





# CAISeg: A Clustering-Aided Interactive Network for Lesion Segmentation in 3D Medical Imaging

Yukang Sun , Shujun Zhang , Jinsong Li , Qi Han , and Yuhua Qin

**Abstract**—Accurate lesion segmentation in medical imaging is critical for medical diagnosis and treatment. Lesions' diverse and heterogeneous characteristics often present a distinct long-tail distribution, posing difficulties for automatic methods. Currently, interactive segmentation approaches have shown promise in improving accuracy, but still struggle to deal with tail features. This triggers a demand of effective utilizing strategies of user interaction. To this end, we propose a novel point-based interactive segmentation model called Clustering-Aided Interactive Segmentation Network (CAISeg) in 3D medical imaging. A customized Interaction-Guided Module (IGM) adopts the concept of clustering to capture features that are semantically similar to interaction points. These clustered features are then mapped to the head regions of the prompted category to facilitate more precise classification. Meanwhile, we put forward a Focus Guided Loss function to grant the network an inductive bias towards user interaction through assigning higher weights to voxels closer to the prompted points, thereby improving the responsiveness efficiency to user guidance. Evaluation across brain tumor, colon cancer, lung cancer, and pancreas cancer segmentation tasks show CAISeg's superiority over the state-of-the-art methods. It outperforms the fully automated segmentation models in accuracy, and achieves results comparable to or better than those of the leading point-based interactive methods while requiring fewer prompt points. Furthermore, we discover that CAISeg possesses good interpretability at various stages, which endows CAISeg with potential clinical application value.

**Index Terms**—Deep learning, Interactive segmentation, Medical image analysis.

## I. INTRODUCTION

MEDICAL image segmentation stands as a pivotal task at the intersection of computer vision and medical research, aiming to accurately extract regions of interest from medical images. This provides a foundation for physicians in their

Received 16 December 2023; revised 28 August 2024; accepted 21 September 2024. Date of publication 25 September 2024; date of current version 8 January 2025. This work was supported by the Qingdao Municipal Demonstration Project of Science and Technology Benefiting the People Number 23-2-8-smjk-20-nsh. (Corresponding author: Shujun Zhang.)

Yukang Sun, Shujun Zhang, Jinsong Li, and Qi Han are with the College of Data Science, Qingdao University of Science and Technology, Qingdao 266061, China (e-mail: yakecan@mails.qust.edu.cn; zhangsj@qust.edu.cn; plaitkol@mails.qust.edu.cn; HanQi@mails.qust.edu.cn).

Yuhua Qin is with the College of Information Science and Technology, Qingdao University of Science and Technology, Qingdao 266061, China (e-mail: yqin@qust.edu.cn).

Digital Object Identifier 10.1109/JBHI.2024.3467279

detailed analysis and quantitative assessment of physiological structures and tissues. With advancements in deep learning and computer vision, fully automatic medical image segmentation, such as [1], [2], and [3], has achieved expert-level performance in many tasks. However, these high-performing models predominantly rely on fully supervised training strategies, which means they are heavily dependent on extensive, high-quality annotated datasets. Particularly in the task of lesion segmentation, the immense variability in lesion shapes and locations intensifies the challenge of creating a thorough dataset. Consequently, these automatic models might struggle when encountering lesion features not well-represented within their training datasets—essentially the tail-end features. Interactive segmentation is a promising avenue for tackling this challenge as it allows the segmentation network to capitalize on the doctor's expertise to more effectively identify these tail features.

Interactive segmentation techniques enable users to convey prior knowledge to the network by drawing scribbles, outlining bounding boxes, or clicking on points. Considering the three-dimensional structure of most medical images in clinical settings, the method of interaction through point clicking stands out as the most straightforward and intuitive strategy. In prior research on point-based interactive segmentation of medical images, most approaches drew inspiration from interactive segmentation techniques in the realm of natural images. These methods initially encode user interaction points, employing methods like Euclidean distance encoding [4], disk encoding [5], [6], Gaussian distance encoding [7], geodesic distance encoding [8], [9], etc., and utilize the resulting encoded output as a cue map to guide the network in segmentation. However, the semantic information conveyed by these simple distance or gradient encodings tends to be shallow and simplistic. Therefore, when it comes to tail features in medical images, these methods often fail to make accurate predictions, even when precise interactive points and their associated label information are provided by physicians. Fortunately, recent works [10], [11], [12], [13] that revisit Mask Transformers [14], [15] from the perspective of clustering have inspired us to develop a novel point-based interactive segmentation network, named Clustering-Aided Interactive Segmentation Network (CAISeg). This framework advances interactive segmentation by introducing a two-stage process: cluster assignment and head feature refinement. Initially, we perform cluster assignment to extract semantic feature vectors that closely resemble user interaction points through clustering within the feature map. Following this, head feature refinement is applied to adjust interactive feature vectors along with those

highly similar vectors within their clusters, enabling them to be mapped to the head distribution of the prompted label categories. This interaction-guided approach ensures that CAISeg retains its effectiveness when addressing tail features.

Previous work on point-based interactive segmentation commonly relied on loss functions tailored for general segmentation tasks. These include Binary Cross Entropy Loss (BCE Loss), Dice Loss [16] designed for sparse sample scenarios, and Focal Loss [17] aimed at hard cases. Yet, all of these loss functions were originally conceived for fully automatic segmentation, and they may not take into full account the context of user interactions. This oversight can prevent models from effectively emphasizing areas pointed out by users. In the realm of interactive segmentation, it's crucial for the model to hone in on user-specified regions and achieve heightened accuracy therein. To bridge this gap, we introduced Focus Guided Loss (FG Loss). Distinct from conventional loss functions, FG Loss is intricately crafted to resonate with the specifics of user interactions. By applying a higher penalty to the regions where users click, it bolsters the model's capacity to learn effectively from those interactions, ensuring a profound comprehension and response to such cues. Leveraging this approach, our network not only delineate user-designated structures with enhanced precision but also to swiftly recalibrate in light of user feedback.

We summarize our main contributions as follows:

- To the best of our knowledge, we are the first to apply the concept of clustering to deep learning-based interactive segmentation networks, and the clustering method is effective in aiding CAISeg to seize the regions of interest based on the interaction points. Moreover, the approach of aligning features within these regions with the head features of the prompted categories enables CAISeg to maintain a high recognition capability for tail features.
- We propose a novel loss function, Focus Guided Loss, a tailored loss function that emphasizes the predictive accuracy around the proximity of user interaction points. By assigning higher weights to voxels closer to these interaction points, we encourage CAISeg to focus more on these critical regions, which enhances its ability to deduce and prioritize user intent.
- We compared CAISeg with the state-of-the-art segmentation methods on three distinct lesion segmentation tasks: brain tumors, colon cancer, and lung cancer. CAISeg exhibits superior segmentation quality compared to existing fully automated segmentation models. Compared to current interactive segmentation methods, CAISeg achieves more precise segmentation with fewer, unrestricted interaction points.

## II. RELATED WORK

### A. Automatic Deep Learning-Based Medical Image Segmentation Methods

Applying deep learning to medical image segmentation has long been challenging in computer vision. The Fully Convolutional Network [18] laid the foundation, but the real breakthrough came with U-Net [19], whose symmetric

encoder-decoder structure and skip-connections effectively capture intricate structures in medical images. Numerous U-Net variants ([1], [16], [20], [21]) followed, with nnU-Net [1] standing out for its auto-adjusting capabilities and robustness across various segmentation tasks. The advent of ViT [22] initiated the integration of Transformers with CNNs, as seen in models like UNeTR [23], which combines CNNs for local feature extraction with Transformers for long-range contextual understanding. Further developments ([24], [2]) replaced convolutional modules with the shifted window self-attention mechanism of the Swin Transformer [25], creating a pure Transformer architecture. Liu et al. [3] introduced contrastive language-image pre-training embeddings into segmentation models, leading to the highly generalized Universal Model. Several studies introduced effective data augmentation techniques to enhance medical image segmentation models. Liang et al. [26] generated new samples by extracting inter-patient deformations through learning-based deformable registration, and creating intra-patient deformations using random 3D Thin-Plate-Spline transformations. Furthermore, He et al. [27] increased the diversity of generated samples through KL transform-based statistical analysis. Despite these advancements, handling tail features in medical images remains difficult. Our method, however, leverages interactive segmentation by incorporating expert guidance from physicians to identify and cluster tail features, allowing for more precise delineation of regions misjudged by the network.

### B. Interactive Segmentation Methods

Traditional interactive segmentation methods such as Level Set [28], Graph Cut [29], Random Walker [30], Region Growing [31], and Grow Cut [32] are notably representative. The Level Set method uses partial differential equations to capture target boundaries, excelling in managing topological changes. Graph Cut segments images into foreground and background using the min-cut algorithm and user-annotated seed points. The Random Walker method labels pixels based on the probability of a random walk to seed points, while Region Growing incrementally expands segmentation areas from seed points based on local similarity. Grow Cut, based on cellular automata theory, iteratively updates pixel labels until a stable state is reached. However, despite their innovative designs, these methods face challenges in 3D medical image segmentation, including high computational complexity, sensitivity to noise, difficulties with complex structures, and susceptibility to local optima.

In deep learning-based interactive segmentation, a key challenge is enabling networks to effectively comprehend user interaction cues. Xu et al. [4], the first model to integrate interactive concepts into deep learning for 2D image segmentation, using Euclidean distance maps to encode positive and negative user sample points, thereby providing spatial information to the network. DEXTR [33] required user points at the extreme edges of the target for precise location information. Wang et al. [8] and Luo et al. [9] used geodesic distance transforms to encode interaction points, aiming to enrich the network's prior information. Wang's model [8] started with a coarse segmentation, refining it using geodesic distance encoding, while

Luo [9] placed interaction points near the foreground boundary, utilizing an exponential geodesic distance transform for more accurate edge information. Sofiuk et al. [34] analyzed and refined interaction encoding methods, training strategies, and loss functions, leading to the development of the RITM interactive segmentation network, which introduced sparse encoding and iterative training as a new standard. Jian et al. [7] proposed DINs, integrating sparse Gaussian distance encoding into multi-scale feature maps, which enhanced the transmission of sparse interaction information through the network. Liu et al. [35] used cross-attention between interaction point vectors and feature maps to incorporate category features directly. The VMN developed by Zhou et al. built upon [33] by performing segmentation on 2D slices and then bi-directionally propagating the segmentation mask, incorporating a quality assessment module to refine the process. While these methods encoded user interaction as shallow semantic information, they limited the prior knowledge conveyed to the network. The MedSAM [36], utilizing the pre-trained SAM [37] with transfer learning on a large medical dataset, introduces effective segmentation for medical images. On the other hand, MedLSAM [38] employs an additional network to regress six extreme points of a 3D object to anchor the corresponding bounding box, which is then segmented slice-by-slice using the SAM model. SAM-Med3D [39] further modified the SAM model into a 3D structure, allowing point-based interaction for 3D medical images. However, these SAM-based methods, while effective for organ segmentation, struggle with the heterogeneity of medical lesions, as their output tokens find it difficult to identify tail features within the target based on interaction cues. Our approach focuses on capturing semantic features deeply embedded in the network that closely resemble the vectors corresponding to user interaction points through clustering. For challenging tail features, we reposition them within the head distribution space to improve segmentation accuracy, aligning more closely with the physician's intent.

### C. Mask Transformers

Instead of directly employing Transformers as network backbones for image segmentation, Mask Transformers enhance CNN-based architectures with stand-alone blocks that leverage masked attention mechanisms. In MaX-Deeplab [40], MaskFormer [15], and Mask2Former [41], mask embedding queries are used in the decoder to perform dot-product operations with per-pixel features, generating the predicted binary masks. Building on this, CMT-Deeplab [10] and KMaX-Deeplab [11] introduced the concept of treating queries as clustering centers, incorporating constraints to improve cluster representation learning within the network. Yuan et al. [12] segmented unseen objects in medical scans by identifying outlier features within the clustering results. S2VNet [13] addresses 3D medical image segmentation by processing slices individually, initializing cluster centers based on the clustering results from previous slices, thereby leveraging prior knowledge to assist in the segmentation of current slices. Inspired by these advancements, we extend this clustering-based approach to interactive segmentation with CAISeg. In CAISeg, user-provided interaction points serve as

clustering centers within a designed masked attention module, allowing the model to identify semantically similar regions and more effectively recognize tail features.

## III. METHOD

In this section, we first outline the problem scenario for our task. Then, we will sequentially introduce the overview of CAISeg, the Feature Clustering Module, the method for loss calculation, and the interaction simulation strategy during the training phase.

### A. Problem Scenario

The objective of an interactive segmentation network is to infer the user's area of interest and delineate it from the background based on user-provided interaction cues. Consequently, irrespective of the number of target classes present in the dataset, interactive segmentation tasks consistently revolve around pixel-level or voxel-level binary classification tasks. In our problem, users guide the network to segment all lesion regions specific to a particular disease from three-dimensional medical images by supplying positive and negative interaction points, representing the foreground region of the target lesion and the non-lesion region, respectively. Here, let's denote the three-dimensional medical image as  $I \in \mathbb{R}^{D \times H \times W}$ . The output of the model is a predicted mask of the region of lesions, denoted as  $\hat{Y} \in \{0, 1\}^{D \times H \times W}$ .

### B. Method Overview

The pipeline of CAISeg is depicted in Fig. 1. It comprises three main components: a Feature Encoder-Decoder for semantic feature extraction, a multi-scale Cross-Attention Module to capture foreground and background head features in the image, and an Interaction-Guided Module (IGM). The IGM, a comprehensive component, encompasses both the Feature Clustering Block and the feature adjustment method.

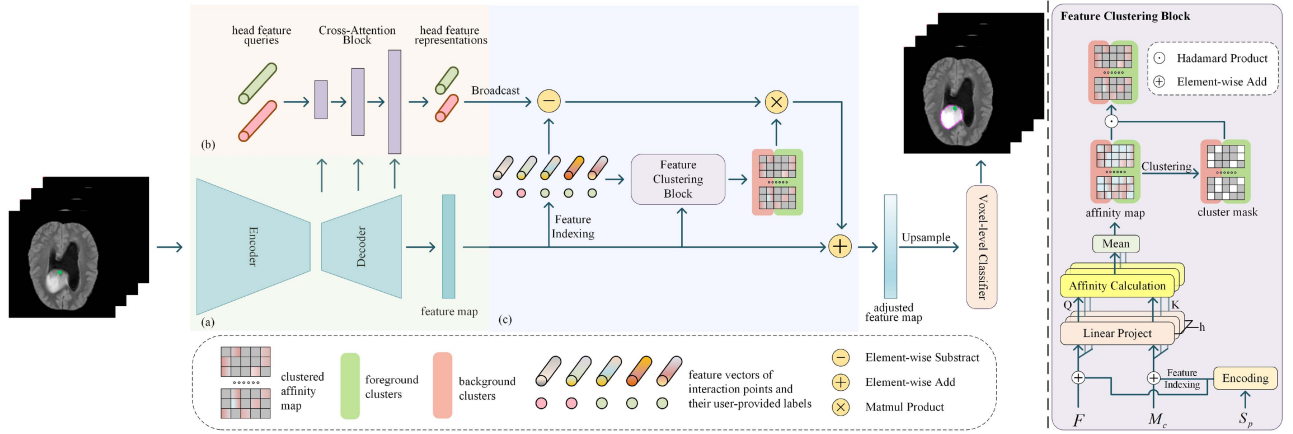
We utilize a CNN backbone to extract semantic features. The feature map undergoes a certain degree of downsampling, maintaining a maximum downsampling rate of  $\frac{1}{2}$  in each direction. After reshaping, this feature map is represented as  $F \in \mathbb{R}^{d_{hw} \times C}$ .

In the multi-scale Cross-Attention Module, we initially define two learnable embedding vectors,  $q_{\text{fore}}$  and  $q_{\text{back}}$ , as head feature queries. These queries engage in cross-attention computations with the feature map  $F'$  at various scales within the decoder of the feature extractor. They selectively integrate head features of foreground and background from the medical image for processing. The definition of the Cross-Attention Module is as follows:

$$q_{\text{cls}} \leftarrow q_{\text{cls}} + MLP \left( q_{\text{cls}} + \text{Softmax} \left( \left( \frac{q_{\text{cls}} \cdot (K_{F'})^T}{\sqrt{d_k}} \right) \cdot V_{F'} \right) \right), \quad (1)$$

where  $q_{\text{cls}}$  represents either  $q_{\text{fore}}$  or  $q_{\text{back}}$ , while  $K_{F'}$  and  $V_{F'}$  are both linear transformations of  $F'$ . Additionally, if necessary, we perform a fully connected layer mapping on  $q_{\text{cls}}$  to change its dimensions, aligning it with the feature map at following scale for cross-attention computation.





**Fig. 1.** The pipeline of CAISeg. Here, we assume that the user has provided a total of 2 background interaction points and 3 foreground interaction points. (a) A CNN backbone for feature extraction; (b) A multi-scale Cross-Attention Module; (c) The Interaction-Guided Module, which encompasses the Feature Clustering Block and a feature adjustment method. In the clustered affinity map, the color gray represents a fixed value of 0.

In the IGM, we first downsample the coordinates of the user's interaction points based on the corresponding axis's downsampling ratio in the feature map  $F$ . Subsequently, using these coordinates, we index the corresponding position's feature vectors from the feature map  $F$  and concatenate them into a new matrix  $M_c \in \mathbb{R}^{n_c \times C}$ , where  $n_c$  represents the number of user interaction points. It should be noted that the two vectors obtained from (1) respectively represent the primary feature of the lesion category and the non-lesion category in the instance to be segmented, which are the targets for adjusting the tail features indicated by medical experts that the model cannot correctly identify. Therefore, for any row vector  $v_c$  in  $M_c$ , its corresponding adjustment vector  $v_{adj}$  is defined as:

$$v_{adj} = q_{cls} - v_c. \quad (2)$$

Here, the value of  $q_{cls}$  is contingent upon the user's annotation for  $v_c$ . If  $v_c$  is labeled as foreground by the user, then  $q_{cls}$  takes the value  $q_{fore}$ ; on the other hand, if  $v_c$  is labeled as background,  $q_{cls}$  is  $q_{back}$ . These adjustment vectors are allocated to the feature vectors within their respective clusters. After being adjusted by the adjustment vectors, those potential tail features that closely align with user interactions are mapped to a more appropriate position within the feature space. Specifically:

$$F_{new} = F + CA \times M_{adj}, \quad (3)$$

where  $M_{adj} \in \mathbb{R}^{n_c \times C}$  is a matrix composed of adjustment vectors  $v_{adj}$  from all prior features annotated by the user.  $F_{new}$  is the adjusted new feature map.  $CA \in \mathbb{R}^{dhw \times n_c}$  is a clustered affinity map output by the Feature Clustering Block, exclusively capturing the affinity of each feature vector to its closest foreground and background clustering centers marked by the user. Detailed information on the Feature Clustering Block can be found in Section III-C.

### C. Feature Clustering Block

The Feature Clustering Block is designed to use the features at the user interaction points as cluster centers, assigning each

feature to the clusters of its most similar foreground and background interaction points. The corresponding similarity guides the assignment of adjustment vectors from these cluster centers.

The elucidation of the Feature Clustering Block is delineated in Fig. 1. In this Block, the preliminary step entails the incorporation of the segmentation mask  $S_p$ , acquired from the preceding interaction round, into the feature representations. This mask is initialized with zero values in the first round of segmentation. Following mask fusion, the enriched feature map and the feature vectors corresponding to user interaction points, undergo transformation via a reshape operation and a shared linear layer, culminating into  $Q \in \mathbb{R}^{n_h \times dhw \times \frac{C}{n_h}}$  and  $K \in \mathbb{R}^{n_h \times n_c \times \frac{C}{n_h}}$  respectively. In every head, Each row vector within  $Q$  and  $K$  is henceforth denoted as  $q$  and  $k$ . For tail features, both the direction and magnitude of their vectors bear significant importance, rendering the Euclidean distance preferable over cosine similarity under such circumstances. The spatial proximity between  $q$  and  $k$  also impinges on the similarity assessment. Therefore, the affinity metric between  $q$  and  $k$  is indirectly manifested through a linear combination of their Euclidean and spatial distances. Subsequently, a Gaussian Radial Basis Function kernel is employed to transmute this combined distance measure into a similarity metric, nurturing a more nuanced comprehension of feature interrelations. The affinity between  $q$  and  $k$  is defined as:

$$D(q, k) = \|q - k\|_2 + \alpha \times norm(\|(\chi_q - \chi_k) \cdot s^T\|_2), \quad (4)$$

$$A(q, k) = e^{-D^2(q, k)}, \quad (5)$$

where  $\chi_q$  and  $\chi_k$  are the coordinates of  $q$  and  $k$  in the original feature map before the reshape operation, respectively.  $\|\cdot\|_2$  refers to min-max normalization.  $\alpha$  is a learnable parameter, and  $norm(\cdot)$  represents the L2 norm operator.  $s^T$  is the distance correction vector that adjusts voxel distances to actual distances, providing more accurate spatial relationships.  $s^T$  will be detailed in Section IV-B1. We then merge the affinity maps of multiple heads into one by averaging.

Furthermore, we classify/categorize clustering centers based on user annotations. Those marked as the foreground are classified into the set  $S_{\text{fore}}$ , and those as the background into  $S_{\text{back}}$ . Consequently, the element  $mask_{q,k}$  in the cluster mask is defined as

$$mask_{q,k} = \begin{cases} 1, & A(q,k) = \max_{k' \in S_k} A(q,k') \\ 0, & \text{otherwise} \end{cases}, \quad (6)$$

where the  $S_k$  refers to the specific set to which  $k$  belongs, namely  $S_{\text{fore}}$  or  $S_{\text{back}}$ . Lastly, the Feature Vector Clustering Module produces an output termed the clustered affinity map  $CA$ , in which the element  $a_{q,k}$  is defined as

$$a_{q,k} = A(q,k) \times mask_{q,k}. \quad (7)$$

Obviously, when the user provides only one category interaction point, the clustered affinity map  $CA$  measures the similarity between each feature vector in the feature map and the most semantically similar cluster center. However, when the user provides both foreground and background category interaction points, the  $CA$  assesses the similarity between each feature and the two distinct clusters that are most semantically similar. This clustering strategy minimizes potential class imbalances resulting from varying quantities of user-provided foreground and background interaction points.

#### D. Focus Guided Loss

A robust interactive segmentation network should fully leverage user interaction information as much as possible. Achieving this demands not only efficient encoding of interaction details and meticulous design of network structure and modules but also the fine-tuning of network parameters. In practical applications, we expect the network to offer more precise predictions for voxels close to interaction points in space. This necessitates that network layers responsible for integrating prior information better understand user priors and predict user interaction intent. This inductive bias can be instilled in the network through the design of loss functions. Just as Dice Loss [16] focuses more on classes with fewer samples and Focal Loss [17] concentrates on challenging samples, our proposed Focus Guided Loss (FG Loss) prioritizes voxels near interaction points during training. Thus, for a specific voxel with a label  $y$  and a predicted value  $\hat{p}$ , the FG Loss  $\ell_{FG}$  is defined as:

$$\ell_{FG}(y, \hat{p}) = \left( 1 + \left( \frac{d_{\max} - d}{d_{\max}} \right)^\lambda \right) \times \ell_{BCE}(y, \hat{p}). \quad (8)$$

Wherein,  $d$  is the distance between the voxel and the nearest user interaction point;  $d_{\max}$  is the maximum distance among the distances to the voxel from all user interaction points closest to that voxel on this image;  $\lambda$  is a hyperparameter;  $\ell_{BCE}$  refers to the Binary Cross Entropy Loss. The proposed FG Loss places more emphasis on the prediction accuracy of the interaction point area than other general loss functions, thereby making the trained network have a stronger preference for user interaction information.

#### E. The Loss Calculation for CAISeg

Ideally, our network needs to have the following capabilities:

- The adjusted  $q_{\text{fore}}$  and  $q_{\text{back}}$  should be as close as possible to the representative foreground and background features in the instance.
- The feature clustering block should be able to measure the similarity between feature vectors and prior features as accurately as possible.

The former determines the adjustment target of tail features, while the latter determines the degree of adjustment of the feature vectors. Simply computing the loss between the network output and the ground truth is insufficient to train the network to this ideal state. To address this, We perform voxel-level classification on the adjusted feature map  $F_{\text{new}}$  based on the adjusted  $q_{\text{fore}}$  and  $q_{\text{back}}$ , thereby achieving deep supervision for IGM. Specifically, we first perform cross-attention between  $F_{\text{new}}$  and the adjusted  $q_{\text{fore}}$  and  $q_{\text{back}}$ , i.e.

$$q_h = \text{concat}(q_{\text{fore}}, q_{\text{back}}), \quad (9)$$

$$\hat{P}_{\text{cls}} = \text{Softmax} \left( \frac{F_{\text{new}} \cdot (q_h)^T}{\sqrt{d_k}} \right). \quad (10)$$

Continuing, we reshape  $\hat{P}_{\text{cls}}$  and then upsample it through interpolation to restore it to the original image shape. The resulting new  $\hat{P}_{\text{cls}}$  will be involved in the computation of network loss.

In our work, we linearly combine FG Loss  $\ell_{FG}$  and Dice Loss  $\ell_{Dice}$  to form a comprehensive loss function  $\ell_{com}$  used to calculate the loss between predicted values  $\hat{P}$  and ground truth  $Y$ . Thus, the loss function  $\ell_{com}$  is defined as:

$$\ell_{com}(Y, \hat{P}) = \sum_i (\mu \times \ell_{FG}(y_i, \hat{p}_i) + (1 - \mu) \times \ell_{Dice}(y_i, \hat{p}_i)), \quad (11)$$

and the final loss function used to supervise the training of CAISeg is

$$\ell = \nu \times \ell_{com}(Y, \hat{Y}) + (1 - \nu) \times \ell_{com}(Y, \hat{P}_{cls}), \quad (12)$$

where  $\mu$  and  $\nu$  are hyperparameters.

#### F. Interaction Simulation Strategy

In our network, user interaction is automatically simulated during the training process, eliminating the need for manual involvement. We adopt the iterative training approach from RITM [34], wherein, during training phase, a batch undergoes  $n$  continuous rounds of prediction. Only the output from the final round is compared with the ground truth to compute the loss, which is then back-propagated to adjust network parameters. For the initial round of point-interaction simulation, we adapt strategies from iFCN [4] with a few modifications:

- For foreground points, we don't randomly sample across the entire foreground region. Instead, we first erode the foreground morphologically, then randomly sample from the remaining area.
- For background points, we sample uniformly from the background region within a certain range outside the target boundary.

The first modification is motivated by the observation that user interaction points rarely fall on the object's edge in the initial round, especially in medical images where many boundaries are ambiguous. The second alteration stems from our objective to segment all lesion areas in medical images under physician guidance, rather than selecting specific areas of interest from multiple lesion areas. For interaction simulations beyond the first round, we first compare the differences between the segmentation results from the previous round and the ground truth. Then we select connected regions from these differential areas that exceed a predetermined threshold in size. Subsequently, a point is sampled from each of these selected connected regions. The binary mask of the segmentation result from the previous round is used as  $S_p$  in Fig. 1 for the subsequent round after the first.

#### IV. EXPERIMENT SETTING

##### A. Datasets and Preprocessing

We subject our proposed CAISeg to evaluation using four primary lesion segmentation tasks from the MSD2018 and 2020 challenges [42]: Brain Tumor, Colon Cancer, Lung Cancer and Pancreas Cancer segmentation. Here, we present the partitioning and preprocessing strategies for four datasets. The preprocessing steps implemented through TorchIO's [43] provided interface.

*Brain Tumor dataset* comprising 484 cases, each is equipped with lesion annotations. Every case offers four magnetic resonance imaging (MRI) sequences: Flair, T1w, t1gd, and T2w. The annotations delineate into four distinct categories: background, edema, non-enhancing tumor, and enhancing tumor. Aligning with our experimental objectives, we selectively employ the Flair sequence for the prediction of the entire brain tumor region, which amalgamates the sections of edema, non-enhancing tumor, and enhancing tumor. The 484 cases were randomly stratified into three segments: 338 (or 70%) for training, 73 (or 15%) for testing, and another 73 (or 15%) for validation. For computational feasibility, voxel spacings underwent a resampling from an initial  $1.0 \text{ mm} \times 1.0 \text{ mm} \times 1.0 \text{ mm}$  to  $1.5 \text{ mm} \times 1.5 \text{ mm} \times 1.5 \text{ mm}$ , yielding a uniform scan resolution of  $104 \times 160 \times 160$ . Extraneous portions of the image were excised by cropping from the scan's center to attain dimensions of  $96 \times 128 \times 128$ . Additionally, each MRI scan was subjected to a series of augmentations: a 50% probability of the introduction of random MRI bias field artifacts, a 50% likelihood of Gaussian noise addition, a 50% chance of undergoing random flips, and a 50% propensity for affine transformations that might include scaling between 0.9 and 1.1 times and rotations within a  $\pm 10^\circ$  spectrum. Each scan, in the last leg of preprocessing, was individually subjected to z-score normalization.

*Colon Cancer, Lung Cancer, and Pancreas Cancer datasets* were also utilized in our study. The Colon Cancer dataset consists of 126 cases, divided into 88 for training (70%), 19 for testing (15%), and 19 for validation (15%). Preprocessing involved truncating image signal intensities between 0.5% and 99.5%, followed by resampling voxel spacings to a median of  $5.00 \text{ mm} \times 0.78 \text{ mm} \times 0.78 \text{ mm}$ , resulting in an average resolution of  $89 \times 512 \times 512$ . Each CT image was then cropped

or padded to a final size of  $80 \times 512 \times 512$ . The Lung Cancer dataset comprises 63 CT cases, split into 37 for training (60%), 13 for testing (20%), and 13 for validation (20%). The preprocessing steps included truncating the CT signal between 0.5% and 99.5%, adjusting voxel spacings to a median of  $1.24 \text{ mm} \times 0.79 \text{ mm} \times 0.79 \text{ mm}$ , and yielding an average resolution of approximately  $264 \times 515 \times 515$  after resampling. The final images were then cropped or padded to a size of  $256 \times 512 \times 512$ . The Pancreas Cancer dataset includes 281 CT cases, divided into 197 for training (70%), 42 for testing (15%), and 42 for validation (15%). Preprocessing involved truncating signal intensities between 0.5% and 99.5%, resampling voxel spacings to a median of  $2.50 \text{ mm} \times 0.80 \text{ mm} \times 0.80 \text{ mm}$ , resulting in an average resolution of  $102 \times 520 \times 520$ . Each CT image was then cropped or padded to  $96 \times 512 \times 512$ . For all three CT datasets, the same data augmentation techniques were applied as those used in the Brain Tumor dataset, including random flips, Gaussian noise, affine transformations, and other standard augmentations, with the exception of MRI-specific artifacts.

##### B. Implementation Details

1) *Network Architecture*: We adopt the VNet [16] as our backbone network. Since the voxel spacing in all three directions in the Brain Tumor dataset is the same during preprocessing, the feature map size we use is  $\frac{D}{2} \times \frac{H}{2} \times \frac{W}{2}$ . However, for the Colon Cancer and Lung Cancer datasets, where the voxel spacing in the vertical direction is larger than the other two directions, the feature map size we use is  $D \times \frac{H}{2} \times \frac{W}{2}$ . Four Cross-Attention Modules are integrated to facilitate cross-attention with the VNet decoder outputs at strides of 16, 8, 4, and 2, with head counts set to 16, 16, 8, and 4, respectively. All the parameters in the network are initialized using He initialization [44]. Additionally, given the varying pixel distance ratios in medical images across the three dimensions, actual distances, rather than voxel distances, are utilized for the distance parameters in (8). Similarly, the distance correction vector  $s^T$  in (4) is used to convert the distance of feature vectors in the feature map to actual distances, where the value of  $s^T$  is directly related to the downsampling ratio of the feature map and the voxel spacing after resampling.  $s^T$  is defined as

$$s^T = (s_D \times r_D, s_H \times r_H, s_W \times r_W)^T, \quad (13)$$

where  $s_D$ ,  $s_H$ , and  $s_W$  represent the voxel spacing in the three directions;  $r_D$ ,  $r_H$ , and  $r_W$  are the downsampling ratios of the feature map compared to the original image in the three directions.

2) *Setting of Interactive Simulation Strategy*: In the training phase, we set the iterative count  $n$  as mentioned in Section III-F to 3. During the first round of interactive sampling, the network randomly samples between 1 to 10 points from the foreground region and 0 to 20 points from the background region. In subsequent rounds of interactive point simulation, the threshold mentioned in Section III-F is set to 10% of the lesion volume in the image being processed. During validation, we do not provide any interaction information to CAISeg in the initial segmentation



round, while subsequent rounds exclusively sample one point from the largest connected region of prediction errors.

**3) Training Strategy Configuration:** Our model was trained on a single RTX 3090 GPU with 24 GB of memory. The CPU used was an Intel Xeon Gold 6326. The training procedure employed the preprocessed images directly for the Brain Tumor dataset. However, for the Colon Cancer and Lung Cancer datasets, due to the high resolution of the preprocessed images, we employed a Patch-based training and inference scheme provided by TorchIO [43]. The patch sizes were set to  $48 \times 128 \times 128$  for both Colon Cancer and Pancreas Cancer, and  $64 \times 128 \times 128$  for Lung Cancer. The batch size during training for all datasets was consistently set to 2. The fully connected layers in multi Cross-Attention Module were implemented with the drop path strategy [45], having a regularization probability of 0.2. Training was conducted using the Adam optimizer [46] with an initial learning rate of 0.001, coupled with L2 regularization with a hyperparameter value of 0.0001. The learning rate adjustment followed a step decay strategy. Initially, the VNet was pre-trained for 200 epochs using a combined loss of DiceLoss [16] and BCELoss. Subsequently, CAISeg was trained for another 200 epochs using the loss described in III-D. The hyperparameter  $\lambda$  in (8) were set as constant values of 2. For (11), the hyperparameter  $\mu$  was dynamically incremented with epochs, progressing from 0.3 to 0.6 by the end of training. For (12), the hyperparameter  $\nu$  was set at a constant value of 0.5.

### C. The State-of-The-Art Approaches for Comparison

We assess CAISeg alongside various methodologies through qualitative and quantitative evaluations, categorizing the methods into three distinct groups. The first encompasses deep learning-based fully automatic segmentation techniques, including nnUNet [1] and Universal Model [3]. The second category consists of traditional interactive segmentation algorithms such as Random Walker [30] and Graph Cuts [29]. Lastly, we examine deep learning-based interactive segmentation approaches, among which are 3D RITM [34], MIDeepSeg [9], DINs [7], VMN [6], SAM-Med3D [39], MedLSAM [38], and MedSAM [36]. This structured comparison allows for a comprehensive evaluation across different segmentation methodologies. Given the diverse interaction requirements of these methods, we specify their interaction simulation strategies in our experiments:

- For Graph Cuts, Random Walker, RITM, DINs, and SAM-Med3D, seed points located at the lesion's central area are provided in the first segmentation round, while subsequent interaction strategies remain consistent with CAISeg.
- MIDeepSeg requires users to provide points describing the target contour in the first interaction. We sample six points near the inner boundary of each lesion in different directions within a maximum distance of three voxels from corresponding extreme points on the scan in the first round.
- VMN requires users to indicate four extreme points on the slice that delineate the target of interest. Our sampling strategy for these extreme points is consistent with that of MIDeepSeg. The slices needed for the next round of

interaction are selected by the assessment module within the model.

- MedLSAM and MedSAM encode the bounding box for interactive segmentation. We simulate user interaction by jittering each connected lesion's adjacent bounding box outward by 1-3 voxels in each direction.

Given the different interaction strategies and approaches of these methods, it is impossible to conduct an entirely fair comparison. To mitigate this issue, we used a shared random seed, ensuring that all methods are evaluated under similar conditions, thereby making the comparison results more reliable.

### D. Evaluation Metrics

To comprehensively evaluate the performance of the network architecture, we employ several key metrics: the Dice Similarity Coefficient (DSC), Recall, Precision, Jaccard Index, 95th percentile Hausdorff Distance ( $Hd_{95}$ ), and Average Surface Distance (ASD). For interactive segmentation methods, where the segmentation accuracy can be influenced by the positioning of interaction points, we conducted 10 rounds of validation using different random seeds. The mean and standard deviation of these ten results are reported to provide a detailed assessment of performance under varying conditions.

Additionally, we use the mean number of clicks (mNoC) to assess the efficiency of the interactive image segmentation algorithm. This metric represents the average number of clicks required to attain the specified DSC. A smaller mNoC indicates a reduced interaction burden needed to achieve the desired accuracy level.

## V. RESULTS AND DISCUSSION

### A. Comparison With Other Methods

Tables I and II present the segmentation performance metrics and mNOC values for various segmentation algorithms across three datasets. The performance metrics reflect the best results obtained after sampling 20 points for all point-based interactive methods, or the results from a single round using the bounding box interaction method. mNOC@x indicates the number of clicks required by an algorithm to achieve x% DSC. The summarized results are as follows:

- CAISeg demonstrated strong performance across multiple key metrics, particularly in Dice coefficient and mNOC. For instance, CAISeg achieved Dice scores of 93.4% on the Brain Tumor dataset and 81.1% on the Colon Cancer dataset, showcasing its effectiveness in accurately capturing and segmenting target regions based on user guidance in complex medical image segmentation tasks. In terms of mNOC, CAISeg performed exceptionally well across all datasets, especially in the Pancreas Cancer dataset, where it required only 2.9 clicks to achieve high segmentation accuracy. Additionally, CAISeg exhibited superior Precision and Recall on certain datasets, indicating its advantage in reducing false positives and false negatives. However, it is worth noting that on some datasets and metrics, such as  $Hd_{95}$  and ASD on the Lung Cancer dataset, CAISeg's

**TABLE I**  
COMPARISON OF CAISEG AGAINST OTHER APPROACHES ACROSS BRAIN TUMOR AND COLON CANCER SEGMENTATION TASKS IN TERMS OF VARIOUS METRICS

Method	Interaction	"Brain Tumor" from MRI					"Colon Cancer" from CT				
		Dice(%)	mNOC@90	HD <sub>95</sub> (mm)	ASD(mm)	Jaccard	Dice(%)	mNOC@75	HD <sub>95</sub> (mm)	ASD(mm)	Jaccard
nnU-Net <sub>2021</sub> [1]	No	87.8	-	4.24	1.67	0.78	58.7	-	17.35	2.48	0.42
Universal Model <sub>2023</sub> [3]	No	89.6	-	3.35	1.31	0.81	61.1	-	14.63	1.77	0.44
Graph Cuts <sub>2001</sub> [29]	points, bounding boxes	79.2±8.5	>20	8.49±4.21	3.49±1.82	0.66±0.16	53.4±16.9	>20	20.02±10.89	3.19±1.75	0.36±0.15
Random Walker <sub>2006</sub> [30]	points, bounding boxes	80.1±9.4	>20	8.61±4.19	3.75±2.26	0.67±0.18	58.6±12.4	>20	17.35±11.42	2.74±1.94	0.41±0.12
MIDeepSeg <sub>2021</sub> [9]	points	90.3±6.4	15.7	4.50±3.67	1.41±1.19	0.82±0.12	77.9±4.7	10.4	3.98±0.95	0.75±0.09	0.64±0.08
DINs <sub>2022</sub> [7]	points	91.2±1.9	10.3	3.35±1.21	1.09±0.21	0.84±0.05	70.9±6.2	>20	4.56±0.22	0.72±0.28	0.55±0.09
3D RITM <sub>2022</sub> [34]	points	91.3±1.6	9.7	3.18±1.18	0.97±0.19	0.84±0.04	71.3±5.8	>20	5.31±1.20	0.84±0.11	0.55±0.08
VMN <sub>2023</sub> [6]	points	<u>92.6±1.4</u>	<u>4</u>	2.59±0.85	0.99±0.25	<u>0.86±0.04</u>	<u>80.2±5.1</u>	11.1	3.91±0.98	<b>0.41±0.08†</b>	<u>0.67±0.10</u>
SAM-Med3D <sub>2023</sub> [39]	points	92.1±2.1	4.8	<u>2.47±0.98</u>	<u>0.86±0.43</u>	0.85±0.06	79.5±7.7	<u>8.2</u>	<u>3.76±1.33</u>	0.68±0.29	0.66±0.15
MedLSAM <sub>2023</sub> [38]	bounding boxes	90.4±1.0	-	2.66±0.44	1.37±0.28	0.82±0.03	68.7±2.9	-	15.78±3.55	1.83±0.84	0.52±0.04
MedSAM <sub>2024</sub> [36]	bounding boxes	90.2±1.3	-	3.02±0.41	1.19±0.37	0.82±0.04	69.5±4.2	-	14.96±4.83	1.52±0.76	0.53±0.06
CAISEG(Ours) <sub>2024</sub>	points	<b>93.4±2.2†</b>	<b>2.7</b>	<b>2.12±1.23</b>	<b>0.80±0.45</b>	<b>0.88±0.06†</b>	<b>81.1±7.3</b>	<b>4.6</b>	<b>3.12±0.81</b>	<u>0.53±0.13</u>	<b>0.68±0.15</b>

Here, the gray font indicates fully automated segmentation methods. †indicates that CAISEG outperforms The best-performing method among other approaches for the corresponding metric with a t-test significance level of  $p < 0.05$ .

**TABLE II**  
COMPARISON OF CAISEG AGAINST OTHER APPROACHES ACROSS LUNG CANCER AND PANCREAS CANCER SEGMENTATION TASKS IN TERMS OF VARIOUS METRICS

Method	Interaction	"Lung Cancer" from CT					"Pancreas Cancer" from CT				
		Dice(%)	mNOC@80	Precision(%)	Recall(%)	Jaccard	Dice(%)	mNOC@80	Precision(%)	Recall(%)	Jaccard
nnU-Net <sub>2021</sub> [1]	No	66.2	-	75.1	67.9	0.49	55.4	-	78.7	44.3	0.4
Universal Model <sub>2023</sub> [3]	No	77.8	-	88.5	68.6	0.64	64.5	-	77.3	57.5	0.48
Graph Cuts <sub>2001</sub> [29]	points, bounding boxes	56.1±14.1	>20	66.9±14.4	59.7±7.5	0.39±0.13	52.2±9.5	>20	75.2±6.6	40.4±16.3	0.36±0.08
Random Walker <sub>2006</sub> [30]	points, bounding boxes	61.8±10.6	>20	63.7±16.9	69.7±6.9	0.48±0.12	59.5±10.2	>20	71.9±2.4	51.2±14.9	0.42±0.10
MIDeepSeg <sub>2021</sub> [9]	points	80.6±5.8	9.8	85.4±5.6	76.2±15.1	0.68±0.12	81.5±1.8	11.9	83.8±3.8	79.4±7.5	0.69±0.03
DINs <sub>2022</sub> [7]	points	77.1±5.9	>20	<u>88.5±2.6</u>	68.1±7.5	0.63±0.10	72.7±2.6	>20	76.4±1.8	69.4±3.9	0.57±0.03
3D RITM <sub>2022</sub> [34]	points	78.1±5.4	>20	80.7±3.5	75.6±4.8	0.64±0.09	74.5±2.2	>20	79.2±6.1	70.5±4.2	0.59±0.04
VMN <sub>2023</sub> [6]	points	<u>81.8±4.1</u>	13.6	83.2±2.9	<u>80.1±5.8</u>	<u>0.69±0.08</u>	<u>82.9±2.9</u>	<u>6.8</u>	81.1±8.3	<u>84.8±3.1</u>	<u>0.71±0.06</u>
SAM-Med3D <sub>2023</sub> [39]	points	82.6±6.5	<u>7.6</u>	<b>89.5±5.8</b>	76.7±6.2	0.70±0.14	80.4±3.7	17.6	<b>85.9±5.6</b>	75.7±12.3	0.67±0.07
MedLSAM <sub>2023</sub> [38]	bounding boxes	77.4±3.7	-	85.2±6.3	71.2±2.2	0.63±0.07	81.7±2.1	-	82.6±3.5	79.5±1.2	0.69±0.04
MedSAM <sub>2024</sub> [36]	bounding boxes	76.9±4.6	-	77.3±5.2	76.7±2.7	0.62±0.09	78.8±2.3	-	84.8±0.6	72.6±7.3	0.65±0.04
CAISEG(Ours) <sub>2024</sub>	points	<b>83.3±7.1</b>	<b>5.4</b>	85.7±5.9	<b>81.0±8.4</b>	<b>0.71±0.14</b>	<b>86.1±3.2†</b>	<b>2.9</b>	<u>85.8±1.9</u>	<b>86.8±4.0†</b>	<b>0.76±0.08†</b>

Here, the gray font indicates fully automated segmentation methods. †indicates that CAISEG outperforms the best-performing method among other approaches for the corresponding metric with a t-test significance level of  $p < 0.05$ .

performance was comparable to or slightly lower than other advanced methods.

- T-test analysis shows that CAISEG's improvements in key metrics, such as Dice coefficient and mNOC on the Brain Tumor and Colon Cancer datasets, are statistically significant ( $p < 0.05$ ) compared to other methods. However, it is important to note that these improvements are not statistically significant in all cases. For instance, while CAISEG performed well on the Lung Cancer and Pancreas Cancer datasets, its improvements over the best-performing methods in these metrics did not reach statistical significance.
- CAISEG exhibited low standard deviation on certain metrics, reflecting a high degree of consistency under different interaction sampling conditions, particularly in mNOC and Dice coefficient on the Brain Tumor dataset. However, CAISEG showed higher standard deviation on other metrics, such as Precision and Recall on the Lung Cancer dataset, indicating some variability in performance under these conditions.

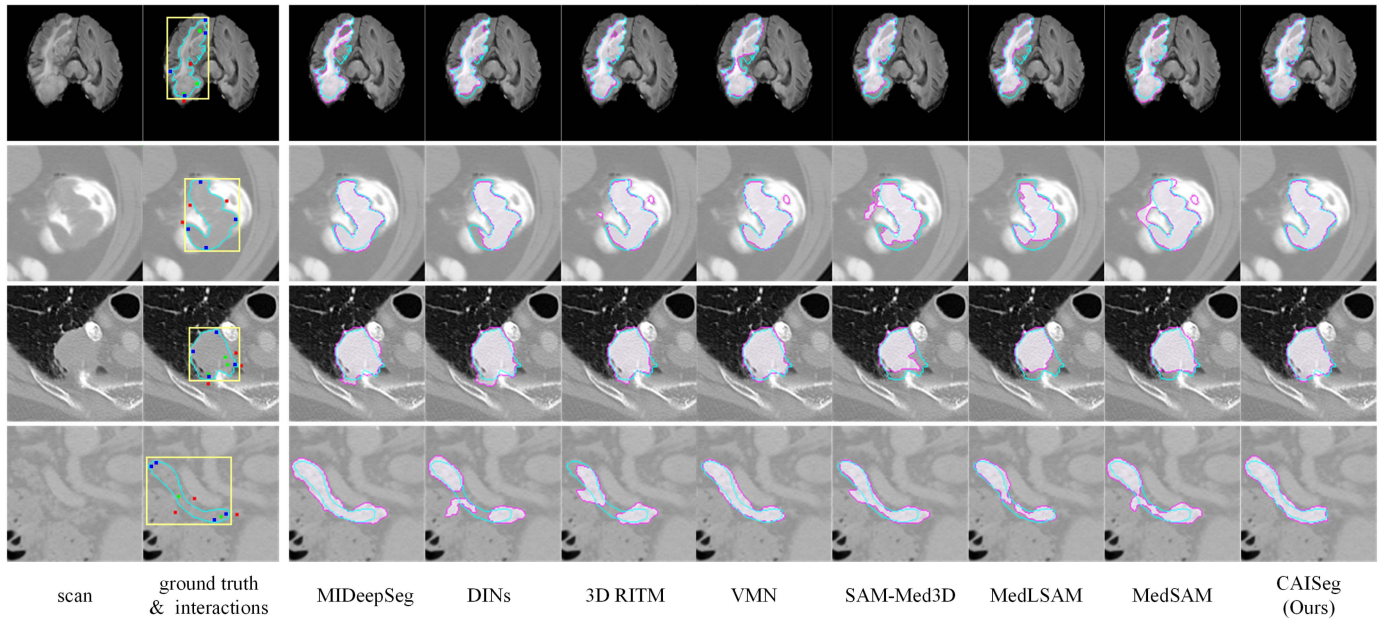
Fig. 2 presents the segmentation outcomes from eight interactive neural networks after user interaction on the same tissue slice. The analysis illustrates that MIDeepSeg, VMN,

SAM-Med3D, MedLSAM, and MedSAM exhibit certain constraints in terms of interaction flexibility. Notably, MedLSAM and MedSAM lack the capacity to refine segmentation results effectively. Concurrently, MIDeepSeg and VMN impose positional requirements on user-provided interaction points, limiting interaction flexibility and indirectly raising the effort for users to interact proficiently. In contrast, DINs, 3D RITM, SAM-Med3D, and CAISEG are more straightforward and unrestricted, allowing users to place interaction points anywhere on the scan. This increase in freedom, however, may lead to a decrease in the amount of information conveyed through interactions. Among these, CAISEG utilizes clustering analysis to capture the deep semantic similarity in user interactions, offering a more accurate understanding and response to user intent compared to DINs, 3D RITM and SAM-Med3D. Fig. 3 demonstrates the capability of CAISEG to understand user interaction intent through the segmentation results of multiple adjacent slices.

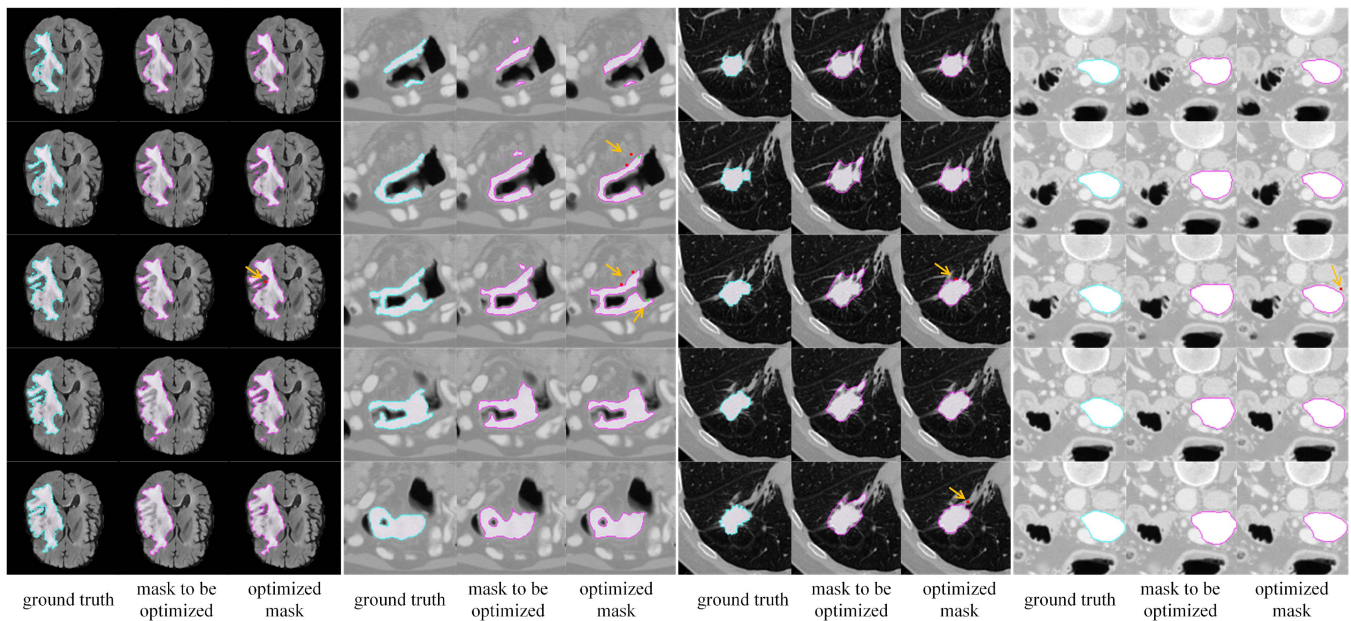
## B. Ablation Studies

To gain more insights into our model, we investigate the influence of essential components in CAISEG.





**Fig. 2.** Demonstrations of segmentation results from different interactive neural networks for different lesion segmentation tasks. In the figure, from top to bottom, the lesions sequentially represent brain tumors, colon cancer, lung cancer, and pancreas cancer. In the second column, the yellow boxes represent the bounding boxes required by MedSAM and MedLSAM; blue interaction points denote the extreme points needed by MiDeepSeg and VMN; red and green points indicate the foreground and background interaction points required by DINs, 3D RITM, SAM-Med3D and CAISeg, respectively. The regions outlined by cyan lines represent the ground truth, while the regions enclosed by magenta lines correspond to the predicted masks by the respective methods.



**Fig. 3.** Examples of CAISeg's mask refinement using interaction points. The series showcases corrections applied by CAISeg to a prior round's segmentation mask across five adjacent slices. Regions enclosed within cyan and magenta lines represent the actual lesion masks and the network's prediction, respectively. Annotations on the optimized masks indicate the locations and types of user interaction points.

**1) Quantitative Analysis of Feature Clustering Block:** We conduct a quantitative analysis on the affinity computation approach implemented in our Feature Clustering Block. Diverging from common practices of absolute position embedding to articulate positional relationships, our module directly assesses

the actual distances between voxels and interaction points. As for measuring vector similarity, we opted for an encoding method combining Euclidean distance with a Gaussian radial basis function, instead of the usual cosine similarity. Table III showcases the appropriateness of our chosen similarity computation

TABLE III  
ABLATION EXPERIMENTS ON THE FEATURE CLUSTERING BLOCK ON BRAIN TUMOR DATASET

No.	settings			DSC(%)	mNOC%90
	position encoding	similarity measurement	multi-head		
1	None/Absolute Positional Embedding	Euclidean Distance + Gaussian KRF	×	-	-
2	Actual Euclidean Distance	Cosine	×	88.6±1.9*	Unable
3	Actual Euclidean Distance	Euclidean Distance + Gaussian KRF	×	91.6±3.1*	7.1
4	Actual Euclidean Distance	Euclidean Distance + Gaussian KRF	✓	93.4±2.2	3.7

\*Indicates a t-test significance level of  $p < 0.05$  compared to the best performance.

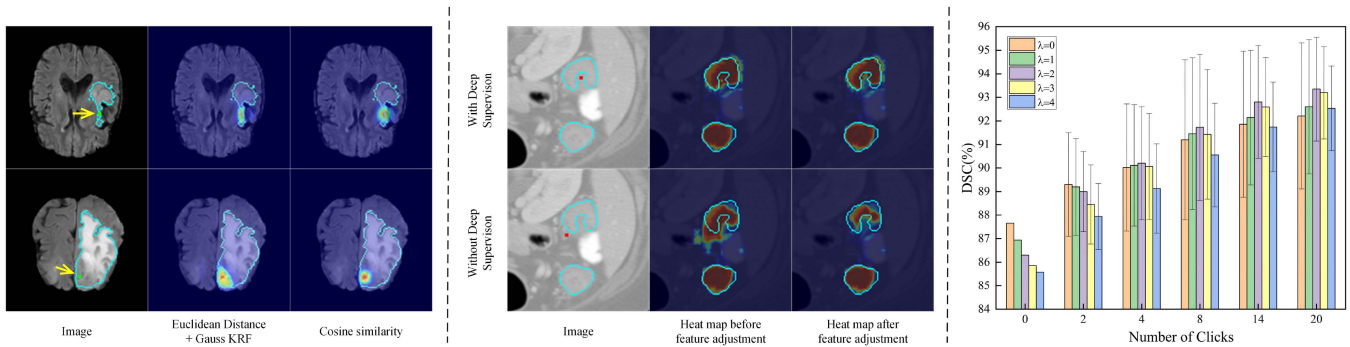


Fig. 4. Left: Two examples of heatmap results yielded by the Feature Clustering Block utilizing different similarity calculation methods. In the first column, the cyan line is the outline of the ground truth; the green points are the foreground interaction points; Middle: Heatmap results of the CAISeg with and without deep supervision. The cyan line in the picture is the outline of the ground truth, and the red points are the background interaction points; Right: Experiment results of different  $\lambda$  in the Focus Guided Loss.

strategy within the context of our specific task. Insights from Table III reveal that:

- Positional information plays a pivotal role in assessing the semantic similarity between feature vectors and interaction points, to the extent that it is a key determinant in the network's trainability. Furthermore, given that CAISeg exclusively operates with sparse feature vectors corresponding to interaction points for cross-computation against the feature map, learning the parameters for absolute position embedding becomes particularly challenging.
- The fusion of Euclidean distance with Gaussian RBF significantly ( $p < 0.05$ ) enhances the delineation of feature vectors that are strikingly similar to the interaction points. This is because cosine similarity only accounts for vector orientation—pattern similarity—whereas identifying inconspicuous tail-end features requires evaluating both vector direction and magnitude. Euclidean distance excels in capturing this comprehensive relationship. Left picture in Fig. 4 shows the clustering heatmap results obtained using these two different methods of calculating vector similarity. Notably, the method that utilizes Euclidean distance in conjunction with a Gaussian radial basis function demonstrates a superior ability to delineate areas that share a high semantic resemblance with the user interaction points.
- Employing a multi-head mechanism to break down high-dimensional feature vectors into several

TABLE IV  
ABLATION EXPERIMENTS ON THE DEEP SUPERVISION OF THE INTERACTION-GUIDED MODULE

deep supervision	DSC(%)		
	Brain Tumor	Colon Cancer	Lung Cancer
×	92.1±3.8	77.6±7.9	80.3±8.8
✓	93.4±2.2	81.1±7.3	83.3±7.1

lower-dimensional ones allows Euclidean distance to compute feature similarities with enhanced accuracy. This configuration achieves the best performance in our experiments, with the highest DSC score and the lowest mNOC metric. The lower variability observed in the results indicates a more consistent and reliable performance across different random seeds, further supporting the effectiveness of this design.

2) *The Effects of Deep Supervision of the Interaction-Guided Module*: Implementing deep supervision on the Interaction-guided Module enhances the model's ability to effectively learn head feature queries and parameters within both the cross-attention and feature clustering modules. Table IV illustrates the effects of deep supervision on the CAISeg's performance across three datasets. It is evident that the use of deep supervision not only improves the quality of the network's segmentation

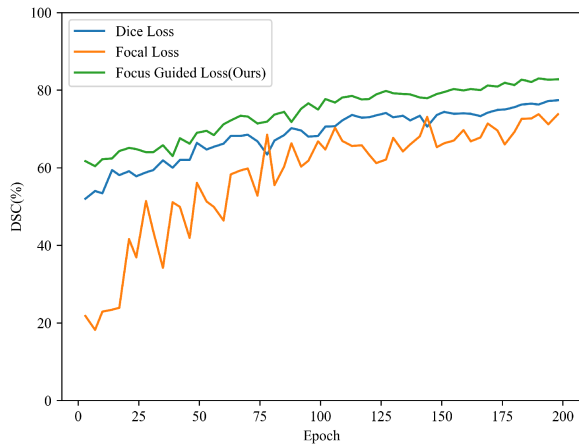


Fig. 5. Validation DSC over training epochs for CAISeg on the Pancreas dataset using different loss functions.

capabilities but also contributes to greater stability in the model's performance. Middle picture in Fig. 4 provides a visual comparison of similarity heatmaps both before and after applying the Interaction-guided Module adjustments, with and without the strategy of deep supervision. The comparative analysis of pre-adjustment heatmaps reveals that CAISeg, with deep supervision, is more adept at learning head features. Meanwhile, the post-adjustment heatmap comparison shows that CAISeg gains a better understanding of the user's interactive intent with deep supervision implemented.

**3) The Effects of the Focus Guided Loss:** The Focus Guided Loss function, by employing a weighted approach, steers the network's attention toward boosting prediction accuracy for voxels near user interaction points. Here, the hyper-parameter  $\lambda$  determines the scope of influence for the significance weighting. In extreme cases, when  $\lambda$  equals zero, the Focus Guided Loss degenerates to BCE Loss. Right picture in Fig. 4 showcases how different values of  $\lambda$  affect the CAISeg's metrics on brain tumor dataset. It is observable that varying  $\lambda$  values can impact both the quality of the network's initial segmentation and its responsiveness to interaction points. With a lower  $\lambda$  value, the network performs better in segmentation quality with fewer interaction points but demonstrates weaker comprehension of user interactions. Conversely, with an overly high  $\lambda$  value, the network's capability to extract visual features is compromised, which in turn affects the results of similarity computations between feature vectors within the Feature Clustering Module, leading to a decline in the network's overall segmentation abilities.

Furthermore, we evaluated the performance of CAISeg by training the model using three different loss functions: Dice Loss, Focal Loss, and our proposed Focus Guided Loss. The results, shown in Fig. 5, reflect the Dice Similarity Coefficient (DSC) values after three rounds of interactions on the validation set across 200 epochs. The model trained with Focus Guided Loss consistently achieved higher DSC values throughout the training process. In comparison, the model trained with Dice Loss displayed moderate performance, reaching a plateau around the 125th epoch. Meanwhile, the model trained with

Focal Loss exhibited greater variability and slower convergence, resulting in significantly lower DSC values. These findings highlight that Focus Guided Loss offers a more stable and effective learning process, leading to superior segmentation performance, particularly in the later stages of training. This emphasizes the practical advantages of Focus Guided Loss for interactive segmentation tasks, where precise alignment with user-guided features is essential.

## VI. CONCLUSION

In this work, we presented CAISeg, an advanced interactive neural network designed for the segmentation of lesions in medical images. CAISeg requires only minimal input from medical experts, such as simple clicks within areas of misprediction on existing segmentation masks. Leveraging a clustering approach, the network effectively captures and realigns semantic features that closely resemble the interaction points, ensuring accurate correction of segmentation errors by mapping these features to the appropriate head feature distribution. Furthermore, our proposed Focus Guided Loss has demonstrated superior effectiveness compared to conventional loss functions in enhancing the performance of point-based interactive segmentation networks. However, it is important to note that the efficacy of CAISeg's Interaction-Guided Module (IGM) is contingent on the availability of a sufficiently large training dataset. Smaller datasets can lead to the network becoming overly sensitive to interaction points, resulting in inconsistent segmentation quality. This highlights the need for adequately sized datasets to fully harness the potential of CAISeg, ensuring stable and reliable segmentation performance across various medical imaging tasks.

## REFERENCES

- [1] F. Isensee, P. F. Jaeger, S. A. Kohl, J. Petersen, and K. H. Maier-Hein, "NNU-Net: A self-configuring method for deep learning-based biomedical image segmentation," *Nat. Methods*, vol. 18, no. 2, pp. 203–211, 2021.
- [2] A. Hatamizadeh, V. Nath, Y. Tang, D. Yang, H. R. Roth, and D. Xu, "Swin unetr: Swin transformers for semantic segmentation of brain tumors in MRI images," in *Proc. Int. MICCAI Brainlesion Workshop*, 2021, pp. 272–284.
- [3] J. Liu et al., "Clip-driven universal model for organ segmentation and tumor detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 21152–21164.
- [4] N. Xu, B. Price, S. Cohen, J. Yang, and T. S. Huang, "Deep interactive object selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 373–381.
- [5] Q. Liu, Z. Xu, Y. Jiao, and M. Niethammer, "isegformer: Interactive segmentation via transformers with application to 3 D knee mr images," in *Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv.*, 2022, pp. 464–474.
- [6] T. Zhou, L. Li, G. Bredell, J. Li, J. Unkelbach, and E. Konukoglu, "Vol-umetric memory network for interactive medical image segmentation," *Med. Image Anal.*, vol. 83, 2023, Art. no. 102599.
- [7] J.-W. Zhang et al., "DINs: Deep interactive networks for neurofibroma segmentation in neurofibromatosis type 1 on whole-body MRI," *IEEE J. Biomed. Health Inform.*, vol. 26, no. 2, pp. 786–797, Feb. 2022.
- [8] G. Wang et al., "Deepigeeos: A deep interactive geodesic framework for medical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1559–1572, Jul. 2019.
- [9] X. Luo et al., "Mideepseg: Minimally interactive segmentation of unseen objects from medical images using deep learning," *Med. Image Anal.*, vol. 72, 2021, Art. no. 102102.
- [10] Q. Yu et al., "CMT-deeplab: Clustering mask transformers for panoptic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 2560–2570.



- [11] Q. Yu et al., "K-means mask transformer," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 288–307.
- [12] M. Yuan et al., "Devil is in the queries: Advancing mask transformers for real-world medical image segmentation and out-of-distribution localization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 23879–23889.
- [13] Y. Ding, L. Li, W. Wang, and Y. Yang, "Clustering propagation for universal medical image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pp. 3357–3369, 2024.
- [14] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.
- [15] B. Cheng, A. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, vol. 34, pp. 17864–17875.
- [16] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis.*, 2016, pp. 565–571.
- [17] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2980–2988.
- [18] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
- [19] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Med. Image Comput. Comput.-Assist. Interv. 2015, 18th Int. Conf.*, 2015, pp. 234–241.
- [20] X. Li, H. Chen, X. Qi, Q. Dou, C.-W. Fu, and P.-A. Heng, "H-DenseUNet: Hybrid densely connected UNet for liver and tumor segmentation from CT volumes," *IEEE Trans. Med. Imag.*, vol. 37, no. 12, pp. 2663–2674, Dec. 2018.
- [21] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet : Redesigning skip connections to exploit multiscale features in image segmentation," *IEEE Trans. Med. Imag.*, vol. 39, no. 6, pp. 1856–1867, Jun. 2020.
- [22] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. ICLR*, 2021.
- [23] A. Hatamizadeh et al., "UNETR: Transformers for 3D medical image segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2022, pp. 574–584.
- [24] H. Cao et al., "Swin-UNet: Unet-like pure transformer for medical image segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 205–218.
- [25] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10012–10022.
- [26] X. Liang, N. Li, Z. Zhang, J. Xiong, S. Zhou, and Y. Xie, "Incorporating the hybrid deformable model for improving the performance of abdominal CT segmentation via multi-scale feature fusion network," *Med. Image Anal.*, vol. 73, 2021, Art. no. 102156.
- [27] W. He et al., "A statistical deformation model-based data augmentation method for volumetric medical image segmentation," *Med. Image Anal.*, vol. 91, 2024, Art. no. 102984.
- [28] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988.
- [29] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in nd images," in *Proc. 8th IEEE Int. Conf. Comput. Vis.*, 2001, vol. 1, pp. 105–112.
- [30] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.
- [31] R. Adams and L. Bischof, "Seeded region growing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 6, pp. 641–647, Jun. 1994.
- [32] V. Vezhnevets and V. Konouchine, "Growcut: Interactive multi-label nd image segmentation by cellular automata," in *Proc. Of Graphicon*, 2005, vol. 1, pp. 150–156.
- [33] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool, "Deep extreme cut: From extreme points to object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 616–625.
- [34] K. Sofiiuk, I. A. Petrov, and A. Konushin, "Reviving iterative training with mask guidance for interactive segmentation," in *Proc. IEEE Int. Conf. Image Process.*, 2022, pp. 3141–3145.
- [35] W. Liu, C. Ma, Y. Yang, W. Xie, and Y. Zhang, "Transforming the interactive segmentation for medical imaging," in *Int. Conf. Med. Image Comput. Comput.- Assist. Interv.*, 2022, pp. 704–713.
- [36] J. Ma, Y. He, F. Li, L. Han, C. You, and B. Wang, "Segment anything in medical images," *Nat. Commun.*, vol. 15, no. 1, 2024, Art. no. 654.
- [37] A. Kirillov et al., "Segment anything," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2023, pp. 4015–4026.
- [38] W. Lei, X. Wei, X. Zhang, K. Li, and S. Zhang, "Medlsam: Localize and segment anything model for 3D medical images," 2023, *arXiv:2306.14752*.
- [39] H. Wang et al., "Sam-Med3d," 2023, *arXiv:2310.15161*.
- [40] H. Wang, Y. Zhu, H. Adam, A. Yuille, and L.-C. Chen, "Max-deeplab: End-to-end panoptic segmentation with mask transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 5463–5474.
- [41] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 1290–1299.
- [42] M. Antonelli et al., "The medical segmentation decathlon," *Nat. Commun.*, vol. 13, no. 1, 2022, Art. no. 4128.
- [43] F. Pérez-García, R. Sparks, and S. Ourselin, "Torchio: A python library for efficient loading, preprocessing, augmentation and patch-based sampling of medical images in deep learning," *Comput. Methods Programs Biomed.*, vol. 208, 2021, Art. no. 106236.
- [44] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 1026–1034.
- [45] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, "Deep networks with stochastic depth," in *Proc. Comput. Vis.-ECCV 2016, 14th Eur. Conf.*, 2016, pp. 646–661.
- [46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.