Shape induced Multi-Class Deep Graph Cut for Hippocampus Subfield Segmentation

Arijit De[1[0000−0003−3503−254X]] and Anands S. Chowdhury[1[0000−0002−5799−3467]]

Jadavpur University, Kolkata, West Bengal 700032, India
{arijitde.etce.rs,as.chowdhury}@jadavpuruniversity.in

Abstract. Automated evaluation of hippocampus volume plays a crucial role in the analysis of various neurodegenerative conditions like Alzheimer's Disease and Epilepsy. Examination of the hippocampus subfields assumes paramount importance as it can reveal early signs of brain abnormalities. However, delineating these subfields becomes extremely challenging due to their intricate nature and the requirement for manually annotated high-resolution magnetic resonance images. In this paper, we propose an innovative deep graph cut approach, boosted by shape information, for automatic segmentation of hippocampus subfields. A deep learned shape term is incorporated in the energy function of the graph cut. A modified $\alpha-\beta$ swap technique, that leverages deep learning, is designed to improve the execution time of the proposed multi-class segmentation algorithm. We demonstrate the efficacy of our solution by outperforming a number of state-of-the-art methods on the publicly available Kulaga-Yoskovitz dataset.

Keywords: Hippocampus subfield segmentation · Multi-class Graph cut · Shape term · Deep Learning.

1 Introduction

The hippocampus (HC) is a paired brain structure situated in the medial temporal lobe adjacent to the brainstem in close proximity to the cerebellum. It plays a crucial role in various cognitive functions, such as, memory and spatial reasoning [23]. Over the past decade, there has been a growing interest in segmenting hippocampal subfields using MRI. Recent research has identified distinct functional roles for these anatomical subregions, with CA1 implicated in memory integration and inference [28], CA3 in memory retrieval [8], and both the dentate gyrus (DG) and CA3 in pattern separation [2]. Clinically, the volume or morphology of the hippocampus and its subfields are closely related to many neurodegenerative diseases like Epilepsy [32] and Alzheimer's disease [17]. So, it is desirable to develop automatic hippocampal subfields segmentation from brain MR image. However, manual delineation of hippocampal subfields is a laborious and time-intensive task, leading to constraints on sample sizes in various studies.

We first discuss some works that use classical techniques for HC subfield segmentation. The authors in [36] used a multi-atlas approach combined with a

similarity-weighted voting and a boosting-based error correction as a solution. They termed their method, ASHS. This method took several hours to produce a segmentation due to exhaustive use of non-linear registrations. More recently, a method named HIPS [26] obtained state-of-the-art results with relatively low processing times. While classical methods integrated domain-specific image features like gradient, intensity, and textures within an energy minimization framework, they are found to heavily depend on initialization, such as manual seeding. As a result, they are prone to segmentation errors caused by uncertain positioning of the markers. Furthermore, these approaches are quite laborious and may not be practical for clinical environments with a heavy workload.

Deep learning approaches have surpassed classical methods, delivering superior segmentation performance in significantly less time. Recently, due to the expansion of deep learning (DL) in medical imaging for tasks like classification [10] and segmentation [11], novel methods based on this technology have been proposed to further improve the accuracy of HC sub-field segmentation. UNet based methods [22], [37] have shown promising results. Shi et al. [29] proposed a Generative Adversarial Network (GAN) to create a segmentation model. But UNet and GAN based methods require a lot of data and medical imaging lacks consistent and sufficient annotated data, making DL algorithms perform poorly in many cases [31]. Although some authors have tried to bypass this problem using multi scaling technique [35] and using higher resolution data [20], the problem continues to exist.

A combination of both classical and DL techniques can achieve better segmentation performance than using DL methods or classical methods in isolation. For example, see the works [27], [24] in lung nodule segmentation. In case of 3D brain tumor segmentation [12], a combination of UNet and graph cut helped circumvent manual seeding problem of the graph cut and undersegmentation of UNet due to scarcity of data.

In this paper, we propose a shape driven multi-class segmentation method using UNet [9] and graph cut. We take inspirations from [12], [11] and [33] to create a state-of-the-art model to segment the HC into three classes, namely, CA1-3, CA4/DG and Subiculum (Sub). In [5], Boykov et al. showed that a two-class segmentation is achievable in polynomial time using graph cuts. However, if the number of labels exceeds 2 (as for the present problem), finding an exact solution becomes an NP-hard problem. They suggested two types of large moves (changing labels of individual pixels/voxels) based on minimal graph cuts, namely, α-expansion and α-β swap. In this work, we use UNet to improve the α-β swap. A number of research papers demonstrated that use of shape information can improve the segmentation accuracy [21], [19], [11]. Shape priors provide valuable guidance by incorporating prior knowledge about the expected shapes of objects in the image. This guidance helps the segmentation algorithm to make more informed decisions about the boundaries and regions of interest [30]. By imposing shape constraints, shape priors help to enforce consistency in the segmented shapes, ensuring that the output conforms to the expected shape characteristics [1]. The shape prior has to be made adaptive in case of substan-

tial noise and intensity variations. Here, we better an adaptive shape prior from deep learned information via UNet. Our main contributions are now summarized below:

1. Propose a new energy function for multi-class segmentation based on graph cut and deep learning (UNet)
2. Incorporate learned information from UNet for optimizing the number of α-β swaps
3. Show how an adaptive shape prior can be learned from UNet

2 Proposed Method

2.1 Deep Graph Cut

Let us define the 3D MRI input as a gray-scale volumetric data, which may be represented as a 3D weighted graph denoted by $G = G(V, E)$. Each vertex is represented by a voxel x in G, and X is the collection of all voxels. We introduce two new vertices, called 'source' and 'sink', represented by s and t respectively. There are two sorts of edges or linkages that we consider: t-links (T) and n-links (N). s and t is linked to very voxel x through t-links. We utilize a compact 26-neighborhood, represented as $Ne(x)$ for every voxel x. Assume that y is a neighbor of x. Therefore, y belongs to the neighborhood of x, and we establish a connection between x and y by an n-link. Therefore, the set V is defined as the union of sets X, s, and t, whereas the set E is defined as the union of sets T and N. Let us establish a segmentation A as a classification of all voxels into two distinct classes: "object" or "background". This classification is done on a voxel-wise basis. Therefore, according to the reference [7], it is necessary to minimize the subsequent energy function:

$$\zeta(A) = B(A) + \lambda R(A) \tag{1}$$

The term $B(A)$ represents the boundary characteristics or smoothness term of A, while $R(A)$ represents the regional properties or data term of A. These terms are represented mathematically as below:

$$B(A) = \sum_{x \in X, y \in Ne(x)} B_{(x,y)} \tag{2}$$

$$R(A) = \sum_{x \in X} R_x \tag{3}$$

In [12], we modified the above energy function (Eq. 1) by incorporating learned information from the 3D UNet [9]. The modified energy function is given below:

$$\begin{aligned} \zeta_{DGC}(A) = & \sum_{x \in X, y \in Ne(x)} B_{DGC}(x, y) + \\ & \lambda_{DGC}(x) \sum_{x \in X} R_{DGC}(x) \end{aligned} \tag{4}$$

2.2 Multi-class Deep Graph Cut

As stated earlier, in this work, we deal with multi-class hippocampus segmentation where a voxel x can belong to any of 'CA1-3', 'CA4/DG' and Subiculum. Following [5], our goal is to find a labeling f that assigns each voxel $x \in X$ a label $f_x \in \mathcal{L}$ and, $|\mathcal{L}| > 2$, where f is both piecewise smooth and consistent with the observed data. Any labeling f can be uniquely represented by a partition of image voxels, $V = V_l | l \in \mathcal{L}$ where $V_l = x \in V | f_x = l$ is a subset of pixels assigned a label l. Hence, Eq. 4 can be rewritten as:

$$\zeta_{DGC}(A_f) = \sum_{x \in X, y \in Ne(x)} B_{DGC}(f_x, f_y) + \lambda_{DGC}(f_x) \sum_{x \in X} R_{DGC}(f_x) \tag{5}$$

where $\zeta_{DGC}(A_f)$ is the energy of the labelling f. The knowledge acquired from the 3D UNet [9] is included into the energy function of the 3D graph cut algorithm in order to achieve precise segmentation. The 3D probability map is obtained from the last convolutional layer for each image. This map is then used to determine the probability, denoted as $Pr(f_x)_{UN}$, of any voxel x belonging to label f_x. The 3D UNet calculates a regression function that maps the voxels of a 3D input to a 3D voxel-wise probability map. This is denoted as $\mathcal{P} : \mathbb{R}^3 \rightarrow (0, 1)$, and it assigns a value between 0 and 1 to each voxel. Additionally, this probability map is used as an automated seed required by 3D graph cut algorithm. With this, we now explain the smoothness and data term in the context of multi label problem as follows. As mentioned in [12], $B_{DGC}(f_x, f_y)$ is a product of four components as shown below-

$$B_{DGC}(f_x, f_y) = K_{(x,y)} \times e^{-(\frac{(I_x - I_y)^2}{2\sigma^2})} \times \frac{1}{d(x,y)} \times \frac{1}{\delta(x,y)_{DGC}} \tag{6}$$

where $d(x, y)$ represents the Euclidean distance between two voxels x and y having intensity values I_x and I_y respectively. The term $K_{(x,y)}$ is based on the probabilities of x and y to have the labeling f_x and f_y and is mathematically represented as:

$$K_{(x,y)} = 1 - |Pr(f_x)_{UN} - Pr(f_y)_{UN}| \tag{7}$$

where $f_x = f_y$. The factor σ is the standard deviation of voxel intensities of the image [19]. The term $\delta(x, y)_{DGC}$ denotes the sum of differences between probabilities of neighbouring voxels x and y to belong to f_x and f_y where $f_x \neq f_y$. This can be expressed as:

$$\begin{aligned} \delta(x,y)_{DGC} = &|Pr(f_x = \alpha)_{UN} - Pr(f_y = \alpha)_{UN}| \\ &+ |Pr(f_x = \beta)_{UN} - Pr(f_y = \beta)_{UN}| \end{aligned} \tag{8}$$

where $\alpha, \beta \in \mathcal{L}$. The data term $R_{DGC}(f_x)$ is dependent on the probability map of UNet as shown below-

$$R_{DGC}(f_x) = -\ln Pr(f_x = \alpha)_{UN} \tag{9}$$

***α-β* Swap** As mentioned in Boykov et al.'s article [5], segmenting a binary image is possible in polynomial time using graph cut. But if the number of labels is more than 2 (as in our case), finding exact solution becomes an NP-hard problem. Therefore, Boykov et al. proposed two types of moves based on minimal graph cuts - α-expansion move and α-β swap move. A standard move means changing the label of a single vertex (voxel in our case). We now discuss how and why we modify the optimization function of α-β swap moves. The choice of which type of move to select depends on whether the smoothness term of the energy function is a metric or a semi-metric [5]. If the smoothness term is metric, α-expansion can be used, otherwise α-β swap move needs to be used. For a function $V(\alpha, \beta)$ to be metric, it has to satisfy the following constraints:

1. $V(\alpha, \beta) \Leftrightarrow \alpha = \beta$
2. $V(\alpha, \beta) = V(\beta, \alpha) \geq 0$
3. $V(\alpha, \beta) \leq V(\alpha, \gamma) + V(\gamma, \alpha)$

for any labels $\alpha, \beta, \gamma \in \mathcal{L}$ [5]. If $V(\alpha, \beta)$ satisfies only the constraints (1) and (2) but not (3), then it is called a semi-metric. We have chosen α-β swap moves to optimize our energy function, as our smoothness term ($B_{DGC}(f_x, f_y)$) is a semi-metric. We explicitly show in the appendix that the smoothness term is indeed a semi-metric.

Deep learned *α-β* Swap If a move from a partition V_l to a new partition V_l' has labels α, β, then $V_l' = V_l$ for all labels $l \neq \alpha, \beta$. This is known as a α-β swap [5]. So, the only thing that's different between V_l and V_l' is that some voxels that were labeled as α in V_l are now labeled as β, and the other way around. The main idea is to use graph cuts to separate all α voxels from β voxels one by one. Each time through the algorithm, the $\alpha - \beta$ mix will be different. The program will keep going through all the possible combinations until it converges with the minimum energy. The algorithm is guaranteed to converge in $O(V)$ time, but when there are a lot of vertices, the whole segmentation process takes a long time.

As reported in [5], segmenting a 384×288 image with $\alpha - \beta$ swap takes 35 seconds. In our case, the image size is $182 \times 218 \times 182$ which is far greater than the images used in [5]. Hence, there is a dire need to optimise the move algorithm to speed up the overall segmentation.

For this, we turn to Pseudo-Boolean optimization techniques used in [6]. As mentioned in [6], we encode the moves of the $\alpha - \beta$ swap algorithm as a vector of binary variables $t = t_i, \forall_i \in V$. $t_i = 0$ means the label of voxel i changed to α and $t_i = 1$ means the label changed to β. The transformation function $T(f^c, t)$ of a move algorithm takes the current labelling f^c and a move t and returns a new labelling f^n that has been induced by the move. The transformation function $T_{\alpha\beta}()$ for an $\alpha - \beta$ swap transforms f^c as

$$f_i^n = T_{\alpha\beta}(f_i^c, t_i) = \begin{cases} f_i^c, & \text{if } f_i^c \neq \alpha \text{ and } f_i^c \neq \beta, \\ \alpha, & \text{if } f_i^c = \alpha \text{ or } \beta \text{ and } t_i = 0, \\ \beta, & \text{if } f_i^c = \alpha \text{ or } \beta \text{ and } t_i = 1. \end{cases} \tag{10}$$

If the current labelling f_i^c is neither α nor β, we don't change it. The energy of the move t is the energy of labelling f^n that the move t induces, i.e., $E_m(t0 = E(T(f^c mt))$. Further details about the pseudo boolean energy of the swap move can be found in Sec. 3.3 of [4].

We modify Eq. 10 by adding the label probability information derived from UNet as follows-

$$f_i^n = T(f_i^c, t_i) = \begin{cases} f_i^c, & \text{if } f_i^c \neq \alpha \text{ and } f_i^c \neq \beta, \\ & \\ & \text{if } f_i^c = \alpha \text{ or } \beta, t_i = 0, [Pr(f_x = \alpha) - Pr(f_x = \beta) > \tau] \\ \alpha, & \text{, or } [Pr(f_x = \beta) - Pr(f_x = \alpha) < (1 - \tau)], \\ & \\ & \text{if } f_i^c = \alpha \text{ or } \beta, t_i = 1, [Pr(f_x = \beta) - Pr(f_x = \alpha) > \tau] \\ \beta, & \text{, or } [Pr(f_x = \alpha) - Pr(f_x = \beta) < (1 - \tau)], \end{cases} \tag{11}$$

We added two more constraints when deciding the new labelling to be α or β. We wanted the confidence of the UNet model to decide whether a label should be swapped or should be kept the same. We define confidence of prediction as the difference between the probabilities of a voxel to have label α and β, i.e. $[Pr(f_x = \alpha) - Pr(f_x = \beta)]$. Generally, a model is said to predict a label (say α) with high confidence if the probability of the voxel to belong α is much higher than that of the voxel to belong to another label (say β), i.e., if $[Pr(f_x = \alpha) - Pr(f_x = \beta)] > \tau$ or $[Pr(f_x = \beta) - Pr(f_x = \alpha) < (1 - \tau)]$ where τ is some threshold. So, if for any voxel i, $[Pr(f_x = \alpha) - Pr(f_x = \beta)] > \tau$ and f_i^c, the current label of i is either α or β, then the label of voxel i will be changed to α. Similarly, the decision to change a label to β if the confidence of the UNet model for that voxel to have label β is greater than τ.

2.3 Deep Learned Shape Information

In this section, we first discuss what is the significance of addition of a shape term in HC segmentation, then we briefly mention the importance of adaptive shape term and how the incorporation of UNet's probability map helps in creating an adaptive shape term suitable for 3D HC segmentation.

Need of an Adaptive Shape term In cases where images are affected by substantial noise and intensity variations, the necessity for a shape prior can vary across different pixels. Consequently, assigning a uniform weight to the shape prior term for all pixels may not be suitable. In our case, 3D MRI images do suffer from noise and intensity variations and in many places the HC and non HC region of the brain has very low contrast as shown in Fig. 1. Segmentation tasks use the adaptive shape term to selectively impose shape constraints based on pixel labeling difficulty to give flexibility and local adaptation. This adaptability allows the system to modify shape prior strength based on local image properties,

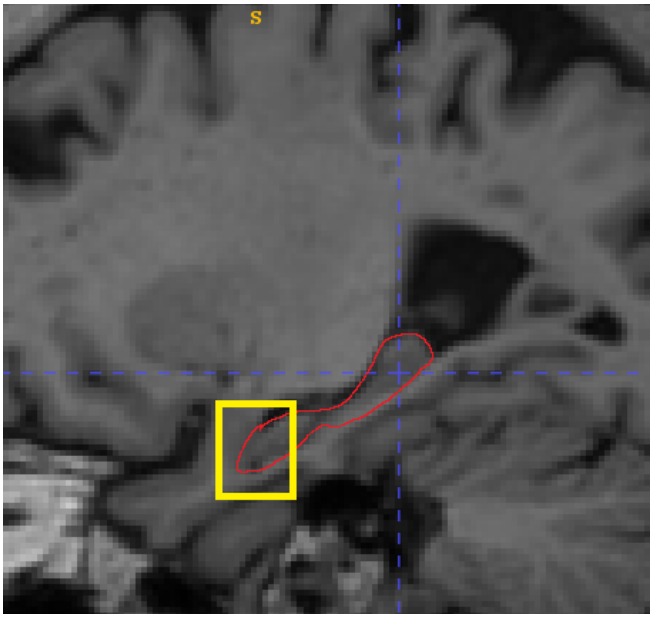

Fig. 1. A sagittal slice view of a brain MRI showing the hippocampus bounded in red. The region marked in yellow shows that the contrast is less between hippocampus and its surrounding region which poses a challenge in segmentation.

applying shape restrictions where they are most useful. The adaptive shape term dynamically adjusts shape priors based on image intensity, resulting in more accurate and context-aware segmentation results [33].

Improved Adaptive Shape Term with UNet We improve Wang et al.'s [33] adaptive shape prior formulation using learned information from 3D UNet. Following their approach, we add to the smoothness term $B_{DGC}(x, y)$, a shape term of the form $S_{DGC}(x, y)$ with η as the shape weight. Note that the authors in [33] defined $\eta = e^{-(Pr(x)-Pr(y))^2}$. $Pr(k)$ is the likelihood of a pixel k belonging to the foreground. They determined this likelihood by using an unsupervised technique like applying Gaussian filter.

Unlike in [33], where the authors used 2D images and performed binary segmentation, we deal with 3D images and multi-class segmentation in this work. So we redefine η as:

$$\eta = e^{-(Pr(f_x)-Pr(f_y))^2} \tag{12}$$

where, $Pr(f_k)$ denotes the likelihood of voxel k to have labelling f_k. Further, $Pr(f_k)$ is obtained from the probability map of UNet as mentioned in Sec. 2.1. This ensures that we have better probability values than that obtained from using unsupervised techniques, as in [33].

The shape term, S_{DGC}, can be formulated as the unsigned distance function (as used in [16]) of the segmentation obtained after thresholding probability map $\mathcal{P}$. Let the segmentation obtained by thresholding $\mathcal{P}$ is $\mathcal{G}$ with a threshold value of κ. Then,

$$S_{DGC} = \bar{\phi}_{\mathcal{G}} \left(\frac{x+y}{2} \right) \tag{13}$$

where, $\bar{\phi}_{\mathcal{G}} : \Re^3 \rightarrow \Re$ is the distance function on $\mathcal{G}$ and is such that $\bar{c} = x \in \Re^3 : \bar{\phi}(x) = 0$; $\bar{c}$ being the set of points that form the boundary of the shape. The energy will be low if $\bar{\phi}_{\mathcal{G}} \left(\frac{x+y}{2} \right) \approx 0$ for all neighboring voxels x and y and $f_x \neq f_y$. If a voxel x lies near the shape template, then it will satisfy $\bar{\phi}(x) \approx 0$. Since, $\left(\frac{x+y}{2} \right)$ is roughly a point on the boundary of the segmented object, the condition for S_{DGC} to be small is the same as the condition that the boundary of the segmented object lies near the shape template.

2.4 Shape driven Multi-class Deep Graph Cut

We started with Eq. 4 which is the deep graph cut for energy function for binary segmentation. Then we modified it to adapt to multi class segmentation in Eq. 14. We then modified the α-β swap moves using information from UNet as described in Sec. 2.2. We compute the data term $R_{DGC}(f_x)$, smoothness term $B'_{DGC}(f_x, f_y)$ and Finally, after incorporating the two terms described in the previous section and shown in Eq. 12 and Eq. 13 in the energy function of multi class deep graph cut (Eq. 5), we get the final energy function for Shape induced Multi class Deep Graph Cut (SMDGC) method as shown below-

$$\begin{aligned} \zeta_{SMDGC} = & \sum_{x \in X, y \in Ne(x), f_x \neq f_y} B_{DGC}(f_x, f_y) + \eta S_{DGC} + \\ & \lambda_{DGC}(f_x) \sum_{x \in X} R_{DGC}(f_x) \end{aligned} \tag{14}$$

The algorithm for our overall workflow is shown below in Algorithm 1 followed by a discussion on its time complexity-

2.5 Analysis of Time-complexity

The alpha-beta swap algorithm, used for multi-label graph cuts, iteratively optimizes the graph $G = G(V, E)$ by swapping labels between pairs (α, β) to minimize the energy function. The time-complexity of this algorithm is influenced by factors, such as, the number of labels, the number of pixels (or nodes), and the underlying max-flow algorithm used. The time-complexity analysis of the proposed SMDGC algorithm is as follows:

1. Max-flow Computation: The time-complexity of each max-flow computation depends on the specific max-flow algorithm used. We have used Edmond Karp's approach [14] in the Ford-Fulkerson algorithm [15], where augmenting paths are computed using the Breadth First Search. It has a time-complexity of $O(VE^2)$.

Algorithm 1: SMDGC

Input: 3D UNet model M trained on training set of the data, Graph G represented as a 3D grid of voxels

Output: G_{out} with desired segmentation and same dimensions as G

Compute data term $R_{DGC}(f_x)$ for each voxel in $x \in G$ as shown in Equation 9

Compute smoothness term $B_{DGC}(f_x, f_y)$ for each voxel x and its neighbor $y \in N_e(x)$ in G as shown in Equation 6

Compute η and S_{DGC} as shown in Equations 12 and 13 respectively

Compute the modified transformation function for optimizing the number of α-β swap moves using Equation 11.

Compute the final energy function using Equation 14, perform Graph cut and store the result in G_{out}.

return G_{out}

2. Number of Labels: The alpha-beta swap considers all pairs of labels. So, the number of iterations is proportional to $\binom{\mathcal{L}}{2} = \frac{\mathcal{L}(\mathcal{L}-1)}{2}$, which is $O(\mathcal{L}^2)$, where $\mathcal{L}$ denotes the number of labels.

Combining the two factors, we can say that the time-complexity of our algorithm is $O(\mathcal{L}^2 V E^2)$.

3 Experimental Results

In this section, we first describe the dataset used for the experimentation. Necessary details of the parameters, and, hyperparameters used, are provided next. To showcase the significance of different components of our solution, we then present a number of ablation studies. Finally, we show performance comparisons with several state-of-the-art approaches.

3.1 Dataset

We have used a publicly available 3D MRI dataset, described in [18]. We henceforth abbreviate this Kulaga-Yoskovitz dataset as the KY dataset. It comprises 25 healthy adult subjects aged between 21 and 53 years, with a mean age of 31.2 ± 7.5 years and a male-to-female ratio of 12 : 13. The data were acquired using a 3T Siemens Tim Trio MRI scanner equipped with a 32-channel head coil. Submillimeter T1 and T2 images were obtained for all participants. The 3D MPRAGE T1 image had a spatial resolution of $0.6 \times 0.6 \times 0.6mm^3$ (isotropic voxel size). The matrix size was 336 x 384, with a field of view (FOV) of 201 mm x 229 mm and 240 axial slices at a slice thickness of 0.6 mm. The T2 image was acquired using a 2D turbo spin-echo sequence with a matrix size of 512 x 512, an FOV of 203 mm x 203 mm, and 60 coronal slices angled perpendicular to the hippocampal long axis, with a slice thickness of 2 mm, resulting in a voxel size of $0.4 \times 0.4 \times 2.0mm^3$. The manual segmentation protocol for this dataset categorized the hippocampus into three labels: subiculum (SUB), a combination of CA1, CA2, and CA3 (CA1-3), and a combination of CA4 and DG (CA4/DG).

3.2 Preprocessing

We preprocessed the data using the steps described in [22] that included cropping along the HC area and data augmentation by left - right flipping. Finally, we got 50 samples having 100 axial slices with the length and breadth same as mentioned earlier for T1 and T2 images. Since, the number of samples is less, we did 5-fold cross validation to assess the results.

3.3 Experimental settings

We implemented our 3D UNet network in PyTorch [25]. The training process was executed on a HP-Z640 workstation having Intel Xeon processor with 14 Cores, a Random Access Memory (RAM) with capacity of 128GB along with a dedicated graphics processor unit (GPU) of 24GB with model name NVIDIA Titan RTX. The network is trained for 100 epochs with initial learning rate of 0.0001, weight decay of 0.00001 and mini-batch size equal to 2 samples. We have used Adam Optimizer and dice loss in the process. Our energy function does not have any parameters that need to be set manually, as all the information is being provided by a trained UNet model. The only parameter κ, which is used for thresholding the UNet probabilities to create a segmentation (as described in Sec. 2.3) is set to, 0.5 which is the most common value as mentioned in [34].

We have used Dice score [13] as the metric to compare the segmentation performance, as this was the metric used by most other works on segmentation [29, 37, 20].

3.4 Ablation Studies

As mentioned earlier, we present three ablation studies for providing a better understanding of our solution. Table 1 shows the results of our first ablation

Table 1. Ablation Study I: Comparison of segmentation performance of multi-class graph cut, multi-class UNet, and the proposed method. Mean Dice Score of each competing approach over all three classes are reported. Best values are shown in **bold**.

Method	**Dice Score**
Multi-class Graph Cut	0.64000 ± 0.073
Multi-class UNet	0.82000 ± 0.047
SMDGC (Ours)	**0.91467 ± 0.009**

study. Here, we demonstrate the improvement our model brings over baseline multi-class graph cut and multi-class UNet, applied in isolation. In Table 2, we first show the performance of DGC [12] for multi-class segmentation, by adding to it traditional α-β swap. It is then demonstrated how the execution time

Table 2. Ablation Study II: Impact of deep learned α-β swap on the segmentation performance. Mean Dice Score of each competing approach over all three classes are reported. Best values are shown in **bold**.

Method	**Dice Score**	**Time (in secs)**
DGC [12] with α-β swap	0.88230 ± 0.032	15
DGC [12] with deep learned α-β swap	**0.89860 ± 0.008**	**8**

improves due to optimization of α-β swap strategy using learned information from UNet. The computation time improves drastically improves by almost 50% when we use modified α-β swap. DGC with normal α-β swap takes on average 15 seconds to segment one 3D sample, whereas, the modified α-β swap achieves the same goal in 8 seconds. A slight improvement in the segmentation accuracy can also be noticed as more informed decision is taken to change a label during a move due to the confidence of UNet incorporated into the optimization strategy (Eq. 11).

Table 3. Ablation Study III: Impact of deep learned shape on segmentation performance. Mean Dice Score of each competing approach over all three classes are reported. Best values are shown in **bold**.

Method	**Dice Score**
Graph cut with adaptive shape term [33]	0.85297 ± 0.026
SMDGC (Ours)	**0.91467 ± 0.009**

We then analyze the impact of a shape term in a graph cut setup through Table 3. The first row shows the results, where an adaptive shape term is used but without deep learning. For that, we re-implement [33] and add α-β swap moves, as that work was originally developed for binary segmentation. We compare the performance of this method with ours, where we have employed deep learned shape information (Eq. 14). The values of the Dice Scores clearly illustrate the benefits of a deep learned shape information.

Qualitative comparisons of different strategies used in the three ablation studies are shown in figure 2. We only include the second method of Table 2 as the improvement there is more in terms of execution time to achieve desired segmentation, rather than the segmentation accuracy per se. In Fig. 2, the visual improvements in segmentation performances clearly corroborate the quantitative results. We specifically highlight how multi-class UNet and graph cut with adaptive shape term suffers from over segmentation of CA1-3 and SUB, as shown in the yellow boxes of the sagittal slices. DGC with modified α-β swap also finds it difficult to decide among the CA1-3 and SUB, as shown in the yellow box

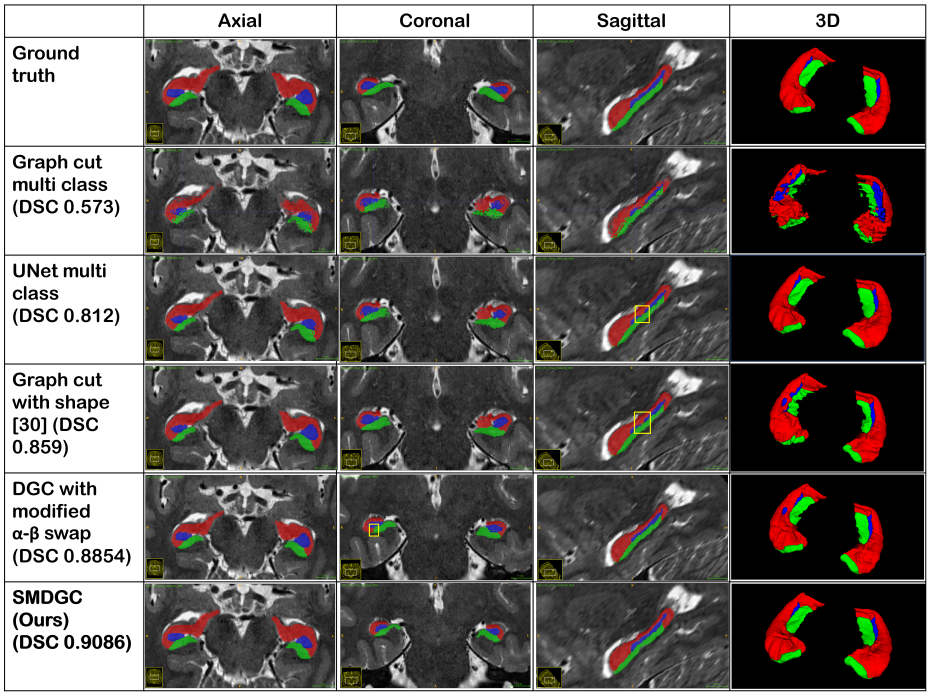

Fig. 2. Qualitative ablation of our method. GT represents the ground truth. Segmentation with color red represents CA1-3 class, blue represents the CA4/DG class and green represents the SUB class.

of its coronal slice. In general, CA4/DG is relatively difficult to segment by all methods, as it is the smallest region among the three classes under consideration.

3.5 Comparison with State-of-the-art Methods

We compare our proposed method with five state-of-the-art approaches (papers published within the last five years). These methods are [20], [35], [37], [22], and, [29]. We showed comparisons with only DL based approaches, as we did not come across any work on multi-class HC segmentation using primarily graph cuts. As can be clearly seen from the Table 4, our method has yielded the highest mean Dice Score, which is marginally better than [29]. We are marginally behind [29] in the SUB subfield segmentation, the most complex object to segment within the HC. However, our model requires much less computational resource, as we used only plain 3D UNet, and, graph cut while, other approaches have used sophisticated DL models that take a lot of time and resources to train.

Table 4. Comparison with state-of-the-art methods. Mean Dice Score ± standard deviation of Dice Score is reported for each class. Additionally, the overall mean Dice Score is reported for each method. Best values are shown in **bold**.

Method	CA1-3	CA4/DG	SUB	Mean
Syn SegNet (2023) [20]	0.865 ± 0.005	0.821 ± 0.014	0.821 ± 0.013	0.835 ± 0.007
CAST (2020) [35]	0.917 ± 0.011	0.89 ± 0.017	0.881 ± 0.021	0.906 ± 0.014
ResDUNet (2019) [37]	0.92 ± 0.011	0.879 ± 0.02	0.888 ± 0.018	0.896
UNet CNN (2022) [22]	0.9245 ± 0.01	0.8887 ± 0.023	0.898 ± 0.015	0.9
GANs (2019) [29]	0.919	0.903	**0.906**	0.88
SMDGC (Ours)	**0.933 ± 0.007**	**0.9078 ± 0.013**	0.903 ± 0.008	**0.9146 ± 0.009**

4 Conclusion

Hippocampus subfield segmentation is a crucial step in the diagnosis of many diseases like Alzheimer's, Epilepsy as the treatment depends on the analysis of volumetric atrophy of the subfields. Automating this process will greatly enhance the treatment experience for both doctors and patients. In this work, we proposed a state-of-the-art method of subfield segmentation using a combination of multi-class graph cuts with shape information and deep learning. In particular, we showed how deep learning can boost the shape knowledge, and the $\alpha - \beta$ swap move. Comparisons with a number of state-of-the-art methods on a publicly available dataset clearly establish the efficacy of our proposed solution. In the future, we plan to include other datasets that contain more subfields to achieve a more fine-grained segmentation of the hippocampus. We also plan to introduce explainability [3] into our proposed hippocampus segmentation model so that it can be more effectively used in the real-world clinical settings.

Acknowledgements Arijit De was supported by Tata Consultancy Services Research Scholar Program (TCS-RSP).

Appendix

Lemma 1. *The product of a semi-metric and a metric function is semi-metric.*

Proof. Let ρ_1 be a semi-metric function and ρ_2 be a metric function defined on some set X. Then, for any $x, y, z \in X$:

1. $\rho_1(x,z) \leq \rho_1(x,y) + \rho_1(y,z) = \Psi\left(\rho_1(x,y) + \rho_1(y,z)\right)$
2. $\rho_2(x,z) > \rho_2(x,y) + \rho_2(y,z) = \Theta\left(\rho_2(x,y) + \rho_2(y,z)\right)$

where $\Psi \leq 1$ and $\Theta > 1$. Now, consider the product

$$\begin{aligned}
\rho(x,z) &= \rho_1(x,z) \cdot \rho_2(x,z) \\
&= \Big(\Psi\left(\rho_1(x,y) + \rho_1(y,z)\right)\Big) \cdot \Big(\Theta\left(\rho_2(x,y) + \rho_2(y,z)\right)\Big) \\
&= \Psi\Theta\rho_1(x,y)\rho_2(x,y) + \Psi\Theta\rho_1(x,y)\rho_2(y,z) \\
&\quad + \Psi\Theta\rho_2(y,z)\rho_1(x,y) + \Psi\Theta\rho_2(y,z)\rho_1(y,z)
\end{aligned}$$

Now, $\Psi\Theta > 1$ when $\Psi = 1$. Hence,

$$\begin{aligned}
\rho(x,z) &> \rho_1(x,y)\rho_2(x,y) + \rho_1(y,z)\rho_2(y,z) + \rho_1(x,y)\rho_2(y,z) + \rho_2(y,z)\rho_1(x,y) \\
&> \rho(x,y) + \rho(y,z) + \omega_1 + \omega_2
\end{aligned}$$

where $\omega_1 \geq 0$ and $\omega_2 \geq 0$.
Therefore, $\rho(x,z) > \rho(x,y) + \rho(y,z)$ when $\Psi = 1$ which means $\rho(x,z)$ does not obey triangle inequality for some particular cases. Thus, the product of a semi-metric and a metric function remains a semi-metric function.

Theorem 1. *$B_{DGC}(f_x, f_y)$ is a semi-metric.*

Proof. B_{DGC} is a product of four components as shown in Eq. 6. Among them, $K_{(x,y)}$ and $\delta(x,y)_{DGC}$ depends on probabilities. From Eq. 7, it is evident that $K_{(x,y)}$ lie between $[0,1]$ whereas, from Eq. 8, $\delta(x,y)_{DGC}$ lie between $[0,2]$. Both these functions satisfy points (1) and (2) of Sec. 2.2, i.e., $K_{x,y} \Leftrightarrow x = y$, $K_{x,y} = K_{y,x} \geq 0$ and similarly for $\delta(x,y)_{DGC}$. But they do not satisfy the triangle inequality (point 3). If we consider three voxels x, y and z, then $K_{x,y}$, $K_{y,z}$ and $K_{x,z}$ can take any value between $[0,1]$ and hence, there will be cases where $K_{(x,z)} > K_{(x,y)} + K_{(y,z)}$ for some x, y and z. A similar situation can also occur in the case of $\delta(x,y)_{DGC}$. Therefore, these functions are semi metric. For example, if we consider $K_{(x,y)} = 0.2$, $K_{(y,z)} = 0.3$ and $K_{(x,z)} = 0.7$, then $K_{(x,z)} > K_{(x,y)} + K_{(y,z)}$. Now, we consider the term $e^{-(\frac{(I_x - I_y)^2}{2\sigma^2})}$, which is based on image intensities I_x and I_y. The intensity value lies between $[0, 255]$. We can similarly argue that $e^{-(\frac{(I_x - I_z)^2}{2\sigma^2})} > e^{-(\frac{(I_x - I_y)^2}{2\sigma^2})} + e^{-(\frac{(I_y - I_z)^2}{2\sigma^2})}$ for some I_x, I_y and I_z.
Thus, $K_{(x,y)}$, $e^{-(\frac{(I_x - I_y)^2}{2\sigma^2})}$ and $\delta(x,y)_{DGC}$ are semi metric in nature, $\frac{1}{d(x,y)}$ is metric as $d(x,y)$ is the Euclidean distance. Hence, from Lemma 1 it follows that $B_{DGC}(x,y)$ is a semi metric.

References

1. Ali, A.M., Farag, A.A., El-Baz, A.S.: Graph cuts framework for kidney segmentation with prior shape constraints. In: MICCAI. pp. 384–392. Springer (2007)
2. Baker, S., et al.: The human dentate gyrus plays a necessary role in discriminating new memories. Current Biology **26**, 2629–2634 (2016)
3. Band, S.S., Yarahmadi, A., Hsu, C.C., Biyari, M., Sookhak, M., Ameri, R., Dehzangi, I., Chronopoulos, A.T., Liang, H.W.: Application of explainable artificial intelligence in medical health: A systematic review of interpretability methods. Informatics in Medicine Unlocked **40**, 101286 (2023)
4. Blake, A., Kohli, P., Rother, C.: Markov random fields for vision and image processing. MIT Press (2011)
5. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE TPAMI **23**(11), 1222–1239 (11 2001)
6. Boykov, Y., Veksler, O., Zabih, R.: Optimizing Multilabel MRFs Using Move-Making Algorithms. Markov Random Fields for Vision and Image Processing pp. 51–64 (12 2011)
7. Boykov, Y.Y., Jolly, M.P.: Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In: ICCV. vol. 1, pp. 105–112. IEEE (2001)
8. Chadwick, M.J., Bonnici, H.M., Maguire, E.A.: Ca3 size predicts the precision of memory recall. Proceedings of the National Academy of Sciences **111**(29), 10720–10725 (2014)
9. Çiçek, Ö., et al.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: MICCAI. pp. 424–432. Springer (2016)
10. De, A., Mhatre, R., Tiwari, M., Chowdhury, A.S.: Brain tumor classification from radiology and histopathology using deep features and graph convolutional network. In: 2022 26th International Conference on Pattern Recognition (ICPR). pp. 4420–4426 (2022)
11. De, A., Tiwari, M., Chowdhury, A.S.: 3d hippocampus segmentation using a hog based loss function with majority pooling. In: IEEE ICIP. pp. 2260–2264. IEEE (2023)
12. De, A., Tiwari, M., Grisan, E., Chowdhury, A.S.: A deep graph cut model for 3d brain tumor segmentation. In: EMBC. pp. 2105–2109. IEEE (2022)
13. Dice, L.R.: Measures of the amount of ecologic association between species. Ecology **26**(3), 297–302 (1945)
14. Edmonds, J., Karp, R.M.: Theoretical improvements in algorithmic efficiency for network flow problems. Journal of the ACM (JACM) **19**(2), 248–264 (1972)
15. Ford, L., Fulkerson, D.: Flows in Networks. Princeton Landmarks in Mathematics and Physics, Princeton University Press (2015), https://books.google.co.in/books?id=fw7WCgAAQBAJ
16. Freedman, D., Zhang, T.: Interactive graph cut based segmentation with shape priors. In: IEEE CVPR. vol. 1, pp. 755–762. IEEE (2005)
17. Hobbs, K.H., Zhang, P., Shi, B., Smith, C.D., Liu, J.: Quad-mesh based radial distance biomarkers for Alzheimer's disease. Proceedings - International Symposium on Biomedical Imaging **2016-June**, 19–23 (6 2016)
18. Kulaga-Yoskovitz, J., et al.: Multi-contrast submillimetric 3 Tesla hippocampal subfield segmentation protocol and dataset. Scientific Data 2015 2:1 **2**(1), 1–9 (11 2015)
19. Li, G., et al.: Automatic liver segmentation based on shape constraints and deformable graph cut in ct images. IEEE Transactions on Image Processing **24**(12), 5315–5329 (12 2015)

20. Li, X., et al.: Syn_SegNet: A Joint Deep Neural Network for Ultrahigh-Field 7T MRI Synthesis and Hippocampal Subfield Segmentation in Routine 3T MRI. IEEE Journal of Biomedical and Health Informatics **27**(10), 4866–4877 (10 2023)
21. Malcolm, J., Rathi, Y., Tannenbaum, A.: Graph cut segmentation with nonlinear shape priors. In: IEEE ICIP. vol. 4, pp. IV–365. IEEE (2007)
22. Manjón, J.V., Romero, J.E., Coupe, P.: A novel deep learning based hippocampus subfield segmentation method. Scientific Reports **12**(1) (12 2022)
23. Milner, B.: Psychological defects produced by temporal lobe excision. Res. Publ. Assoc. Res. Nerv. Ment. Dis. **36**, 244–257 (1958)
24. Mukherjee, S., Huang, X., Bhagalia, R.R.: Lung nodule segmentation using deep learned prior based graph cut. In: IEEE ISBI. pp. 1205–1208. IEEE (2017)
25. Paszke, A., et al.: PyTorch: an imperative style, high-performance deep learning library. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems. Curran Associates Inc., Red Hook, NY, USA (2019)
26. Romero, J.E., Coupé, P., Manjón, J.V.: HIPS: A new hippocampus subfield segmentation method. Neuroimage **163**, 286–295 (12 2017)
27. Roy, R., Chakraborti, T., Chowdhury, A.S.: A deep learning-shape driven level set synergism for pulmonary nodule segmentation. Pattern Recognition Letters **123**, 31–38 (5 2019)
28. Schlichting, M.L., Zeithamova, D., Preston, A.R.: CA1 subfield contributions to memory integration and inference. Hippocampus **24**(10), 1248–1260 (10 2014)
29. Shi, Y., Cheng, K., Liu, Z.: Hippocampal subfields segmentation in brain MR images using generative adversarial networks. BioMedical Engineering Online **18**(1) (1 2019)
30. Slabaugh, G., Unal, G.: Graph cuts segmentation using an elliptical shape prior. In: IEEE ICIP. vol. 2, pp. II–1222. IEEE (2005)
31. Suganyadevi, S., Seethalakshmi, V., Balasamy, K.: A review on deep learning in medical image analysis. International Journal of Multimedia Information Retrieval 2021 11:1 **11**(1), 19–38 (9 2021)
32. Voets, N.L., Bernhardt, B.C., Kim, H., Yoon, U., Bernasconi, N.: Increased temporolimbic cortical folding complexity in temporal lobe epilepsy. Neurology. **76**(2), 138–144 (1 2010)
33. Wang, H., Zhang, H., Ray, N.: Adaptive shape prior in graph cut image segmentation. Pattern Recognition **46**(5), 1409–1414 (5 2013)
34. Xu, Y., Gao, F., Wu, T., Bennett, K.M., Charlton, J.R., Sarkar, S.: U-net with optimal thresholding for small blob detection in medical images. In: 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE). pp. 1761–1767 (2019)
35. Yang, Z., Zhuang, X., Mishra, V., Sreenivasan, K., Cordes, D.: CAST: A multi-scale convolutional neural network based automated hippocampal subfield segmentation toolbox. NeuroImage **218**, 116947 (9 2020)
36. Yushkevich, P.A., et al.: Automated volumetry and regional thickness analysis of hippocampal subfields and medial temporal cortical structures in mild cognitive impairment. Human Brain Mapping **36**(1), 258–287 (1 2015)
37. Zhu, H., Shi, F., Wang, L., Hung, S.C., Chen, M.H., Wang, S., Lin, W., Shen, D.: Dilated dense U-net for infant hippocampus subfield segmentation. Frontiers in Neuroinformatics **13** (4 2019)