Interactive Medical Image Segmentation with Self-Adaptive Confidence Calibration

Wenhao Li[*], **Chuyun Shen**[*], **Qisen Xu**[*]
School of Computer Science and Technology
East China Normal University
Shanghai, China 200062
{whli,cyshen,qsxu}@stu.ecnu.edu.cn

Bin Hu
Huashan Hospital
Fudan University
Shanghai, China 200062
08301010188@fudan.edu.cn

Bo Jin[†]
School of Computer Science and Technology
East China Normal University
Shanghai, China 200062
bjin@cs.ecnu.edu.cn

Haibin Cai
Software Engineering Institute
East China Normal University
Shanghai, China 200062
hbcai@sei.ecnu.edu.cn

Fengping Zhu, Yuxin Li
Huashan Hospital
Fudan University
Shanghai, China 200062
{zhufengping,liyuxin}@fudan.edu.cn

Xiangfeng Wang[†]
School of Computer Science and Technology
East China Normal University
Shanghai, China 200062
xfwang@cs.ecnu.edu.cn

Abstract

Interactive medical segmentation based on human-in-the-loop is a novel paradigm that draws on human expert knowledge to assist medical image segmentation. However, existing methods often fall into what we call the *interactive misunderstanding*, the essence of which is the dilemma in trade-off *short-* and *long-term* interaction information. To better utilize the interactive information at various timescales, we propose an interactive segmentation framework, called interactive **ME**dical segmentation with self-adaptive **C**onfidence **CA**libration (**MECCA**), which combines the action-based confidence learning and multi-agent reinforcement learning. A novel confidence network is learned by predicting the alignment level of the action with the short-term interactive information. A confidence-based reward shaping mechanism is then proposed to explicitly incorporate the confidence into the policy gradient calculation, thus directly correcting the model's interactive misunderstanding. Furthermore, MECCA also enables user-friendly interactions by reducing the interaction intensity and difficulty via label generation and interaction guidance, respectively. Numerical experiments on different segmentation tasks show that MECCA can significantly improve short- and long-term interactive information utilization efficiency with remarkably fewer labeled samples. The demo video is available at `https://bit.ly/mecca-demo-video`.

1 Introduction

The thorny data-hungry issue posed by machine learning has spawned a boom in human-in-the-loop (HITL) that incorporates human domain knowledge into the data annotation, model training, or

[*]First three authors contribute equally.
[†]Corresponding Author.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

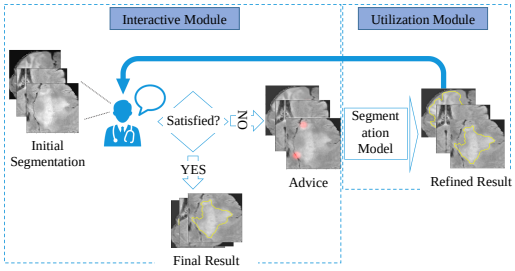
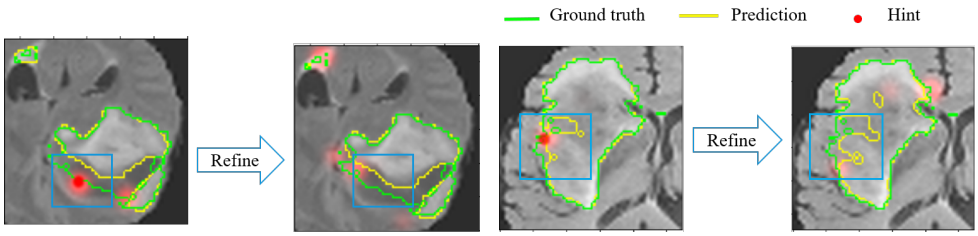


Figure 1: The interactive segmentation process. Interactive module: the expert observes current (or initial) segmentation and provides further correction information (red hints); Utilization module: new segmentation is refined based on the correction information.

system design process in computer vision [1–3]. Meanwhile, deep learning-based segmentation algorithms have significantly boosted automatic medical image segmentation performance. However, the accuracy of existing algorithms usually fails to meet clinical demands due to the pathological variability, dark lesion areas, and the uneven quality of the training data [4–6]. To further refine the relatively inaccurate segmentation results, HITL-based interactive image segmentation algorithms that take advantage of interactive correction information (*e.g.*, clicks, scribbles, or bounding boxes) increasingly popular [4–11]. The general interactive segmentation process containing an interactive module and a utilization module is depicted in Figure 1.



(a) Correction information is ignored (b) Refined result becomes worse

Figure 2: Segmentation refining failure after long-term interactions on BraTS2015: a) The segmentation model can not fully understand or ignore the hint information. b) The interactive segmentation model could misuse the expert's interaction correction, which results in a worse result;

Intuitively, the performance of an interactive segmentation algorithm mainly depends on the quality of its understanding of human-infused sequence knowledge. Unfortunately, existing methods often fall into what we call the *interactive misunderstanding*, the essence of which is the dilemma of algorithms in understanding *short-term* and *long-term* interaction information. Paying too much attention to short-term information can easily cause shocks in the segmentation effect [6, 11]; otherwise, short-term information may be ignored or misused. We implement the popular interactive segmentation algorithm, InterCNN [10], which focuses more on long-term information, and evaluate it on the BraTS2015 benchamark [12]. As can be seen from Figure 2(a), the algorithm ignores the expert's correction information and can even be adversely affected, as shown in Figure 2(b).

In this paper, we propose a novel interactive segmentation algorithm for 3D medical images called interactive **ME**dical segmentation with self-adaptive **C**onfidence **CA**libration (MECCA, see Figure 1) to alleviate the interactive misunderstanding problem. We follow the multi-agent reinforcement learning (MARL) framework of [6, 11] to model each voxel [‡] as an agent and formulate the iterative interactions process as a partially observed stochastic game. Each voxel learns an optimal policy to adjust its likelihood based on expert interaction information and adjacent voxel information through a trial-and-error process that continuous interaction with the external environment (i.e., the fixed offline training dataset). To enable the MARL framework, which is better able to handle long-term information, to achieve a balance between short-term and long-term information, we introduce a novel action-based confidence network to predict the alignment level of the action with the short-term interactive information. Based on this confidence network, we further propose a

[‡]The smallest processable unit in three-dimensional medical image space.

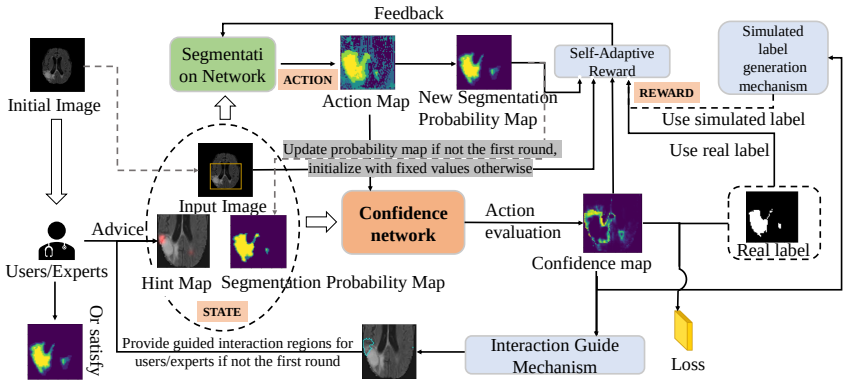


Figure 3: MECCA's architecture. The segmentation module outputs actions to change the segmentation probability of each voxel(agent) at each interaction step. Meanwhile, the confidence network will estimate the confidence of actions, which will generate the self-adaptive reward and simulated label. The confidence map can provide advice regions for the next interaction step with experts.

self-adaptive reward function to explicitly incorporate the alignment level into the policy gradient calculation, thus directly correcting the model's interactive misunderstanding. Numerical experiments on different segmentation tasks show that MECCA can significantly improve efficiency in short-term and long-term interactive information utilization.

Interestingly, the proposed action-based confidence network enables user-friendly interactions in addition to alleviating the interactive misunderstanding problem. On the one hand, the confidence network can generate high-quality labels for unlabeled data, thus reducing the interaction *intensity*; on the other hand, one can generate advice interactive regions (i.e., low-confidence segmentation regions) based on the confidence network to reduce the interaction *difficulty*. Additional numerical experiments in the semi-supervised setting show that MACCA is able to approximate the segmentation accuracy of SOTA methods using significantly fewer labeled samples.

2 Related Works

Interactive image segmentation. Before the significant development of the deep learning-type segmentation method, many traditional interactive image segmentation methods have been proposed [13]. The classical random Walk [14] can create a weight map with pixels as vertices and segment the image based on user interactions. GrabCut [15] and GraphCut [16] could associate image segmentation with the maximum flow and minimum cut algorithms on graphs, respectively, while Geos [17] was proposed to measure the similarity between pixels' geodesic distance. In recent years, deep learning-based interactive image segmentation methods are popularly discussed. [8] segments images based on CNN interactively. DeepCut [7] and ScribbleSup [9] employed weakly supervised expert hints to establish interactive image segmentation methods. DeepIGeoS [5] employed geodesic distance metric to construct a hint map. Polygon-RNN [18] and Polygon-RNN+ [19] fundamentally segmented each target as a polygon and iteratively chose the polygon vertexes. SeedNet [20] trained an expert interaction generation RL model. IteR-MRL [6] and BS-IRIS [11] both modeled the dynamic interaction process as an MDP and employed MARL models to segment images. In addition, IFSL [21] and IOG [22] aim to reduce the annotation cost of interactive image segmentation. These interactive methods often fall into the interactive misunderstanding due to the difficulty of effectively utilizing experts' short-term and long-term interaction information simultaneously.

Uncertainty estimation for image segmentation Uncertainty estimates are helpful in the context of deployed machine learning systems as they are capable of detecting when a neural network is likely to make an incorrect prediction. Traditionally, many algorithms are inspired by Bayesian statistics [23, 24], such as Monte-Carlo dropout [25], multiplicative normalizing flows [26], and stochastic batch normalization [27]. The main disadvantage of these BNN approximations is that they require massive sampling to generate the output distributions. An alternative to BNNs is ensembling methods [28–32], which propose a frequentist approach to the problem of uncertainty estimation by training many models and observing the variance in their predictions. However, this technique is still quite resourcing intensive, requiring inference from multiple models to produce the uncertainty

estimate. A promising alternative to the sampling-based approach is to let a neural network learn its uncertainty.[33–40]. These methods commonly consist of a segmentation and confidence network and are more computationally efficient than other techniques. Different from the previous work on learning uncertainty through imitation, joint training, or adversarial learning, MECCA adopts a simple but powerful mechanism by introducing an auxiliary task [35, 37] to predict the alignment level of the action with the short-term supervised interactive information.

3 MARL-based Interactive Segmentation Framework

This paper employs the MARL framework similar with [6, 11, 41] to formulate the interactive segmentation process and continuously give error-corrective actions at each interaction step. Let $\mathbf{x} = (x_1, \cdots, x_N)$ denotes the input image and x_i denotes the i-th voxel. In the setting of MARL, every voxel x_i is treated as an agent with its own refinement policy $\pi_i({a_i}^{(t)}, {s_i}^{(t)})$. At time step t, agent x_i gets action ${a_i}^{(t)}$ from the segmentation network according to its current state ${s_i}^{(t)}$. After taking the action, the agent will receive a reward ${r_i}^{(t)}$ according to the segmentation result.

The **state** ${s_i}^{(t)}$ for agent x_i is concatenated by its voxel value b_i, its current segmentation probability ${p_i}^{(t)}$ and the value ${h_i}^{(t)}$ on the hint map. In particular, the segmentation probabilities of all agents are initialized to 0.5 and range from 0 to 1. The hint map $h^{(t)}$ is transformed from the user's hints which are in the form of edge points. At each step, users click on some edges, which are not correctly predicted, as hints. The **action** ${a_i}^{(t)}$ for agent x_i is sampled from its policy and used to adjust its previous segmentation probability:

$$\begin{aligned} a_i^{(t)} &\sim \pi_\theta(a_i^{(t)}|s_i^{(t)}), \\ p_i^{(t+1)} &= \mathrm{clip}(p_i^{(t)} + a_i^{(t)}, 0, 1), \end{aligned} \tag{1}$$

where ${a_i}^{(t)} \in \mathcal{A}$ and the $clip$ operation modifies the probability to the interval [0,1]. The action set $\mathcal{A}$ contains actions of different scales, allowing the agent to select the proper action. In our setting, the $\mathcal{A} = \{\pm 0.1, \pm 0.2, \pm 0.4\}$. The **reward** ${r_i}^{(t)}$ is the feedback (positive or negative) of the action and is used to update the refinement policy. The reward design is a significant part of our algorithm, and we will introduce it in detail in Section 4.2.

We adopt the independent learning algorithm framework to solve this MARL problem and use the popular actor-critic algorithm, A3C [42] to optimize each agent's refinement policy. The segmentation network $\boldsymbol{S}$, which adopts the P-Net[5] as the backbone, is used to parameterize the actor and critic an $\boldsymbol{S}$ contains two output heads, i.e., policy head and value head. Two heads share the first three 3D convolutional blocks to extract low-level features. Each of the blocks has two convolution layers and the size of the convolution kernel is fixed as 3×3 in all these convolution layers. All the convolution kernels are dilated convolution, which can reduce the loss of resolution. Both of the heads have another two 3D convolutional blocks to extract specific high-level features. The policy head output the policy $\pi_i({a_i}^{(t)}, {s_i}^{(t)})$, which is the probability distribution of action under current state. Moreover, the value head estimates the value of the current state, which evaluates how good the current state is and estimates the expected return. The goal of the policy head is to maximize the expected return by selecting proper actions in state $\boldsymbol{s}^{(t)}$.

4 Self-Adaptive Confidence Calibration

As we mentioned in the Introduction, there will be some situations where the segmentation model misunderstands or ignores the hint information. This challenge is what we call the *interactive misunderstanding* phenomenon, and the formal definition is shown in the following.

Definition 1 (*Interactive Misunderstanding*)**.** *For a binary classification problem, the sign of the foreground label $y = 1$ is denoted as positive, and the sign of the background label $y = 0$ is denoted as negative accordingly. In an interactive medical image segmentation task (i.e. a voxel-wise binary classification problem), for any voxel i, if the sign of the change of segmentation probability output by algorithm for two consecutive interaction steps sign($\triangle(p^{(i)})$) is not equal to sign(y_i), then this phenomenon is defined as interactive misunderstanding.*

To better utilize the interactive information at various timescales and alleviate the interactive misunderstanding, we propose an interactive segmentation framework called interactive medical segmentation with self-adaptive confidence calibration (**MECCA**, Algorithm 1), which combines the action-based confidence learning and multi-agent reinforcement learning. A novel **action-based confidence network** is learned by predicting the alignment level of the action with the short-term interactive information. A **confidence-based reward shaping** mechanism is then proposed to explicitly incorporate the confidence into the policy gradient calculation, thus directly correcting the model's interactive misunderstanding. Furthermore, MECCA also enables user-friendly interactions by reducing the interaction intensity and difficulty via **label generation** and **interaction guidance** respectively. The four key modules mentioned above will be described separately below.

Algorithm 1 Interactive Medical Image Segmentation with Self-Adaptive Confidence Calibration

Initialize the segmentation network $\boldsymbol{S}$ with θ and the confidence network $\boldsymbol{C}$ with w;
for every sample in labeled datasets **do**
  Set the segmentation probability of each voxel to 0.5, $s^{(0)} \leftarrow (x, p^{(0)}, h^{(0)})$;
  for every interaction time step t **do**
    Take action $a^{(t)} \leftarrow S(s^{(t)})$, get reward $r^{(t)}$ and observe the next state $s^{(t+1)}$;
    Compute the gradient of $\boldsymbol{S}$ and $\boldsymbol{C}$;
  Get a sample (x', y') from unlabeled dataset and initialize the state: $s'^{(0)} \leftarrow (x', p'^{(0)}, h'^{(0)})$;
  for every interaction time step t **do**
    Take action $a'^{(t)} \leftarrow S(s'^{(t)})$ and generate the simulated label $\hat{y}^{(t)}$ by (6);
    Observe the reward and next state $s'^{(t+1)}$ and compute the gradient of $\boldsymbol{S}$.

4.1 Action-based Confidence Network

Our proposed confidence network learns the confidence of given actions to enable the MARL framework to balance short-term and long-term information and alleviate interactive misunderstanding. The confidence network uses the previous state and action as input and a confidence map as output and is optimized by minimizing the summation of binary cross-entropy loss over actions at each time step t. Here we use $\boldsymbol{C}$ to denote the confidence network, $w_{\boldsymbol{C}}$ denotes the parameters of the confidence network, while L_{BCE} denotes the binary cross-entropy loss:

$$L_{\boldsymbol{C}}(s^{(t)}, a^{(t)}; w_{\boldsymbol{C}}) = L_{BCE}(\boldsymbol{C}(s^{(t)}, a^{(t)}), \mathrm{g}^{(t)}) + L_{BCE}(\boldsymbol{C}(s^{(t)}, -a^{(t)}), 1 - \mathrm{g}^{(t)}), \tag{2}$$

where

$$\mathrm{g}^{(t)} = \begin{cases} 0 & \text{if } a^{(t)} \oplus y^{(t)} == 1, \\ 1 & \text{otherwise}, \end{cases} \tag{3}$$

and $\mathrm{g}^{(t)}$ means whether the direction of action is consistent with the label. $a \oplus b$ is defined as the statement being only true if either $a > 0$ or $b > 0$, but not both. One potential issue when training the confidence network is the imbalance of samples. Inspired by the discriminator learning in generative adversarial networks [43], we introduce symmetric samples into the Equation (2) to speed up training.

4.2 Confidence-based Reward Shaping

The previously described action-confidence learning provides the segmentation model with a confidence map to alleviate the interactive misunderstanding phenomenon. By this confidence map, hard-or-easy samples can be better recognized as the confidence values for these 'hard regions' are relatively lower than in other regions. We formulate this voxel-level action-aware as the self-adaptive reward function, $r^{(t)}$, which is shown in (4), to adapt this mechanism to the training of MARL:

$$r^{(t)} = \textstyle\sum_{i=1}^{N} \alpha \left(2 - c_i\right)^{\beta} gain_i^{(t)}, \tag{4}$$

where c_i is the value on the confidence map. The setting of hyperparameters α and β are described in Section 5.1. In addition, $gain_i^{(t)}$ denotes the relative gain of cross-entropy:

$$\mathrm{gain}_i^{(t)} = \mathcal{X}_i^{(t-1)} - \mathcal{X}_i^{(t)}, \mathcal{X}_i^{(t)} = -y_i \log(p_i^{(t)}) - (1 - y_i) \log(1 - p_i^{(t)}), \tag{5}$$

where $\mathcal{X}_i^{(t)}$ denotes the cross-entropy between current segmentation probability and ground truth. If an agent gets a positive reward, its current action is good, and the refined segmentation result is closer to the ground truth. With the self-adaptive reward function in (4), the confidence value c_i of these wrong actions is lower, and they will be punished more when training our segmentation model.

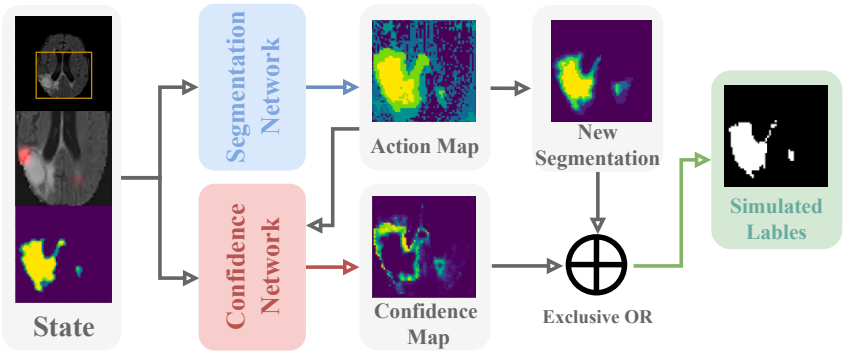


Figure 4: The simulated label generation. The simulated label generation mechanism utilizes the confidence map and the action map to generate the simulated label. The confidence map is used to calibrate the action, and the direction of the calibrated action is the simulated label of each voxel.

4.3 Label Generation and Interaction Guidance

Interestingly, the proposed action-based confidence network can enable user-friendly interactions in addition to alleviating the interactive misunderstanding problem. On the one hand, the confidence network can generate high-quality labels for unlabeled data (see Figure 4), thus reducing the interaction *intensity*. Specifically, for unlabeled data, MECCA can be used to leverage the action confidence to generate simulated labels:

$$\hat{y}^{(t)} = \begin{cases} 1 & \text{if } a^{(t)} \oplus c^{(t)} == 1, \\ 0 & \text{otherwise }, \end{cases} \tag{6}$$

where $\hat{y}^{(t)}$ is the simulated voxel-level label generated from the confidence map. Then one can train model by using the unlabeled data with simulated labels. But in order to stabilize the training process, the gradients of unlabeled data backpropagate only when the action confidence exceeds the threshold δ. Unlike traditional pseudo-label training, the supervised signal does not come from the segmentation but from the confidence network. Thus, These filtered data with hint information are more valuable and provide more accurate supervised signals.

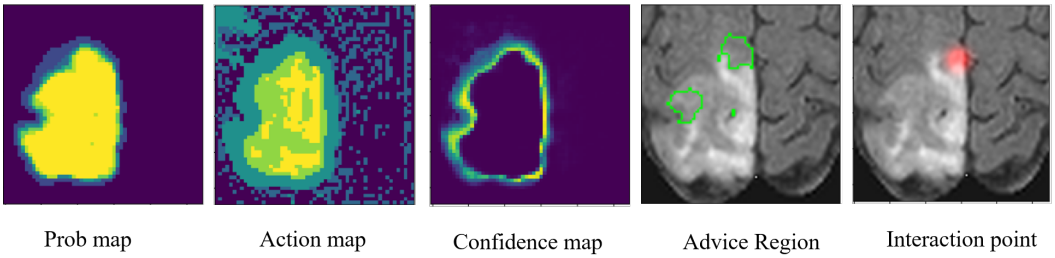


Figure 5: The interaction guide mechanism. The areas surrounded by the green lines are advice regions, and the red point is the real hint selected from advice regions. The color closer to yellow, the larger the positive value; otherwise, the smaller the negative value.

On the other hand, one can generate advice regions based on the confidence network to reduce the interaction *difficulty*. After refinement, our framework can suggest some possible regions as advice regions for users to interact with in the next round by filtering out those areas with low action confidence (see Figure 5). Firstly, the original 3D image will be segmented with super voxels, which can be regarded as a group of voxels with common characteristics. In this paper, we use simple linear iterative clustering (SLIC) [44] technique with *spacing* $= [2, 2, 2]$, *compactness* $= 0.1$ to generate super voxels, and the number of initial super voxels equals to 100 and gradually declines during the refinement iterations for training and testing. Secondly, the proposed algorithm will compute the mean action confidence in each super voxel and sort them in descending order. Finally, the top 5 super voxels will be marked and recommended to users.

5 Numerical Results

5.1 Experiment Settings

The proposed MECCA is evaluated on 4 3D medical image segmentation benchmarks: (1) **BraTS2015**: Brain Tumor Segmentation Challenge 2015 [12] contains $274(234/40)$ multiparametric MRI(Flair, T1, T1C, T2) from brain tumor patients. we only use the Flair image and segment the whole brain tumor, and the **BraTS2020** dataset is used the same way. (2) **BraTS2020**: Brain Tumor Segmentation Challenge 2020 [12] contains $285(235/50)$ multiparametric MRI(Flair, T1, T1C, T2) from brain tumor patients. (3) **MM-WHS**: Multi-Modality Whole Heart Segmentation [45] contains $24(20/4)$ multi-modality whole heart images covering the whole heart substructures. We choose to segment the left atrium blood cavity only. (4) **Medical Segmentation Decathlon**: A generalisable 3D semantic segmentation datasets containing different organ segmentation tasks [46]. We use the spleen and liver datasets, which provide $61(41/20)$ and $106(96/10)$ CT images respectively. The Dice score [47] and the average symmetric surface distance (ASSD) [48] are used to evaluate the performance of the segmentation result.

We compare MECCA with 5 state-of-the-art interactive segmentation methods: DeepIGeos [5] , InterCNN [10] , IteR-MRL [6] and BS-IRIS [11]. We also introduce the U-Net [49] as a comparable baseline. For a fair comparison, all CNN-based methods adopt the same network structure (P-Net) proposed in [5]. We generate the hint map for all tasks by randomly selecting the wrong segmentation points on the edges with the ground truth. For MECCA, we set learning rates initialized as $1e-4$, $T=5$, $\gamma=0.95$, $\alpha=0.8$, $\beta=1$. The training time of the proposed method with one Nvidia 2080ti GPU varies from 5 to 13 hours for different datasets. The hyperparameters of baselines are consistent with those in the open source library.

5.2 Main Results

Table 1: Quantitative comparison of 3D medical image datasets segmentation by different methods. In particular, the P-Net is the method without hint information. Significant improvement (p-value < 0.05) is marked in bold.

Methods	BraTS2020		BraTS2015		MM-WHS		Spleen		Liver	
	Dice(%)	ASSD(pixels)	Dice(%)	ASSD(pixels)	Dice(%)	ASSD(pixels)	Dice(%)	ASSD(pixels)	Dice(%)	ASSD(pixels)
P-Net	83.67±8.35	5.78±4.01	84.00±12.01	5.10±3.72	81.40±1.48	3.28±0.45	88.08±2.25	4.25±2.07	35.89±2.61	34.46±23.82
U-Net	84.72±10.42	4.09±3.89	84.66±11.25	6.17±4.69	80.96±1.65	3.72±0.39	87.95±2.87	5.12±1.09	56.00±1.93	22.38±22.42
DeepIGeos	88.54±.97	2.11±1.30	88.32±5.34	2.28±1.24	88.48±0.71	1.53±0.18	91.97±1.51	0.93±0.46	48.57±2.52	10.28±3.45
InterCNN	88.39±6.01	2.01±1.09	88.26±7.07	1.81±2.09	87.85±1.15	0.80±0.15	93.52±0.94	0.54±0.83	59.92±2.20	5.95±2.76
IteR-MRL	89.22±4.65	2.07±0.91	88.94±4.81	**1.41±0.22**	89.55±0.87	0.90±0.11	91.50±1.34	0.67±0.21	62.29±1.93	**0.87±0.59**
BS-IRIS	90.47±5.23	1.82±0.33	89.74±3.86	1.61±0.42	89.12±0.98	1.19±0.16	92.35±1.13	0.54±0.19	67.25±2.01	4.34±1.18
MECCA	**91.02±5.86**	**1.15±0.20**	**90.29±5.07**	1.50±0.33	**90.39±5.89**	**0.80±0.01**	**94.96±1.44**	**0.30±0.16**	**71.46±1.41**	2.36±0.99

We can see from Table 1 and Figure 6 that our proposed MECCA outperforms other state-of-the-art methods on all datasets. To demonstrate that our method can take advantage of hint information more efficiently, we also compare the relative improvement of different methods at each interaction step using the same amount of hints. The experimental results are shown in Figure 6.

These results suggest that MECCA can effectively use hint information and improve dice scores on most steps. MECCA and IteR-MRL are RL-based methods, while others are CNN-based methods, and we can find the main advantage of RL-based methods is that they can always keep notable improvement. However, we should realize that the RL-based methods still can not guarantee the high confidence of the corrective actions. As we can see in Figure 6, the performance of IteR-MRL is not as good as other methods at the beginning, which is caused by numerous incorrect actions. On the contrary, our proposed MECCA with self-adaptive reward can perform well at each step and consecutive refine the result.

5.3 Weakly-supervised Interactive Segmentation

As mentioned in Section 4.3, MECCA can reduce the interaction intensity by using the simulated label generated from the action confidence map. We validate the proposed method on the BraTS2015 dataset by randomly selecting different proportions of samples as fully labeled data and using the rest of the training images as unlabeled data, which only provided hint information when interacting. Table 2 shows the results of different methods. Only MECCA and another semi-supervised method

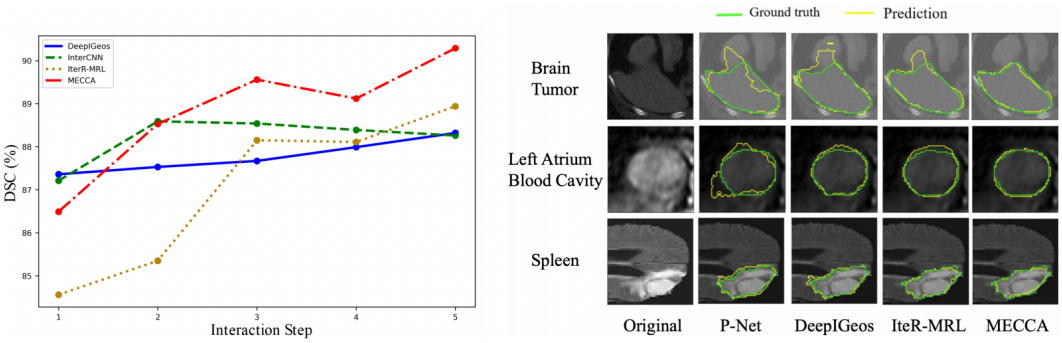


Figure 6: **Left:** Visualization of the performance improvement of different methods at the different interaction steps. All these testing results are performed on the BraTS2015 dataset. **Right:** Visualization of segmentation results from different methods. The green lines represent ground truth boundaries, and the yellow lines represent the predictive boundaries.

Table 2: Quantitative comparison between MECCA and other methods on BraTS2015 Dataset of different sizes. Significant improvement (p-value < 0.05) is marked in bold.

Data amount	1/8	1/4	1/3	1/2	1/1	$\triangle(1/8, 1/1)$
P-Net	75.86	80.83	80.88	83.02	84.00	10.73%
DeepIGeos	85.50	85.90	87.70	87.60	88.32	3.30%
IteR-MRL	84.36	86.54	87.38	88.66	88.94	5.07%
UA-MT	83.08	84.47	84.62	84.39	84.66	**1.90%**
MECCA	**87.14**	**88.23**	**88.31**	**89.17**	**90.29**	3.60%

UA-MT [50] uses unlabeled data, and the remaining three baselines only use a fixed proportion of labeled data. As seen from the figure, MECCA achieves a Dice score of 87.14% with only 12.5% labeled data, and it performs better than baselines with 25% labeled data.

Table 3: DICE of the algorithm under different amounts of interactions and labeled data percentage.

% Labeled Data	Interaction Steps				
	3	4	5	6	7
25%	84.03	86.18	88.23	87.32	88.24
33%	85.32	87.60	88.31	87.99	88.53
50%	85.39	87.07	89.17	88.93	89.13
100%	89.93	90.40	90.29	89.85	90.79

We also test MECCA's performance under different interaction times, and labeled data percentages and the results are shown in Table 3. Based on the experimental results, we can obtain the following conclusions: 1) if there is enough labeled data, MECCA can achieve good results after a few interactions; 2) the number of interactions required by MECCA to achieve the same performance is roughly inversely proportional to the number of labeled data; 3) although MECCA requires more interactions (about 2-3 times the number of interactions) when only part of the labeled data is available, it can approach the algorithm's performance trained with all labeled data in the end.

6 Closing Remarks

This paper presents a novel action-based confidence learning method for interactive 3D image segmentation. Specifically, this paper proposes to learn the confidence of actions that continuously refine the segmentation result during the interaction process so that hint information can be used more effectively. Based on this, the self-adaptive reward is proposed for the segmentation module, which can prevent the interactive misunderstanding phenomenon during the interaction and help reduce the time cost of interaction by providing users with the advice regions to interact next. Besides, the confidence map can also replace the ground truth to generate feedback for the unlabeled samples for

the segmentation module. These samples without voxel-level annotations can also be used to train our model. Experiments on different segmentation tasks show that MECCA can significantly improve efficiency in short-term and long-term interactive information utilization. Additional numerical results in the semi-supervised setting show that MACCA is able to approximate the segmentation accuracy of SOTA methods using significantly fewer labeled samples.

Acknowledgement. This work was supported in part by Shanghai Project (No. 22511106000, 22QB1402100, 22511106004) and NSFC (No. 12071145).

Our approach's limitations and potential societal impacts are as follows.

Limitations. This work investigates interactive medical segmentation under a uni-modality setting based on human-in-the-loop with self-adaptive confidence calibration. However, 3D segmentation with the multi-modality medical image is more realistic and clinically instructive. We will discuss interactive segmentation methods for multimodal images in our future work. In addition, although MECCA requires 45 points for interaction, it is still a burden for users. Therefore, it might be possible to reduce the number of interaction points using generated interaction points in future work.

Ethics Statement. Our method is not a generative model, nor does it involve super large-scale models. The training data is sampled in the simulated environments, so it does not involve fairness issues. Our method also does not involve model or data stealing and adversarial attacks.

References

[1] Cranor, L. F. A framework for reasoning about the human in the loop. In *Proceedings of the 1st Conference on Usability, Psychology, and Security*, pages 1–15. 2008.

[2] Zanzotto, F. M. Human-in-the-loop artificial intelligence. *Journal of Artificial Intelligence Research*, 64:243–252, 2019.

[3] Wu, X., L. Xiao, Y. Sun, et al. A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems*, 2022.

[4] Wang, G., W. Li, M. A. Zuluaga, et al. Interactive Medical Image Segmentation using Deep Learning with Image-specific Fine Tuning. *IEEE Transactions on Medical Imaging*, 37(7):1562–1573, 2018.

[5] Wang, G., M. A. Zuluaga, W. Li, et al. DeepIGeoS: A Deep Interactive Geodesic Framework for Medical Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1559–1572, 2018.

[6] Liao, X., W. Li, Q. Xu, et al. Iteratively-Refined Interactive 3D Medical Image SegmentaWenhaotion with Multi-Agent Reinforcement Learning. In *CVPR*. 2020.

[7] Rajchl, M., M. C. Lee, O. Oktay, et al. Deepcut: Object Segmentation from Bounding Box Annotations using Convolutional Neural Networks. *IEEE Transactions on Medical Imaging*, 36(2):674–683, 2016.

[8] Xu, N., B. Price, S. Cohen, et al. Deep Interactive Object Selection. In *CVPR*. 2016.

[9] Lin, D., J. Dai, J. Jia, et al. Scribblesup: Scribble-supervised Convolutional Networks for Semantic Segmentation. In *CVPR*. 2016.

[10] Bredell, G., C. Tanner, E. Konukoglu. Iterative Interaction Training for Segmentation Editing Networks. In *MLMI International Workshop*. 2018.

[11] Ma, C., Q. Xu, X. Wang, et al. Boundary-aware supervoxel-level iteratively refined interactive 3d image segmentation with multi-agent reinforcement learning. *IEEE Transactions on Medical Imaging*, 40(10):2563–2574, 2021.

[12] Menze, B., A. Jakab, S. Bauer, et al. The Multimodal Brain Tumor Image Segmentation Benchmark (BraTS). *IEEE Transactions on Medical Imaging*, 34(10):1993–2024, 2015.

[13] Zhao, F., X. Xie. An Overview of Interactive Medical Image Segmentation. *Annals of the BMVA*, 2013(7):1–22, 2013.

[14] Grady, L. Random Walks for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.

[15] Rother, C., V. Kolmogorov, A. Blake. "GrabCut": interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, 2004.

[16] Boykov, Y. Y., M.-P. Jolly. Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in ND Images. In *ICCV*. 2001.

[17] Criminisi, A., T. Sharp, A. Blake. GeoS: Geodesic Image Segmentation. In *ECCV*. 2008.

[18] Castrejon, L., K. Kundu, R. Urtasun, et al. Annotating Object Instances with a Polygon-RNN. In *CVPR*. 2017.

[19] Acuna, D., H. Ling, A. Kar, et al. Efficient Interactive Annotation of Segmentation Datasets with Polygon-RNN++. In *CVPR*. 2018.

[20] Song, G., H. Myeong, K. Mu Lee. SeedNet: Automatic Seed Generation with Deep Reinforcement Learning for Robust Interactive Segmentation. In *CVPR*. 2018.

[21] Feng, R., X. Zheng, T. Gao, et al. Interactive few-shot learning: Limited supervision, better medical image segmentation. *IEEE Transactions on Medical Imaging*, 40(10):2575–2588, 2021.

[22] Zhang, S., J. H. Liew, Y. Wei, et al. Interactive object segmentation with inside-outside guidance. In *CVPR*. 2020.

[23] MacKay, D. J. A Practical Bayesian Framework for Backpropagation Networks. *Neural Computation*, 4(3):448–472, 1992.

[24] Neal, R. M. *Bayesian learning for neural networks*, vol. 118. Springer Science & Business Media, 2012.

[25] Gal, Y., Z. Ghahramani. Dropout as a Bayesian Approximation: Representing model Uncertainty in Deep Learning. In *ICML*. 2016.

[26] Louizos, C., M. Welling. Multiplicative normalizing flows for variational bayesian neural networks. In *ICML*. 2017.

[27] Atanov, A., A. Ashukha, D. Molchanov, et al. Uncertainty estimation via stochastic batch normalization. In *ICLR*. 2018.

[28] Dietterich, T. G. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[29] Kamnitsas, K., W. Bai, E. Ferrante, et al. Ensembles of multiple models and architectures for robust brain tumour segmentation. In *International MICCAI brainlesion workshop*, pages 450–462. 2017.

[30] Mehrtash, A., M. Ghafoorian, G. Pernelle, et al. Automatic needle segmentation and localization in mri with 3-d convolutional neural networks: application to mri-targeted prostate biopsy. *IEEE Transactions on Medical Imaging*, 38(4):1026–1036, 2018.

[31] Lakshminarayanan, B., A. Pritzel, C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *NIPS*. 2017.

[32] Mehrtash, A., W. M. Wells, C. M. Tempany, et al. Confidence calibration and predictive uncertainty estimation for deep medical image segmentation. *IEEE Transactions on Medical Imaging*, 39(12):3868–3878, 2020.

[33] Kendall, A., Y. Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *NeurIPS*. 2017.

[34] DeVries, T., G. W. Taylor. Learning confidence for out-of-distribution detection in neural networks. *arXiv*, 2018.

[35] Robinson, R., O. Oktay, W. Bai, et al. Real-time prediction of segmentation quality. In *MICCAI*. 2018.

[36] DeVries, T., G. W. Taylor. Leveraging uncertainty estimates for predicting segmentation quality. *arXiv*, 2018.

[37] Jungo, A., M. Reyes. Assessing reliability and challenges of uncertainty estimations for medical image segmentation. In *MICCAI*. 2019.

[38] Moeskops, P., M. Veta, M. W. Lafarge, et al. Adversarial Training and Dilated Convolutions for Brain MRI Segmentation. In *DLMIA*. 2017.

[39] Hung, W.-C., Y.-H. Tsai, Y.-T. Liou, et al. Adversarial Learning for Semi-supervised Semantic Segmentation. In *BMVC*. 2018.

[40] Nie, D., L. Wang, L. Xiang, et al. Difficulty-aware Attention Network with Confidence Learning for Medical Image Segmentation. In *AAAI*. 2019.

[41] Furuta, R., N. Inoue, T. Yamasaki. PixelRL: Fully Convolutional Network With Reinforcement Learning for Image Processing. *IEEE Transactions on Multimedia*, 22(7):1704–1719, 2020.

[42] Mnih, V., A. P. Badia, M. Mirza, et al. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937. PMLR, 2016.

[43] Sampath, V., I. Maurtua, J. J. Aguilar Martín, et al. A survey on generative adversarial networks for imbalance problems in computer vision tasks. *Journal of big Data*, 8(1):1–59, 2021.

[44] Achanta, R., A. Shaji, K. Smith, et al. SLIC Superpixels Compared to State-of-the-Art Superpixel Methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012.

[45] Zhuang, X., J. Shen. Multi-scale Patch and Multi-modality Atlases for Whole Heart Segmentation of MRI. *Medical Image Analysis*, 31:77–87, 2016.

[46] Simpson, A. L., M. Antonelli, S. Bakas, et al. A Large Annotated Medical Image Dataset for the Development and Evaluation of Segmentation Algorithms. *arXiv preprint*, 2019.

[47] Carass, A., S. Roy, A. Gherman, et al. Evaluating white matter lesion segmentations with refined sørensen-dice analysis. *Scientific reports*, 10(1):1–19, 2020.

[48] Yeghiazaryan, V., I. D. Voiculescu. Family of boundary overlap metrics for the evaluation of medical image segmentation. *Journal of Medical Imaging*, 5(1):015006, 2018.

[49] Ronneberger, O., P. Fischer, T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *MICCAI*. 2015.

[50] Yu, L., S. Wang, X. Li, et al. Uncertainty-aware Self-ensembling Model for Semi-supervised 3D Left Atrium Segmentation. In *MICCAI*. 2019.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Please refer to Appendix. 6
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] Please refer to Appendix.6
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]

(b) Did you include complete proofs of all theoretical results? [N/A]

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] The code is available at `https://anonymous.4open.science/r/MECCA-C9C7/`.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Please see 5.1 for details.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] Please refer to 5.1.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] Please refer to 5.1. Besides, We use DeepIGeos by `https://github.com/taigw/GeodisTK`, InterCNN by `https://github.com/gbredell/interCNN`, and U-Net by `https://github.com/liyun-lu/unet_and_vnet`.
   (b) Did you mention the license of the assets? [Yes] We mention that the open-sourced repository DeepIGeos is licensed under MIT license, while InterCNN and U-Net are under no license.
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] The codebases we use are open-sourced.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] The datasets we use are publicly available and desensitized.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]