Model-Informed Deep Learning for Surface Segmentation in Medical Imaging

Xiaodong Wu[1,2(✉)], Leixin Zhou[1], Fahim Zaman[1], Bensheng Qiu[3], and John M. Buatti[2]

[1] Department of Electrical and Computer Engineering, The University of Iowa, Iowa City, IA 52242, USA
{leixin-zhou,fahim-zaman}@uiowa.edu

[2] Department of Radiation Oncology, University of Iowa, Iowa City, IA 52242, USA
{xiaodong-wu,john-buatti}@uiowa.edu

[3] School of Information Science and Technology, The University of Science and Technology of China, Hefei 230027, China
bqiu@ustc.edu.cn

Abstract. Automated surface segmentation is an important tool for utilizing medical image data in modern precision medicine for routine clinical practice and research. Deep-learning based methods have been developed for various medical image segmentation tasks. The inherent classification nature of those methods yet limits their capability of modeling global spatial dependency, which poses great challenges in incorporating geometric priors for segmentation, such as surface shape and surface smoothness, significantly compromising the accuracy and robustness of segmentation performance. To solve this problem, we propose integrating the graph-based optimal surface segmentation model into a new form of Convolutional Neural Networks (CNNs) that unifies the strengths of both deep learning and the graph segmentation model. To this end, we propose to parameterize the graph-based surface segmentation model and formulate the optimal surface segmentation as a quadratic programming problem, which admits an efficient inference for globally optimal solutions. The resulting network fully unifies graph segmentation modeling with CNNs, making it possible to train the whole deep network end-to-end with the usual back-propagation algorithm. Our experiments on two medical image segmentation applications demonstrated high performance of the proposed method with respect to segmentation accuracy, demands for annotated training data, and robustness to adversarial noise.

1 Introduction

Highly-automated and consistently accurate quantitative analysis of volumetric medical image data is a pre-requisite to utilize medical image data in modern precision medicine. Surface segmentation, which aims to accurately define the boundary surfaces of tissues captured by image data, is becoming increasingly necessary in quantitative image analysis. Many surface segmentation methods have been developed, including parametric deformable models, geometric deformable models, and atlas-guided approaches.

A. Frangi et al. (Eds.): IPMI 2023, LNCS 13939, pp. 822–834, 2023.
https://doi.org/10.1007/978-3-031-34048-2_63

As one of the prominent surface segmentation approaches, the graph-based optimal surface segmentation method (Graph-OSSeg) [1] has demonstrated efficacy in the medical imaging field [2]. It is capable of simultaneously detecting multiple interacting surfaces with global optimality with respect to the energy function designed for the target surfaces with geometric constraints, which define the surface smoothness and interrelations. It also enables sub-pixel accurate surface segmentation [3]. The method solves the surface segmentation problem by transforming it to compute a minimum s-t cut in a derived arc-weighted directed graph, which can be solved optimally with a low-order polynomial time complexity. The major limitation of Graph-OSSeg is associated with the need for handcrafted features to define the parameters of the underlying graph model.

Armed with superior data representation learning capability, deep learning (DL) methods are emerging as powerful alternatives to current segmentation algorithms for many medical image segmentation tasks [4]. The state-of-the-art DL segmentation methods in medical imaging include fully convolutional networks (FCNs) [5] and U-net based frameworks [6,7], which model the segmentation problem as a pixel-wise or voxel-wise classification problem. Those convolutional neural network (CNN) methods have some critical limitations that restrict their use in the medical setting: (i) *Training data demand:* current schemes often need extensive training data, which is an almost insurmountable obstacle due to the risk to patients and high cost. (ii) *Difficulty in exploiting prior information* (shape, boundary smoothness and interaction): the methods are classification-based in nature, and the output probability maps are relatively unstructured. (iii) *Vulnerability to adversarial perturbations:* recent research has demonstrated that, compared to the segmentation CNNs alone, the integration of a graphical model such as conditional random fields (CRFs) into CNNs enhances the robustness of the method to adversarial perturbations [8].

To address those limitations, many model-based attempts have been proposed. One natural way is to use CNNs to learn the probability maps and then apply the traditional model-based methods such as graph cuts and deformable models to incorporate the prior information for segmentation [9,10]. In this scheme, feature learning by CNNs is, in fact, disconnected from the segmentation model; the learned features thus may not be truly appropriate for the model. Recent works introduce the energy function of a segmentation model into the loss function to guide CNNs for more model-specific feature learning, and improved segmentation performance has been demonstrated [11,12]. The model is not yet explicitly enforced while inferring the segmentation solutions with the trained network. In Zheng *et al.*'s work [13], the CRFs model is implemented as a recurrence neural network (RNN) and is integrated with an FCN for feature learning in a single neural network to achieve end-to-end learning. Arnab *et al.* [14] and Vemulapalli *et al.* [15] have demonstrated that the CRF-RNN framework outperforms other DL methods for semantic segmentation in computer vision. However, the CRFs inference is computationally intractable, thus no optimal solutions can be guaranteed – the solutions can be far from the optimal one at any scale, which may confuse the network during training and

may contribute to its known high training complexity. In fact, the CRF-RNN method has not been widely used in medical image segmentation.

In this study, we propose unifying the powerful feature learning capability of DL with the successful graph-based optimal surface segmentation (Graph-OSSeg) model in a single deep neural network for end-to-end learning to achieve globally optimal segmentation. In this model-informed deep-learning segmentation method for optimal surface segmentation (MiDL-OSSeg), the known model is integrated into the DL network, which provides an advanced "attention" mechanism to the network. The network does not need to learn the prior information encoded in the model, reducing the demand of labeled data, which is critically important for medical imaging where scarcity of labeled data is common. Our major contributions are, as follows. (i) We model the graph-based optimal surface segmentation as a quadratic programming, blending learning and inference in a deep structured model while achieving global optimality of the segmentation solutions. (ii) The parameters of the graph-based optimal surface segmentation model are parameterized and learned by leveraging deep learning with a U-net as the backbone. (iii) Our experiments have demonstrated the high performance of our proposed method with high segmentation accuracy, less labeled data demand, and high robustness to adversarial perturbations.

2 Method

In this section, we present our MiDL-OSSeg method, merging the strength of both DL and Graph-OSSeg. We first formally define the optimal surface segmentation problem, which is formulated as a quadratic programming problem by parameterizing the Graph-OSSeg model. The proposed MiDL-OSSeg network is then depicted in detail, followed by its training strategy.

2.1 Quadratic Programming Formulation of Surface Segmentation

To present our method in a comprehensible manner, we consider a task of single *terrain-like* surface segmentation while incorporating the shape priors. Note that this simple principle used for this illustration is directly applicable to more complex surface segmentation (see Sect. 3.1 for prostate segmentation).

Let $\mathcal{I}(X, Y, Z)$ of size $X \times Y \times Z$ be a given 3-D volumetric image. For each (x, y) pair (i.e., $(x, y) \in X \times Y$), the voxel subset $\{\mathcal{I}(x, y, z)|0 \leq z < Z\}$ forms a column parallel to the **z**-axis, denoted by $p(x, y)$. Each column has a set of neighboring columns for a certain neighboring setting $\mathcal{N}$, e.g., the four-neighbor relationship. Our goal is to seek a terrain-like surface $\mathcal{S}$, which intersects each column $p(x, y)$ at exactly one voxel. Thus, the terrain-like surface $\mathcal{S}$ can be defined as a function $\mathcal{S}(x, y)$, mapping $p(x, y)$ pairs to their z-values z_p.

In the Graph-OSSeg model [1], each voxel $\mathcal{I}(x, y, z)$ is associated with an on-surface cost $c(x, y, z)$ for the sought surface $\mathcal{S}$, which is inversely related to the likelihood that the desired surface $\mathcal{S}$ contains the voxel, and is computed based on handcrafted image features. The on-surface cost function $c(x, y, z)$ for

each column $p(x, y)$ (i.e., $z = 0, 1, \ldots, Z-1$) can be an arbitrary function in the Graph-OSSeg model (Fig. 1a). However, an ideal cost function $c(x, y, z)$ should express a certain type of convexity: as we aim to formulate surface segmentation as a minimization problem, $c(x, y, z)$ should be low at the surface location for the column $p(x, y)$; while the distance increases from the surface location along the column, the cost should increase proportionally. We propose to make use of a Gaussian distribution $\mathcal{G}(\mu_p, \sigma_p)$ to model the likelihood of the column voxels on the target surface $\mathcal{S}$, and to define the on-surface cost function $c(x, y, z)$ for each column $p(x, y)$ as $c(x, y, z) = \frac{(z-\mu_p)^2}{2\sigma_p^2}$ $(0 \le z \le Z-1)$ (Fig. 1b). Thus, the on-surface cost functions for all columns are parameterized with $(\boldsymbol{\mu}, \boldsymbol{\sigma})$. In the Graph-OSSeg model, it is at least nontrivial to determine $(\boldsymbol{\mu}, \boldsymbol{\sigma})$ based on the handcrafted features. In this work, we propose to leverage DL for the on-surface cost parameterization with Gaussians.

It is critically important to incorporate shape priors in the segmentation model. In the Graph-OSSeg model [2], the shape changes of surface $\mathcal{S}$ are defined as the surface position changes between pairs of neighboring columns. Specifically, for any pair of neighboring columns p and q, the shape change of $\mathcal{S}$ between the column pair (p, q) is $d_{p,q} = (z_p - z_q)$ (note that surface $\mathcal{S}$ cuts the columns p and q at z_p and z_q, respectively). Then, $\boldsymbol{d} = (d_{p,q})_{(p,q)\in\mathcal{N}}$ forms a parameterization of the shape prior, which will be dynamically learned with DL for the input image during the inference. We use a quadratic function $((z_p - z_q) - d_{p,q})^2$ to penalize the deviation of the shape change to the prior model $\boldsymbol{d}$.

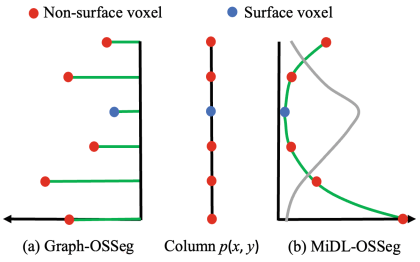

Fig. 1. On-surface cost parameterization with Gaussians. (a) The on-surface cost function in the Graph-OSSeg model defined on voxels for each column based on handcrafted features. The green line segments indicate the magnitudes for the corresponding voxels. (b) The on-surface cost function in the MiDL-OSSeg model (green curve) is computed based on the Gaussian-parameterized likelihood function (grey curve) over the column voxels. (Color figure online)

Thus, the MiDL-OSSeg problem is to find a terrain-like surface $\mathcal{S}$, such that $\mathcal{S}$ intersects each columns $p(x, y)$ at exactly one location z_p $(0 \le z_p \le Z-1)$ while minimizing the energy function $\mathbb{E}(\boldsymbol{z})$, with

$$\mathbb{E}(\boldsymbol{z}) = \sum_{p\in\mathcal{C}} \frac{(z_p - \mu_p)^2}{2\sigma_p^2} + w \sum_{(p,q)\in\mathcal{N}} ((z_p - z_q) - d_{p,q})^2, \tag{1}$$

where $\mathcal{C}$ is the set of all columns, $\mathcal{N}$ is the set of neighboring column pairs, and w is the coefficient. In the problem formulation (1), the surface location vector $\boldsymbol{z}$ is relaxed as continuous variables, that is, $0 \le z_p \le Z-1$ for each $p \in \mathcal{C}$. Hence, instead of keeping the target surface passing the center of a voxel, we allow the target surface $\mathcal{S}$ intersecting each column at any location, which may alleviates the partial volume effect.

2.2 The MiDL-OSSeg Model

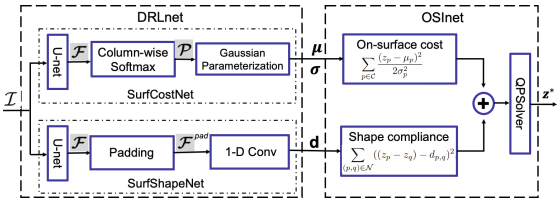

Fig. 2. Inference pipeline of the proposed method.

The proposed MiDL-OSSeg model consists of two integrative components – a data representation learning network (DRLnet) and an optimal surface inference network (OSInet) (Fig. 2). The DRLnet is a DL network aiming to learn data representations in the form of those in the MiDL-OSSeg model, that is, the on-surface cost parameterization $(\boldsymbol{\mu}, \boldsymbol{\sigma})$ and the shape prior parameterization $\boldsymbol{d}$. The OSInet strikes to solve the optimal surface interference by optimizing the energy function $\mathbb{E}(\boldsymbol{z})$. The whole network can then be trained in an end-to-end fashion and output globally optimal solutions for surface segmentation.

The surface cost net (SurfCostNet) for learning the on-surface cost parameterization $(\boldsymbol{\mu}, \boldsymbol{\sigma})$ is illustrated in the upper left panel of Fig. 2. A common U-net architecture is utilized to generate the discrete probability map $\boldsymbol{\mathcal{P}}$ for the input image $\mathcal{I}$. In the proposed method, the softmax layer, taking the feature maps $\boldsymbol{\mathcal{F}}$ from the U-net, works on each *column*, instead of for each voxel. As the target surface intersects with each column exactly once, the probabilities are normalized within each column $q(x, y)$ to obtain the probability vector $\mathcal{P}_q$. Each element $\mathcal{P}_q[z]$ indicates the probability of voxel $\mathcal{I}(x, y, z)$ being on the target surface $\mathcal{S}$, and the total sum of the probabilities of all voxels on the column q equals to 1. Then, $\boldsymbol{\mathcal{P}} = \{\mathcal{P}_q | q \in \mathcal{C}\}$ forms the probability map of the input image. As we intend to parameterize the on-surface costs, the probability vector $\mathcal{P}_q$ for each column q is expected to be in a Gaussian distribution. To regularize the probability map $\boldsymbol{\mathcal{P}}$ output from the U-net with a Gaussian, which mimics the Bayesian learning for each column and shares merits with knowledge distillation and distillation defense.

The *Gaussian parameterization* block is then applied to compute a Gaussian $\mathcal{G}(\mu_q, \sigma_q)$ to best fit to the discrete probability vector $\mathcal{P}_q$ for each column $q(x, y) \in \mathcal{C}$. $\mathcal{P}_q$ can be viewed as a discrete sample of the continuous Gaussian probability density function $\mathcal{G}(\mu_q, \sigma_q)$. We can estimate μ_q and σ_q from the probability vector $\mathcal{P}_q$ by minimizing a weighted mean square error, which admits an analytic solution for backpropogation [16].

The surface shape net (SurfShapeNet) for learning the parameterized shape model $\boldsymbol{d}$ is illustrated in the lower left panel of Fig. 2. It consists of a common U-net for the extraction of representative features $\mathcal{F}$, a padding layer to enable sufficient context information, and one 1-D convolution layer to generate the shape model $\boldsymbol{d}$.

To compute the surface position change $d_{p,q}$ between two adjacent columns p and q in the shape model $\boldsymbol{d}$, we consider a 4-neighborhood setting for the purpose of comprehensible illustration, in which each column $p(x, y)$ has four adjacent columns: $p(x-1, y)$ and $p(x+1, y)$ in the $\boldsymbol{x}$-dimension, and $p(x, y-1)$ and

$p(x, y+1)$ along the $\boldsymbol{y}$-dimension. This simple illustrative principle is directly applicable to an arbitrary neighborhood setting.

Consider two adjacent columns $p(x, y)$ and $p(x+1, y)$ along the $\boldsymbol{x}$-dimension, denoted by p and $p+1$, respectively. In general, we use $p+j$ to denote the column $q(x+j, y)$. For a robust inference of the surface position change $d_{p,p+1}$ between columns p and $p+1$, we consider $N_c > 0$ consecutive neighboring columns of p and $p+1$. The set $\mathcal{F}_p^{pad}$ of feature maps output from U-net for those columns with possible padding are used to infer $d_{p,p+1}$. Then, a 1-D convolution layer with a kernel size 1 and a stride of 1 is applied to the padded feature map $\mathcal{F}_p^{pad}$ to generate the surface position change $d_{p,p+1}$ between any two adjacent columns $p(x, y)$ and $p(x+1, y)$ along the $\boldsymbol{x}$-dimension.

Similarly, the surface position change between any two adjacent columns $p(x, y)$ and $p(x, y+1)$ in the $\boldsymbol{y}$-dimension can be computed. Thus, the parameterized shape model $\boldsymbol{d}$ can be dynamically generated for the input image $\mathcal{I}$.

The optimal surface inference network (OSInet) aims to solve the optimization problem in Eq. (1) with a globally optimal solution. To minimize the energy function $\mathbb{E}(\boldsymbol{z})$, we convert it to a standard quadratic form. For the purpose of comprehensible illustration, we consider a 4-neighborhood setting $\mathcal{N}$ for the adjacency of columns. Then, the grid $X \times Y$ defines the domain of all the columns, that is, every pair (x, y) corresponds an image column. The sought surface positions on $X \times Y$ thus form a matrix $\boldsymbol{z} \in \mathbb{R}^{X \times Y}$. To convert $\mathbb{E}(\boldsymbol{z})$ to a quadratic form, we flatten the matrix $\boldsymbol{z}$ to a vector $\boldsymbol{z'} \in \mathbb{R}^{XY}$, as follows. Each element $\boldsymbol{z}(x, y)$ ($x = 0, 1, \ldots, X-1$ and $y = 0, 1, \ldots, Y-1$) corresponds to $\boldsymbol{z'}(x * Y + y)$. It is equivalent to do a column-major order traversal of $\boldsymbol{z}$. We explicitly maintain the adjacency relationship of each $\boldsymbol{z}(x, y)$ in the flattened vector $\boldsymbol{z'}$ with $\mathcal{N}'$ for the corresponding elements. That is, for any two adjacent columns $p(x, y)$ and $q(x, y)$ with $(p, q) \in \mathcal{N}$, let $k = x * Y + y$ and $\bar{k} = x' * Y + y'$, then $(k, \bar{k}) \in \mathcal{N}'$. The matrix $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are flattened into the vectors $\boldsymbol{\mu'}$ and $\boldsymbol{\sigma'}$, respectively, in the same way as $\boldsymbol{z}$. We then have the following form of the objective function $\mathbb{E}(\boldsymbol{z})$.

$$\mathbb{E}(\boldsymbol{z}) = \mathbb{E}(\boldsymbol{z'}) = \frac{1}{2}\boldsymbol{z'}^T \boldsymbol{H} \boldsymbol{z'} + \boldsymbol{c}^T \boldsymbol{z'} + \text{CONST.}, \tag{2}$$

where $\boldsymbol{H}$ is a Hessian matrix of $\mathbb{E}(\boldsymbol{z'})$. It can be proved that $\boldsymbol{H}$ is positive definite by using the Gershgorin circle theorem [17]. The energy function $\mathbb{E}(\boldsymbol{z'})$ is thus convex. Let the gradient $\nabla = \boldsymbol{H}\boldsymbol{z'} + \boldsymbol{c}$ to be zero, we have the global optimal solution $\boldsymbol{z'}^* = -\boldsymbol{H}^{-1}\boldsymbol{c}$. We thus do not need to make use of a recurrent neural network (RNN) to implement OSInet for a globally optimal solution.

2.3 Training Strategy

Pre-training the Surface Cost Net. To pre-train SurfaceCostNet to obtain the probability map $\mathcal{P}$ for on-surface cost parameterization (Fig. 2), we make use of the ground truth $\mathcal{S}_{gt}$ of the surface segmentation. For each column q, if the voxel is on $\mathcal{S}_{gt}$, then the probability of the voxel is 1; otherwise, it is 0. Thus, the probabilities of all the voxels on q form a delta function, which is Gaussianized by setting the standard deviation σ to be 0.1 times of the column length to

obtain a Gaussian distribution $\hat{\mathcal{P}}_q$ as the ground truth for each column q. Let $\mathcal{P}_q$ be the output probability vector from the column-wise softmax layer for column q. The loss for the pre-training, $Loss_{pre}$, is formulated as the Kullback-Leibler divergence of $\hat{\mathcal{P}}_q$ and $\mathcal{P}_q$.

Training the Surface Shape Net. The reference surface $\mathcal{S}_{gt}$ is first used to generate the ground truth $\hat{\boldsymbol{d}}$ to train the surface shape net (SurfShapeNet). For each pair of adjacent columns $(p, q) \in \mathcal{N}$, compute the surface position change $\hat{d}_{p,q}$ between columns p and q from $\mathcal{S}_{gt}$. Let $\boldsymbol{d}$ be the output of SurfShapeNet. The mean square error of the surface position changes between $\boldsymbol{d}$ and $\hat{\boldsymbol{d}}$, $Loss_{shape}$, is then utilized for the loss function. Note that the surface position changes could be highly erratic, especially when the ground truth surface positions are defined in the discrete voxel space. This hinders SurfShapeNet from learning useful representation and usually the trained SurfShapeNet just generates a constant prediction that is not much useful. We propose smoothing the ground truth $\hat{\boldsymbol{d}}$ by using the sliding window average method for the training of SurfShapeNet. The predicted shape model $\boldsymbol{d}$ by the network trained with the smoothed ground truth $\hat{\boldsymbol{d}}$ is much more accurate.

Fine Tuning. The L_1-loss on surface positioning errors is used for the fine tuning of the whole network. The fine tuning proceeds alternatively between the training of SurfaceCostNet and OSInet. The training data is used for the SurfaceCostNet training, while the validation data is utilized to train OSInet. As OSInet only has one parameter (w) that needs to be trained, the chance of overfitting is low. Note that SurfaceCostNet is not trained on the validation data, the learned parameter w should be more representative in the wild. Otherwise, if we use the training data for the fine tuning of both SurfaceCostNet and OSInet, the learned w tends to be small due to the pre-training process of SurfaceCostNet, which may marginalize the shape term in the energy function $\mathbb{E}(\boldsymbol{z})$. As the shape priors are relatively stable, we freeze the pre-trained SurfaceShapeNet to obtain the shape model during the network fine tuning.

3 Performance Assessment

The performance of the proposed MiDL-OSSeg method was evaluated to determine: segmentation accuracy, annotated data demands for model training, and robustness to adversarial perturbations. The experiments were carried out on medical images from spectral domain optical coherence tomography (SD-OCT) and magnetic resonance imaging (MRI). Assessments of terrain-like and closed surface segmentation were performed.

3.1 Application Experiments

Automated Retinal Layer Segmentation in SD-OCT Images. To demonstrate the utility of our MiDL-OSSeg method in segmenting terrain-like surfaces, automated retinal layer segmentation in SD-OCT images was performed.

Data. 382 SD-OCT scans (114 normal eyes and 268 eyes with intermediate age-related macular degeneration (AMD)) and their respective manual tracings by an expert were obtained from the publicly available repository of datasets [18]. Each OCT volume consists of $400 \times 60 \times 512$ voxels with a size of$6.54 \times 67 \times 3.23$ μm^3. The dataset was randomly divided into 3 sets: 1) training set - 266 volumes (79 normal, 187 AMD), 2) validation set - 57 volumes (17 normal, 40 AMD), and 3) testing set - 59 volumes (18 normal, 41 AMD). The surfaces considered are the Inner aspect of Retinal Pigment Epithelium drusen complex (IRPE) and the Outer aspect of Bruch Membrane (OBM) (Fig. 3). The proposed MiDL-OSSeg model was trained and tested on the 2D B-scans of the OCT volumes.

Prostate Segmentation in MR Images. The proposed MiDL-OSSeg method was evaluated on automated prostate segmentation in 3D MR images to demonstrated its applicability of segmenting irregular surfaces in 3D.

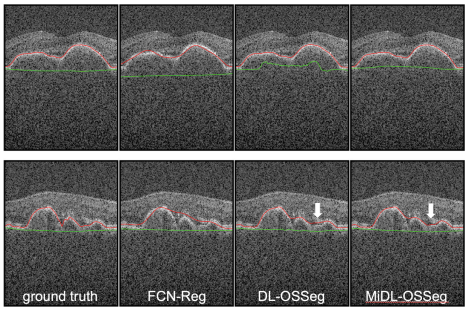

Fig. 3. Illustrations of SD-OCT segmentation results. Red: IRPE; Green: OBM. (Color figure online)

Data. The dataset is provided by the NCI-ISBI 2013 Challenge - Automated Segmentation of Prostate Structures [19]. This dataset has two labels: peripheral zone (PZ) and central gland (CG). We treat both of them as prostate for single surface segmentation. The ground truth surface of the prostate boundary in each image was generated from the PZ and CG labels. The challenge data set consists of the training set (60 cases), the leader board set (10 cases) and the test set (10 cases). 70 cases in total were used as the test set was not available. Ten-fold cross validation was applied on that dataset. For each fold, the training, validation and test sets consist of 58, 5 and 7 cases, respectively. The shape-aware patch generation method [20] was adopted to divide each MRI scan into 6 volumetric patches. Each patch contains a portion of prostate boundary, which is a terrain-like surface in 3D. Our MiDL-OSSeg model was trained and validated on the volumetric patches.

3.2 Segmentation Accuracy

OCT Retinal Layer Segmentation. Unsigned mean surface positioning error (UMSP) was utilized for accuracy assessment of retina OCT segmentation. We compared the proposed MiDL-OSSeg method to the Graph-OSSeg method [2] as well as Shah *et al.*'s FCN-based regression model (denoted by FCN-Reg) [21]. To ensure a fair comparison, we reimplemented Shah *et al.*'s method to make sure that the training, validation and test data splitting was the same for the two compared methods. For the purpose of an ablation study, we showed the segmentation results of our method without incorporating the shape priors,

that is, the means of Gaussians μ output from the Gaussian Parameterization block in Fig. 2 are treated as the predicted surface positions. The method is marked as DL-OSSeg in Table 1. Our MiDL-OSSeg method significantly outperformed all other methods for each surface with the p-value less than 0.05. Specifically, MiDL-OSSeg incorporating the shape priors which was implemented with OSInet yielded significant improvement compared to DL-OSSeg. Sample segmentation results are illustrated in Fig. 3.

Table 1. UMSP errors and standard deviations in μm evaluated on the SD-OCT dataset. Depth resolution is 3.23 µm. Numbers in bold are the best in that row.

Surfaces	Normal				AMD			
	Graph-OSSeg	FCN-Reg	DL-OSSeg	MiDL-OSSeg	Graph-OSSeg	FCN-Reg	DL-OSSeg	MiDL-OSSeg
IRPE	4.55±0.36	3.70±0.69	2.16±0.67	**1.89±0.68**	9.30±1.74	6.45±2.11	3.09±1.52	**2.96±1.91**
OBM	5.59±1.20	3.58±0.38	3.28±0.71	**2.55±0.40**	10.14±5.30	6.43±2.82	5.74±2.51	**4.29±1.71**

Prostate MRI Segmentation. The proposed MiDL-OSSeg method for prostate segmentation was compared to the Graph-OSSeg method [2] and other two CNN-based approaches, U-net [6] and PSNet [22]. PSNet is the state-of-the-art method on the dataset. The Dice similarity coefficient (DSC), Hausdroff distance, and the average surface distance (ASD) between predicted prostate boundary surface and manual delineation for each method are shown in Table 2. With respect to all three metrics, the proposed MiDL-OSSeg significantly outperformed all the compared methods, especially for the surface-based ASD and HD metrics. Figure 4 shows an example segmentation results by MiDL-OSSeg for a 3D prostate MR image in the transverse, sagittal and coronal views.

Table 2. The DSC, ASD and HD with standard deviations evaluated on the prostate dataset. Numbers in bold are the best in that column among all the methods.

Methods	DSC	ASD (mm)	HD (mm)
U-net	0.84±0.05	3.3±1.0	10.1±3.2
PSNet	0.85±0.04	3.0±0.9	9.3±3.5
Graph-OSSeg	0.80±0.04	2.7±0.6	13.9±1.8
MiDL-OSSeg	**0.89±0.03**	**1.36±0.34**	**7.28±3.20**

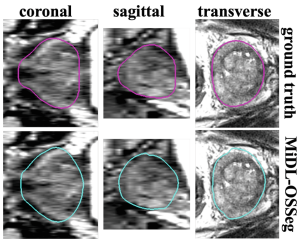

Fig. 4. Example segmentation of a prostate MR image.

3.3 Annotated Data Demands for Training

We evaluated the segmentation performance changes of the proposed method with respect to the different training data sizes. The validation and test datasets

were fixed and the training dataset for model training was randomly sampled with different rates. Each trained segmentation model was applied to the same test dataset for performance evaluation. For the OCT retina layer segmentation, 50%, 30%, and 10% of the training set were randomly generated for model training. The proposed MiDL-OSSeg method was compared to Shah *et al.*'s FCN-Reg model [21] while trained on different sampled training sets. The UMSP errors evaluated on the SD-OCT dataset are shown in Fig. 5 for the two compared methods. The proposed MiDL-OSSeg model trained on each of the reduced training sets (50%, 30%, and 10%) significantly outperformed the FCN-Reg model trained on the same training set for each target surface of normal and AMD subjects. Of note: while the MiDL-OSSeg model was trained on 10% of the whole training set it achieved an even better accuracy, compared to the FCN-Reg trained on the entire training data set.

3.4 Robustness to Adversarial Perturbations

Robustness of the proposed MiDL-OSSeg model was evaluated against adversarial samples [23], which are legitimate samples with human-imperceptible perturbations that attempt to fool a trained model to make incorrect predictions with high confidence. To push the model to its limit for performance degeneration, we adopted the white-box attack methods [24], in which the full knowledge of the network architecture and the model parameters is used to generate adversarial noises. In our experiments, the fast gradient sign method (FGSM) [24] was utilized.

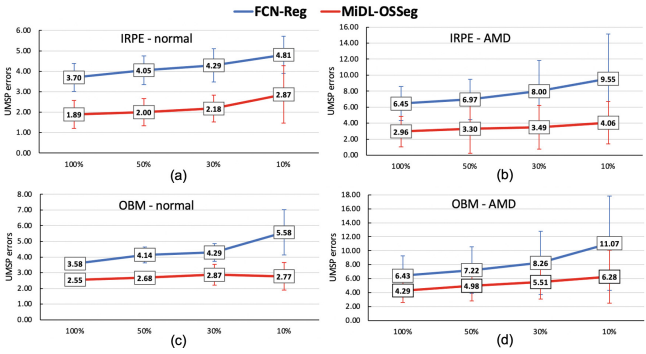

Fig. 5. Segmentation accuracy evaluated on the SD-OCT dataset for the proposed MiDL-OSSeg model, compared to FCN-Reg [21], while trained on 100%, 50%, 30%, and 10% of the training set.

Our robustness experiments were conducted on the retinal OCT dataset for retinal layer segmentation. For each attack level $\epsilon = 0.02, 0.04, 0.06, 0.08, 0.10$, an adversarial sample $\mathcal{I}_{adv}$ was generated for each OCT image $\mathcal{I}$ in the test set, all of which form an adversarial sample set for the corresponding attack level ϵ. The MiDL-OSSeg model trained with the original training and validation sets (without using adversarial samples) was then tested on the adversarial sample set of each ϵ for segmentation accuracy. For comparison, Shah *et al.*'s FCN-Reg method [21] was also evaluated for its segmentation performance on the adversarial sample sets. The segmentation accuracy measured with UMSP errors and standard deviations for both IRPE and OBM surfaces of normal and

AMD subjects are summarized in Fig. 6 for each adversarial attack level ϵ. The proposed MiDL-OSSeg method showed higher robustness to adversarial noise than FCN-Reg, as the UMSP errors increased much slower with respect to the increased attack levels than those of FCN-Reg consistently in all four cases. We attribute this MiDL-OSSeg robustness to the incorporation of the graph-based segmentation model.

4 Conclusion

In this paper, we developed a model-informed deep learning segmentation method for optimal surface segmentation, which unifies DL with the Graph-OSSeg model in a single deep neural network for end-to-end learning, greatly enhancing the strengths of both while minimizing the drawbacks of each. To the best of our knowledge, this is the first study for surface segmentation which can achieve guaranteed globally optimal solutions using deep learning. The proposed method has been validated on two medical image segmentation tasks, demonstrating its efficacy with respect to segmentation accuracy, demands for annotated training data, and robustness to adversarial noise.

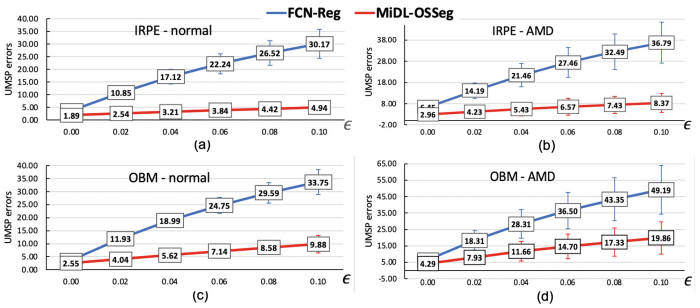

Fig. 6. Segmentation accuracy evaluated on the SD-OCT dataset for the proposed MiDL-OSSeg model, compared to FCN-Reg [21], while testing on different adversarial sample sets.

References

1. Li, K., Wu, X., Chen, D.Z., Sonka, M.: Optimal surface segmentation in volumetric images-a graph-theoretic approach. IEEE Trans. Pattern Anal. Mach. Intell. **28**(1), 119–134 (2006)
2. Song, Q., Bai, J., Garvin, M.K., Sonka, M., Buatti, J.M., Wu, X.: Optimal multiple surface segmentation with shape and context priors. IEEE Trans. Med. Imaging **32**(2), 376–386 (2013)
3. Shah, A., Abramoff, M.D., Wu, X.: Optimal surface segmentation with convex priors in irregularly sampled space. Med. Image Anal. **54**, 63–75 (2019)
4. Litjens, G., et al.: A survey on deep learning in medical image analysis. Med. Image Anal. **42**, 60–88 (2017)
5. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR 2015, pp. 3431–3440 (2015)

6. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
7. Isensee, F., Jaeger, P.F., Kohl, S.A.A., Petersen, J., Maier-Hein, K.H.: nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nat. Methods **18**(2), 203–211 (2021)
8. Arnab, A., Miksik, O., Torr, P.H.: On the robustness of semantic segmentation models to adversarial attacks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 888–897 (2018)
9. Lu, F., Wu, F., Hu, P., Peng, Z., Kong, D.: Automatic 3d liver location and segmentation via convolutional neural network and graph cut. Int. J. Comput. Assist. Radiol. Surg. **12**(2), 171–182 (2017)
10. Liu, F., Zhou, Z., Jang, H., Samsonov, A., Zhao, G., Kijowski, R.: Deep convolutional neural network and 3d deformable approach for tissue segmentation in musculoskeletal magnetic resonance imaging. Magn. Reson. Med. **79**(4), 2379–2391 (2018)
11. Milletari, F., Rothberg, A., Jia, J., Sofka, M.: Integrating statistical prior knowledge into convolutional neural networks. In: Descoteaux, M., Maier-Hein, L., Franz, A., Jannin, P., Collins, D.L., Duchesne, S. (eds.) MICCAI 2017. LNCS, vol. 10433, pp. 161–168. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66182-7_19
12. Ravishankar, H., Venkataramani, R., Thiruvenkadam, S., Sudhakar, P., Vaidya, V.: Learning and incorporating shape models for semantic segmentation. In: Descoteaux, M., Maier-Hein, L., Franz, A., Jannin, P., Collins, D.L., Duchesne, S. (eds.) MICCAI 2017. LNCS, vol. 10433, pp. 203–211. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66182-7_24
13. Zheng, S., et al.: Conditional random fields as recurrent neural networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1529–1537 (2015)
14. Arnab, A., et al.: Conditional random fields meet deep neural networks for semantic segmentation: combining probabilistic graphical models with deep learning for structured prediction. IEEE Signal Process. Mag. **35**(1), 37–52 (2018)
15. Vemulapalli, R., Tuzel, O., Liu, M.-Y., Chellapa, R.: Gaussian conditional random field network for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3224–3233 (2016)
16. Guo, H.: A simple algorithm for fitting a gaussian function [DSP tips and tricks]. IEEE Signal Process. Mag. **28**(5), 134–137 (2011)
17. Horn, R.A., Johnson, C.R.: Matrix Analysis, 2nd edn. Cambridge University Press, Cambridge (2012)
18. Farsiu, S., et al.: Quantitative classification of eyes with and without intermediate age-related macular degeneration using optical coherence tomography. Ophthalmology **121**(1), 162–172 (2014)
19. Bloch, N., Madabhushi, A., Huisman, H., Freymann, J., Kirby, J., Grauer, M.: NCI-ISBI 2013 challenge: automated segmentation of prostate structures. Cancer Imaging Arch. **370** (2015)
20. Zhou, L., Zhong, Z., Shah, A., Qiu, B., Buatti, J., Wu, X.: Deep neural networks for surface segmentation meet conditional random fields (2019). https://arxiv.org/abs/1906.04714

21. Shah, A., Zhou, L., Abrámoff, M.D., Wu, X.: Multiple surface segmentation using convolution neural nets: application to retinal layer segmentation in oct images. Biomed. Opt. Express **9**(9), 4509–4526 (2018)
22. Tian, Z., Liu, L., Zhang, Z., Fei, B.: PSNet: prostate segmentation on MRI based on a convolutional neural network. J. Med. Imaging **5**(2), 021208 (2018)
23. Szegedy, C., et al.: Intriguing properties of neural networks. In: International Conference on Learning Representations (2014)
24. Goodfellow, I., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. In: International Conference on Learning Representations (2015)