



Ricci curvature based volumetric segmentation

Na Lei^a, Jisui Huang^{b,d}, Ke Chen^{c,d}, Yuxue Ren^{b,*}, Emil Saucan^e, Zhenchang Wang^f, Yuanyuan Shang^b

^a Dalian University of Technology, Dalian 116600, China

^b Capital Normal University, Beijing 100048, China

^c University of Strathclyde, Glasgow G1 1XH, United Kingdom

^d University of Liverpool, Liverpool L69 7ZL, United Kingdom

^e Braude College of Engineering, Karmiel 2161002, Israel

^f Capital Medical University, Beijing 100050, China

ARTICLE INFO

Keywords:

Ricci curvature
Variational model
Level set
Image segmentation

ABSTRACT

The level set method has played a critical role among many image segmentation approaches. Several edge detectors, such as the gradient, have been applied to its regularisation term. However, traditional edge detectors lack high-order information and are sensitive to image noise. To tackle this problem, we introduce a method to calculate the Ricci curvature, a vital curvature in three-dimensional Riemannian geometry. In addition, we propose incorporating the curvature into the regularisation term. Experiments suggest that our method outperforms the state-of-the-art level set methods and achieves a comparable result with the Swin UNETR and Segment Anything.

1. Introduction

Image segmentation is an important aspect of image processing and has seen significant advancements [1–3]. From Otsu's thresholding method [4] and energy-based techniques like level set [5] and GrabCut [6], segmentation techniques have now evolved towards deep learning methods [7–10], including the well-known U-Net [11] and the transformer-based model UNETR [12], which have achieved state-of-the-art results in a variety of medical segmentation tasks [13–15].

Despite their remarkable advantages, deep learning-based methods present several non-negligible issues. Firstly, numerous annotations are required, particularly for 3D medical image segmentation where several hundreds of slices [16,17] of a single image should be interpreted [18]. Secondly, deep learning lacks interoperability [19,20], leading to the “black box” nature, while traditional models can offer insights into geometric properties. Therefore, traditional methods remain relevant and require further improvement.

Among the traditional methods, the level set method is recognised as highly effective, falling under edge-based and region-based models.

Through the region-based term, the model can group similar pixels in the foreground and background, respectively. This is evident in the renowned Chan-Vase model [21], which uses the global region term to

cluster different classes. The global region term assumes the foreground and background have homogeneous intensity, which is, however, not the case in real datasets. Several models have used the local region term instead of the global one to deal with heterogeneous intensity. For example, the local statistical information used in [22], the weighted region information [23] calculated using a Gaussian kernel-based convolution, the geodesic distance between two distinct spectral density functions [24], and the geodesic distance [25] between two fitted Gaussian distributions. Nevertheless, the local region term might be sensitive to initialization.

The edge-based term ensures that the energy takes a low value when the contour lies on the edge of the image with a significant gradient, which forces the contour to stop at image boundaries. A seminal model is the geodesic active contours [26], which encourages the contour to pass through edges and remain smooth by using image gradient built into the level set. In a follow-up study, the DRLSE model [27] introduced a refreshing energy term to eliminate the need for initialization. Additionally, Su et al. [28] utilised various gradient information by varying the scale of images. To extend the gradient operator, several researchers have attempted to fuse the gradient information with other forms of information. For example, Yu et al. [29] used the diffusion rate to describe edges. Liu et al. [30] incorporated the local regional fitting

* Corresponding author.

E-mail address: 7029@cnu.edu.cn (Y. Ren).

variances into the gradient information. Furthermore, the multi-local statistical information [31] was employed in noisy image segmentation.

In contrast to extending the gradient operator, recent studies have shown that curvature-based terms [32–35] can facilitate segmentation.

Existing curvature in image segmentation can be classified into one-dimensional curvature and two-dimensional curvature. Euler's elastica is a famous representative of 1D curvature, representing the squared contour curvature of the 2D level set function. Several studies [32,36,37] have shown that Euler's elastica can integrate missing boundaries due to low contrast. Other forms of 1D curvature can be defined using different norms, such as the ℓ_1 norm [38]. In the case of two dimensions, several types of curvature have been investigated, including mean curvature [39] and Gaussian curvature [40–42]. For example, Arif et al. [42] adopted the Gaussian curvature for haze or fog removal.

However, 3D image segmentation requires 3D curvature. Since there are numerous forms and definitions of 3D curvature, the best way to search for a proper one probably relies on certain conditions it may satisfy.

Noise [7,31–33,43] has proved to significantly downgrade the regularisation term, which might cause broken or unnecessary segmentation results. From a geometric perspective, being sensitive to noise implies being sensitive to coordinate transformations. We now provide an example to illustrate this. The curvature of a circle is a constant uniquely determined by its radius, such that a smaller circle has a higher curvature. By contrast, its gradient may vary with coordinate systems. For example, we can change the relationship of the three derivatives at the point p of Fig. 2 by changing the coordinate system. They can become 0 simultaneously, which might be linked to a low contrast region; if we take the derivative with respect to y , the three circles' gradients at p will simultaneously become $+\infty$, which could be considered noise in image processing.

To avoid the sensitivity to coordinate transformations, we desire an operator that satisfies the following conditions: 1) it remains invariant under the transformation of coordinates, such as from spherical coordinates to cylindrical coordinates; 2) it remains invariant under the transformation of the ambient space, such as from the Euclidean space to the Minkowski space; 3) it can capture the edge. To satisfy condition 1, it suffices to conduct calculations on the manifold, independent of the coordinates that describe it. To meet condition 2, we shall only utilise the intrinsic quantities determined by the manifold's intrinsic metric tensor, rather than certain forms related to the ambient space, including the normal vector and the second fundamental form. In condition 3, traditional edge descriptors, such as gradient, typically compare a function with the smoothest possible function, the constant function. In terms of coordinate-free expression, the quantities to be found shall compare a manifold with the smoothest possible manifold, the Euclidean space, which naturally leads to curvature.

Therefore, to satisfy the three conditions, we shall employ a curvature uniquely dependent on the metric tensor. In fact, all these curvatures have proved to depend linearly on the Riemann curvature tensor, a $3 \times 3 \times 3 \times 3 \times H \times W \times D$ tensor for a $H \times W \times D$ 3D image. In the particular case of the three-dimensional continuous Riemannian manifold, the Riemann curvature tensor can be completely described [44] by the Ricci curvature tensor in shape $3 \times 3 \times H \times W \times D$.

A classical approach to discrete Ricci curvature, particularly of networks, was due to Ollivier [45], which is based on the optimal transport theory. However, by its very definition, for an image of size $H \times W \times D$, it requires solving a separate optimal transport problem for each edge, that is, $(H - 1) \times W \times D + H \times (W - 1) \times D + H \times W \times (D - 1)$ linear programms, which is a computationally prohibitive task in 3D image processing. Another viewpoint towards the discretization of the Ricci curvature, based on the connection between curvature and the Laplacian on the manifold, was adopted by Forman [46], which is of linear complexity to be proved. However, all the weights used in the abstract framework of [46] were not specified, and consequently, no experiments

were performed.¹

In this paper, we investigate a level set model that integrates the Ricci curvature for 3D image segmentation. Specifically, we first consider the 3D image $f(x,y,z)$ as a 3D parametric hyper-surface $(x,y,z,f(x,y,z))$ embedded in the 4D Euclidean space, and use the induced metric tensor to make measurements (i.e., volume, area and length) on the Riemannian geometry. The three forms of measurement can then serve as the weight function of [46] to derive the discrete Ricci curvature. Finally, we introduce a level set model where the trace of the Ricci curvature acts as the edge descriptor. Fig. 3 compares our proposed pipeline with prior 3D models, where we replace the length of the gradient with the trace of the Ricci curvature. The proposed discrete Ricci curvature and its absolute value prohibit disproportionate fine details and noise, which stands out in Fig. 1. By contrast, the gradient in Fig. 1(d) shows noticeable noise.

In summary, our contributions are as follows:

- This is the first study to propose a novel 3D level set method with Ricci curvature in the edge term.
- We introduce a formula for calculating the Ricci curvature for 3D images.
- Our experiments demonstrate that our method outperforms commonly used gradient-based level sets and is comparable to deep learning models.

The remainder of the paper is organised as follows: Section 2 briefly reviews related 3D segmentation models. Section 3 is devoted to the 3D

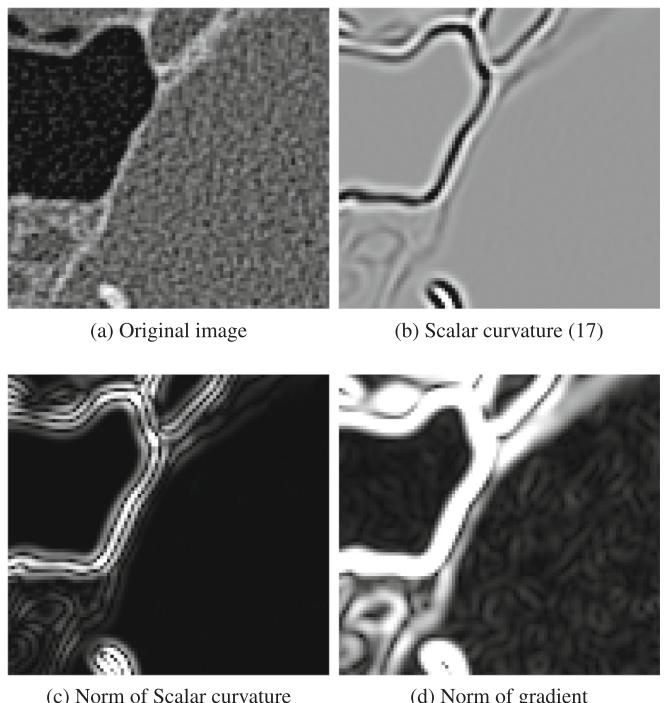


Fig. 1. A region in a real 3D CT image. It can be seen that the Ricci curvature has a continuous change from positive to negative on the boundary of the image, and the absolute value of the curvature is close to 0 in the region where the gradient produces recognisable noise.

¹ However, Forman clearly states in his paper [46] that the motivation of the weights comes from the natural geometric ones, namely length, area, volume, etc. Also, such weights were considered in imaging applications of Forman-Ricci curvature [47–49].

level set model with Ricci curvature as the edge term. [Section 4](#) describes our experiments on real datasets, including the colon, brain, vessel, the necrotic tumour core from the BraTS dataset, and the right kidney from the BTCV dataset. We discuss our limitations and future work in [Section 5](#), followed by a conclusion in [Section 6](#).

2. Related work

2.1. ALF model

Given a 3D image f , the adaptive local fitting model (ALF) [50] finds the optimal surface implicitly represented by the level set $\phi = 0$, satisfying

$$\min_{\Gamma} E(\Gamma) = E_G(\Gamma) + E_F(\Gamma), \quad (1)$$

where $E_G(\Gamma) = \alpha \int g(x)d(x)dx$, $g(x)$ is the commonly used gradient operator

$$g(x) = \frac{1}{1 + a|\nabla f(x)|^2}, \quad a > 0, \quad (2)$$

and

$$d(x) = \prod_{i=1}^{n_p} \left(1 - \exp \left(-\frac{|x - x_i|^2}{2\sigma^2} \right) \right) \quad (3)$$

represents a distance penalty term. Another term $E_F(\Gamma)$ is defined as

$$\begin{aligned} & \lambda_1 \int b_1(\phi(x), \gamma_{in})(f(x) - C_1)^2 dx \\ & + \lambda_2 \int b_2(\phi(x), \gamma_{out})(f(x) - C_2)^2 dx, \end{aligned} \quad (4)$$

where $b_1(\phi(x), \gamma_{in}) = B(\phi(x), \gamma_{in}, 0)$ and $b_2(\phi(x), \gamma_{out}) = B(\phi(x), 0, \gamma_{out})$ with

$$B(\phi, \gamma_{in}, \gamma_{out}) = H(\phi(x) + \gamma_{in})(1 - H(\phi(x) - \gamma_{out})) \quad (5)$$

depicting the narrow band around the level set surface $-\gamma_{in} \leq \phi(x) \leq \gamma_{out}$. Therefore, the level set method can be written as

$$\begin{aligned} & \min_{\phi} \alpha \int_{\Omega} g(x)d(x)|\nabla H(\phi)|dx \\ & + \int_{\Omega} \lambda_1 B(\phi(x), \gamma_{in}, 0)(f(x) - C_1)^2 \\ & + \lambda_2 B(\phi(x), 0, \gamma_{out})(f(x) - C_2)^2 dx. \end{aligned} \quad (6)$$

2.2. LRFI model

The local regional fitting information based level set (LRFI) [30] relies on two key terms. The first term is

$$v(x) = \alpha e^{-\beta|f_{in}(x) - f_{out}(x)|} + k, \quad (7)$$

where $f_{in}(x)$ and $f_{out}(x)$ are the local regional fitting means in foreground and background

$$\begin{aligned} f_{in}(x) &= \frac{G(x-y)^* [f(y)H(\phi(y))]}{A_{in}} \\ f_{out}(x) &= \frac{G(x-y)^* [f(y)(1 - H(\phi(y)))]}{A_{out}}, \end{aligned} \quad (8)$$

respectively, A_{in} and A_{out} are their respective weighted area

$$\begin{aligned} A_{in} &= G(x-y)^* H(\phi(y)) \\ A_{out} &= G(x-y)^* (1 - H(\phi(y))), \end{aligned} \quad (9)$$

and $G(x)$ is the Gaussian kernel used for the convolution operator. The second term is the improved edge stop function,

$$g(x) = \frac{1}{1 + |\nabla f| / (\sigma_{in}^2(x) + \sigma_{out}^2(x) + 1)}, \quad (10)$$

where $\sigma_{in}^2(x)$ and $\sigma_{out}^2(x)$ are weighted variance

$$\begin{aligned} \sigma_{in}^2(x) &= G(x-y)^* [(f(y) - f_{in}(x))^2 H(\phi(y))] \\ \sigma_{out}^2(x) &= G(x-y)^* [(f(y) - f_{out}(x))^2 (1 - H(\phi(y)))] \end{aligned} \quad (11)$$

It can be observed that the edge stop function $g(x)$ is still based on the gradient information $|\nabla f|$. The final level set energy has the following form

$$\begin{aligned} E(\phi) &= \mu \int_{\Omega} p(|\nabla \phi(x)|) dx + \lambda \int_{\Omega} g(x) \delta(\phi(x)) |\nabla \phi(x)| dx \\ & + v(x) \int_{\Omega} g(x) H(-\phi(x)) dx, \end{aligned} \quad (12)$$

where the $p(x)$ function is an additional penalty term that can force the level set function to be close to a distance function as soon as possible.

2.3. MLSI model

The multi-local statistical information based level set (MLSI) [31] utilises the entropy $E(x)$ of the neighbourhood of each pixel x . The first term used is the weighted regional coefficient,

$$v(x) = \frac{\alpha + \beta(1 - E_{nor}(x))}{1 + |f_{in}(x) - f_{out}(x)|^2} + k, \quad (13)$$

where $E_{nor}(x)$ is a normalised entropy such that all scaled entropy is within $[0, 1]$, and $f_{in}(x)$ and $f_{out}(x)$ are the same as the LRFI model. The second term is

$$g(x) = \frac{1}{1 + E_{nor}(x) |\nabla f| / \left(\frac{\sigma_{in}(x)}{A_{in}} + \frac{\sigma_{out}(x)}{A_{out}} + 1 \right)}, \quad (14)$$

where $\sigma_{in}(x)$ and $\sigma_{out}(x)$ are the root of the $\sigma_{in}^2(x)$ and $\sigma_{out}^2(x)$ in the LRFI model. We can see that this edge stop function $g(x)$ is also based on the gradient information $|\nabla f|$. Based on the two terms, the final level set method can be described as

$$\begin{aligned} E(\phi) &= \mu \int_{\Omega} p(|\nabla \phi(x)|) dx \\ & + (\lambda E_{nor}(x) + l) \int_{\Omega} g(x) \delta(\phi(x)) |\nabla \phi(x)| dx \\ & + v(x) \int_{\Omega} g(x) H(-\phi(x)) dx. \end{aligned} \quad (15)$$

3. Ricci curvature based level set model

Given a 3D image $f(x, y, z)$, the key point in this section is to calculate the Ricci curvature of the 3D hyper-surface $(x, y, z, f(x, y, z))$ using the metric tensor induced from the 4D Euclidean space. This allows for an application to the proposed level set model. We first reformulate Forman's Ricci curvature based on the 3D array in [Section 3.1](#). Next, we provide a coordinate-free weight function in the curvature formula in [Section 3.2](#). Finally, we integrate the discrete Ricci curvature into the level set for 3D image segmentation in [Section 3.3](#), followed by a discussion of the complexity of our model in [Section 3.4](#).

3.1. Discrete Ricci curvature

Based on the continuous setting of [44] (cf. Appendix A), the discrete approach to the Ricci curvature lies in the discrete analogue of certain differential operators on the manifold, which poses a considerable challenge. Forman [46] studies the discrete Ricci curvature through a

discrete Laplacian, which can be related to the Ricci curvature by Bochner's formula [51].

We describe Forman's original formula in Appendix B in terms of cell complex. More precisely, the discrete Ricci curvature tensor **Ric** at a point is represented by the values associated with its adjacent edges, where the value of an edge is an analogue of the continuous Ricci curvature in that direction. It is easy to see from (B.1) that the discrete curvature is uniquely determined by three different weight functions w , that is, three forms of value associated with a point, edge, and face respectively, which are, however, not specified in Forman's discrete Ricci curvature. In this chapter, we propose using a measure, i.e., length, area and volume, on a Riemann manifold (again, see also [47–49]) to represent the weight functions w , which is invariant under coordinate transformations.

In the particular case of the 3D image, a natural setting of the cell complex is the hexahedral mesh, where each hexahedron has 1 volume, 6 faces, 12 edges, and 8 points. At this point, a natural question is whether a voxel in a 3D image should be considered a point or a volume. In consideration of the fact that a point cannot be assigned a meaningful measure since it is dimensional 0, a voxel should correspond to a volume. Therefore, the original Ricci curvature tensor defined on 6 adjacent edges of a point can now be identified with the Ricci curvature tensor defined on 6 adjacent faces of a volume. In other words, when calculating discrete Ricci curvature, if, for instance, a point's weight is required, we just need to provide the weight of the corresponding volume.

Based on our setting of the hexahedral mesh, we shall now give an array version of the Forman curvature which is adapted to 3D images.

The hexahedral mesh corresponding to a $H \times W \times D$ image can be completely represented by an up-sampled array in shape $(2H+1) \times (2W+1) \times (2D+1)$. To see this, it suffices to observe that a hexahedron is composed of 27 cells: 1 volume, 6 faces, 12 edges, and 8 points, which can exactly fit in a $3 \times 3 \times 3$ array. Hence, if putting a volume, corresponding to a voxel as mentioned, at index $(1, 1, 1)$ of the up-sampled array, all volumes will have subscript indexing composed of three odd numbers, the subscript indexing of each face will have exactly 2 odd numbers and each edge will have array indexing with only one odd number.

Having defined the one-to-one map from the hexahedral mesh to an up-sampled array, the discrete Ricci curvature can be explained using two up-sampled arrays w and **Ric** both in shape $(2H+1) \times (2W+1) \times (2D+1)$, where w can quantify weights (i.e., length, area and volume) and **Ric** can store the Ricci curvature of faces, as follows. Assume the face under consideration has subscript index \mathbf{x} in the up-sampled array, then its Ricci curvature **Ric**(\mathbf{x}) is

$$\begin{aligned} \text{Ric}(\mathbf{x}) = & f_1(\mathbf{x}, e_i) + f_1(\mathbf{x}, -e_i) \\ & + f_2(\mathbf{x}, e_{(i+1)\%3}) + f_2(\mathbf{x}, -e_{(i+1)\%3}) \\ & + f_2(\mathbf{x}, e_{(i+2)\%3}) + f_2(\mathbf{x}, -e_{(i+2)\%3}), \end{aligned} \quad (16)$$

where $\mathbf{e}_0 = (1, 0, 0)$, $\mathbf{e}_1 = (0, 1, 0)$ and $\mathbf{e}_2 = (0, 0, 1)$ are three displacements relative to the index considered, $i = 0, 1, 2$ is an integer indicating the position of the even number in \mathbf{x} , $a\%b$ means the remainder after division of a by b ,

$$f_1(\mathbf{x}, \mathbf{e}) = \frac{w(\mathbf{x})}{w(\mathbf{x} + \mathbf{e})} - \frac{\sqrt{w(\mathbf{x})w(\mathbf{x} + \mathbf{e})}}{w(\mathbf{x} + \mathbf{e})},$$

and

$$f_2(\mathbf{x}, \mathbf{e}) = \frac{w(\mathbf{x} + \mathbf{e})}{w(\mathbf{x})} - \frac{w(\mathbf{x} + \mathbf{e})}{\sqrt{w(\mathbf{x})w(\mathbf{x} + \mathbf{e})}}.$$

For example, given a face located at position $\mathbf{x} = (1, 2, 3)$ of the up-sampled array, $e_i = e_1$ since the even number 2 is the second index, giving rise to $\text{Ric}(\mathbf{x}) = f_1(\mathbf{x}, \mathbf{e}_1) + f_1(\mathbf{x}, -\mathbf{e}_1) + f_2(\mathbf{x}, \mathbf{e}_2) + f_2(\mathbf{x}, -\mathbf{e}_2) + f_3(\mathbf{x}, \mathbf{e}_0) + f_3(\mathbf{x}, -\mathbf{e}_0)$.

After obtaining the Ricci curvature tensor defined on each face, we

calculate its trace to obtain a scalar at each voxel, which serves as an edge descriptor in Section 3.3. Specifically, a voxel has the scalar curvature

$$S(\mathbf{x}) = \sum_{i=0,1,2} \text{Ric}(2\mathbf{x} + 1 - \mathbf{e}_i) + \text{Ric}(2\mathbf{x} + 1 + \mathbf{e}_i), \quad (17)$$

where S is a 3D array with the shape $H \times W \times D$, and $2\mathbf{x} + 1$ is the position of the corresponding volume in the **Ric** array (16). Eq. (17) implies that to compute the scalar curvature of a voxel located at array index \mathbf{x} , we sum up the Ricci curvature tensor of its six adjacent faces.

3.2. Discrete Ricci curvature with coordinate-free weights

From (16) and (17), it is evident that the Ricci curvature of a $H \times W \times D$ image is uniquely determined by the weight function w defined on edges, faces, and volumes, respectively, of the corresponding hexahedral mesh. These have been shown to be equivalent to an upsampled array in shape $(2H+1) \times (2W+1) \times (2D+1)$ in the previous section. In this section, we will define a suitable weight function and discuss its invariant property under coordinate transformations.

A natural coordinate-free weight is a measure (i.e., volume, area, or length) associated with a Riemannian manifold. To assign a measure to the Riemannian manifold determined by a 3D image, we assign a 3×3 positive definite matrix to each voxel. We propose using a metric tensor that is widely studied and used, instead of a manually specified matrix, at each point. Several metric tensors can be employed for 3D images. One way is to use classical three-dimensional metric tensors, including Euclidean and Hyperbolic metrics [52,53]. However, when applying the naive Euclidean metric, the Ricci curvature of each voxel becomes 0. Similarly, if using the Hyperbolic metric, the sectional curvature is -1 everywhere. Another way is to lift the 3D image to a higher dimensional space, such as the 4D Euclidean and Minkowski spaces. Unfortunately, due to its special metric tensor, Minkowski space can lead to an undesired restriction on the magnitude of the gradient, analogous to the observation that no object has a faster speed than light. Therefore, a proper metric tensor of 3D images $f(x, y, z)$ is induced by the metric tensor of the 4D Euclidean space $(x, y, z, f(x, y, z))$ that the image is lifted into.

Specifically, since a 3D image $f(x, y, z)$ is a parametric hyper-surface $\mathbf{r}(x, y, z) = (x, y, z, f(x, y, z))$, it has three tangents at each point: $\mathbf{r}_x = (1, 0, 0, f_x)$, $\mathbf{r}_y = (0, 1, 0, f_y)$ and $\mathbf{r}_z = (0, 0, 1, f_z)$. Recall that the metric tensor is a 3×3 matrix at each point whose elements represent pairwise inner products of the three tangents, yielding the following positive definite matrix at each voxel

$$\mathbf{g} = \begin{bmatrix} 1 + f_x^2 & f_x f_y & f_x f_z \\ f_y f_x & 1 + f_y^2 & f_y f_z \\ f_z f_x & f_z f_y & 1 + f_z^2 \end{bmatrix}, \quad (18)$$

where $1 + f_x^2$, for instance, describes the inner product of \mathbf{r}_x and \mathbf{r}_x at the point considered.

The metric tensor \mathbf{g} allows us to measure a volume on the manifold which will not vary with coordinates

$$\int \sqrt{\det(\mathbf{g})} dx dy dz.$$

It follows from the linear algebra identity

$$\det(\mathbf{I} + \mathbf{u}\mathbf{v}^T) = 1 + \mathbf{u}^T \mathbf{v},$$

where \mathbf{u} and \mathbf{v} are column vectors, that the volume of a volume on the 3D manifold reads

$$\int \sqrt{1 + f_x^2 + f_y^2 + f_z^2} dx dy dz. \quad (19)$$

Here the derivatives f_x , f_y , and f_z represent the gradient of volumes, which we have identified with voxels of the 3D image, and thus can be directly calculated by applying the centre difference to the original 3D image.

Regarding faces, their 2×2 metric tensor will vary with their direction as follows

$$\begin{bmatrix} 1+f_x^2 & f_x f_y \\ f_x f_y & 1+f_y^2 \end{bmatrix}, \begin{bmatrix} 1+f_y^2 & f_y f_z \\ f_y f_z & 1+f_z^2 \end{bmatrix}, \begin{bmatrix} 1+f_x^2 & f_x f_z \\ f_x f_z & 1+f_z^2 \end{bmatrix}.$$

Consequently, their areas can be analogously represented as:

$$\begin{aligned} & \int \sqrt{1+f_x^2+f_y^2} dx dy, \\ & \int \sqrt{1+f_y^2+f_z^2} dy dz, \\ & \int \sqrt{1+f_x^2+f_z^2} dx dz. \end{aligned} \quad (20)$$

The unknown derivatives f_x , f_y , and f_z , defined on faces, can be interpolated from two adjacent volumes (i.e., correspond to the voxels of the original image) using interpolation coefficients equal to the distances to the two volumes' centres. For instance, a face perpendicular to the x axis has two interpolation coefficients, $\sqrt{1+f_x^2(a)} dx$ and

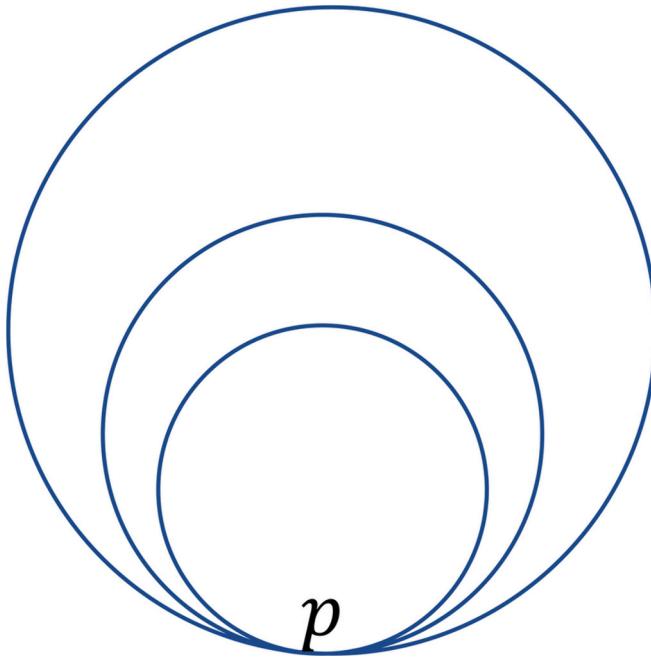


Fig. 2. Geometrical perspective of noise. Many traditional edge operators, such as gradient, are sensitive to noise. For example, the gradients of the three circles at point p may have an arbitrary relationship in a designed coordinate system. They can simultaneously reach 0, which might be identified with the low contrast region, if we take the derivative with respect to x . They can also take the value $+\infty$, which might emerge from the high noise region, if we take the derivative with respect to y . However, it is not essential to describe a quantity on a curve (or surface) with the assistance of a coordinate system. Manifold is an abstract surface on which certain quantities are independent of the choice of coordinate systems, although it is typically described using different functions, such as a parametric function or an implicit function (e.g., a level set), or using different coordinate systems, such as the polar coordinates, the cylindrical coordinates or a user-specified coordinate system. For example, although we typically use different forms of function to describe a circle, the curvature of a circle is dependent only on its radius rather than the function itself. In this paper, we propose using the 3D Ricci curvature of 3D images to serve as an edge descriptor of the level set model for 3D image segmentation.

$\sqrt{1+f_x^2(b)} dx$, where a and b are its two adjacent volumes, as depicted in Fig. 4.

Regarding edges, they possess three possible metric tensors:

$$[1+f_x^2], \quad [1+f_y^2], \quad [1+f_z^2].$$

Consequently, their lengths are given by

$$\int \sqrt{1+f_x^2} dx, \quad \int \sqrt{1+f_y^2} dy, \quad \int \sqrt{1+f_z^2} dz. \quad (21)$$

Similarly, since an edge divides the rectangle bounded by the centres of the four adjacent volumes (i.e., correspond to the voxels of the original image) of the edge into four smaller rectangles, we can apply bilinear interpolation to estimate its gradient. The interpolation coefficients are the areas of the four smaller rectangles, as shown in Fig. 5.

To provide some insight into the weight setting, we conclude this section by investigating its coordinate-free property. Specifically, we will demonstrate that the following equality holds for any arbitrary transformation:

$$\int \sqrt{\det(\tilde{\mathbf{g}})} d\tilde{x} d\tilde{y} d\tilde{z} = \int \sqrt{\det(\mathbf{g})} dx dy dz,$$

where $(\tilde{x}, \tilde{y}, \tilde{z})$ is the image under an arbitrary transformation of the coordinates (x, y, z) , and $\tilde{\mathbf{g}}(\tilde{x}, \tilde{y}, \tilde{z})$ is the 3×3 metric tensor of the new coordinates.

Recall the formula for change of variables in multiple integrals:

$$\begin{aligned} & \int f(\tilde{x}, \tilde{y}, \tilde{z}) d\tilde{x} d\tilde{y} d\tilde{z} = \\ & \int f(\tilde{x}(x, y, z), \tilde{y}(x, y, z), \tilde{z}(x, y, z)) \frac{\partial(\tilde{x}, \tilde{y}, \tilde{z})}{\partial(x, y, z)} dx dy dz, \end{aligned}$$

where

$$\frac{\partial(\tilde{x}, \tilde{y}, \tilde{z})}{\partial(x, y, z)} = \begin{vmatrix} \frac{\partial \tilde{x}}{\partial x} & \frac{\partial \tilde{x}}{\partial y} & \frac{\partial \tilde{x}}{\partial z} \\ \frac{\partial \tilde{y}}{\partial x} & \frac{\partial \tilde{y}}{\partial y} & \frac{\partial \tilde{y}}{\partial z} \\ \frac{\partial \tilde{z}}{\partial x} & \frac{\partial \tilde{z}}{\partial y} & \frac{\partial \tilde{z}}{\partial z} \end{vmatrix}$$

is the Jacobian of this transformation, which we will denote by J for simplicity of notation.

To apply the change of variables, it suffices to derive the new metric tensor $\tilde{\mathbf{g}}$ in terms of (x, y, z) . By applying the chain rule to the function $\mathbf{r} = (x(\tilde{x}, \tilde{y}, \tilde{z}), y(\tilde{x}, \tilde{y}, \tilde{z}), z(\tilde{x}, \tilde{y}, \tilde{z}), f(\tilde{x}, \tilde{y}, \tilde{z}))$ where f is a 3D image, we have:

$$\begin{bmatrix} r_{\tilde{x}} \\ r_{\tilde{y}} \\ r_{\tilde{z}} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \tilde{x}} r_x + \frac{\partial y}{\partial \tilde{x}} r_y + \frac{\partial z}{\partial \tilde{x}} r_z \\ \frac{\partial x}{\partial \tilde{y}} r_x + \frac{\partial y}{\partial \tilde{y}} r_y + \frac{\partial z}{\partial \tilde{y}} r_z \\ \frac{\partial x}{\partial \tilde{z}} r_x + \frac{\partial y}{\partial \tilde{z}} r_y + \frac{\partial z}{\partial \tilde{z}} r_z \end{bmatrix} = J^{-T} \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix}.$$

Since the metric tensor is the pairwise inner products of \mathbf{r}_x , \mathbf{r}_y and \mathbf{r}_z , the metric tensor $\tilde{\mathbf{g}}$ can be expressed in terms of \mathbf{g} as:

$$\tilde{\mathbf{g}} = \begin{bmatrix} r_{\tilde{x}} \\ r_{\tilde{y}} \\ r_{\tilde{z}} \end{bmatrix} \begin{bmatrix} r_{\tilde{x}} & r_y & r_z \end{bmatrix} = J^{-T} \begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} \begin{bmatrix} r_x & r_y & r_z \end{bmatrix} J^{-1} = J^{-T} \mathbf{g} J^{-1}.$$

Therefore,

$$\begin{aligned} \sqrt{\det(\tilde{\mathbf{g}})} d\tilde{x} d\tilde{y} d\tilde{z} &= \sqrt{\det(J^{-T} \mathbf{g} J^{-1})} \det(J) dx dy dz \\ &= \sqrt{\det(\mathbf{g})} dx dy dz, \end{aligned}$$

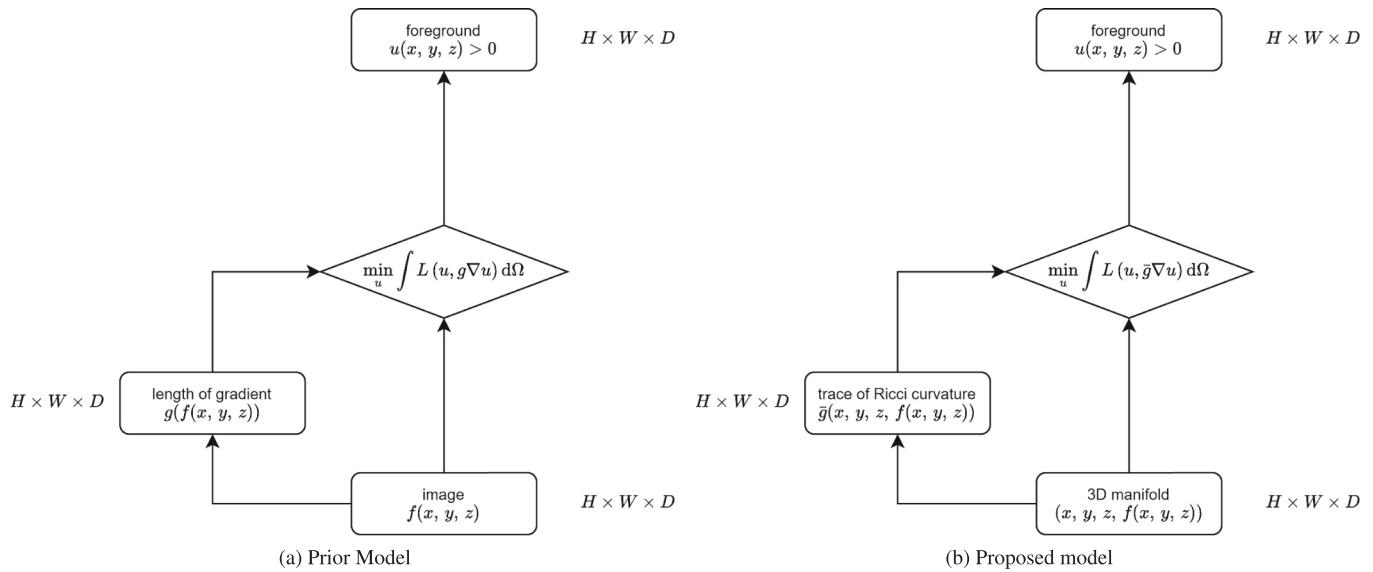


Fig. 3. Comparison of our model with prior models. The gradient operator is commonly used in prior models and is recognised as noise-sensitive. To address this, we propose a 3D level set model with the substitution of the Ricci curvature for the traditional gradient operator in the 3D level set model. Specifically, given a 3D image $f(x,y,z)$, we derive the Ricci curvature of the 3D parametric hyper-surface $(x,y,z,f(x,y,z))$ whose metric tensor is induced by the 4D Euclidean space. Thus, the coordinate-free operator can supersede the gradient, which is sensitive to coordinate transformations.

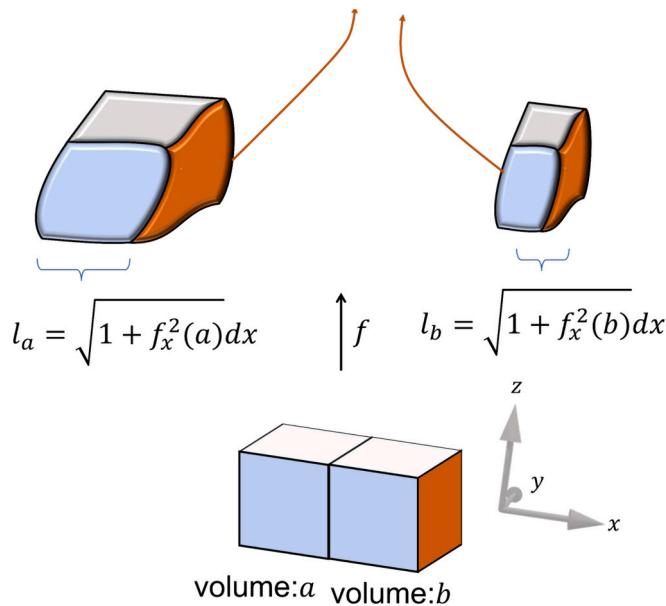


Fig. 4. Calculation of the weight of a face, which is perpendicular to the x -direction and is the boundary of volume a and volume b . The face's f_y and f_z are the weighted average of the corresponding values of its adjacent volumes. The smaller the length l_a of volume a than the length l_b of volume b in the x -direction, the closer the f_y and f_z of the face to $f_y(a)$ and $f_z(a)$.

as desired.

3.3. Level set model

In this section, we will present our proposed level set model that integrates Ricci curvature.

Inspired by the model in [50], we solve for the function $\phi(x,y,z)$ through minimising the functional

$$\min_{\phi} E(\phi) = E_G(\phi) + E_F(\phi), \quad (22)$$

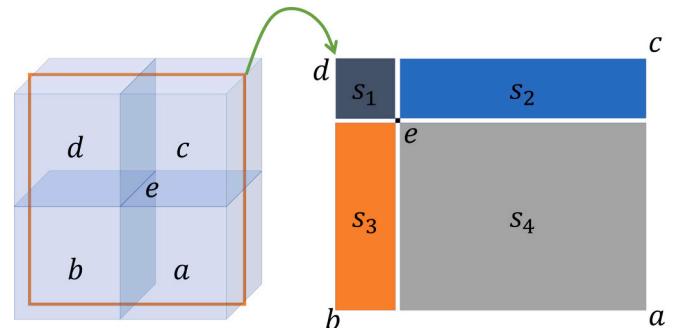


Fig. 5. Calculation of edge e 's weight, given the f_x , f_y , and f_z of the four volumes adjacent to it. If e is along the x -direction, seen from the yz plane, this is equivalent to a bilinear interpolation: given the known f_x of four corners (a, b, c, d), find the unknown f_x of a point inside the rectangle. The unknown point divides the rectangle into four small rectangles with areas s_1, s_2, s_3 , and s_4 , respectively, calculated by the formula $\sqrt{1 + f_y^2 + f_z^2} dy dz$.

where the level set function $\phi(x, y, z)$ partitions the 3D image $f(x, y, z)$ into the foreground $\phi(x, y, z) > 0$ and the background $\phi(x, y, z) < 0$.

E_G is the edge term expressed as

$$E_G(\phi) = \int \alpha g(x, y, z) |\nabla H(\phi)| \, dx \, dy \, dz, \quad (23)$$

where $\alpha > 0$ is a hyperparameter, and $H(x)$ is the smoothed Heaviside step function, defined as

$$H_\epsilon(x) = \frac{1}{2} + \frac{1}{\pi} \arctan\left(\frac{\pi x}{\epsilon}\right).$$

The key function $g(x,y,z)$ is capable of terminating the contour evolution at the image boundary, typically described by the image gradient as in (2). However, gradients are known to be sensitive to noise. As indicated in Fig. 2, a noise point may have a gradient of 0 in another coordinate system, suggesting that the appearance of noise might be due to an over-reliance on coordinates. To mitigate this effect, we propose the function $g(x,y,z)$ in the form

$$g(x, y, z) = \frac{1}{1 + |S(x, y, z)|}, \quad (24)$$

where $S(x, y, z)$ (17) is the trace of the Ricci curvature tensor.

$E_F(\phi)$ represents the data fidelity term, defined as

$$E_F(\phi) = \lambda_1 \int_{S_{in}} (f - C_{in})^2 dx dy dz + \lambda_2 \int_{S_{out}} (f - C_{out})^2 dx dy dz, \quad (25)$$

where λ_1 and λ_2 are coefficients. S_{in} is a band region with the width γ_{in} inside the isosurface $\phi = 0$, and S_{out} is a band region with the width γ_{out} outside the isosurface $\phi = 0$. Both S_{in} and S_{out} control the image region that introduces statistical information into the region term. We assume that ϕ takes positive values at the voxels interior to $\phi = 0$, so S_{in} satisfies $0 < \phi(S_{in}) < \gamma_{in}$, and analogously, $0 > \phi(S_{out}) > -\gamma_{out}$.

The remaining parameters C_{in} and C_{out} are determined as follows. If $\phi(x, y, z)$ is fixed, E_F will attain the minimum value when

$$C_{in} = \frac{\int H(\phi)H(\gamma_{in} - \phi)f dx dy dz}{\int H(\phi)H(\gamma_{in} - \phi) dx dy dz}, \quad (26)$$

and

$$C_{out} = \frac{\int H(-\phi)H(\phi + \gamma_{out})f dx dy dz}{\int H(-\phi)H(\phi + \gamma_{out}) dx dy dz}. \quad (27)$$

Ultimately, the optimal ϕ can be obtained from the Euler–Lagrange equation:

$$\begin{aligned} \phi_t = & \delta(\phi)\alpha \nabla \cdot \left(\frac{g \nabla \phi}{|\nabla \phi|} \right) + \delta(\phi)\lambda_2 H(\phi)H(\gamma_{in} - \phi)(f - C_{out}) \\ & - \delta(\phi)\lambda_1 H(-\phi)H(\phi + \gamma_{out})(f - C_{in}). \end{aligned} \quad (28)$$

Here, $\delta(x)$ represents the derivative of the Heaviside step function $H(x)$, which is the Dirac delta function. To solve Eq. (28) in a 3D image, we discretize it using central differences and apply gradient descent for the numerical solution [50]. The segmentation algorithm is illustrated in Algorithm 1.

Algorithm 1. Ricci curvature based segmentation.

Require: 3D medical image $f(x, y, z)$ of shape $H \times W \times D$, a boolean array $I(x, y, z)$ in shape $H \times W \times D$ where the true values indicate the initial points, parameter $\gamma_{in} > 0$ and parameter $\gamma_{out} > 0$.

Ensure: The level set function $\phi(x, y, z)$ in shape $H \times W \times D$ where the area $\phi > 0$ represents the foreground.

- 1: Calculate the trace of Ricci curvature $S(x, y, z)$ in shape $H \times W \times D$ following Algorithm 2.
 - 2: Initialise a signed distance function (SDF) $\phi(x, y, z)$ in shape $H \times W \times D$ such that: 1) its values at the initial region's boundary are 0; 2) ϕ takes positive values in the interior of the boundary and negative values in the exterior; 3) the absolute value of ϕ at each voxel indicates the distance between the voxel and the boundary.
 - 3: Recognise the narrow band B_0 such that $-\gamma_{out} < \phi(B_0) < \gamma_{in}$.
 - 4: Within the narrow band, track the zero isosurface Γ_0 such that each voxel of Γ_0 is encompassed by voxels having both positive and negative values in ϕ .
 - 5: **for** each i starting with $i = 0$ and $\phi_0 = \phi$ **do**
 - 6: Apply gradient descent (28) only to the voxels of Γ_i and obtain a new level set ϕ_{i+1} .
 - 7: Propagate the updated level set value $\phi_{i+1}(\Gamma_i)$ in an SDF fashion until the values are beyond the interval $[-\gamma_{out}, \gamma_{in}]$ and denote this region by B_{i+1} .
 - 8: Within the new narrow band B_{i+1} , track the zero isosurface Γ_{i+1} .
 - 9: **end for**
-

Algorithm 2. Computation of scalar curvature.

Require: 3D image $f(x, y, z)$ of size $H \times W \times D$.

Ensure: Scalar curvature $S(x, y, z)$ of size $H \times W \times D$.

- 1: Initialize an array $w(x, y, z)$ of size $(2H + 1) \times (2W + 1) \times (2D + 1)$ to store the weights.
 - 2: Initialize an array $\text{Ric}(x, y, z)$ of size $(2H + 1) \times (2W + 1) \times (2D + 1)$ to store the Ricci curvature tensor.
 - 3: **for** each volume with array index x consisting of 3 odd numbers **do**
 - 4: Store its volume (19), using the gradient of the original image f , into $w(x)$.
 - 5: **end for**
 - 6: **for** each face with array index x consisting of 2 odd numbers and 1 even number **do**
 - 7: Store its area (20), using the gradient linearly interpolated from the gradients of 2 adjacent volumes, into $w(x)$.
 - 8: **end for**
 - 9: **for** each edge with array index x consisting of 1 odd number and 2 even numbers **do**
 - 10: Store its length (21), using the gradient bilinearly interpolated from the gradients of 4 adjacent volumes, into $w(x)$.
 - 11: **end for**
 - 12: **for** each face with array index x consisting of 2 odd numbers and 1 even number **do**
 - 13: Compute $\text{Ric}(x)$ using (16).
 - 14: **end for**
 - 15: **for** each x **do**
 - 16: Compute $S(x)$ using (17).
 - 17: **end for**
-

3.4. Complexity

In this section, we present the complexity of our model. We shall mention that the calculation of discrete Ricci curvature (17) and the evolution of the level set are independent. In other words, the Ricci curvature can be calculated only once before starting the level set and remains unchanged during iteration. Therefore, we discuss their complexity separately.

Firstly, the Ricci curvature introduces a linear computational complexity. To see this, it suffices to observe that if the two functions f_1 and f_2 in (16) depend linearly on the array $w(x, y, z)$, the Eq. (16) reduces to a 3D convolution on the image $w(x, y, z)$ with the receptive field $5 \times 5 \times 5$. Although f_1 and f_2 are nonlinear functions, this does not increase the order of its complexity. Furthermore, Algorithm 2 shows that the eventual edge descriptor $S(x, y, z)$ requires five for loops, which implies it is five times slower than the gradient operator $|\nabla f|$, assuming the cost of the gradient operator is one for loop. Thus, precisely, the computational complexity is $O(5N)$, where N is the number of voxels in the 3D image, compared to the $O(N)$ for the gradient operator.

Next, we discuss the complexity of the level set model. Due to the fact that the Dirac delta function $\delta(\phi)$ in (28) takes nonzero values only around $\phi = 0$, the gradient descent (28) is essentially conducted on a region around the voxels corresponding to zero level set values $\phi = 0$. In application, we make this region coincide with the parameters γ_{in} and γ_{out} . That is, in this region $\gamma_{in} > \phi > -\gamma_{out}$, we employ all computations, including the calculation of the region information (26) and (27) to be used in gradient descent, gradient descent restricted to the zero level set $\phi = 0$ as in Algorithm 1.6, recognising a new narrow band after each gradient descent as in Algorithm 1.7, and tracking a new zero level set within the new narrow band as in Algorithm 1.8.

Among these operations, the primary computational burden is introduced by the reinitialization of the signed distance function (SDF) [27, 54], by which we mean the process to propagate the gradient descent result to a new narrow band as in Algorithm 1.7. To make our process more computationally attractive, we make use of two Breadth-

first searches (BFS) separately, one starting at the updated zero level set and proceeding inward, and the other starting at the same voxels but moving outward. We terminate the process when the new narrow band $\gamma_{\text{in}} > \phi > -\gamma_{\text{out}}$ has been recognised. During BFS, each voxel inherits SDF value from its unique parent voxel, and then we add to it 1 or -1 , depending on whether the voxel is located in the interior or exterior, respectively. It is necessary to mention that although the BFS only adds an integer to the distance, the updated zero level set is a continuous floating point, thus leading to a floating point propagation, a practical yet good approximation to the SDF.

Therefore, in each iteration, the level set model brings computational complexity $O(N_{\text{band}})$ where N_{band} indicates the voxel number within $\gamma_{\text{in}} > \phi > -\gamma_{\text{out}}$. In practice, a satisfactory segmentation result routinely favours a small narrow band (e.g. $\gamma_{\text{in}}, \gamma_{\text{out}} = 1$ or 2), which implies our method can offer a respectable computational advantage.

4. Experimental results

In this section, we conduct a comprehensive evaluation of our proposed segmentation model on five diverse datasets: brain, vessel, colon, Brain Tumour Segmentation (BraTS), and Beyond the Cranial Vault (BTCV) dataset. The brain and vessel datasets are provided by The Beijing Friendship Hospital. The colon dataset [17,55,56], BraTS [16,57,58], and BTCV dataset [59,60] are publicly available online.

Before delving into the segmentation results, we highlight the advantages of Ricci curvature over the traditional gradient in terms of resistance to noise and low contrast regions in Section 4.1. Subsequently, we compare our method with traditional 3D level set segmentation models across brain, vessel, and colon datasets, followed by a comparison of our model with specific deep learning models on BTCV and BraTS datasets. The traditional 3D level sets include Adaptive Local Fitting (ALF) [50], Local Regional Fitting Information based Level Set (LRFI) [30], Multi-local Statistical Information based Level Set (MLSI) [31], Wasserstein Distance based Level Set (WDLS) [61], Relative Entropy based Level Set (RELS) [62], and GrabCut [6]. ALF uses the traditional gradient in the edge term, while LRFI and MLSI employ an improved gradient. RELS and WDLS use improved global region terms without resorting to edge descriptors. GrabCut is another classical segmentation model using the maximum flow model. The deep learning models involved are Swin UNETR [63] and Segment Anything [64]. Swin UNETR is a transformer-based model that achieved state-of-the-art results on the BraTS dataset. Segment Anything is a well-known segmentation model developed by Meta AI, recognised as one of the most fundamental models in the field of computer vision.

4.1. Comparison of Ricci curvature and gradient

Traditional gradient-based edge terms of level set models have been shown to be susceptible to low contrast and noise, which could degrade segmentation results. In this section, we demonstrate the benefits of Ricci curvature in addressing these issues.

First, we show that our discrete Ricci curvature can capture intrinsic edge information while the gradient recognises it as low contrast, namely the 0 gradient. Fig. 6(b) shows a 3D function $f = x^2 + y^2 + z^2$ whose value is described by the image intensity. Following the definition of continuous Ricci curvature, its trace is $\frac{8(4x^2+4y^2+4z^2+3)}{(4x^2+4y^2+4z^2+1)^2}$, indicating that it reaches a maximum value at the origin $(0, 0, 0)$. Fig. 6(c) shows its discrete counterparts (17), revealing that its value at the origin is the maximum of discrete Ricci curvature, coinciding with the continuous setting. Fig. 6(d) shows the norm of its gradient $(2x, 2y, 2z)$, which reaches a minimum 0 at the origin. However, as mentioned before, a zero gradient represents the particular xyz coordinates that agree with the tangent plane of the surface, which heavily relies on choices of coordinates rather than the intrinsic edge information.

Next, we demonstrate the advantage of Ricci curvature in terms of its

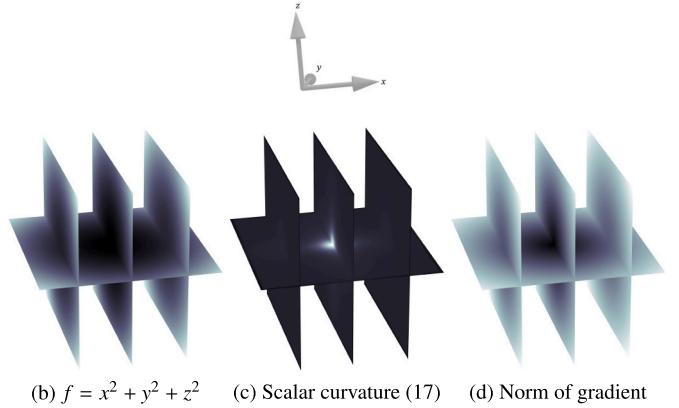


Fig. 6. The function $f = x^2 + y^2 + z^2$, its discrete Ricci curvature, and gradient. It is easy to see that the curvature reaches the maximum value at the origin. According to the Riemannian geometry, its Ricci curvature is $\frac{8(4x^2+4y^2+4z^2+3)}{(4x^2+4y^2+4z^2+1)^2}$, also having a maximum at the origin. So, the discrete curvature can capture the intrinsic curving information which does not only rely on the intensity change. By contrast, based on the intensity change, the gradient $(2x, 2y, 2z)$ reaches its minimum length at the origin. However, as mentioned, the gradient only reflects the coordinates we use to describe the image rather than the image itself.

insensitivity to noise in a real image. Fig. 7 shows a slice of a CT image, along with its curvature (17) and the length of its gradient. It is worth mentioning that the curvature may have positive or negative values, so Fig. 7(c) also displays its absolute value. To see it clearly, we focus on a subregion of it in Fig. 1. It can be observed that the Ricci curvature does not show discernible noise, while the gradient displays noticeable noise.

The two important features of Ricci curvature can significantly benefit our proposed level set model. While the level set function is

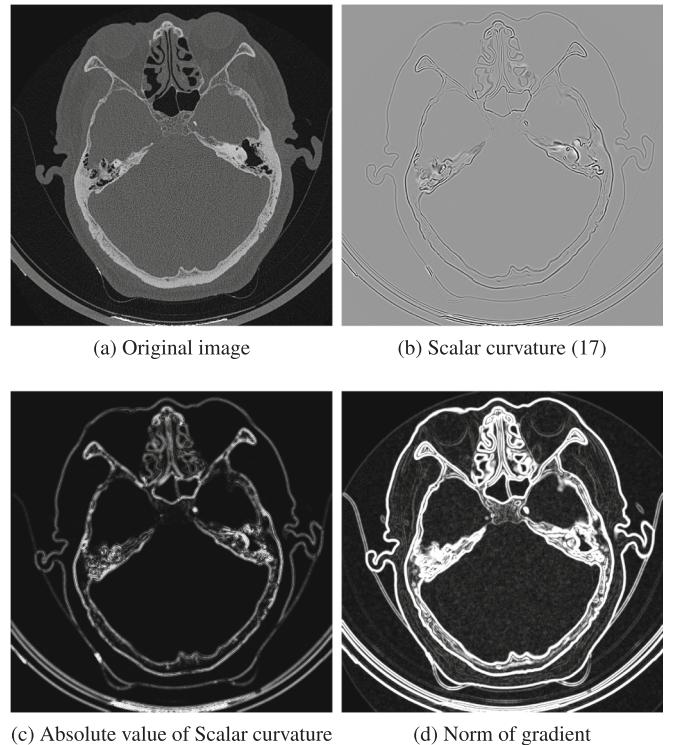


Fig. 7. One slice of a 3D image, its 3D curvature, and gradient. Since the curvature can be positive or negative, its absolute value is also displayed. It can be observed that using curvature can avoid undesired noise. This means applying the intrinsic information can mitigate the noise problem.

evolving, the edge descriptor in the edge term of the level set plays a vital role in stopping the contour when it lies in the image boundary. A regularisation term with intrinsic edge information, produced by the Ricci curvature, can be expected to drive the contour properly. By contrast, the gradient routinely produces a biased curvature estimate, which may generate spurious inward regularisation force.

4.2. Vessel

When dealing with the vessel data, we manually selected some initial points for all methods. Because LRFI and MLSI methods rely solely on edge information, we used the ground truth morphologically eroded by a kernel of size $7 \times 7 \times 7$ for initial points. For the GrabCut method, we used the same foreground seeds as that of the LRFI model, and the background seeds were all air voxels obtained using a thresholding method. For the remaining methods, which all possess region terms, we chose a significantly small region inside the vessel area to make the segmentation task challenging.

Our Ricci curvature-based model and the gradient-based model (the ALF model) utilised a completely identical framework except for the edge descriptor, and thus we used the same parameter configuration. Since making the region information as local as possible [22] used in the region term has been recognised as one of the most efficient approaches to the heterogeneous region, we should make the two parameters γ_{in} and γ_{out} related to the region term as small as possible by setting $\gamma_{\text{out}} = 2$, $\gamma_{\text{in}} = 2$. Now we give the other three parameters α , λ_1 , and λ_2 , where $\alpha > 0$ and $\lambda_1 > 0$ allow the control of the internal force, and $\lambda_2 > 0$ forces the contour to evolve outward. In practice, to keep the region force and the

regularisation force on the same order, we normalised the region force generated by λ_1 and λ_2 . Therefore, the ratio $\frac{\lambda_2}{\lambda_1}$ in fact controlled the level of dilation which was expected to be relatively high to grow the contour outward from a small initial region. On the other hand, our parameters should remain unchanged during contour evolution, so the dilation force should not be too high to expand excessively. After experimenting with varied parameters, we choose the most proper parameters: $\alpha = 0.8$ and $\frac{\lambda_2}{\lambda_1} = 1.05$.

However, 3D images usually have too many voxels to run the GrabCut completely. In the actual implementation, we divided the 3D image into 5,000,000 3D superpixels [65] and ran the GrabCut based on the graph of these superpixel nodes, as implemented in Matlab.

Fig. 8 is an example of the contour evolution, whose initial points are shown in the first image. **Fig. 9** shows the result of three different methods in this example: Our method (dubbed as Ricci in the picture), ALF (dubbed as gradient) and GrabCut. Our method is shown to outperform the gradient in accurately identifying delicate vessel branches. This can be attributed to the low contrast regions between certain vessel branches, where the regularisation force yielded by the gradient-based edge term becomes relatively large and thus prohibits further contour dilation.

A natural problem arises as to whether it is possible to tune the parameters of the gradient-based model to improve its performance. To demonstrate this, it would be instructive to enumerate all possible parameters. As mentioned before, since we should make the region information as local as possible, we do not intend to increase γ_{out} and γ_{in} . Thus, **Fig. 20** compares the best Dice's coefficient (DSC) of our model

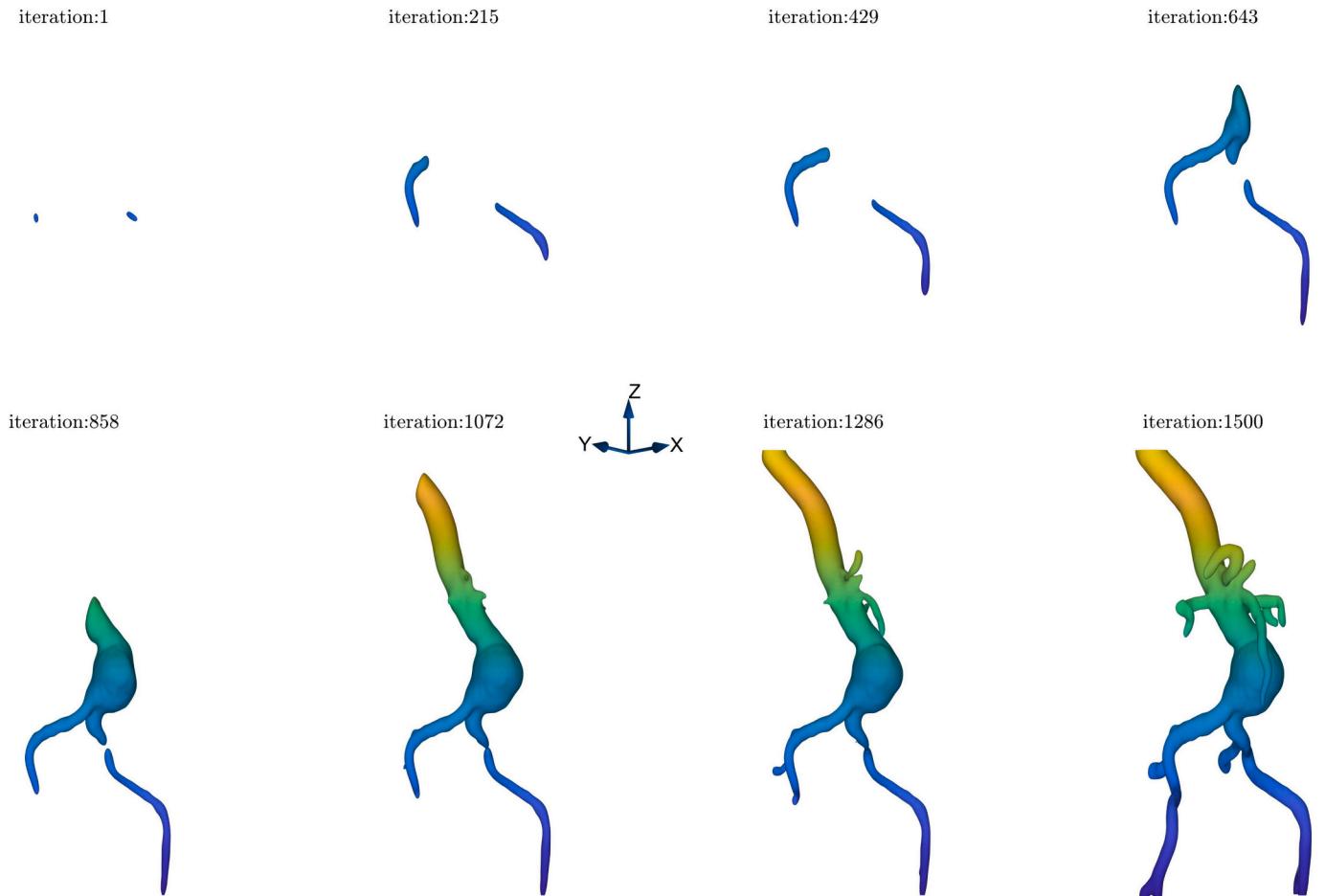


Fig. 8. The contour evolution of our algorithm in the vessel data, where the first iteration refers to the initial area. The contour starts from two small initial areas on the two narrow ends of the vessel, gradually expanding outward and eventually segmenting the whole vessel.

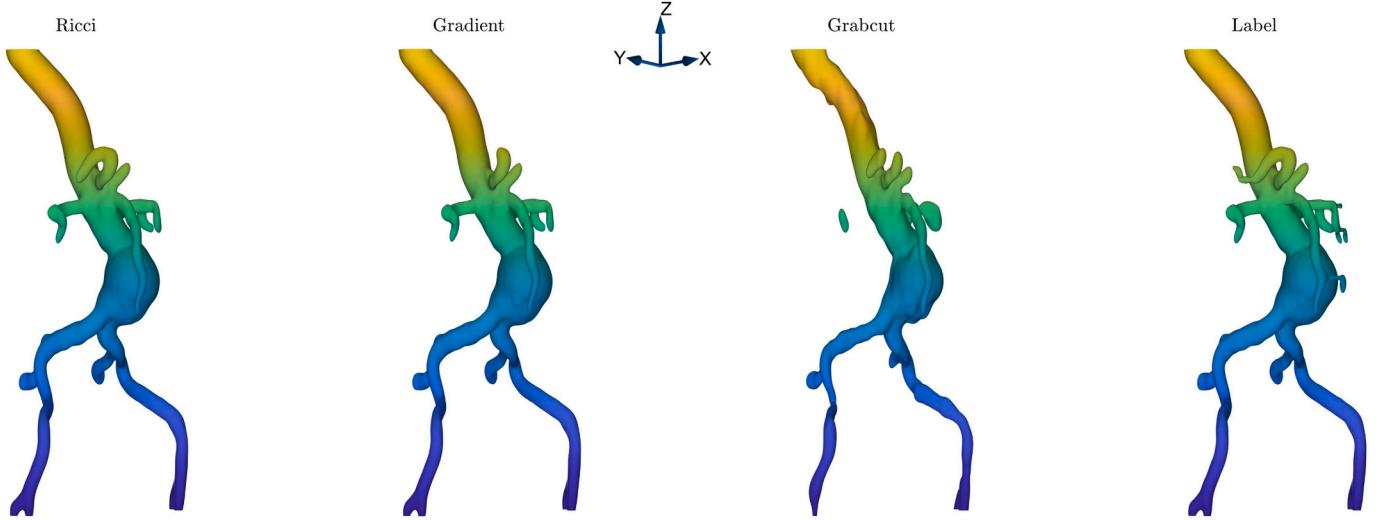


Fig. 9. The comparison of our model with the other two methods for the vessel dataset. It can be observed that our method can recognise smaller vessel branches than the gradient method.

during 2000 iterations with the gradient-based model across varied $\frac{\lambda_2}{\lambda_1}$ (dubbed as dilation) and α (dubbed as regularisation), both ranging from 0.5 to 1.5. This indicates that the maximum of our model and gradient model are 91 and 84, respectively, which implies that it is impossible to improve the gradient result by simply “fine-tuning” parameters. Although it is possible to obtain a narrow branch by growing the contour through, for instance, increasing $\frac{\lambda_2}{\lambda_1}$ or decreasing α , we should mention that the price paid for the local improvement is a global outward movement of the contour, which will give rise to much unnecessary region and decrease the DSC.

It can also be observed that GrabCut’s segmentation approach can mistakenly classify a portion of vessels as the background, resulting in fragmented vessels. This problem may stem from using 3D superpixels in GrabCut’s processing rather than using the original image’s voxels. If some superpixels contain a small part of blood vessels and a large part of the background by mistake, GrabCut might regard it as the background.

We summarise the comparison of all models in **Table 1** across 50 images using three metrics: precision, recall and dice. While LRFI and MLSI models enhance the gradient, they cannot show favourable results due to the absence of region terms in their energy function. The real CT data poses a challenge as it comprises various objects, including organs, tissues, air, and bone, making it difficult to isolate specific objects using edge information alone. Although WDLS and RELS have well-designed global region terms, they only assume the image has two regions, which is usually not satisfied in the real datasets. By contrast, our method is based on the ALF model, which boasts an edge term and local region term, which can deal with inhomogeneous images as well as weak boundaries.

4.3. Colon

For the colon dataset, the ground truth was eroded by a $7 \times 7 \times 7$ kernel and then served as the foreground seeds of LRFI, MLSI, and GrabCut. We selected all air around the body and lungs as background seeds for GrabCut. We chose a small region inside the colon as the initial contour for the other models, expecting it to evolve outward gradually. For our model and the gradient-based model, we set $\gamma_{\text{out}} = 2$, $\gamma_{\text{in}} = 1$ because the two parameters controlling the “receptive field” of the region term should be sufficiently small to handle inhomogeneous region distribution. Additionally, we set $\frac{\lambda_2}{\lambda_1} = 1.2$ to ensure that the contour started with a larger outward region force than the inward region force. Accordingly, we set $\alpha = 1$.

Table 1
Average metrics for different models.

	Precision	Recall	Dice
Vessel			
OURS	90.45 ± 1.42	88.68 ± 1.86	89.55 ± 1.71
ALF	85.96 ± 2.18	84.64 ± 2.42	85.29 ± 2.33
LRFI	83.96 ± 3.05	67.62 ± 7.40	74.90 ± 5.35
MLSI	85.97 ± 2.45	71.81 ± 6.24	78.25 ± 4.31
WDLS	83.81 ± 2.85	72.98 ± 5.66	78.02 ± 4.60
RELS	77.67 ± 4.45	76.08 ± 5.08	76.86 ± 4.77
GrabCut	80.41 ± 3.49	65.03 ± 8.68	71.90 ± 6.35
Colon			
OURS	98.22 ± 1.23	98.77 ± 1.43	98.49 ± 1.40
ALF	96.71 ± 1.61	93.17 ± 2.42	94.90 ± 1.81
LRFI	90.11 ± 2.90	87.02 ± 3.74	88.53 ± 3.47
MLSI	95.61 ± 1.68	93.86 ± 2.19	94.72 ± 1.73
WDLS	87.59 ± 3.72	85.92 ± 3.92	86.74 ± 3.85
RELS	82.46 ± 4.99	85.67 ± 4.39	84.03 ± 4.55
GrabCut	94.95 ± 2.08	87.50 ± 3.88	91.07 ± 2.62
Brain			
OURS	89.90 ± 1.37	98.23 ± 0.41	93.88 ± 1.01
ALF	77.78 ± 13.96	98.15 ± 0.11	86.78 ± 7.92
LRFI	63.89 ± 8.72	98.08 ± 0.14	77.37 ± 4.78
MLSI	75.52 ± 5.14	99.99 ± 0.03	86.04 ± 2.35
WDLS	68.14 ± 7.42	99.51 ± 0.27	80.89 ± 3.50
RELS	60.44 ± 10.30	99.27 ± 0.35	75.13 ± 5.01
GrabCut	83.00 ± 11.23	99.10 ± 0.23	90.34 ± 5.86
BraTS Necrotic Tumour Core			
OURS	86.77 ± 5.19	89.76 ± 6.68	87.32 ± 5.72
Swin UNETR	86.61 ± 5.89	82.71 ± 7.67	83.97 ± 5.55
SAM	80.03 ± 11.38	89.97 ± 4.27	82.59 ± 9.06
BTCV Kidney			
OURS	89.63 ± 7.64	97.02 ± 3.55	92.02 ± 4.32
Swin UNETR	90.09 ± 7.23	93.29 ± 2.41	91.55 ± 4.22
SAM	83.85 ± 12.86	97.15 ± 6.95	90.60 ± 9.27

Fig. 11 presents the comparison results for a sample colon, while **Fig. 10** shows our initial points (iteration 1) and detailed contour evolution. Negligible differences can be observed in **Fig. 11** because, for a normal colon, more straightforward methods, such as region growing [66], can effectively separate the colon from the image. However, difficulties arise when the small intestine adheres to the colon [67–69],

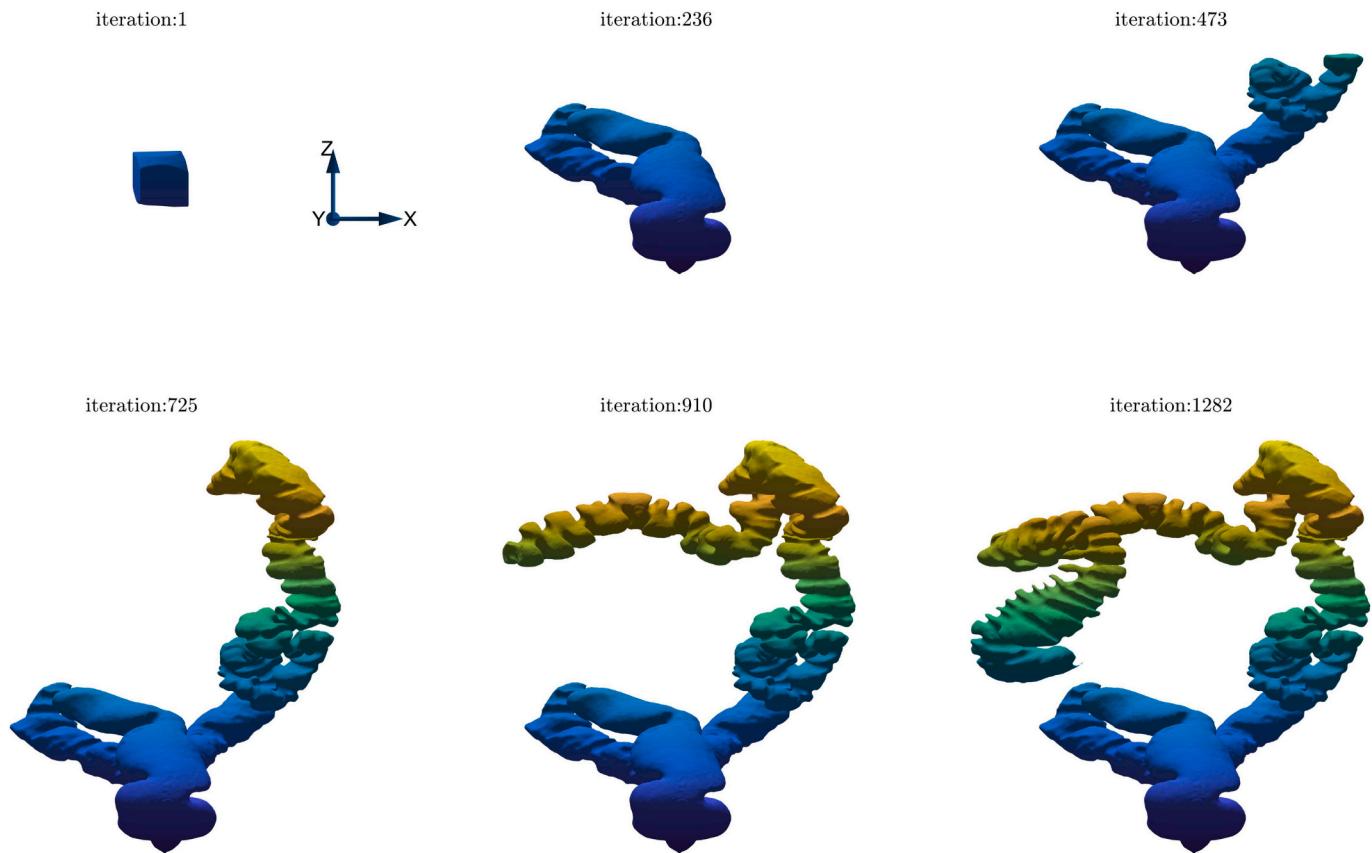


Fig. 10. The contour evolution of our algorithm in the colon data, where the first iteration refers to the initial area. The initial area is a small connected component within the colon.

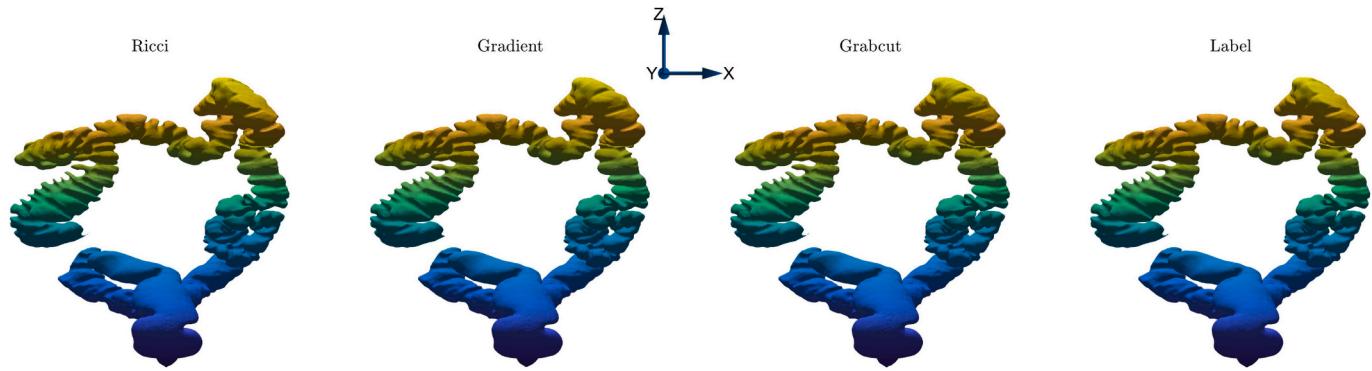


Fig. 11. The comparison for a normal colon. It can be observed that the differences between various methods are minor.

primarily due to poor CT image quality (rather than colon lesions). In Fig. 12, we present a segmentation result for an abnormal colon where distinguishing between the colon and the small intestine is challenging. Our method outperforms the other methods by achieving the best results, while the other methods wrongly consider some small intestines as colon.

This results from noise between the colon and the small intestine, where the regularisation force yielded by the gradient-based term is smaller than that of our model, which fails to be balanced against the outward force. Fig. 21 shows their respective best Dice's coefficient on this sample during 2000 iterations, where our model reaches the maximum 99 and the gradient model reaches the maximum 94. This implies that increasing regularisation will not improve the gradient-based model, since a high inward force generated from the regularisation term in combination with the internal region term will obscure the

contour dilation. Similarly, decreasing the dilation parameter $\frac{\lambda_2}{\lambda_1}$ might also contract some parts of the contour, although the other parts of the contour still expand. A trick is to distribute the initial points over the center-line of the colon and use a high regularisation force, namely, increasing α .

Table 1 compares all models across 50 cases using three metrics. It appears that many models can reach a DSC higher than 0.9. We should mention that in colon segmentation, a high Dice's coefficient (e.g., 0.9) does not reflect a reasonable segmentation result. This is because the colon is similar in appearance to a cylinder, whose radius is routinely much larger than the small intestine, which means the introduction of the small intestine cannot result in a significant decrease in the Dice's coefficient. Table 1 indicates that LRFI and MLSI perform better than the previous dataset. The rationale behind this is that the colon only consists

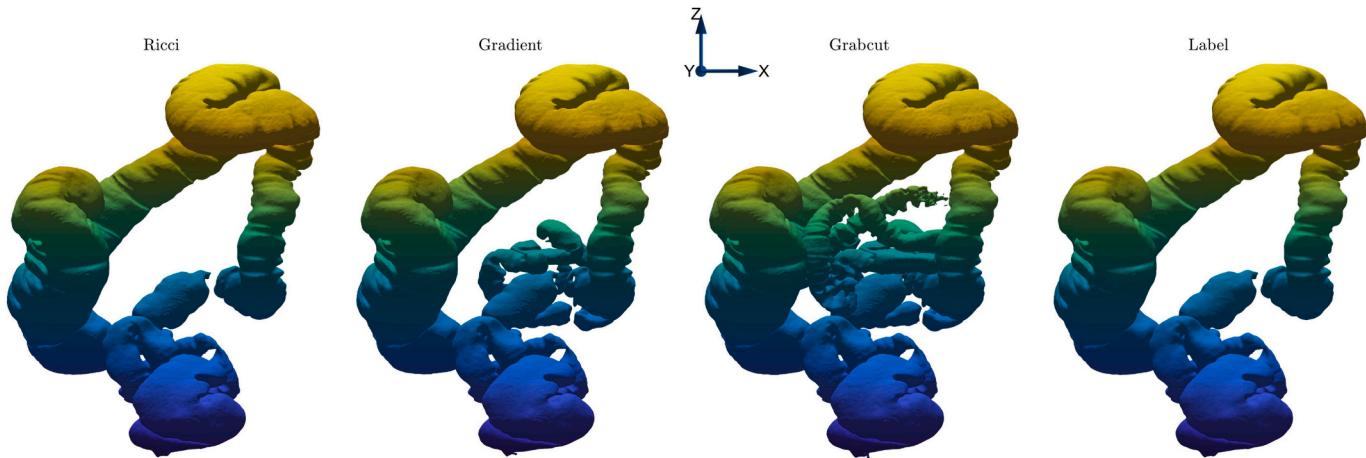


Fig. 12. The comparison for an abnormal colon where some undesired small intestine is attached to the colon. While the gradient-based model can't distinguish between the colon and small intestine, our method can achieve the best result.

of air and is surrounded by tissue, which possesses grayscale values distinct from those of the air. Thus, using an edge descriptor to identify the colon is a more appropriate approach than a global region-based method such as WDLS and RELS. The global region-based method

does not show a promising result because some objects having the same intensity as the colon (e.g., the small intestine) and the significant intensity variance beyond the colon (e.g., the lung, tissue, water, and air), cannot fulfil the assumption of the global region term. As for the

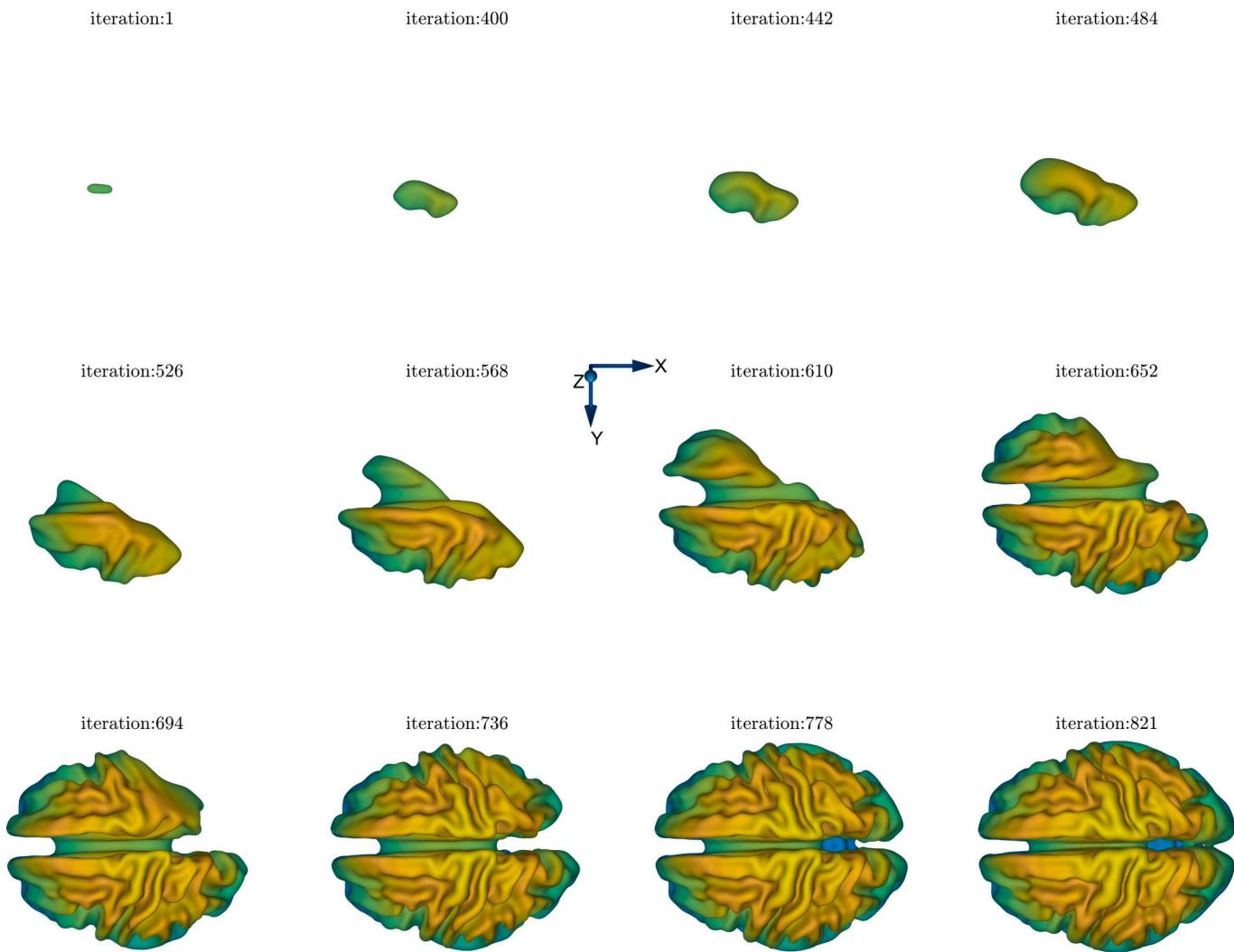


Fig. 13. The contour evolution of our algorithm in the brain data, where the first iteration refers to the initial area. The contour starts from a small part of the left brain and gradually expands to the entire brain.

GrabCut, the main reason for its unsatisfactory result is that the small intestine and colon have the same intensity, which leads to an identical superpixel.

4.4. Brain

We now turn to the segmentation of the brain. The processes for generating the foreground seeds of LRFI, MLSI, and GrabCut were identical to those of the preceding datasets. The background seeds of GrabCut were the air areas outside the brain. We chose a small initial region located in the interior of the brain for the other models. For our model and the gradient-based model, we set $\alpha = 1$, $\frac{\lambda_2}{\lambda_1} = 1.05$, $\gamma_{\text{out}} = 2$, and $\gamma_{\text{in}} = 1$.

The contour evolution of the Ricci method is illustrated in Fig. 13, where the model begins from a small area of the left brain and gradually proceeds outward from left to right until it reaches an area close to the ground truth. The results of a comparative analysis between the Ricci, gradient, and GrabCut methods for this case are shown in Fig. 14. Additionally, Fig. 22 compares the best DSC of our model on this case during 1000 iterations with the gradient-based model, where the maximum of our model and the gradient model are 96 and 68, respectively.

It is evident that the Ricci method outperforms the gradient-based model. Initially, due to $\frac{\lambda_2}{\lambda_1} = 1.05$, the level set evolution is dominated by the outward region term, which allows the contour to grow gradually. Upon reaching the vicinity of the brain boundary, the outward region force is approximately balanced against the inward region force. Once the contour proceeds to move beyond the boundary, the edge term integrating Ricci curvature will smooth the level set, pulling it back. However, the relatively high gradient cannot produce enough force to pull the level set back due to noise. Although the Gaussian filter is widely used to smooth noise, the trade-off is a smoothed gradient, which also leads to a high gradient around the boundary. Decreasing the dilation force, namely, $\frac{\lambda_2}{\lambda_1}$, will not serve our purpose, as shown in Fig. 22. A small dilation force may not ensure that the contour grows globally. In other words, some parts of the contour might stop growing and instead fluctuate. A trick is to give the gradient model a low dilation force or high regularisation and start the contour from Fig. 14, rather than our specified initial points. However, adjusting parameters during the iteration process is not allowed in our comparison.

Table 1 compares all models across 50 cases using three metrics. It demonstrates that using an improved gradient alone, like MLSI, can achieve reasonable performance since the grayscale is similar within the brain area and gradually transitions from light to dark at the boundary. Therefore, utilising an efficient edge descriptor can aid in brain segmentation. For the global region-based level set, the grayscale outside the brain is much more inhomogeneous than the intensity of the brain, which will always produce an outward force. Thus, the contour will always grow, leading to a large recall and small precision. Analogous to the preceding dataset, the failure of GrabCut lies in the superpixel's

inability to recognise fine detail around the brain boundary.

4.5. Necrotic tumour core from BraTS 2021

Our approach, similar to other classical methods, can be freely used in various situations without the need for an extensive amount of ground truths for fitting. This is exemplified in ITK-SNAP, a medical software that uses active contour to segment medical images, and 3D Slicer, a renowned software using Growcut to interactively separate a 3D medical image into the foreground and background. However, deep learning has become increasingly important in medical image segmentation. Therefore, we present a comparison of our method with Swin UNETR [63] and Segment Anything (SAM) [64] on the BraTS 2021 Task1 dataset.

The BraTS dataset consists of various multi-parametric MRI scans, which use four different parameters: T1, weighted T1, weighted T2, and T2 fluid attenuated inverse recovery. The annotation comprises three labels: necrotic tumour core (label 1), peritumoral oedematous/invasive tissue (label 2), and GD-enhancing tumour (label 4). The Swin UNETR models pre-trained on BraTS were available online at MONAI Research Contributions [70], where the model was validated through 5-fold cross-validation. Here, we utilised the model trained from the first fold and compared the three models on 50 cases of the first fold.

MONAI's specialised Swin UNETR class maintained the same default setting as in [63]. Its default input was of size $B \times 4 \times 128 \times 128 \times 128$, and it can perform sliding window inference. We first normalised the non-zero value on each channel separately of the original image of size $4 \times 240 \times 240 \times 155$, cropped it to size $4 \times 128 \times 128 \times 128$, and finally, the output of the network was a tensor of size $B \times 3 \times 128 \times 128 \times 128$. Here, the first channel of the output was the union of Label 1 and Label 4, the second channel represented all foreground, and the last channel indicated Label 4, which implied Label 1 corresponded to (1, 1, 0). Thus, we applied the sigmoid function to the output tensor and generated a binary tensor using the threshold 0.5. We then converted all points with (1, 1, 0) to 1, representing Label 1, and other points to 0, representing anything else.

Segment Anything [64] provided a more convenient Python function powerful enough to predict foreground and background using a few seeds from them, respectively. The Meta AI Research released three settings of model checkpoints: vit_h (2.4 GB), vit_l (1.2 GB), and vit_b (358 MB). In our experiment, we applied the vit_h configuration. Its input was only restricted to a 3 channel unsigned char 8 bit 2D image. Therefore, for each slice, i.e., with size $4 \times 240 \times 240 \times 1$, we first normalised the image to [0, 1] globally and converted it to [0, 255]. We found the segmentation result primarily lies in the selection of initial points, especially the background seeds, rather than which 3 channels are used. Therefore, we chose the first 3 channels as input and randomly selected 3 points within Label 1 as the foreground seeds. As for the background seeds, since putting them around the boundary of Label 1 can significantly improve the segmentation, we randomly selected 20 points within the other two labels.

We now turn to the parameter setting of our model. Following the

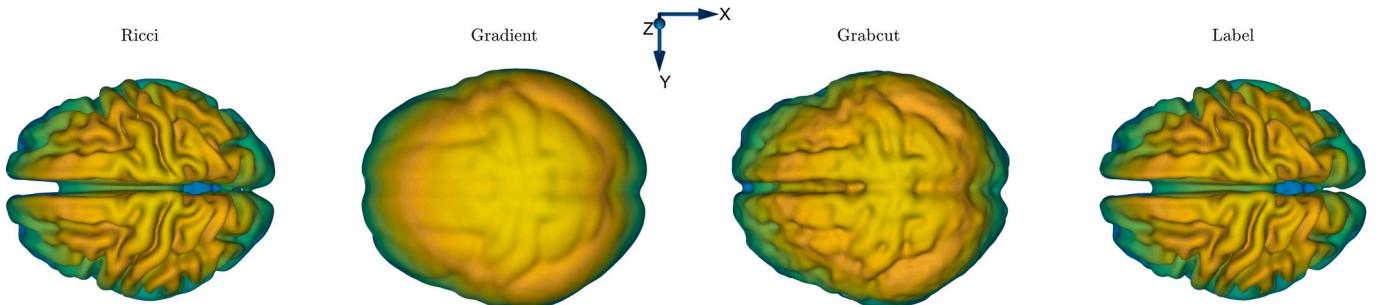


Fig. 14. The comparison for the brain data. For the gradient-based model, the unsatisfactory result is partly due to the small initial region, which dictates a high initial dilation force. In application, a larger initial region can mitigate this problem.

preceding convention, we set $\alpha = 0.8$, $\frac{\lambda_2}{\lambda_1} = 1.1$, $\gamma_{\text{out}} = 2$, $\gamma_{\text{in}} = 2$. A concern with our model was the channel of the image because we only developed our model based on a 1 channel 3D image. However, it was convenient to adjust our code to compensate for increased channels of the 3D image. In application, this meant looping through each channel using (26) and (27) within the narrow band $\gamma_{\text{in}} > \phi > -\gamma_{\text{out}}$ per loop iteration. When the curvature (17) was required, we could also take their mean. Our C++ code was packaged using Matlab's mexFunction and was compiled into a mexw64 file, which could directly run on Matlab script and handle sufficiently many channels of the input image.

Fig. 15 shows the contour evolution of our model and **Fig. 16** compares our model with the other models on the case BraTS2021_01001. Although Swin UNETR reaches the DSC 91.48, it brings numerous small broken parts. Segment Anything features only one connected component, but it does not show a promising result on the first slice of the 3D image because it is a 2D model, and the difference between foreground and background on the first slice is indiscernible. By contrast, our model does not show additional spurious parts and can handle the first slice. However, this does not mean our model cannot handle complex topology. Our model can merge two different connected components, such as **Fig. 8**, but also can split one, such as **Fig. 15**, during iteration. This results from our way of gradient descent: we track the contour as shown in **Algorithm 1** rather than update the level set globally.

Table 1 shows the comparison result on the 50 cases. It is essential to mention that most deep learning models, including Swin UNETR, are evaluated on the tumour core, which refers to the union of Label 1 (necrotic tumour core) and Label 4, rather than only the necrotic tumour core, which we use in this table. We evaluated the pre-trained model across all 251 cases of the first fold, and found that the DSC of the tumour core is 89.33, which is consistent with the original paper. However, the DSC of the necrotic tumour core is only 75.82, which might result from its small volume.

4.6. Right kidney from BTCV

The Beyond the Cranial Vault (BTCV) abdomen dataset, an important benchmark for abdominal organ segmentation tasks, comprises 30 training CT scans and 20 test scans. It includes labels for 13 anatomical structures, such as the spleen, kidneys, liver, gallbladder, and more. In

this section, we compare our model's segmentation of the right kidney with Swin UNETR and Segment Anything.

Compared to BraTS, the scale of the BTCV dataset was not large. Therefore, [71] suggested fine-tuning the Swin UNETR, which was extensively self-supervised trained on other 5 varied datasets where the ground truth was not required. To train Swin UNETR, the 30 scans with ground truth were split into 24 training scans and 6 validation scans. The model was trained using the AdamW optimiser for 30,000 epochs with the loss function as the weighted sum of the dice loss and cross-entropy loss, as detailed in [70]. The final model was available online at MONAI Model Zoo [72].

For Segment Anything, which only supported an 8-bit unsigned char data type, we converted the 3D image into a 3 channel 3D image. Specifically, noting that the kidney was within the intensity range [0, 255], we replaced values less than 0 with 0 and values larger than 255 with 255, followed by repeating the array 3 times. When constructing the foreground and background seeds, we found that the background seeds played a more critical role in the segmentation. For example, a background point far from the foreground could barely contribute to the segmentation result. Therefore, for each slice, we randomly chose 20 points within the kidney and 20 points outside the kidney but with a distance less than 10.

Our model followed a similar strategy to that of the preceding dataset. We still used the local regional information, which meant we set $\gamma_{\text{in}} = 2$ and $\gamma_{\text{out}} = -2$, and the dilation strategy, requiring a larger $\frac{\lambda_2}{\lambda_1}$ than 1 which, on the other hand, should not be too large to grow excessively. Therefore, we set $\frac{\lambda_2}{\lambda_1} = 1.1$ and $\alpha = 1$ after experimenting with varied configurations.

We compared our model with the other two models on the 6 validation cases of Swin UNETR since the test set's ground truth was unavailable. **Table 1** shows the three models' precision, recall, and DSC. For one case, **Fig. 17** shows our contour iteration, which starts from the initial points and expands gradually until it reaches the kidney boundary. **Fig. 18** visualises the three segmentation results and the ground truth. The Swin UNETR demonstrate jagged edges and protruding and concave regions. The first feature in fact makes it closer in appearance to the ground truth. The last two features, on the other hand, degrade the overall performance (DSC 93.11). By contrast, our model can address the last two issues yielding the DSC 93.71. The conclusion can be further

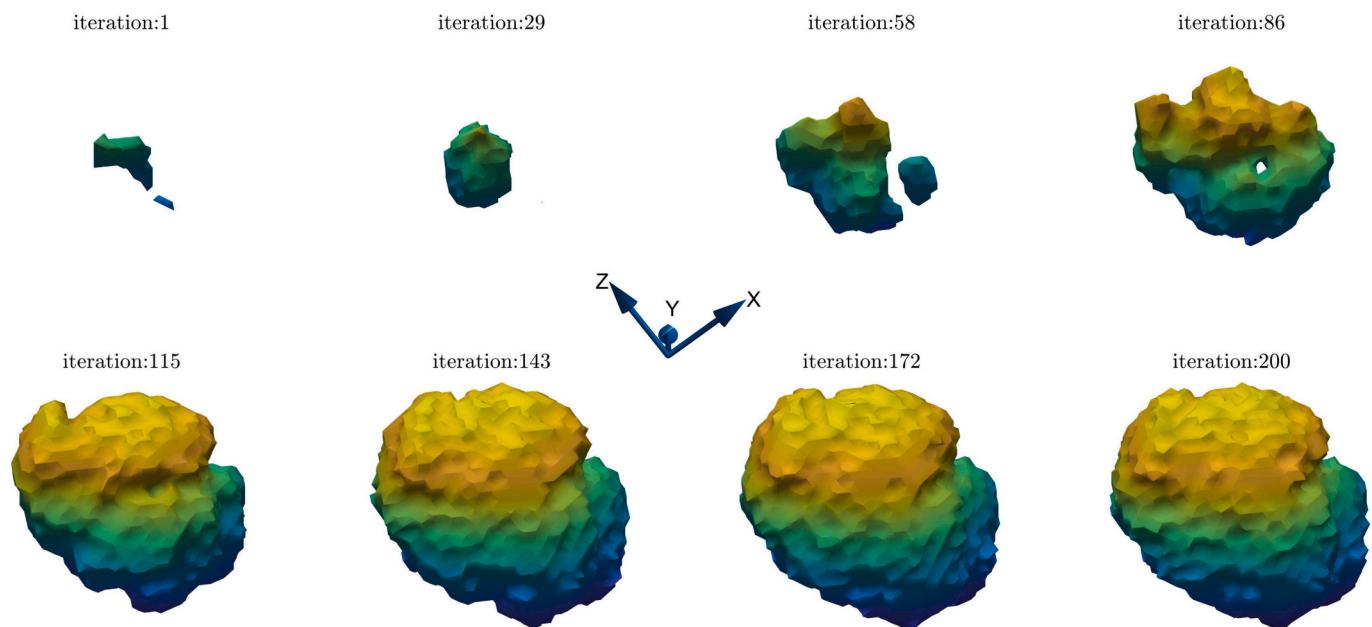


Fig. 15. The contour evolution of our algorithm in the BraTS 2021 dataset, where the first iteration refers to the initial area. The algorithm starts from a small part of label 1, and then gradually achieves a promising result.

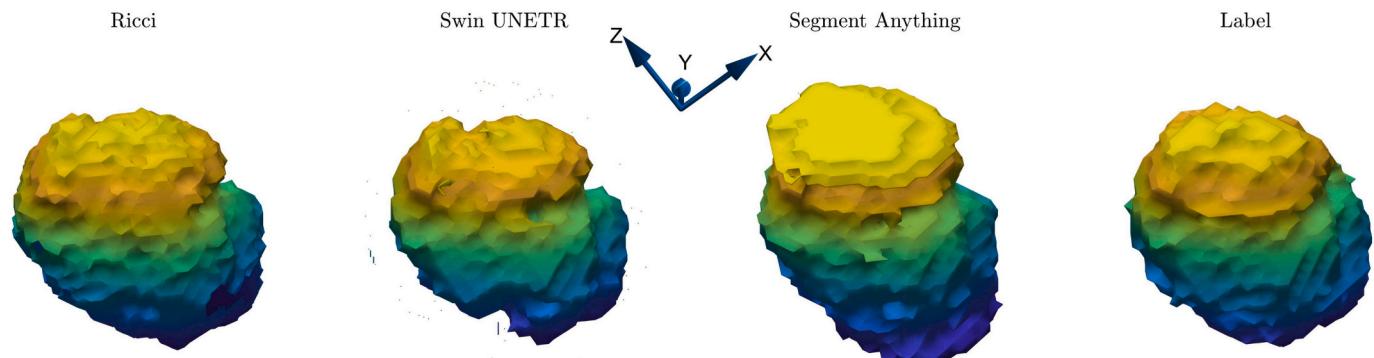


Fig. 16. The comparison on one case of the BraTS 2021 dataset. The DSC from left to right are 92.84, 91.48, and 91.00. The example is in the folder BraTS2021_01001, which is in the first fold of the 5-fold validation set of the officially pre-trained Swin UNETR model. It can be observed that although Swin UNETR can achieve a DSC of 91.48, it introduces additional broken and spurious parts. By contrast, our result has a single connected component leading to the DSC of 92.84. In addition, since Segment Anything is a 2D model, it cannot deal with some slices where the foreground and background are hard to discern, as shown in the uppermost part of its result.

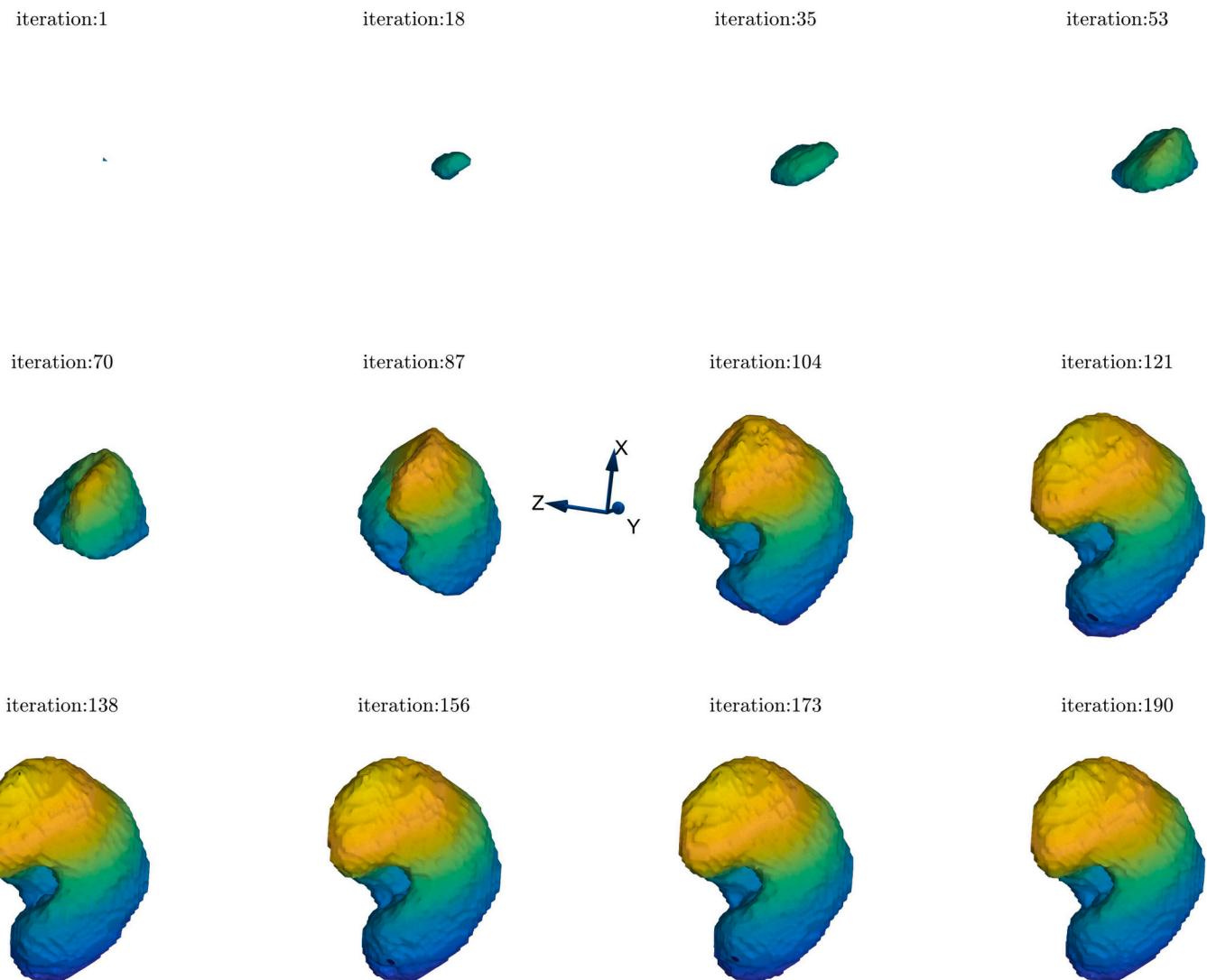


Fig. 17. The contour evolution of our algorithm for the right kidney of the case imagesTr/img0035 from the BTCV dataset, where the first iteration refers to the initial area. The algorithm starts from a small part of label 2, and then gradually achieves a promising result.

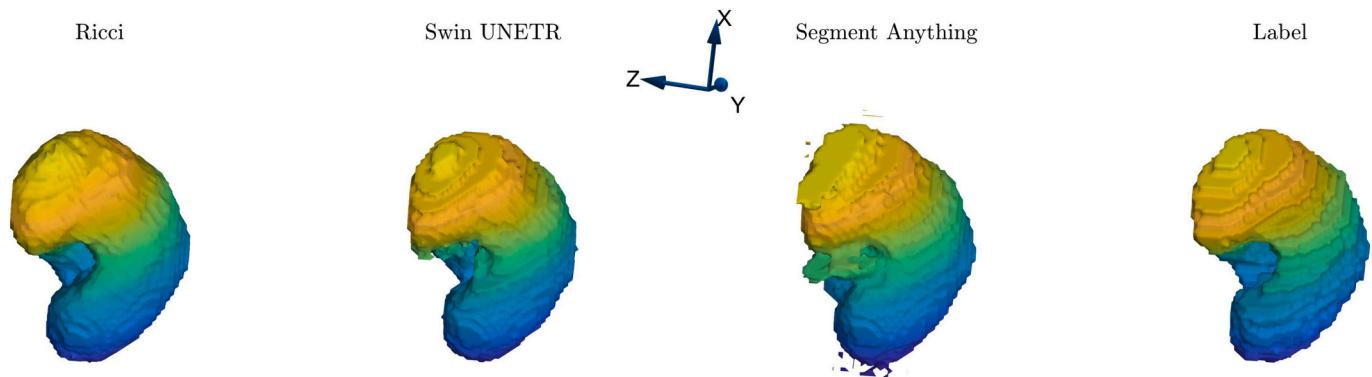


Fig. 18. The comparison on the right kidney of the case imagesTr/img0035 from the BTCV dataset. The DSC from left to right are 93.71, 93.11, and 90.01. The Swin UNETR model is fine-tuned from a self-supervised pre-trained model on 5 other datasets and is available online at the MONAI Model Zoo. The Swin UNETR produces jagged edges, which, interestingly, makes it resemble in appearance the ground truth. It also has noticeable convex(protruding) and concave regions. By contrast, our model exhibits smoothness resulting from the regularisation term. Segment Anything also shows some fragments which result from, by an analogous argument to the preceding dataset, the first as well as the last slice. The kidney differs from the preceding organs by a huge concave in the ground truth and the view of it is blocked. Thus, we partition the kidney and visualise the lower part in Fig. 19.

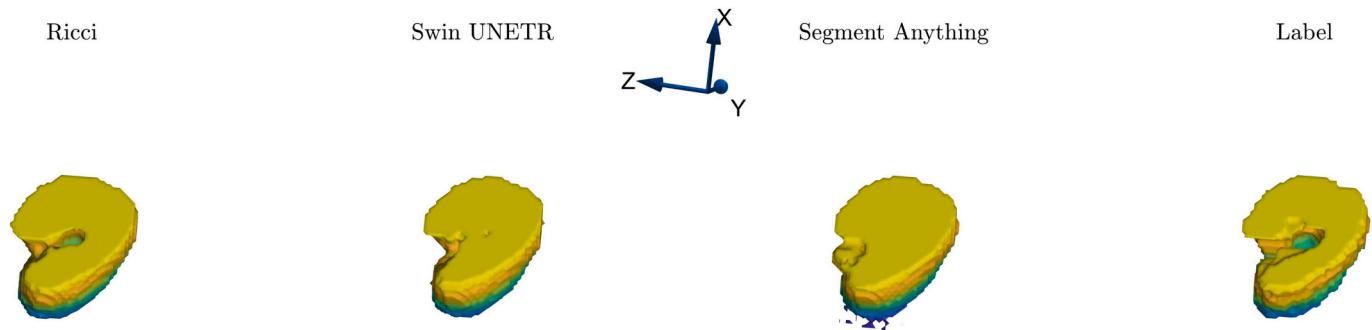


Fig. 19. Investigation of the cavity of the kidney. We notice in Fig. 18 that the view of the concave part of the kidney is blocked. To visualise it, we partition the kidney into two parts and this is the lower part. The camera position and rotation of the camera around the viewing axis are identical to the preceding figure. It can be seen that the ground truth has a hole, which is only apparent in our segmentation result.

confirmed in Fig. 19, which shows the interior of the kidney. This is not totally unexpected. The local regional information of our model only assumes the region around the boundary $2 > \phi > -2$ is homogeneous. The Ricci curvature-based regularisation can segment the organ if it is not substantially connected to the other organ whose intensity is almost the same as the ground truth. However, although Segment Anything can reach the dice 90.01, it has fragments. This is because we randomly select the foreground seeds and background seeds using Python, and some background seeds do not lie in the region corresponding to the fragments.

5. Discussion

This paper explores an intrinsic quantity, namely the Ricci curvature, that can function as an edge operator. The intrinsic property is the ability to remain invariant under coordinate and ambient transformations, which prevents a biased estimate of curvature. In contrast, the gradient might overestimate or underestimate the measure, as shown in Figs. 1 and 6.

Such inaccurate curvature estimate can lead to significant issues in image segmentation. For instance, a high $E_G(\phi)$, resulting from a small gradient typically found in low contrast regions, can generate a higher inward regularisation force than expected, obscuring critical details such as the elongated vessel in Fig. 9. Conversely, a small $E_G(\phi)$, caused by a high gradient or noise, might result in unnecessary segmentation results, as seen in Figs. 12 and 14. In contrast, the Ricci curvature-based model shows promising results, even when compared to the Swin

UNETR, as shown in Fig. 16 and Fig. 18.

Our model requires two steps for implementation. First, we calculate two auxiliary arrays, **Ric** and **w**, both in shape $(2H + 1) \times (2W + 1) \times (2D + 1)$. Here, the Ricci curvature tensor (16) **Ric** takes values only on elements corresponding to faces, and **w** stores measures on the 3D manifold $(x, y, z, f(x, y, z))$ intended for the weight function of (16). Second, we apply our proposed level set model that integrates scalar curvature (17), a function defined on each voxel, to 3D image segmentation.

In practice, we tend to dilate the level set rather than erode it. This is because the initial points can provide the level set model with the location of the object to be segmented. Otherwise, if we select an initial region much larger than the organ of interest, the model might not be able to distinguish the organ of interest from other organs that have the same intensity. In other words, the model itself cannot identify whether we need only one of them or all of them. To address this, a potential solution is to introduce prior information. For example, in the software ITK-SNAP, the contour can be driven by the force generated from a function, which is generated from user-specified labels using random forest or Gaussian mixture models. Therefore, across all datasets, we first choose a small part of the ground truth, as shown in Figs. 8, 10, 13, 15, and 17, and then make the outward region force larger than the inward region force. The model will then achieve a balance between the inner regularisation force and the outer region force when it lies on the object's edge.

A comparable segmentation result cannot be achieved by simply fine-tuning the parameters of the gradient-based model, as shown in

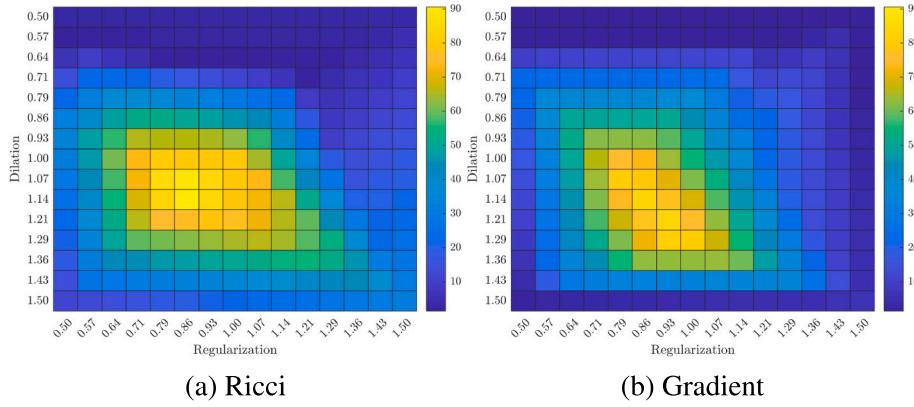


Fig. 20. The DSC of the Ricci curvature-based model and the gradient-based model for the vessel. The maximum DSC of the left hand and right image are 90.55 and 84.21, respectively.

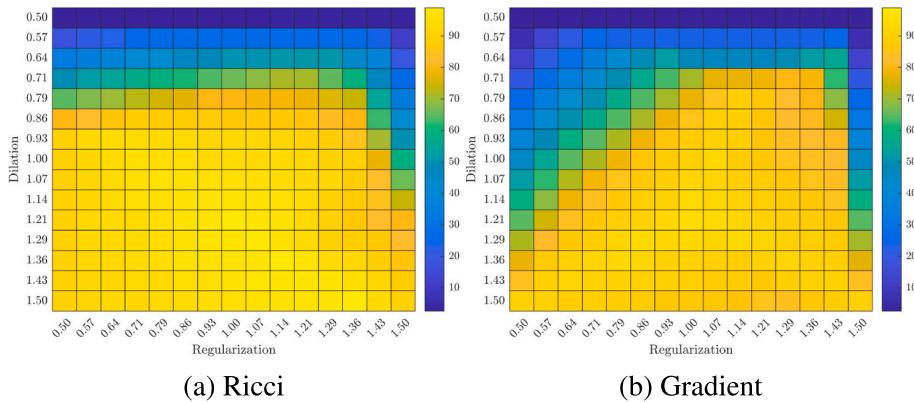


Fig. 21. The DSC of the Ricci curvature-based model and the gradient-based model for the colon. The maximum DSC of the left and right images are 98.97 and 94.32, respectively.

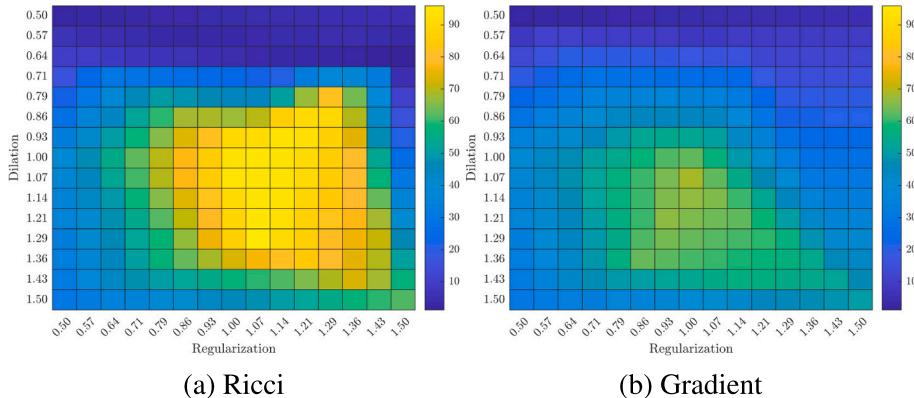


Fig. 22. The DSC of the Ricci curvature-based model and the gradient-based model for the brain dataset. The maximum DSC of the left and right images are 95.96 and 68.47, respectively.

Figs. 22, 21, and 20. This is primarily due to our small initial region and fixed hyperparameters. However, this does not mean the gradient-based model cannot be further improved. In fact, a graphical user interface (GUI) can bring significant improvement. For example, when dealing with missing delicate branches, experienced radiologists would split the whole 3D image into several regions of interest (ROIs), and use a separate set of parameters to segment each ROI independently using the gradient-based level set model in ITK-SNAP software. When removing the small intestine, one of the best approaches is through post-processing techniques. The user can specify one point of a triangle

known to be located at the small intestine, and then remove the connected component containing the point, adjusting the threshold or morphology erosion if necessary, as with the Colon VCAR software by GE HealthCare. As for brain segmentation, we can specify a larger initial region closer to the ground truth than ours.

In addition to its application to the level set model, the proposed Ricci curvature has considerable potential to act as a more general edge descriptor in the field of image processing. Its traditional counterparts, including the famous Canny and Sobel operators, have been frequently used in image software. Our proposed Ricci curvature can not only

preserve the same order of linear computational complexity but also remain independent of the choices of coordinates and space used to describe the image.

However, the Ricci curvature as an edge descriptor is not its most powerful application. A natural way to introduce Ricci curvature is to emerge from a gradient of a form of designed energy (loss function or action). For example, the gradient of the Einstein–Hilbert action $\int R \sqrt{\det(\mathbf{g})} dx dy dz$, where R (17) is the trace of the Ricci curvature \mathbf{Ric} (16), which we have used to replace the length of the gradient in (24), and \mathbf{g} is the metric tensor as in (18), has proved to boast the gradient $\frac{\partial \mathbf{g}}{\partial t} = -\mathbf{Ric} + \frac{1}{2}R\mathbf{g}$. Another example is Hamilton’s Ricci flow, probably the most renowned curvature flow across a variety of fields, which possesses another gradient configuration $\frac{\partial \mathbf{g}}{\partial t} = -\mathbf{Ric}$, and leads to a successful practical application in Computational Conformal Geometry [73]. In other words, if we intend to introduce the energy into, for instance, deep learning, we shall update the matrix \mathbf{g} (18) in this way, and recover the original 3D image $f(x,y,z)$ from the new metric tensor. Note that, discrete Forman-Ricci curvature has already been employed to this end [74].

Unfortunately, recovering a surface from the metric tensor is generally ill-posed, which can be evident in the fact that, if we cut a cylinder along a generator, we may unroll it onto a plane without changing the pairwise geodesic distances between points.

One way to deal with this is to recover certain predefined surfaces from the metric. For example, from the uniformization theorem [75] that any 2D surface with genus greater than 1 is equivalent to a unit disk, it follows that recovering a surface from a given metric is equivalent to constructing a unit disk from numerous triangles with given lengths, as shown in the discrete Ricci flow in [73].

However, this does not seem to extend to a general 3D image application, and thus our future work aims to adopt an alternative approach. We find it seems feasible to derive the Euler–Lagrange equation with respect to, instead of the metric \mathbf{g} (18), another variable, followed by a reconstruction of the 3D image $f(x,y,z)$ from this variable.

Appendix A. Continuous Ricci curvature

Given a manifold M of n dimension, the Ricci curvature tensor \mathbf{Ric} is an extension of the Gaussian curvature [52,53], namely, a $n \times n$ matrix at each point, defined as the contraction of the Riemann curvature tensor

$$\mathbf{Ric}(X, Y) = \text{trace}(Z \mapsto R(X, Z)Y) \quad (\text{A.1})$$

where X , Y and Z are three tangent vectors in M . Here, $R(X, Z)Y$ represents the renowned Riemann curvature tensor, a $n \times n \times n \times n$ tensor at each point, possessing the form

$$R(X, Y)Z = \nabla_X \nabla_Y Z - \nabla_Y \nabla_X Z - \nabla_{[X,Y]} Z \quad (\text{A.2})$$

with the $\nabla_X Y$ representing the “directional derivative” of Y with respect to X .

For the most widely used manifold, namely, Euclidean space, the Riemann curvature tensor (A.2) becomes 0 identically, which lies in the fact that different directional derivatives can be arbitrarily interchanged, such as $\partial_x \partial_y = \partial_y \partial_x$. Therefore, (A.2) measures the degree to which the image differs locally from the Euclidean space. Since the Euclidean space is flat, the difference from Euclidean space prohibits smooth space, which implies the emergence of edges. Therefore, edges in images are intrinsically associated with curvature.

Riemann curvature tensor, in fact, unifies all operators that measure the degree of curving independent of coordinate transformations, including the typical translation and rotation, and ambient space transformations, compared to other widely used operators, such as gradient, which could be determined by the choice of coordinates considered.

In the particular 3D image application, it has been shown that the Riemann curvature tensor depends linearly on the Ricci curvature tensor. Thus, the discrete Ricci curvature should suffice for the 3D image processing.

Appendix B. Formal curvature function

The Forman curvature function is designated for the cell complex, in which, for example, a point is a 0-cell, and an edge is a 1-cell. Before illustrating the curvature function, the relationships of various cells are required to comprehend the formulation. we write $\alpha < \beta$ or $\beta > \alpha$ if

6. Conclusion

The paper presents two notable contributions. Firstly, we introduce an energy function that incorporates the Ricci curvature term. This addition enhances the performance of the level set model used for medical image segmentation. Secondly, we propose a novel method for computing the Ricci curvature tensor. This method effectively detects intrinsic edges in 3D images independent of the choice of coordinates. Our experimental results demonstrate that the Ricci curvature-based level set model surpasses the previous gradient-based level set model in terms of segmentation accuracy and robustness.

CRediT authorship contribution statement

Na Lei: Resources. **Jisui Huang:** Software. **Ke Chen:** Writing – original draft, Formal analysis. **Yuxue Ren:** Data curation. **Emil Sau-can:** Writing – original draft, Conceptualization. **Zhenchang Wang:** Funding acquisition. **Yuanyuan Shang:** Writing – original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Acknowledgments

This research was supported by National Key R&D Program of China 2020YFA0712203, 2021YFA1003003; the National Natural Science Foundation of China 61936002, T2225012; GIF Grant No. I-1514-304.6/2019; the program of China Scholarships Council 202307300007, Beijing Natural Sciences Foundation Z210003.

α is contained in β 's boundaries. If α_1, α_2 are p -cells, α_1 and α_2 are neighbours if

1. α_1 and α_2 share a $(p+1)$ -cell. That is, there is a $(p+1)$ -cell β with $\beta > \alpha_1$ and $\beta > \alpha_2$, or
2. α_1 and α_2 share a $(p-1)$ -cell. That is, there is a $(p-1)$ -cell γ with $\gamma < \alpha_1$ and $\gamma < \alpha_2$.

We say that α_1 and α_2 are parallel neighbours, or $\alpha_1 \parallel \alpha_2$, if either 1) or 2) is true but not both.

Based on these relationships, the curvature function $\mathcal{F} : \{1\text{-cells}\} \mapsto \mathbb{R}$ for a 1-cell α is defined as

$$\mathcal{F}(\alpha) = w_\alpha \left[\left(\sum_{\substack{\beta^{(p+1)} \\ \beta^{(p+1)} > \alpha}} \frac{w_\alpha w_\beta}{w_\beta} + \sum_{\substack{\gamma^{(p-1)} \\ \gamma^{(p-1)} < \alpha}} \frac{w_\alpha w_\gamma}{w_\gamma} \right) - \sum_{\substack{\tilde{\alpha}^{(p+1)} \\ \tilde{\alpha}^{(p+1)} > \alpha}} \frac{\sqrt{(w_\alpha w_{\tilde{\alpha}})}}{w_{\tilde{\alpha}}} \right] \quad (\text{B.1})$$

where \parallel denotes parallel relationship, $\sum_{\substack{\beta^{(p+1)} > \alpha \\ \beta^{(p+1)} > \tilde{\alpha}}} w_\beta$ denotes the sum over all 2-cells β containing both boundaries α and $\tilde{\alpha}$, and $\sum_{\substack{\gamma^{(p-1)} < \alpha \\ \gamma^{(p-1)} < \tilde{\alpha}}} w_\gamma$ denotes all 0-cells γ , which are boundaries of both α and $\tilde{\alpha}$.

References

- [1] J. Ma, X. Yang, Automatic dental root cbct image segmentation based on cnn and level set method, in: Medical Imaging 2019: Image Processing vol. 10949, International Society for Optics and Photonics, 2019, p. 109492N.
- [2] Z.-Q. Zhao, H. Glotin, Z. Xie, J. Gao, X. Wu, Cooperative sparse representation in two opposite directions for semi-supervised image annotation, IEEE Trans. Image Process. 21 (9) (2012) 4218–4231.
- [3] D.-S. Huang, Systematic theory of neural networks for pattern recognition, in: Publishing House of Electronic Industry of China, Beijing 201, 1996.
- [4] L.A. Vese, C. Le Guyader, Variational Methods in Image Processing, CRC Press, Boca Raton, FL, 2016.
- [5] B.V. Kumar, S. Sabareeswaran, G. Madumitha, A decennary survey on artificial intelligence methods for image segmentation, in: Advanced Engineering Optimization through Intelligent Techniques, Springer, 2020, pp. 291–311.
- [6] C. Rother, V. Kolmogorov, A. Blake, “Grabcut” interactive foreground extraction using iterated graph cuts, ACM Trans. Graph. (TOG) 23 (3) (2004) 309–314.
- [7] N.A. Valente, Z. Mao, C. Niezrecki, Holistically nested edge detection and particle filtering for subtle vibration extraction, Mech. Syst. Signal Process. 204 (2023) 110753, <https://doi.org/10.1016/j.ymssp.2023.110753>.
- [8] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
- [9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, arXiv (2020) preprint arXiv: 2010.11929.
- [10] Z. Zhou, M.M.R. Siddiquee, N. Tajbakhsh, J. Liang, Unet++: redesigning skip connections to exploit multiscale features in image segmentation, IEEE Trans. Med. Imaging 39 (6) (2019) 1856–1867.
- [11] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18, Springer, 2015, pp. 234–241.
- [12] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, H.R. Roth, D. Xu, Unetr: Transformers for 3d medical image segmentation, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 574–584.
- [13] S. Bakas, M. Reyes, A. Jakab, S. Bauer, M. Rempfler, A. Crimi, R.T. Shinohara, C. Berger, S.M. Ha, M. Rozycki, et al., Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge, arXiv (2018) preprint arXiv:1811.02629.
- [14] F. Isensee, P.F. Jäger, P.M. Full, P. Vollrath, K.H. Maier-Hein, nnu-net for brain tumor segmentation, in: Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries: 6th International Workshop, BrainLes 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4, 2020, Revised Selected Papers, Part II 6, Springer, 2021, pp. 118–132.
- [15] A.L. Simpson, M. Antonelli, S. Bakas, M. Bilello, K. Farahani, B. Van Ginneken, A. Kopp-Schneider, B.A. Landman, G. Litjens, B. Menze, et al., A large annotated medical image dataset for the development and evaluation of segmentation algorithms, arXiv (2019) preprint arXiv:1902.09063.
- [16] B.H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al., The multimodal brain tumor image segmentation benchmark (brats), IEEE Trans. Med. Imaging 34 (10) (2014) 1993–2024.
- [17] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle, et al., The cancer imaging archive (tcia): maintaining and operating a public information repository, J. Digit. Imaging 26 (2013) 1045–1057.
- [18] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, J. Finet, J.-C. Fillion-Robin, S. Pujol, C. Bauer, D. Jennings, F. Fennessy, M. Sonka, et al., 3d slicer as an image computing platform for the quantitative imaging network, Magn. Reson. Imaging 30 (9) (2012) 1323–1341.
- [19] X. Li, H. Xiong, X. Li, X. Wu, X. Zhang, J. Liu, J. Bian, D. Dou, Interpretable deep learning: interpretation, interpretability, trustworthiness, and beyond, Knowl. Inf. Syst. 64 (12) (2022) 3197–3234.
- [20] Q. Teng, Z. Liu, Y. Song, K. Han, Y. Lu, A survey on the interpretability of deep learning in medical diagnosis, Multimedia Systems 28 (6) (2022) 2335–2355.
- [21] T.F. Chan, L.A. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2) (2001) 266–277.
- [22] X.-F. Wang, D.-S. Huang, H. Xu, An efficient local chan–vese model for image segmentation, Pattern Recogn. 43 (3) (2010) 603–618.
- [23] C. Li, C.-Y. Kao, J.C. Gore, Z. Ding, Implicit active contours driven by local binary fitting energy, in: 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2007, pp. 1–7.
- [24] N. Li, F. Liu, J. Li, J. Yang, T. Zheng, A variational method using riemannian metric for sar image segmentation, in: 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), IEEE, 2017, pp. 787–790.
- [25] C. Lenglet, M. Rousson, R. Deriche, O. Faugeras, S. Lehericy, K. Ugurbil, A riemannian approach to diffusion tensor images segmentation, in: Biennial International Conference on Information Processing in Medical Imaging, Springer, 2005, pp. 591–602.
- [26] V. Caselles, R. Kimmel, G. Sapiro, Geodesic active contours, Int. J. Comput. Vis. 22 (1) (1997) 61–79.
- [27] C. Li, C. Xu, C. Gui, M.D. Fox, Distance regularized level set evolution and its application to image segmentation, IEEE Trans. Image Process. 19 (12) (2010) 3243–3254.
- [28] Y. Su, K. He, D. Wang, T. Peng, An improved level set method on the multiscale edges, Symmetry 12 (10) (2020) 1650.
- [29] H. Yu, F. He, Y. Pan, A novel segmentation model for medical images with intensity inhomogeneity based on adaptive perturbation, Multimed. Tools Appl. 78 (9) (2019) 11779–11798.
- [30] C. Liu, W. Liu, W. Xing, An improved edge-based level set method combining local regional fitting information for noisy image segmentation, Signal Process. 130 (2017) 12–21.
- [31] C. Liu, W. Liu, W. Xing, A weighted edge-based level set method based on multi-local statistical information for noisy image segmentation, J. Vis. Commun. Image Represent. 59 (2019) 89–107.
- [32] W. Zhu, X.-C. Tai, T. Chan, Image segmentation using euler's elastica as the regularization, J. Sci. Comput. 57 (2) (2013) 414–438.
- [33] Tan Lu, Li Ling, Liu Wanquan, Sun Jie, Zhang Min, A novel euler's elastica-based segmentation approach for noisy images using the progressive hedging algorithm, J. Math. Imag. Vision 62 (1) (2020) 98–119.
- [34] T. Lu, L. Wanquan, L. Ling, P. Zhenkuan, A fast computational approach for illusory contour reconstruction, Multimed. Tools Appl. (2018).
- [35] L. Tan, W. Liu, Z. Pan, Color image restoration and inpainting via multi-channel total curvature, Appl. Math. Model. 61 (2018) 280–299, <https://doi.org/10.1016/j.apm.2018.04.017>.
- [36] X. He, W. Zhu, X.-C. Tai, Segmentation by elastica energy with l1 and l2 curvatures: a performance comparison, Numer. Math. Theory Methods Appl. 12 (06) (2019).
- [37] L.-J. Deng, R. Glowinski, X.-C. Tai, A new operator splitting method for the euler elastica model for image smoothing, SIAM J. Imag. Sci. 12 (2) (2019) 1190–1230.

- [38] Q. Zhong, Y. Li, Y. Yang, Y. Duan, Minimizing discrete total curvature for image processing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9474–9482.
- [39] W. Zhu, T. Chan, Image denoising using mean curvature of image surface, *SIAM J. Imag. Sci.* 5 (1) (2012) 1–32.
- [40] H. Liu, X.-C. Tai, R. Glowinski, An operator-splitting method for the gaussian curvature regularization model with applications to surface smoothing and imaging, *SIAM J. Sci. Comput.* 44 (2) (2022) A935–A963.
- [41] C. Brito-Loeza, K. Chen, V. Uc-Cetina, Image denoising using the gaussian curvature of the image surface, *Numer. Methods Partial Different. Eq.* 32 (3) (2016) 1066–1089.
- [42] M. Arif, N. Badshah, T.A. Khan, A. Ullah, H. Rabbani, H. Atta, N. Begum, A new gaussian curvature of the image surface based variational model for haze or fog removal, *PLoS One* 18 (3) (2023) e0282568.
- [43] Y. Chen, G. Courbebaisse, J. Yu, D. Lu, F. Ge, A method for giant aneurysm segmentation using euler's elastica, *Biomed. Sign. Proc. Control* 62 (2020) 102111.
- [44] S.W. Hawking, G.F. Ellis, *The Large Scale Structure of Space-Time*, Cambridge University Press, 2023.
- [45] Y. Ollivier, Ricci curvature of markov chains on metric spaces, *J. Funct. Anal.* 256 (3) (2009) 810–864.
- [46] R. Forman, Bochner's method for cell complexes and combinatorial ricci curvature, *Discret. Comput. Geom.* 29 (3) (2003) 323–374.
- [47] E. Saucan, E. Appleboim, G. Wolanski, Y. Zeevi, Combinatorial ricci curvature for image processing, in: MICCAI 2008 Workshop Manifolds in Medical Imaging: Metrics, Learning and Beyond, Citeseer, 2008.
- [48] E. Saucan, G. Wolanski, E. Appleboim, Y.Y. Zeevi, Combinatorial ricci curvature and laplacians for image processing, in: 2009 2nd International Congress on Image and Signal Processing, IEEE, 2009, pp. 1–6.
- [49] V. Barkanass, W. Chen, N. Lei, E. Saucan, Geometric graph measures for textures classification, 2023.
- [50] J. Zhang, K. Chen, D.A. Gould, A fast algorithm for automatic segmentation and extraction of a single object by active surfaces, *Int. J. Comput. Math.* 92 (6) (2015) 1251–1274.
- [51] P. Petersen, *Riemannian Geometry* vol. 171, Springer, 2006.
- [52] M.P. Do Carmo, J. Flaherty Francis, *Riemannian Geometry* Vol. 6, Springer, 1992.
- [53] H.W. Guggenheimer, *Differential Geometry*, Courier Corporation, 2012.
- [54] C. Li, C. Xu, C. Gui, M.D. Fox, Level set evolution without re-initialization: a new variational formulation, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) Vol. 1, IEEE, 2005, pp. 430–436.
- [55] K. Smith, K. Clark, W. Bennett, T. Nolan, J. Kirby, M. Wolfsberger, J. Moulton, B. Vendt, J. Freymann, Data from ct colonography. the cancer imaging archive, 2015.
- [56] C.D. Johnson, M.-H. Chen, A.Y. Toledano, J.P. Heiken, A. Dachman, M.D. Kuo, C. O. Menias, B. Siewert, J.I. Cheema, R.G. Obregon, et al., Accuracy of ct colonography for detection of large adenomas and cancers, *N. Engl. J. Med.* 359 (12) (2008) 1207–1217.
- [57] U. Baid, S. Ghodasara, S. Mohan, M. Bilello, E. Calabrese, E. Colak, K. Farahani, J. Kalpathy-Cramer, F.C. Kitamura, S. Pati, et al., The rsna-asnr-miccai brats 2021 benchmark on brain tumor segmentation and radiogenomic classification, arXiv (2021) preprint arXiv:2107.02314.
- [58] S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycski, J.S. Kirby, J.B. Freymann, K. Farahani, C. Davatzikos, Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features, *Sci. Data* 4 (1) (2017) 1–13.
- [59] Z. Xu, C.P. Lee, M.P. Heinrich, M. Modat, D. Rueckert, S. Ourselin, R.G. Abramson, B.A. Landman, Evaluation of six registration methods for the human abdomen on clinically acquired ct, *IEEE Trans. Biomed. Eng.* 63 (8) (2016) 1563–1572.
- [60] B. Landman, Z. Xu, J. Iglesias, M. Styner, T. Langerak, A. Klein, Miccai multi-atlas labeling beyond the cranial vault—workshop and challenge, in: Proc. MICCAI Multi-Atlas Labeling Beyond Cranial Vault—Workshop Challenge Vol. 5, 2015, p. 12.
- [61] K. Ni, X. Bresson, T. Chan, S. Esedoglu, Local histogram based segmentation using the wasserstein distance, *Int. J. Comput. Vis.* 84 (1) (2009) 97–111.
- [62] Y. Song, Y. Wu, Y. Dai, A new active contour remote sensing river image segmentation algorithm inspired from the cross entropy, *Digit. Signal Process.* 48 (2016) 322–332.
- [63] A. Hatamizadeh, V. Nath, Y. Tang, D. Yang, H.R. Roth, D. Xu, Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images, in: International MICCAI Brainlesion Workshop, Springer, 2021, pp. 272–284.
- [64] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A.C. Berg, W.-Y. Lo, et al., Segment anything, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 4015–4026.
- [65] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, S. Süsstrunk, Slic superpixels compared to state-of-the-art superpixel methods, *IEEE Trans. Pattern Anal. Mach. Intell.* 34 (11) (2012) 2274–2282.
- [66] H. Li, P. Santiago, Automatic colon segmentation with dual scan ct colonography, *J. Digit. Imaging* 18 (2005) 42–54.
- [67] L. Lu, J. Zhao, An automatic method for colon segmentation in virtual colonoscopy, in: 2011 4th International Conference on Biomedical Engineering and Informatics (BMEI) vol. 1, IEEE, 2011, pp. 105–108.
- [68] H. Frimmel, J. Nappi, H. Yoshida, Centerline-based colon segmentation for ct colonography, *Med. Phys.* 32 (8) (2005) 2665–2672.
- [69] L. Lu, D. Zhang, L. Li, J. Zhao, Fully automated colon segmentation for the computation of complete colon centerline in virtual colonoscopy, *IEEE Trans. Biomed. Eng.* 59 (4) (2011) 996–1004.
- [70] Project-MONAI, Research-contributions/swinunetr at main project-monai/research-contributions, 2024. URL, <https://github.com/Project-MONAI/research-contributions/tree/main/SwinUNETR>.
- [71] Y. Tang, D. Yang, W. Li, H.R. Roth, B. Landman, D. Xu, V. Nath, A. Hatamizadeh, Self-supervised pre-training of swin transformers for 3d medical image analysis, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 20730–20740.
- [72] Project-MONAI, Link, URL, <https://monai.io/model-zoo.html>, 2024.
- [73] M. Jin, X. Gu, Y. He, Y. Wang, Conformal geometry, in: Computational Algorithms, 2018.
- [74] E. Appleboim, E. Saucan, Y.Y. Zeevi, Ricci curvature and flow for image denoising and super-resolution, in: 2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO), IEEE, 2012, pp. 2743–2747.
- [75] W. Abikoff, The uniformization theorem, *Am. Math. Mon.* 88 (8) (1981) 574–592.