1 and G^2, respectively, which are calculated by:

$$G^1_{(m,n)} = \max_{s_{i,j} \in S^1} e^{-\frac{4*\left((m-i)^2+(n-j)^2\right)}{R^2}} \quad (1)$$

$$G^2_{(m,n)} = \max_{s_{i,j} \in S^2} e^{-\frac{4*\left((m-i)^2+(n-j)^2\right)}{R^2}} \quad (2)$$

where R is a radius parameter that controls the area of influence of a user click. After the transformation of user clicks, we concatenate the feature maps extracted from

SBox-Net with G^1 and G^2, which are then fed into Click-Net for further processing.

Click-Net

The workflow of Click-Net is shown in Supplementary Fig. 2. On the basis of SBox-Net segmentation, our Click-Net was designed specifically for responding to user clicks when a user is seeking to refine the segmentation result. In order to achieve this, the training data for Click-Net must be collected carefully. The click simulation strategy is described in detail in section "User interaction simulation". In Click-Net, we first transform the positive and negative clicks into two Gaussian centred maps. We then concatenate the transformed Gaussian maps with feature maps extracted from SBox-Net, which are then fed into Click-Net to generate our final segmentation. Contrary to previous works [24–26], we do not concatenate the transformed user clicks with raw images directly but with feature maps instead. The main motivation behind this is to decouple the segmentation process and the refinement process. Besides, it is obvious that user clicks are informative both semantically (positive or negative) and spatially (the absolute position of the clicks inside the surrounding box). Thus, their level of abstraction is more compatible with high-level features instead of low-level features such as raw pixels.

Inspired by the famous ResNet [19], which incorporates residual blocks to tackle the exploding gradient problem and significantly boosts the performance of artificial networks, we designed our Click-Net as a residual refinement network. Before yielding the final segmentation, our Click-Net fuses its output with that of the SBox-Net, which makes it in effect a residual refinement network. The fusion process considers the number and position of user clicks. We transform the user clicks into a weight map using Gaussian distance transformation. Unlike in user interaction transformation depicted in 2.3, we do not differentiate between positive and negative clicks. Besides, instead of setting the pixel value to the maximum Gaussian distance from all click points, we add those distances up. Finally, the radius parameter, R, which controls the area of influence of a user click, is set to a much larger value, allowing each click to adjust the weight of a much broader area. The final weight map is given as:

$$W_{(m,n)} = \sum_{s_{i,j} \in \left(S^1 \cup S^2\right)} e^{-\frac{4*\left((m-i)^2+(n-j)^2\right)}{R^2}} \tag{3}$$

In Formula 3, $W_{(m,n)}$ represents the sum of the Gaussian distances between all the click points and the element at location (m, n) in matrix W. $e^{-\frac{4*\left((m-i)^2+(n-j)^2\right)}{R^2}}$ represents the Gaussian distance from a single click point $s_{i,j}$ in the set $S^1 \cup S^2$ to the element at location (m, n) in matrix W.

After obtaining the weight map, we can fuse the SBox-Net result, denoted B, with the Click-Net result, denoted C, to produce our final result, denoted F, using the formula:

$$F = W * C + B \tag{4}$$

where $*$ is the bitwise multiplication operator and + is the bitwise addition operator.

Results

Datasets

Our method has been trained and evaluated on two publicly available datasets, the ISIC dataset [32] and PH2 [33]. The ISIC dataset was used for a skin image analysis challenge hosted by the International Skin Imaging Collaboration (ISIC). The challenge was hosted in 2018 at the Medical Image Computing and Computer Assisted Intervention (MICCAI) conference in Granada, Spain. The dataset included over 12,500 images across 3 tasks, including lesion segmentation, attribute detection, and disease classification. To train and evaluate our interactive segmentation method, we selected the dataset for lesion segmentation, which consists of 2596 skin lesion images with corresponding segmentation masks. We divide the dataset into two independent and equally distributed parts: one part for training and the other part for model evaluation.

The other dataset, PH2, was provided by a joint collaboration between the University of Porto and University of Lisbon in conjunction with the Department of Dermatology at the Pedro Hispanio Hospital in Matosinos, Portugal. The dataset was mainly created to provide a common dataset that may be used for the performance evaluation of different computer-aided dermoscopic image diagnosis systems. The dataset consists of 200 dermoscopic images with corresponding segmentations, including 80 common nevi, 80 atypical nevi, and 40 malignant melanomas. All images are 8-bit RGB colour images with a resolution of 768*560 px. All dermoscopic images were carefully acquired using a magnification of 20* under unchanged conditions.

Training configuration

Our SBox-Net utilizes the DeepLab V3+ model pretrained on the Pascal VOC dataset with the last inference layer replaced. We then fine-tune our model using stochastic gradient descent with a batch size of 8 objects for 50 epochs. The learning rate is set to 0.01 with a momentum of 0.9 and a weight decay of 0.0005. Inspied by

[34], we exploit the 'poly' learning rate policy, which multiplies the learning rate by $(1 - iter/\max\_iter)^{power}$.

We trained our Click-Net with a learning rate of 0.1, while the other hyper-parameters remained the same. The objects we used to train Click-Net are those that are not accurately segmented by SBox-Net (IoU less than 0.9). We simulate user clicks by randomly sampling the false positives and false negatives of the SBox-Net prediction.

The experiments are conducted using the PyTorch framework. All our networks are trained on a single NVIDIA GeForce GTX TITAN X GPU with 12 GB of memory. The training of SBox-Net takes approximately 11 h, and Click-Net takes 6 h. The hyper-parameters of Sbox-Net and Click-Net are shown in Supplementary Table 1. We trained our Click-Net with a learning rate of 0.1 while the other hyper-parameters remained the same.

Performance evaluation metrics

Our proposed method is composed of two loosely coupled modules with Click-Net using the feature maps extracted from SBox-Net, while our SBox-Net works fairly well without the knowledge of Click-Net. We evaluate our method in two stages. In the first stage, the performance of SBox-Net is evaluated. Then, we will show how our Click-Net improves the segmentation result of SBox-Net. The following performance metrics were used in evaluating our algorithm: the sensitivity (Sen), specificity (Spe), dice coefficient (Dic), accuracy (Acc) and intersection over union (IoU). The sensitivity, also known as the true positive rate, is the number of correctly segmented lesion pixels, and the specificity is the ratio of correctly segmented non-lesion pixels. The Dice coefficient evaluates the similarity between the segmented lesions and the underlying ground truth. The accuracy shows the overall pixel-wise segmentation performance. Finally, IoU, as its name implies, measures the proportion of the intersection over the union between the segmentation and the ground truth. All aforementioned evaluation metrics are calculated by the following formulas:

$$Acc = \frac{TP + TN}{TP + FN + TN + FP} \tag{5}$$

$$Sen = \frac{TP}{TP + FN} \tag{6}$$

$$Spe = \frac{TN}{TN + FP} \tag{7}$$

$$Dic = \frac{2*TP}{2*TP + FN + FP} \tag{8}$$

$$IoU = \frac{Area\ of\ intersection}{Area\ of\ union} \tag{9}$$

Segmentation performance

We demonstrate our algorithm on two publicly available datasets, ISIC 2018 and PH2. First, we present the performance of SBox-Net on those two datasets as it is the first stage of our proposed pipeline. Table 1 summarizes the segmentation performance of SBox-Net. From this table, it is obvious that all the evaluation metrics results on the ISIC dataset are higher than those on the PH2 dataset. The major reason is that the ISIC dataset is larger than the PH2 dataset. As a result, the model with more training data has better performance. As we can see, the proposed model achieves compelling results on both datasets, reaching an accuracy, a sensitivity, a specificity, a Dice coefficient, and an intersection over union of 94.40, 94.27, 91.60, 91.60, and 88.22% on PH2, respectively; compared to 96.23, 97.58, 92.52%, 92.93, and 90.89% on ISIC, respectively. Additionally, Fig. 3 illustrates some examples of the segmentation results of our SBox-Net.

In Table 2, the performance of stacking a Click-Net on top of SBox-Net is listed. The performance improvement is significant considering the relatively small amount of computation required by the light-weight Click-Net. The table shows that the improvement made on the PH2 dataset is larger than that on the ISIC dataset. This is partly because our SBox-Net already did a very good job on the ISIC dataset (reaching a 90% IoU). There is less room left for improvement. Conversely, the PH2 dataset provides a good place for our Click-Net to shine.

In Fig. 4, the overall performance of our method on each single image instance of both datasets is depicted as scatter plots. These plots show that our method achieved very good scores on most of the samples in those two datasets. In the figure, the horizontal axis represents the index for each sample, and the vertical axis represents the evaluation metric score. Some outliers with metric scores lower than 0.7 can be spotted. Each of these outliers corresponds to a sample, and these samples are difficult to segment because the labelling is not precise or the sample is inherently hard.

Interactive performance

In this subsection, we present the performance of our algorithm in terms of user interactions. In Table 3, both

Table 1 Skin lesion segmentation performance of SBox-Net

Datasets	Acc (%)	Sen (%)	Spe (%)	Dic (%)	IoU (%)
PH2	94.40	94.27	91.60	91.60	88.22
ISIC	96.23	97.58	92.52	92.93	90.89

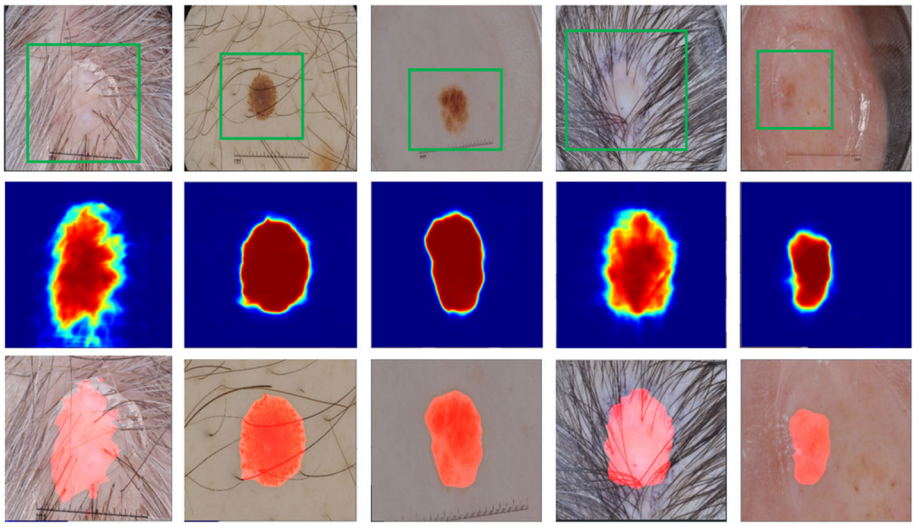

Fig. 3 Example segmentations produced by our SBox-Net. The first row presents the original skin lesion images. The green boxes in those images are surrounding boxes provided by users that serve as guiding signals for SBox-Net. The middle row lists the segmentation confidence map generated by SBox-Net for the region of interest cropped by the surrounding box. The third row shows cropped images with corresponding overlayed segmentations

the number of clicks and estimated time required to achieve a certain IoU on the two datasets are listed. Since our SBox-Net requires at least 2 clicks to draw a surrounding box, we fix the amount of user interaction of our SBox-Net to 2 clicks. Drawing a surrounding box is an easy task that takes less than 1 s on average. Identifying an ill-segmented area and then clicking requires more attention. Our simulation experiments showed that this process takes 1.4 s on average. In the evaluation process, we adopted the same sampling strategy as depicted in section "User interaction simulation".

Residual refinement performance

In Fig. 5, we illustrate our residual refinement process. After a user draws a surrounding box around the target lesion region, our SBox-Net produces an initial segmentation. In cases where the initial segmentation is unsatisfactory, Click-Net will be invoked. After each click a user inputs, Click-Net transforms the clicks into Gaussian distance maps. Concatenated with feature maps extracted from SBox-Net, the newly formed input is then fed into Click-Net to generate its immediate output. Before yielding the final segmentation, our Click-Net first reevaluates its output by multiplying the weight map (see "Click-Net") pixel-wise. The output is then added to that of SBox-Net, reaching the final segmentation. Note that the process can be repeated to refine the result step by step. A few examples are illustrated in Fig. 5.

Ablation study

Global Context Extractor: As a common practice in recent deep segmentation methods, various types of global context extractors have been proven to be very useful. The most successful ones are Atrous Spatial Pyramid Pooling (ASPP) and Pyramid Scene Parsing (PSP). We conducted experiments to evaluate the effectiveness of ASPP and PSP in our interactive segmentation task. As a comparison, we have also experimented with a naive convolution layer as our global context extractor. The results of our experiments are shown in Table 4. As the table shows, if we only use a naive convolution layer as our global context extractor, its performance only reaches an MIoU of 83.23%. We add the PSP module to obtain more abundant global features and significantly improved segmentation results. It is important to note that ASPP is superior to PSP in the experimental results, and ASPP achieves the optimal performance, reaching an MIoU of 88.22% with SBox-Net alone and an MIoU of 90.36% when combined with Click-Net.

Table 2 Skin lesion segmentation performance of SBox-Net + Click-Net with the number of clicks restricted to 2. The numbers in parentheses are the relative improvements made by adding a Click-Net

Datasets	Acc (%)	Sen (%)	Spe (%)	Dic (%)	IoU (%)
PH2	95.53(+ 1.13)	96.34(+ 2.07)	93.82(+ 2.22)	93.12(+ 1.52)	90.36(+ 2.14)
ISIC	96.86(+ 0.63)	98.48(+ 0.90)	92.63(+ 0.11)	94.06(+ 1.13)	92.31(+ 1.42)

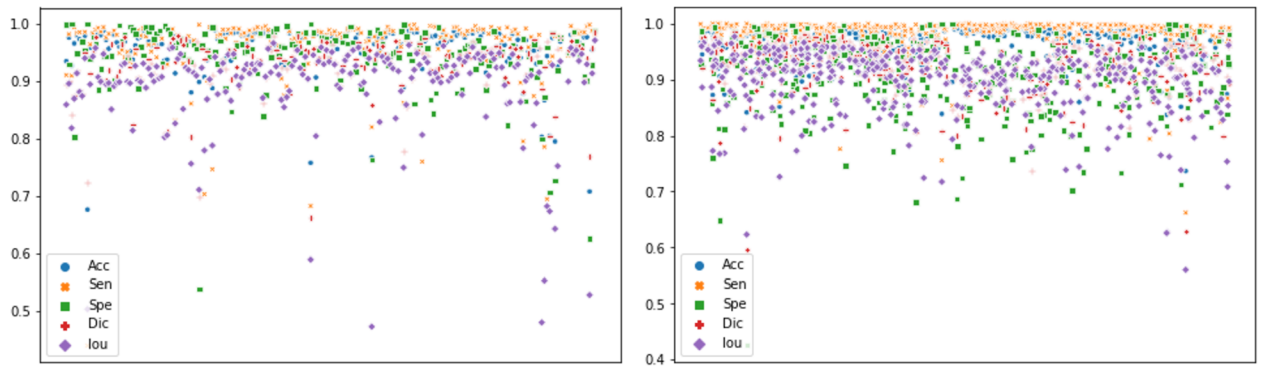

Fig. 4 The distribution of performance metrics on both datasets. We can see from the scatter plot that apart from a few outliers, our method achieved very good scores on most of the samples, concentrated at approximately 0.9 and above

Discussion

Related works

Image Segmentation: For the first time, an FCN [9] adopted convolutional neural networks (CNNs) for dense prediction by replacing fully connected layers with convolutional layers. This innovation enables the FCN to process different sized input images and produce segmentation maps accordingly. Almost all the subsequent state-of-the-art approaches for segmentation followed this paradigm. To address the resolution loss due to pooling layers in CNNs, two types of architectures have emerged. The first one is the encoder-decoder architecture. U-Net [10] is one of the representatives of this class. The encoder module gradually reduces the spatial resolution with pooling layers while the decoder recovers the object details and spatial dimension using shortcut connections. Another branch of architectures drops pooling layers altogether and instead uses a special type of convolution layer, called dilated/atrous convolution. Representatives of this class include DeepLab [35] and PSPNet [36].

Interactive Segmentation: Various types of user interaction have been exploited for interactive segmentation. Xu et al. [24] proposed a method to incorporate user clicks into an FCN [9] model by transforming user clicks into Euclidean distance maps, which are then concatenated with the RGB channel of the original image and fed into the iFCN model. This paradigm is followed by subsequent interactive segmentation architecture. A follow-up model by Liew et al., called RIS-Net [26], improved the result of Xu et al. [24] by focusing on local regions that are poorly segmented. RIS-Net exploits the local regional context around the user click pair along with multiscale global contextual information to improve the segmentation result. DEXTR [25], a more recent method proposed by Maninis et al., uses four extreme points on the object as a supervisory signal to guide the segmentation. Slightly different from previous methods, DEXTR encodes these points as a Gaussian map. DeepIGeoS [37] strictly followed the iFCN paradigm but transformed user-provided clicks and scribles into a geodesic distance map instead of a Euclidean distance map for interactive medical image segmentation. In the research of Wang et al. [38], a method for image-specific fine-tuning at test time for a CNN model was proposed.

To segment an object in an image, a box around the object, which we called the surrounding box, is first drawn by users. Unlike a tight, accurate bounding box, the surrounding box is much looser and thus a user-friendlier version of the bounding box. The surrounding box is a user-provided guiding signal to exclude most of the background from the image. After feeding the patch of the image cropped by the surrounding box into SBox-Net, an initial segmentation result is directly returned. If the result is satisfactory, no more user interaction is required. Otherwise, users provide positive and negative clicks, which are then transformed into Gaussian-centred maps. Those maps together with feature maps extracted from SBox-Net are then fed into Click-Net for further refinement. Through careful architectural design, our method requires only one feed-forward pass through SBox-Net while Click-Net reuses the feature maps computed by SBox-Net in the following refinement iterations. Another key improvement that separates our method from the existing methods is that our Click-Net

Table 3 Number of clicks and estimated time to achieve a certain IoU on 2 datasets

Datasets	IoU (%)	Clicks	Time (seconds)
PH2	90	3.67	3.4
	95	6.72	7.6
ISIC	90	2	< 1
	95	4.84	4.9

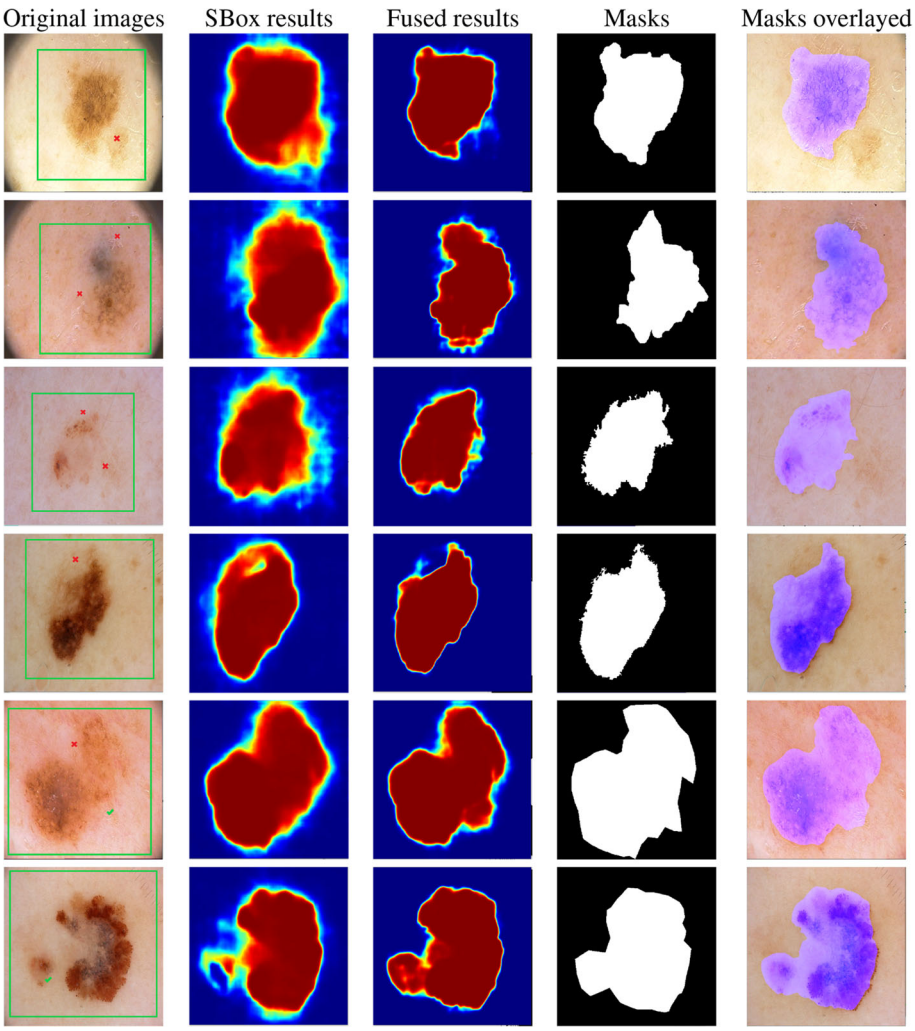

Fig. 5 Residual refinement process of our proposed method. In the rare situation of our SBox-Net producing unsatisfactory segmentation results, our Click-Net can be invoked to refine the results. First, a surrounding box (the green box in column a) is drawn by the user to select the region of the target lesion. Based on the surrounding box, a preliminary result is returned by SBox-Net. While the results are not satisfactory, the user can provide positive clicks (the green check marks in column a) and negative clicks (the red cross marks in column a) to further refine the segmentations. The third column shows the fused segmentation maps of Click-Net and SBox-Net compared to the SBox-Net results and ground truth masks in the second and fourth columns, respectively. In the final column, the fused segmentations overlayed on the cropped original images are presented

is designed specifically for responding to user clicks when a user seeks to refine an unsatisfactory result. We achieve this by applying a simple, straightforward, yet effective sampling strategy in the training process of Click-Net. User intentions are well captured by our Click-Net. Apart from addressing the issues stated above, the modulization design of our architecture brings us several advantages. First, since SBox-net operates directly on raw images, instead of (image, transformed user interaction) pairs as in [24–26], our SBox-Net can smoothly repurpose pre-trained, state-of-the-art semantic segmentation models for our task by simply replacing the inference layer. This strategy saves us a huge amount of training time and computational resources. Second, as SBox-net and Click-Net are only loosely coupled, we could conveniently test different Click-Nets in a plug-and-play style. For instance, we could test an aggressive Click-Net at some time and a moderate Click-Net at others. Alternatively, to push the limit even further, we could test a totally different refinement network with another type of user interaction, such as scribbles. Our architecture, therefore, can be seen more

Table 4 Comparison of different global context extractors

Global Context Extractor	PSP	ASPP	Naive
SBox IoU (%)	0.8632	0.8822	0.8323
SBox + Click IoU (%)	0.8942	0.9036	0.8532

broadly as a flexible framework for interactive skin lesion segmentation and can be readily extended to accommodate various types of user interactions. However, this paper focuses only on clicks, leaving others for future research.

In summary, the key contributions of this paper are summarized as follows:

- We combined a deep convolutional neural network with a grabcut-like user interaction to tackle the interactive skin lesion segmentation problem.
- We decoupled the refinement task with the initial segmentation task using the modularized design of the network architecture, which greatly enhances the flexibility of our model, facilitates reusing computations at inference time and allows our Click-Net to be trained in a way that fully captures the intention of users.
- We exploit the pre-trained, state-of-the-art semantic segmentation model for SBox-Net. With few changes in the model and a small training time budget, we could achieve a compelling result.

Conclusion

In this paper, we present an interactive method for skin lesion segmentation. We approach the problem as a two-stage pipeline. First, a user uses a surrounding box to select the skin lesion of interest. A preliminary segmentation result is returned by our SBox-Net. Then, if the result in the first step is unsatisfactory, a light-weight Click-Net is invoked to further refine the segmentation. Extensive experiments on two public datasets, PH2 and ISIC, proved the effectiveness of our approach.

Abbreviations

ROI: Region of interest; CNNs: Convolutional neural networks; ISIC: International Skin Imaging Collaboration; ASPP: Atrous Spatial Pyramid Pooling; PSP: Pyramid Scene Parsing; FCN: Fully convolutional networks; DCNN: Deep convolutional neural network; SAL: Stacked adversarial learning

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s13326-021-00255-z.

Additional file 1 : **Supplementary Figure 1.** The workflow of Sbox-Net. In SBox-Net, in order to integrate the shallow and deep features in the encoder, the features extracted from the encoder should be 'concatenated' first. Then, the 3×3 convolution is used to refine the features, and the refined features have deeper semantic features.

Additional file 2 : **Supplementary Figure 2.** The workflow of Click-Net. In Click-Net, we first transform the positive and negative clicks into two Gaussian centred maps. We then concatenate the transformed Gaussian maps with the feature maps extracted from SBox-Net, which are then fed into Click-Net to generate our final segmentation.

Additional file 3 : **Supplementary Table 1.** Hyper-parameters of SBox-Net and Click-Net.

Acknowledgements

Not applicable.

Authors' contributions

D.J. wrote the manuscript, H.M., W. F, W.Z. analyzed the data, P.Z. and Z.T. designed the study and Y.W., F.Z. revised the manuscript. The author(s) read and approved the final manuscript.

Funding

This work was supported by the National Natural Science Foundation of China (82074208), Natural Science Foundation of Zhejiang Province (LY20H160033), National Major Scientific and Technological Special Project for Significant New Drugs Development during the Thirteenth Five-year Plan Period 2020ZX 09201–003.

Availability of data and materials

All the data can be downloaded at https://challenge2018.isic-archive.com.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

All authors declare no conflict of interest.

Author details

[1]Echocardiography and Vascular Ultrasound Center, The First Affiliated Hospital, Zhejiang University School of Medicine, Hangzhou, CN, China. [2]College of Computer Science and Technology, Zhejiang University, Hangzhou, CN, China. [3]Department of Urology, The First Affiliated Hospital, Zhejiang University School of Medicine, Hangzhou, CN, China. [4]Department of Medical Oncology, The First Affiliated Hospital, Zhejiang University School of Medicine, Hangzhou, CN, China.

Received: 6 December 2020 Accepted: 11 November 2021
Published online: 18 December 2021

References

1. Duan Y, Sun X, Che H, Cao C, Li Z, Yang X. Modeling data, information and knowledge for security protection of hybrid IoT and edge resources. Ieee Access. 2019;7:99161–76. https://doi.org/10.1109/ACCESS.2019.2931365.
2. Salonen R, Haapanen A, Salonen JT. Measurement of intima-media thickness of common carotid arteries with high-resolution b-mode ultrasonography - interobserver and intraobserver variability. Ultrasound Med Biol. 1991;17(3):225–30. https://doi.org/10.1016/0301-5629(91)90043-V.
3. Eugenio Iglesias J, Sabuncu MR. Multi-atlas segmentation of biomedical images: a survey. Med Image Anal. 2015;24(1):205–19. https://doi.org/10.1016/j.media.2015.06.012.
4. Peng B, Zhang L, Zhang D. A survey of graph theoretical approaches to image segmentation. Pattern Recogn. 2013;46(3):1020–38. https://doi.org/10.1016/j.patcog.2012.09.015.
5. Petitjean C, Dacher J-N. A review of segmentation methods in short axis cardiac MR images. Med Image Anal. 2011;15(2):169–84. https://doi.org/10.1016/j.media.2010.12.004.
6. Otsu N. Threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern. 1979;9(1):62–6. https://doi.org/10.1109/TSMC.1979.4310076.
7. Rai HGN, Nair TRG. Gradient based seeded region grow method for CT angiographic image segmentation. CoRR. 2010;1:1–6. https://arxiv.org/abs/1001.3735.
8. Rother C, Kolmogorov V, Blake A. "GrabCut" - interactive foreground extraction using iterated graph cuts. ACM Trans Graph. 2004;23(3):309–14. https://doi.org/10.1145/1015706.1015720.
9. Long J, Shelhamer E, Darrell T, Ieee. Fully convolutional networks for semantic segmentation. 2015 Ieee conference on computer vision and pattern recognition; 2015. p. 3431–40.

10. Ronneberger O, Fischer P, Brox T. U-Net: convolutional networks for biomedical image segmentation. In: Navab N, Hornegger J, Wells WM, Frangi AF, editors. Medical image computing and computer-assisted intervention, Pt Iii. Lecture Notes in Computer Science. 9351; 2015. p. 234–41.
11. Chen L, Bentley P, Mori K, Misawa K, Fujiwara M, Rueckert D. DRINet for medical image segmentation. IEEE Trans Med Imaging. 2018;37(11):2453–62. https://doi.org/10.1109/TMI.2018.2835303.
12. Yushkevich PA, Piven J, Hazlett HC, Smith RG, Ho S, Gee JC, et al. User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability. Neuroimage. 2006;31(3):1116–28. https://doi.org/10.1016/j.neuroimage.2006.01.015.
13. Grady L. Random walks for image segmentation. IEEE Trans Pattern Anal Mach Intell. 2006;28(11):1768–83. https://doi.org/10.1109/TPAMI.2006.233.
14. Boykov YY, Jolly MP, Ieee Computer Society; Ieee Computer S. Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images; 2001. p. 105–12.
15. Bi L, Feng D, Fulham M, Kim J, Ieee. Improving skin lesion segmentation via stacked adversarial learning. 2019 Ieee 16th international symposium on biomedical imaging. IEEE International Symposium on Biomedical Imaging; 2019. p. 1100–3.
16. Liu L, Mou L, Zhu XX, Mandal M, editors. Skin lesion segmentation based on improved U-net. 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE). IEEE; 2019. p. 1–4.
17. Qin K, Sun D, Zhang S, Zhao H, editors. Asymmetric encode-decode network with two decoding paths for skin lesion segmentation. 2020 5th International Conference on Biomedical Imaging, Signal Processing. 2020.
18. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, et al. ImageNet large scale visual recognition challenge. Int J Comput Vis. 2015;115(3):211–52. https://doi.org/10.1007/s11263-015-0816-y.
19. He K, Zhang X, Ren S, Sun J, Ieee. Deep residual learning for image recognition. 2016 Ieee conference on computer vision and pattern recognition. IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 770–8.
20. Zhao H, Shi J, Qi X, Wang X, Jia J, Ieee. Pyramid scene parsing network. 30th Ieee Conference on Computer Vision and Pattern Recognition. IEEE Conference on Computer Vision and Pattern Recognition; 2017. p. 6230–9.
21. Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Trans Pattern Anal Mach Intell. 2018;40(4):834–48. https://doi.org/10.1109/TPAMI.2017.2699184.
22. Bai X, Sapiro G. Geodesic matting: a framework for fast interactive image and video segmentation and matting. Int J Comput Vis. 2009;82(2):113–32. https://doi.org/10.1007/s11263-008-0191-z.
23. Vezhnevets V, Konouchine V, editors. GrowCut: Interactive multi-label ND image segmentation by cellular automata. proc of Graphicon: Citeseer; 2005.
24. Xu N, Price B, Cohen S, Yang J, Huang T, Ieee. Deep interactive object selection. 2016 Ieee conference on computer vision and pattern recognition. IEEE Conference on Computer Vision and Pattern Recognition; 2016. p. 373–81.
25. Maninis KK, Caelles S, Pont-Tuset J, Van Gool L, Ieee. Deep extreme cut: from extreme points to object segmentation. 2018 Ieee/Cvf conference on computer vision and pattern recognition. IEEE Conference on Computer Vision and Pattern Recognition; 2018. p. 616–25.
26. Liew J, Wei Y, Xiong W, Ong S, Feng J, editors. Regional Interactive Image Segmentation Networks. 2017 IEEE International Conference on Computer Vision (ICCV). 2017.
27. Hu Y, Soltoggio A, Lock R, Carter S. A fully convolutional two-stream fusion network for interactive image segmentation. Neural Netw. 2019;109:31–42. https://doi.org/10.1016/j.neunet.2018.10.009.
28. Feng R, Liu X, Chen J, Chen DZ, Gao H, Wu J. A deep learning approach for colonoscopy pathology WSI analysis: accurate segmentation and classification. IEEE J Biomed Health Inform. 2020;25(10):3700–8. https://doi.org/10.1109/JBHI.2020.3040269.
29. Chen J, Ying H, Liu X, Gu J, Feng R, Chen T, et al. A transfer learning based super-resolution microscopy for biopsy slice images: the joint methods perspective. IEEE/ACM Trans Comput Biol Bioinform. 2020:1. https://doi.org/10.1109/TCBB.2020.2991173.
30. Chen T, Xu J, Ying H, Chen X, Feng R, Fang X, et al. Prediction of extubation failure for intensive care unit patients using light gradient boosting machine. IEEE Access. 2019;7:150960–8. https://doi.org/10.1109/ACCESS.2019.2946980.
31. Lin B, Deng S, Gao H, Yin J. A multi-scale activity transition network for data translation in EEG signals decoding. IEEE/ACM Trans Comput Biol Bioinform. 2020;18(5):1699–709. https://doi.org/10.1109/TCBB.2020.3024228.
32. Codella N, Rotemberg V, Tschandl P, Celebi ME, Dusza S, Gutman D, et al. Skin lesion analysis toward melanoma detection 2018: a challenge hosted by the international skin imaging collaboration (isic). 2019.
33. Mendonca T, Celebi M, Mendonca T, Marques JJD. Ph2: A public database for the analysis of dermoscopic images; 2015.
34. Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille ALJ. Semantic image segmentation with deep convolutional nets and fully connected crfs; 2014.
35. Chen L C, Papandreou G, Kokkinos I, et al. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs[J]. Comp Sci. 2014;(4):357–61.
36. Zhao H, Shi J, Qi X, Wang X, Jia J. Pyramid scene parsing network; 2017. p. 6230–9.
37. Wang G, Zuluaga MA, Li W, Pratt R, Patel PA, Aertsen M, et al. DeepIGeoS: a deep interactive geodesic framework for medical image segmentation. IEEE Trans Pattern Anal Mach Intell. 2018;41(7):1559–72. https://doi.org/10.1109/TPAMI.2018.2840695.
38. Wang G, Li W, Zuluaga MA, Pratt R, Patel PA, Aertsen M, et al. Interactive medical image segmentation using deep learning with image-specific fine tuning. IEEE Trans Med Imaging. 2018;37(7):1562–73. https://doi.org/10.1109/TMI.2018.2791721.

Publisher's Note