

# 3D Manifold Topology Based Medical Image Data Augmentation

**Jisui Huang**

2200501032@CNU.EDU.CN

*Beijing Advanced Innovation Center for Imaging Theory and Technology, Capital Normal University, Beijing*

**Na Lei** (corresponding author)

NALEI@DLUT.EDU.CN

*Key Laboratory for Ubiquitous Network and Service Software of Liaoning Province, Dalian University of Technology, Dalian*

**Editors:** Emtiyaz Khan and Mehmet Gönen

## Abstract

Data augmentation is an effective and universal technique for improving the generalization performance of deep neural networks. Current data augmentation implementations usually involve geometric and photometric transformations. However, none of them considers the topological information in images, which is an important global invariant of the three-dimensional manifold. In our implementation, we design a novel method that finds the generator of the first homology group, i.e. closed loops cannot shrink to a point, of 3D image and erases the bounding box of a random loop. To the best of our knowledge, it is the first time that data augmentation based on the first homology group of the three-dimensional image is applied in medical image augmentation. Our numerical experiments demonstrate that the proposed approach outperforms the state-of-the-art method.

**Keywords:** Data Augmentation, Topology, Homology, Medical Image Segmentation

## 1. Introduction

Data augmentation is a popular technique that generates new instances by processing available training data to increase its amount and variance. Data augmentation improves the network’s generalization, performance, and robustness for image segmentation tasks. Some augmentation techniques use handcrafted operations (e.g., scale, rotation, colour changes, and so on), while others learn the desired operations from these operations that can achieve the most accurate results (e.g., Cubuk et al. (2019); Xu et al. (2020); Li et al. (2020)). Despite the improvements achieved by these techniques, no matter for handcrafted data augmentation or policy search-based data augmentation, the final augmented image changes only under geometric and photometric transformations, without considering the topology transformation. This limitation can hurt the model’s generalization ability and robustness.

Generally, a manifold (surface) has two important invariants, topology and geometry. Topological information is global information, and geometric information is local information. For example, all the curvature information on the three-dimensional manifold (Riemann curvature tensor, Ricci curvature tensor, scalar curvature, section curvature) is essentially the complicated combination of the first derivative and the second derivative, only describing the local bend of the surface. The topological information is global, rep-

resenting the manifold's invariant information under large-scale global deformation. For example, no matter how we stretch or deform the surface in Figure 1, it always has three "holes". In fact, the origin of topology is precisely to characterize intuition about "holes" in strict concepts. Homology is an important topological concept whose original definition relies on the so-called complex, such as triangulation of common 2D meshes. However, its final result does not depend on how the manifold is decomposed. For the three-dimensional manifold represented by the three-dimensional image, we first need to define the three-dimensional decomposition, such as the hexahedral mesh of the three-dimensional space, corresponding to the image voxel. Then we calculate the first homology group on the hexahedral mesh, the closed loops that cannot shrink to a point. The bounding boxes of these closed loops will be used for random erasure during data augmentation.

This paper proposes a topology-based data augmentation framework by finding the first homology group, i.e. the closed loops that cannot shrink to a point. It is the first automatic data augmentation work involving homology basis in 3D image. The contributions of our paper are as follows:

- Following the classical mathematical definition of homology, we design a method to find the first homology group of a three-dimensional manifold, i.e. a three-dimensional image. To the best of our knowledge, no one has proposed this approach before.
- We take full advantage of the topological information of 3D images and design a topology-based data augmentation method. After computing all homology groups offline, it performs like any common method.
- Experiment shows that our method outperforms CutOut [DeVries and Taylor \(2017\)](#), CutMix [Yun et al. \(2019\)](#), MixUp [Zhang et al. \(2017\)](#).

Therefore, our paper is organized as follows: Section 2 presents more detailed background information on data augmentation and homology, Section 3 presents the representation of hexahedral meshes for arbitrary regions of 3D images, Section 4 presents our original method for finding the first homology group in hexahedral meshes, Section 5 is the pseudocode, Section 6 is the detailed implementation on a private dataset and a public dataset, and Section 7 is the conclusion.

## 2. Related Work

### 2.1. Data Augmentation

Deep neural networks have accomplished incredible progress in medical image segmentation tasks and promoted successful computer-assisted intervention development in the past few years. This has benefitted research and clinical treatment of disease diagnosis, treatment design, and prognosis evaluation. Researchers proposed various 3D medical image segmentation models for supervised or semi-supervised tasks given the training data. However, the conduct of deep learning models profoundly relies on adequate well-labeled data. Common image data augmentation techniques are geometric and photometric transformations such as flipping, cropping, rotation, translation, noise injection, kernel filters, mixing images, random erasing, and so on [Moreno-Barea et al. \(2018\)](#); [Jurio et al. \(2010\)](#); [Taylor and](#)

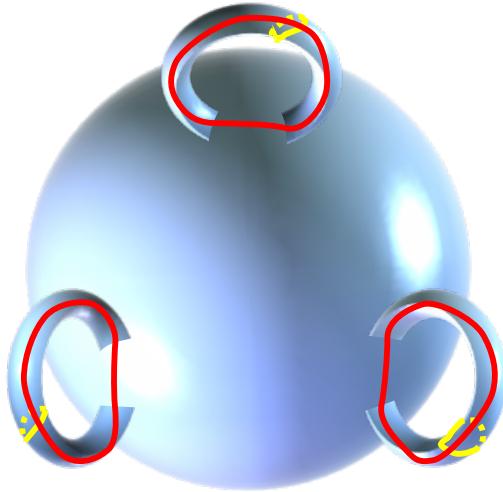


Figure 1: The 2D surface has three "holes" and thus six closed loops (in red and yellow) that cannot contract to a point, so its first homology group has six generators. However, if the surface is considered a solid 3D surface (or 3D manifold more formally), then the three yellow curves can contract to a point inside the solid. Therefore, only three red lines remain in the generator of the first homology group of the three-dimensional manifold. This paper will find the first homology group of 3D manifold, which is naturally equivalent to 3D image.

Nitschke (2018); Kang et al. (2017); Inoue (2018); Summers and Dinneen (2019); Takahashi et al. (2019); Zhong et al. (2020); DeVries and Taylor (2017); Yun et al. (2019); Zhang et al. (2017). Besides these classical methods, GAN-based data augmentation Zhu et al. (2018); Li et al. (2018) and Neural Style Transfer Gatys et al. (2015); Jackson et al. (2019) are also conducted for generating extra data. However, these methods do not explicitly consider the topology of the image. Recently, reinforcement learning and evolution based date augment have been proposed due to the appearance of neural architecture search. The neural architecture search-based data augmentation either searches from the policy space of the geometric and photometric transformations, such as Xu et al. (2020); Cubuk et al. (2019); Terauchi and Mori (2021); Lim et al. (2019), or searches from the predefined net structure such as Neural Style Transfer Perez and Wang (2017), the structure providing optimal mixing of images Lemley et al. (2017). As mentioned before, geometric and photometric transformations (policies to be searched) and predefined net structures do not explicitly consider the topological information about the image, especially in 3D images.

## 2.2. Homology

Homology theory can be said to start with the Euler polyhedron formula, or Euler characteristic Stillwell (2012). This was followed by Riemann's definition of genus and n-fold connectedness numerical invariants in 1857 and Betti's proof in 1871 of the independence of "homology numbers" from the choice of basis Rotman (2008). Homology itself was de-

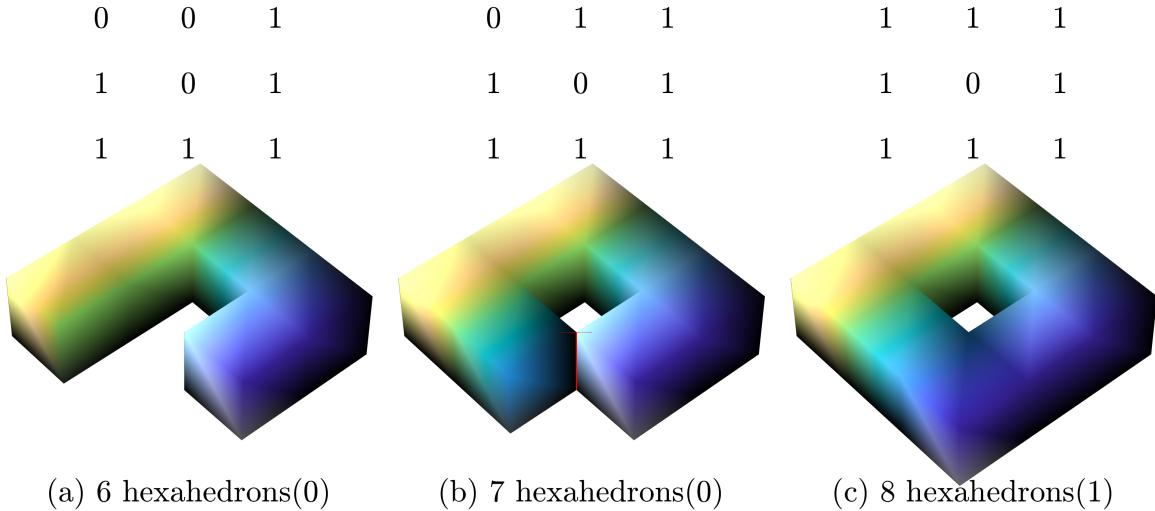


Figure 2: Different hexahedrons and ranks of  $H_1$ . The first row shows three different 3D bool arrays, each with size  $3 \times 3 \times 1$ . The second row shows their corresponding hexahedral mesh. Each sub caption shows the number of hexahedral mesh and the rank of the first homology group. The middle mesh is not considered stable, since the edge in red colour can be perturbed such that the mesh would have the same shape as the left mesh. So our definition of hexahedral mesh is consistent with intuition, although it has a complicated form.

veloped to analyze and classify manifolds according to their cycles – closed loops (or more generally submanifolds) that can be drawn on a given  $n$  dimensional manifold but not continuously deformed into each other [Richeson \(2019\)](#).

Intuitively, all closed loops on a sphere can be contracted to a point with continuous deformation. For more complex surfaces, not all closed loops have such trivial properties. As an example, a sphere with three rings, as the 2D surface in Figure 1, has six closed curves that can't contract continuously to a point. These six curves are the inherent invariants of the surface, reflecting its important global structure. Informally, these nontrivial curves, and their properties, are the main content of the homology study.

Medical images are usually 3D manifolds, and the homology of 3D manifolds is slightly different from 2D. If regarding Figure 1 as a solid, in other words, the so-called two-dimensional manifold in the above paragraph is exactly the boundary of the three-dimensional shape. Then the solid shape has three special closed loops (in red) because all yellow loops can contract to a point through the interior. It is exactly this three-dimensional homology that this paper refers to.

### 3. Hexahedral Mesh

There is a natural structure of hexahedral mesh in three dimensional CT image. More precisely, an arbitrary 3D bool array is equivalent to a hexahedral mesh. Two simple examples of the bool array with size  $3 \times 3 \times 1$  and corresponding hexahedral mesh are

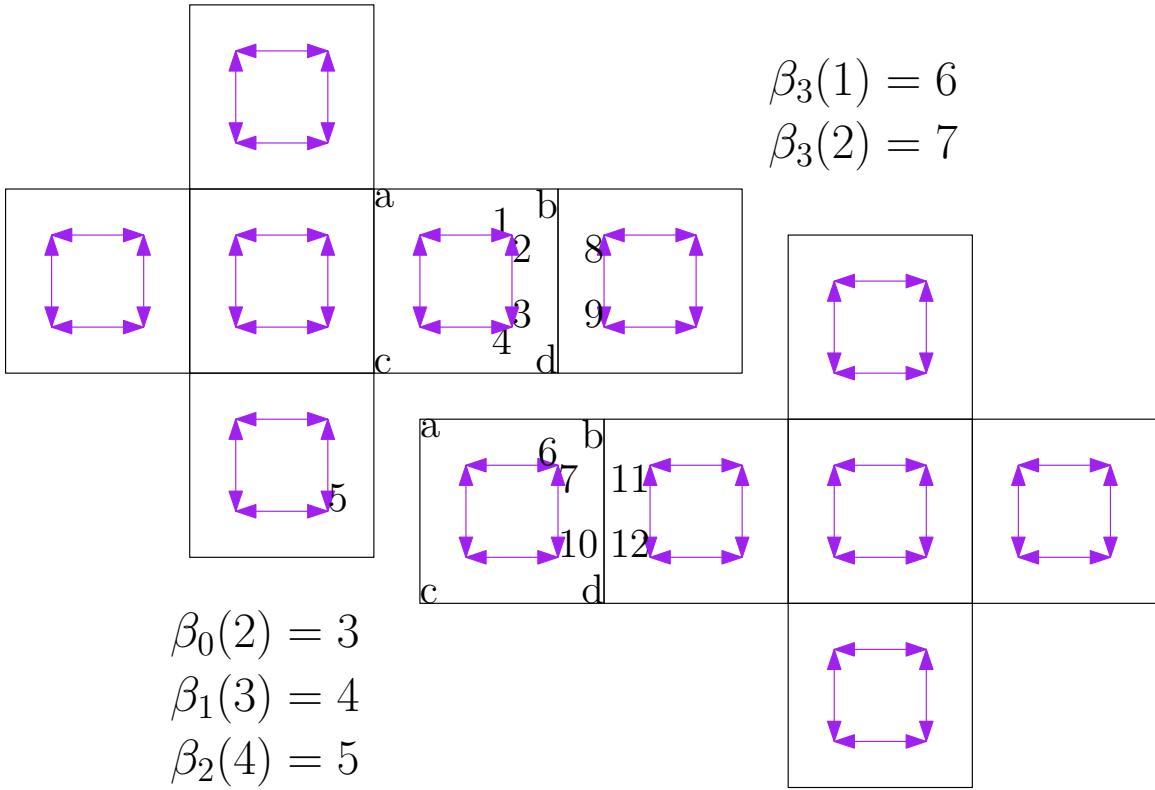


Figure 3: Net of two connected hexahedrons with the common face  $abcd$ . Each arrow is a 4-tuple with struct (vertex, edge, face, volume), and four different relationship operations  $\beta_0$ ,  $\beta_1$ ,  $\beta_2$ ,  $\beta_3$  have been shown on the image.

shown in Figure 2a and Figure 2c, respectively. It is easy to see that in mesh of Figure 2a, rank of first homology group is 0, and its rank in mesh of Figure 2c is 1. However, the array in Figure 2b with a shape between Figure 2a and Figure 2c confuses the corresponding hexahedral mesh. How to define the edge of the mesh on the red line? Or, is there one edge or two edges on the red line? The rank of the homology group is 1, which is topologically equivalent to Figure 2c, in the former case, or 0, which is topologically equivalent to Figure 2a, in the latter case. One important thing to consider when defining hexahedral mesh is to reduce the topological noise as much as possible, i.e., reduce the rank of the homology group in a mesh like Figure 2b. So there are two edges in the red line of Figure 2b in our definition. The way we define it is also intuitive because we can intuitively feel that the mesh in Figure 2b is unstable; thus, it can be continuously deformed to Figure 2a.

It is easy to see that the definition of hexahedral mesh is equivalent to counting the number of vertexes, edges, faces, and volumes on the mesh in a given mesh. For example, in Figure 2b, the red edge is counted twice, and its two endpoints are counted twice. So the number of them are 32, 60, 36, and 7.

Our method is inspired by G. Damiand and P. Lienhardt [Damiand and Lienhardt \(2014\)](#), the implementation of which can be found in CGAL Library [The CGAL Project \(2021\)](#).

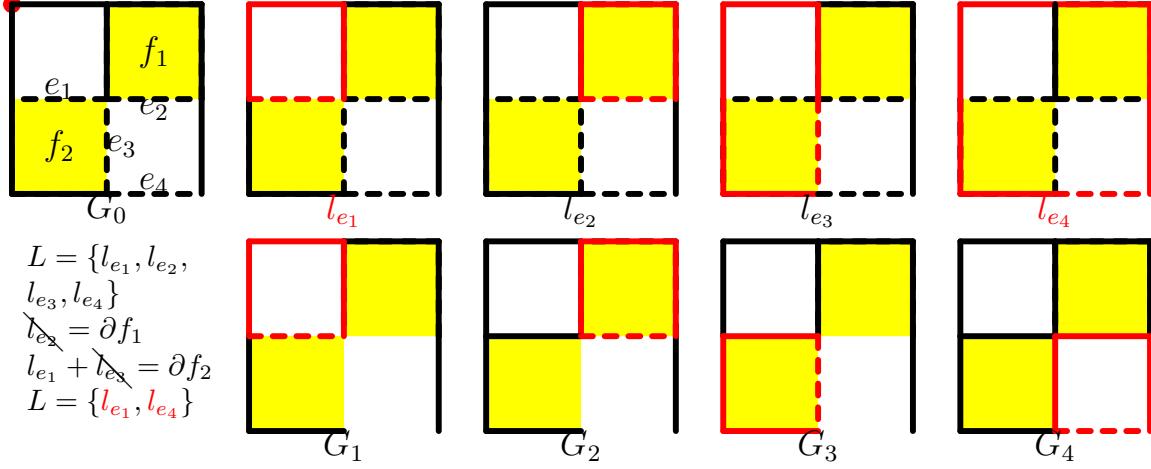


Figure 4: Algorithm design

Denote by  $X$  the set of 4-tuple, referring to (vertex, edge, face, volume), or (0-cell,1-cell,2-cell,3-cell). The Figure 3 represents the net of two connected hexahedrons, and the adjacent face is  $abcd$ . Then each arrow represents a tuple; that is, there are 48 darts in each hexahedron. We marked some darts with numbers.

Define the relational operations on a tuple:  $\beta_0, \beta_1, \beta_2, \beta_3$ .  $\beta_i(t_0) = t_1$  means that tuple  $t_0$  and  $t_1$  are same except the  $i$ -cell, for example  $\beta_0((b),(bd),(abcd),\text{the left cube}) = ((d),(bd),(abcd),\text{the left cube})$ , or equivalently,  $\beta_0(2) = 3$ , in Figure 3. Define four different finitely generated groups  $G_i$  that has the generating set  $\{\beta_j | j \neq i\} = \{\beta_0, \dots, \beta_{i-1}, \beta_{i+1}, \dots, \beta_3\}$ , every element of  $G_i$  can be written as the combination (under the group operation) of finitely many elements of the finite set  $\{\beta_j | j \neq i\}$  and of inverse of such elements. Let  $t$  be a tuple associated with an  $i$ -cell, and all the tuples  $G_i(t)$  are the same  $i$ -cell, i.e., counted once. This means all of  $G_i(t)$  are equivalent. The number of the equivalent class of  $i$ -cell is equal to the number of orbits under the group action  $G_i$ . For example, in Figure 3, there are 8 tuples associated with the edge  $bd$ , the orbit of them under action of  $G_1 = <\beta_0, \beta_2, \beta_3>$  is itself: 2, 3, 8, 9, 7, 10, 11, 12. Then we count the edge  $bd$  only once in the counting problem. However, in Figure 2, there are two orbits of tuples associated with red line under action of  $G_1$ . So the red line is counted twice.

#### 4. First Homology Group

As mentioned above, we usually decompose manifold to study homology, such as the common triangulation of two-dimensional manifolds and the hexahedral mesh of three-dimensional manifolds in this paper. Once the manifold has a decomposition, the topological problem becomes an algebraic problem, which is precisely the tool that the original mathematical definition of homology used.

For algebraic problems, the usual critical step is how to choose a basis, that is, all elements are linear combinations of these basis elements. Therefore, the first step in this problem is to find some closed loops, so that all closed loops can be represented as linear combinations of these loops. We use a simplest example in Figure 4 to illustrate these

concepts. It shows a graph  $G$  with 9 vertexes, 12 edges, 2 faces (yellow face), 2 holes (white face) and a breadth-first search (BFS) tree (solid line)  $G_0$ , with red vertex as root. There are two types of edges in  $G$ , one of which is the edge in  $G_0$  (solid line), and the other is the remaining edges (dashed line). Each edge  $e \notin G_0$  corresponds to a unique cycle  $l_e$  in  $e \cup G_0$ , i.e.  $l_{e_1}, l_{e_2}, l_{e_3}, l_{e_4}$  in Figure 4. These closed loops are exactly the basis we are looking for, i.e. all closed loops can be represented as their linear combination.

The first homology group is essentially looking for closed loops that cannot contract to a point, equivalent to deleting all closed loops that can contract to a point. Note that the boundary of each quadrilateral (i.e. the closed loop comprised of its four edges) can obviously contract to a point through the quadrilateral's face. Since the smallest face elements are quadrilaterals in hexahedral mesh, we enumerate all quadrilaterals based on this property, removing all trivial loops in the base.

Let  $L = \{l_e | e \notin G_0, l_e \text{ is not removed}\}$ . When a quadrilateral  $f$  appears, the following relationship occurs:

$$\sum l_{e_i} = \partial f, l_{e_i} \in L \quad (1)$$

where the symbol “=” means homologous. We expect to remove one of the  $l_{e_i}$  while keep remaining closed loop as short as possible, so we remove the one with the largest perimeter among all the closed loops  $l_{e_i}$  in Equation (1). More precisely, we remove such  $l_e$  that  $\forall i, \text{length}(l_e) \geq \text{length}(l_{e_i})$ . When the algorithm terminates,  $L$  is the generator of first homology group. In other words, each closed loop in  $L$  can't shrink to a point, as the  $l_{e_1}$  and  $l_{e_4}$  in Figure 4. We expect the obtained closed loop to be as short as possible; however, the closed path  $l_{e_4}$  now does not meet this requirement because it can be simplified further.

As mentioned above,  $G_0$  is a BFS tree, the elements in  $L$  are the shortest loops that passes through the root vertex. Next, we shorten each element in  $L$  by constructing a sequence of subgraphs  $G_i$ : For an tuple  $(l_{e_1}, l_{e_2}, \dots)$ ,  $i < j \Rightarrow \text{length}(l_{e_i}) \leq \text{length}(l_{e_j})$ , then  $G_i = G_{i-1} \cup l_{e_i}$ . We denote  $l'_{e_i}$  the shortest circle containing edge  $e_i$  in  $G_i$ . If  $l_{e_i}$  is the shortest circle that passes through a certain hole and the root node, which is obviously also an element in  $L$ , then  $l'_{e_i}$  must be a non-trivial circle that passes through the same hole with perimeter much less than  $l_{e_i}$ .

Now we have the closed loops  $L' = \{l'_e | l_e \in L\}$ , as the red circles in  $G_1$  and  $G_4$  in Figure 4. In this example, it is already the global optimal solution. Figure 5 shows the results of our algorithm on two other hexahedral meshes, where the red lines represent the closed loops we wish to find. Note that the left graph does not have any closed loop because it is not stable and can continuously deform to a solid sphere. Some closed loops can shrink to a point on the right mesh, each having the shortest length.

## 5. Algorithm

We give two algorithms in this section. Algorithm 1 is a method of calculating the homology basis, and Algorithm 2 is our main algorithm for data augmentation. Note that the organs/tissues/tumours to be segmented in our dataset and all 10 datasets of the Medical Segmentation Decathlon are topologically trivial, i.e. they can all continuously deform to a solid sphere. Therefore, our data augmentation method theoretically does not mask the ground truth.

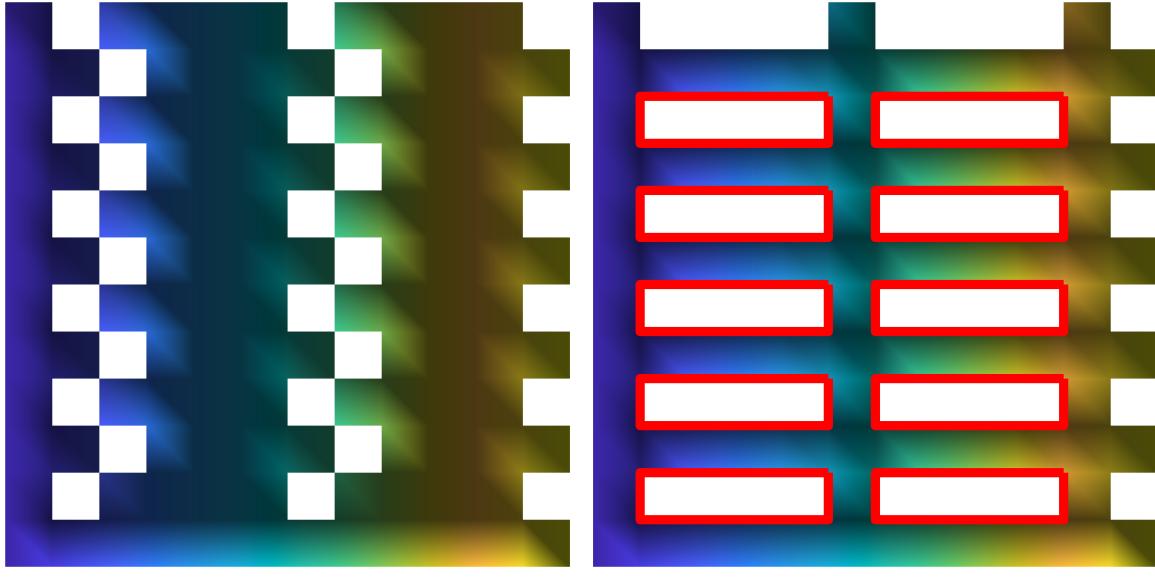


Figure 5: First homology group (red lines) in different hexahedral meshes. Note that according to the definition of the hexahedral mesh in Section 3, there is no red line in the left mesh, because the "hole" of it is unstable, which is also consistent with human intuition.

---

**Algorithm 1** FirstHomologyGroup

---

**Input:** hexahedral mesh

**Output:** closed loops

- 1: construct shortest path tree  $G$ .
  - 2: obtain cycle basis  $C = \{C_{e_1}, C_{e_2}, \dots, C_{e_n}\}$  in order of decreasing length.
  - 3: obtain homology basis  $C^* = \{C_{e_{i_1}}, C_{e_{i_2}}, \dots, C_{e_{i_k}}\}$  by solving (1)
  - 4: **for** edge  $e_j \in \{e_n, e_{n-1}, \dots, e_1\}$  **do**
  - 5:    $l'_{e_j}$  = shortest cycle containing  $e_j$  in  $G$
  - 6:    $G = G \cup e_j$
  - 7: **end for**
  - 8: **return**  $\{l'_{e_j} | j \in i_1, i_2, \dots, i_k\}$
- 

**Algorithm 2** DataAugmentation

---

**Input:** 3D CT image  $img$

**Output:** new image  $img$ , new label  $L$

- 1: Choose the bounding box of a random closed loop.
  - 2: Initialize mask  $M$  with the same size as the image.
  - 3:  $M(\text{bounding box}) = 0$
  - 4:  $img = img \otimes M$
  - 5:  $L(\text{bounding box}) = 0$
-

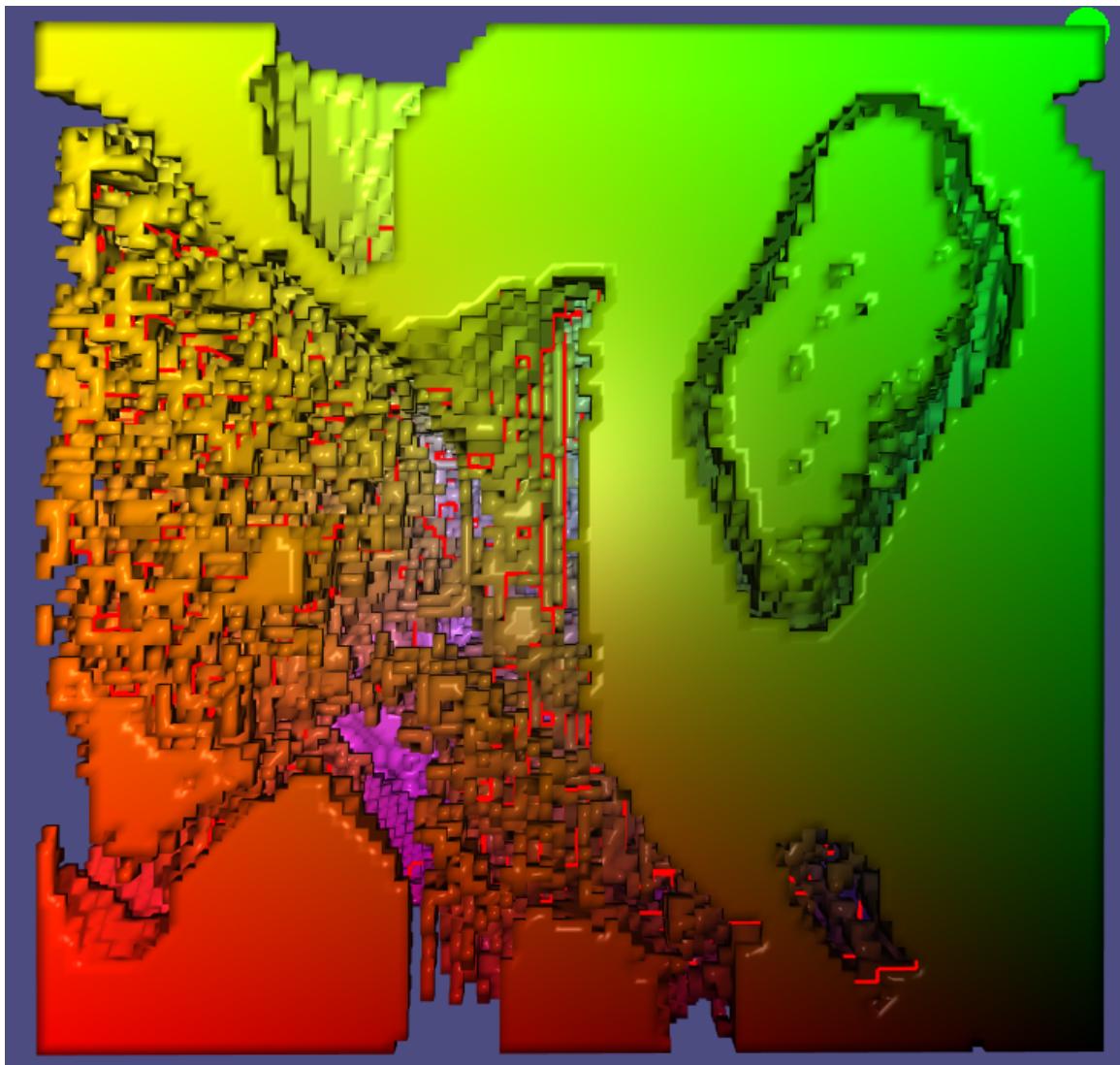


Figure 6: A hexahedral mesh of a large region in the ossicles dataset and its first homology group, where the red lines represent all elements of the first homology group. It is worth emphasizing again that this is a three-dimensional manifold, so it is a solid, and its interior is also composed of hexahedrons. It has a lot of small "holes" and therefore a lot of topological noises. This is because we are using a simple threshold to get the area, and the noise of the image will cause some small holes. The easiest way to remove such small holes is to use morphological techniques in digital image processing, although we currently use the perimeter of the closed loops to filter the noise.

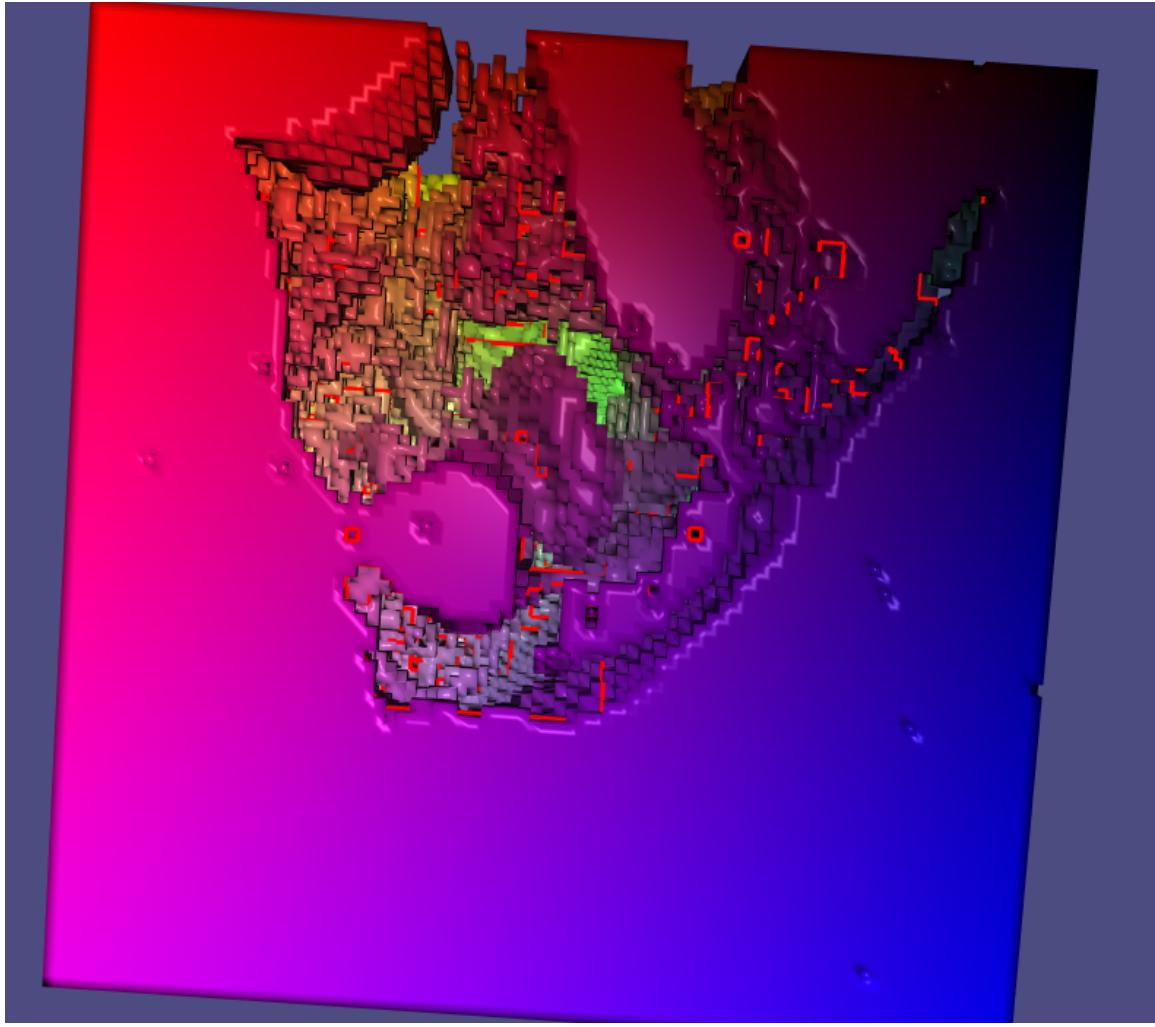


Figure 7: Another side of the mesh in Figure 6, there are obviously some closed paths of length 4, which are the smallest closed paths that can exist in our hexahedral mesh, usually due to noise.

## 6. Experiment

**Dataset** We conduct the proposed method on two datasets respectively. One is the Heart dataset in Medical Segmentation Decathlon. The data set consists of 30 mono-modal MRI scans of the entire heart acquired during a single cardiac phase (free breathing with respiratory and electrocardiogram (ECG) gating). The corresponding target ROI was the left atrium. The other dataset is a private dataset, consisting of 350 3D CT images from different patients, 250 of whom were diagnosed with normal ossicles, and 100 were diagnosed with deficient ossicles. The corresponding target ROI was the ossicles and the voxel space is  $0.2734 \times 0.2734 \times 0.335$ . The images have dimensions of 150 pixels in height and 200 pixels in width.

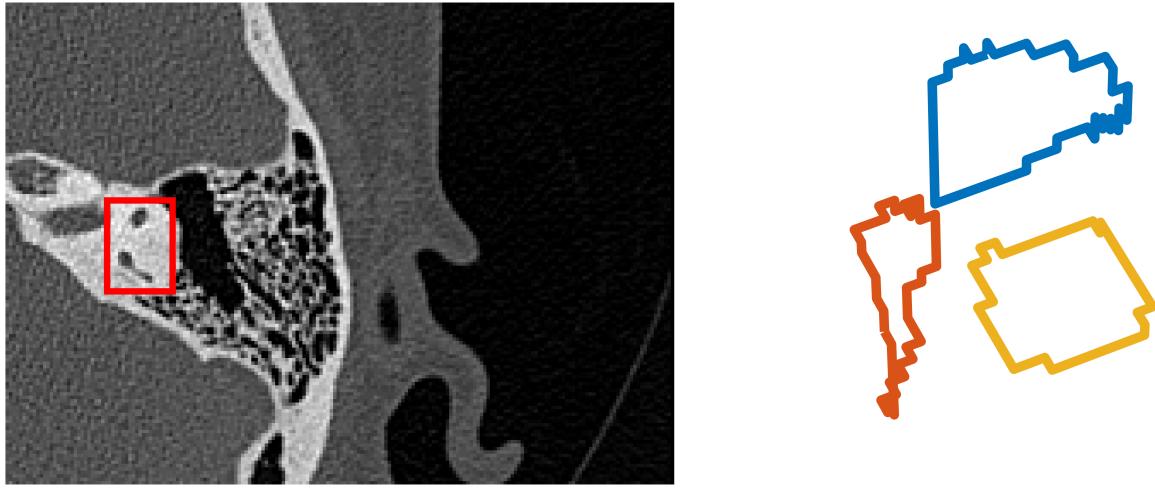


Figure 8: A closed loop within a smaller area of the image. The left picture shows the position of the cube region in one of the slices, and the right picture shows the three closed paths within it obtained by our algorithm.

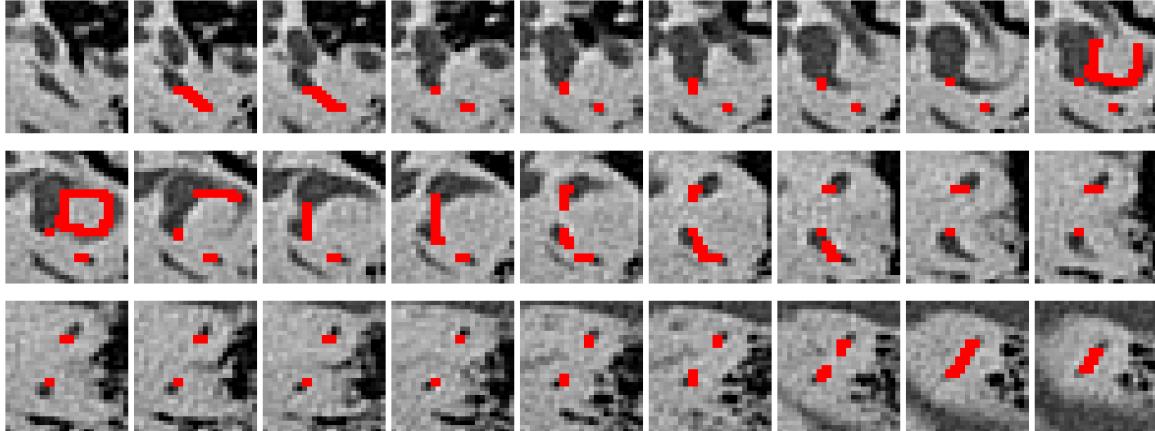


Figure 9: The real positions in CT image of the three closed loops from Figure 8.

**Data augmentation** We first need to divide the CT image into different regions to build their hexahedrons. We adopted a simple thresholding method because grayscale in medical images usually has specific anatomical meanings. That is, voxels in different grayscale intervals usually have different anatomical meanings.

For the ossicles dataset, we divide the grayscale into three intervals,  $[0, 542]$ ,  $[542, 1704]$ , and  $[1704, +\infty]$ , which represent air, tissue, and bone, respectively. Since our data augmentation method is somewhat equivalent to erasing an area and setting it to air, we do not consider air intervals. Reconstruct the hexahedral meshes of the tissue and bone areas with the method of Section 3, and obtain the generators of the first homology groups of them, that is, the closed loops on them that cannot shrink into a point, which will be

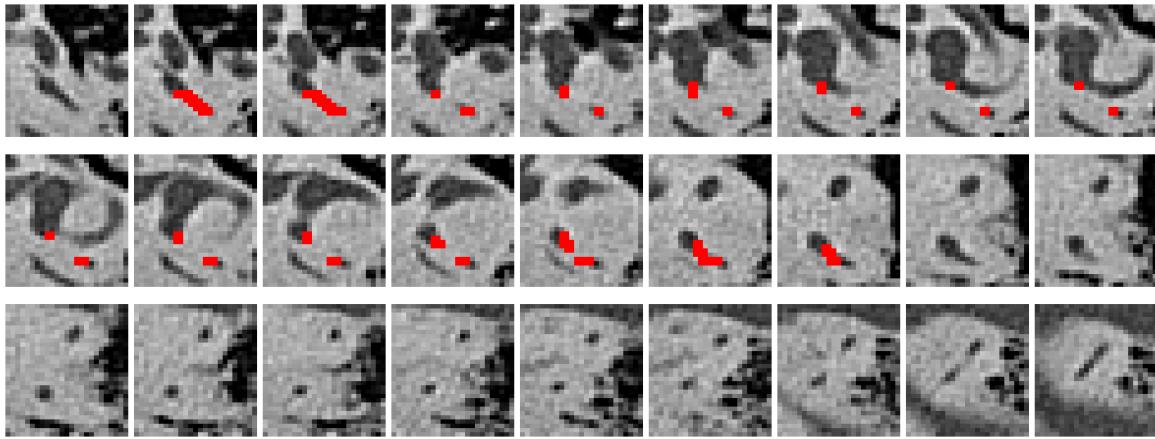


Figure 10: The true position of one of the 3 closed loops from Figure 8.

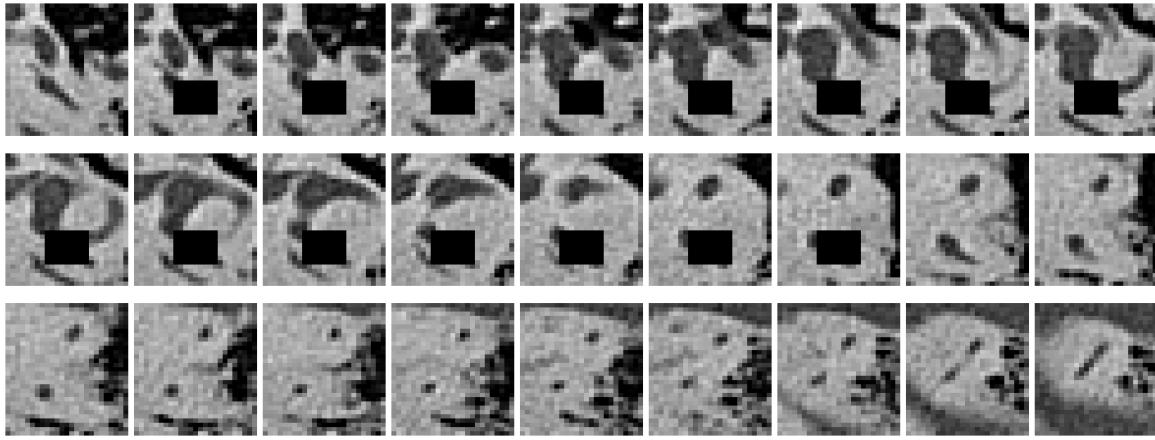


Figure 11: Erase the bounding box of the closed loop in figure 10 for data augmentation. From our observations, in both our dataset and all 10 public datasets in the Medical Segmentation Decathlon, the organs to be segmented are topologically trivial, i.e. equivalent to solid spheres. Therefore, our region removal does not theoretically remove the organ to be segmented.

erased during data augmentation. Figure 6 and 7 show different sides of the hexahedral mesh constructed from a big region in the ossicles dataset, where all voxels have grayscale in interval [542, 1704]. The red lines represent all generators of the First Homology Group, that is, all closed curves that cannot shrink to a point. In order to show their effect more clearly, we selected a small 3D region from this data and showed its closed loops. The left image of Figure 8 shows the region's position (inside the red rectangle) in a slice of the CT image, and the right image shows a total of 3 closed loops within this region. Figure 9 shows the actual locations of the 3 loops in the CT image. Although the CT image seen from the 2D perspective limits our 3D imagination, we can still observe that the red region of Figure 9 does have nontrivial topological properties, i.e. its three special closed loops. It

is worth pointing out that the three curves in Figure 8 don't intersect, but the red lines in Figure 9 intersect. This is because a point on the hexahedral mesh established in Section 3 has 8 hexahedrons adjacent to it. Therefore, in Figure 9, we coloured in red all 8 adjacent hexahedrons of each point in the closed loops from Figure 8.

Figure 10 only retains one closed loop in Figure 9. In our data augmentation process, we select the bounding box of a random closed loop as in Figure 10 to erase, as shown in Figure 11. Our method can automatically select bounding boxes based on the topological properties of the data instead of randomly selecting bounding boxes as in previous algorithms (e.g. CutOut or CutMix). More importantly, from a practical perspective, after offline processing, our algorithm can be computed as fast in pytorch as methods like CutOut without any additional processing.

Notice that there are some closed curves with very short lengths, such as some loops of length 4 in Figure 7, which are the theoretically shortest loops that can appear in hexahedron mesh. These small loops usually represent a kind of topological noise, and we set an empirical threshold to remove them. For example, we only consider closed loops with perimeters greater than 10.

Finally, we store all closed loops for each data in corresponding files. In data augmentation, we randomly select one of the closed paths stored before, and set the interior of its bounding box to 0, instead of generating a rectangle with arbitrary size at a random position each time as the CutOut or CutMix method. Since our algorithm computes all closed paths to files offline, the final data augmentation process consumes the same time as any other common data augmentation method.

For the heart dataset, we divide the grayscale into three intervals,  $[0, 100]$ ,  $[100, 500]$ , and  $[500, +\infty]$ . The remaining process is the same as before.

In summary, our fundamental principle is to analyze the topological properties of different anatomical structures and use the topological properties to guide the region removal.

**Compared Methods** We used MultiResUNet [Ibtehaz and Rahman \(2020\)](#) as net struct, and compared our metod with three similar data augmentation methods CutOut, MixUp, CutMix. We unify the identical voxel spacing resampling them to  $0.7 \text{ mm} \times 0.7 \text{ mm} \times 0.7 \text{ mm}$  firstly. The patches are generated with a sliding window moving across the entire image with a stride of half patch size. The input batch consists of two patches with size  $128 \times 128 \times 128$ . The network is implemented in Pytorch 1.11.0 on an NVIDIA V100 GPU. For both dataset, We evaluate the performance with 5-fold cross validation. The ADAM optimizer is applied with an initial learning rate of  $3 \times 10^{-4}$  and a weight decay of  $10^{-5}$ , which will be reduced by 80% if the training loss is not reduced over 30 epochs. We terminate the training once the learning rate is below  $10^{-8}$ . Table 1 shows our comparison results. We found that the CutOut method does not work as well as expected, and the MixUp method does not perform well on the ossicles dataset. However, our method outperforms the other three methods and achieves good results on all datasets.

## 7. Conclusion

We have proposed an automatic data augmentation strategy involving topology to accommodate 3D medical image segmentation tasks. The image's topology information can be calculated by constructing a hexahedral mesh followed by a method to find the first homol-

Table 1: Comparison of four data augmentation methods, CutOut, MixUp, CutMix, and ours, on the two datasets, our ossicles dataset and heart dataset in Medical Segmentation Decathlon. The metric is the Dice similarity coefficient.

	Ossicles	Heart
MultiResUNet	$80.28 \pm 4.03$	$86.47 \pm 0.66$
MultiResUNet+CutOut	$76.16 \pm 2.49$	$84.32 \pm 1.21$
MultiResUNet+MixUp	$79.42 \pm 0.72$	$87.18 \pm 1.96$
MultiResUNet+CutMix	$82.01 \pm 1.75$	$87.43 \pm 1.05$
MultiResUNet+OURS	<b><math>83.13 \pm 0.31</math></b>	<b><math>89.83 \pm 0.56</math></b>

ogy group. The numerical results for 3D segmentation tasks show that it can improve the performance of the segmentation model. By adding it to the search space of some automatic augmentation methods, We believe that incorporating our method with existing methods could further boost the segmentation’s performance.

## Acknowledgments

This research was supported by the National Key R & D Program of China 2020YFA0712203, National Natural Science Foundation of China under Grant No. 61936002, T2225012, and Beijing Natural Science Foundation Z210003

## References

- Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019.
- Guillaume Damiand and Pascal Lienhardt. *Combinatorial maps: efficient data structures for computer graphics and image processing*. CRC Press, 2014.
- Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.
- Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.
- Nabil Ibtehaz and M Sohel Rahman. Multiresunet: Rethinking the u-net architecture for multimodal biomedical image segmentation. *Neural networks*, 121:74–87, 2020.
- Hiroshi Inoue. Data augmentation by pairing samples for images classification. *arXiv preprint arXiv:1801.02929*, 2018.

- Philip TG Jackson, Amir Atapour Abarghouei, Stephen Bonner, Toby P Breckon, and Boguslaw Obara. Style augmentation: data augmentation via style randomization. In *CVPR Workshops*, pages 83–92, 2019.
- Aranzazu Jurio, Miguel Pagola, Mikel Galar, Carlos Lopez-Molina, and Daniel Paternain. A comparison study of different color spaces in clustering based image segmentation. In *International conference on information processing and management of uncertainty in knowledge-based systems*, pages 532–541. Springer, 2010.
- Guoliang Kang, Xuanyi Dong, Liang Zheng, and Yi Yang. Patchshuffle regularization. *arXiv preprint arXiv:1707.07103*, 2017.
- Joseph Lemley, Shabab Bazrafkan, and Peter Corcoran. Smart augmentation learning an optimal data augmentation strategy. *Ieee Access*, 5:5858–5869, 2017.
- Shuangtao Li, Yuanke Chen, Yanlin Peng, and Lin Bai. Learning more robust features with adversarial training. *arXiv preprint arXiv:1804.07757*, 2018.
- Yonggang Li, Guosheng Hu, Yongtao Wang, Timothy Hospedales, Neil M Robertson, and Yuxin Yang. Dada: Differentiable automatic data augmentation. *arXiv preprint arXiv:2003.03780*, 2020.
- Sungbin Lim, Ildoo Kim, Taesup Kim, Chiheon Kim, and Sungwoong Kim. Fast autoaugment. *Advances in Neural Information Processing Systems*, 32:6665–6675, 2019.
- Francisco J Moreno-Barea, Fiammetta Strazzera, José M Jerez, Daniel Urda, and Leonardo Franco. Forward noise adjustment scheme for data augmentation. In *2018 IEEE symposium series on computational intelligence (SSCI)*, pages 728–734. IEEE, 2018.
- Luis Perez and Jason Wang. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.
- David S Richeson. *Euler’s gem: the polyhedron formula and the birth of topology*. Princeton University Press, 2019.
- Joseph J Rotman. *An introduction to homological algebra*. Springer Science & Business Media, 2008.
- John Stillwell. *Classical topology and combinatorial group theory*, volume 72. Springer Science & Business Media, 2012.
- Cecilia Summers and Michael J Dinneen. Improved mixed-example data augmentation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1262–1270. IEEE, 2019.
- Ryo Takahashi, Takashi Matsubara, and Kuniaki Uehara. Data augmentation using random image cropping and patching for deep cnns. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(9):2917–2931, 2019.

- Luke Taylor and Geoff Nitschke. Improving deep learning with generic data augmentation. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1542–1547. IEEE, 2018.
- Akira Terauchi and Naoki Mori. Evolutionary approach for autoaugment using the thermodynamical genetic algorithm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9851–9858, 2021.
- The CGAL Project. *CGAL User and Reference Manual*. CGAL Editorial Board, 5.3 edition, 2021. URL <https://doc.cgal.org/5.3/Manual/packages.html>.
- Ju Xu, Mengzhang Li, and Zhanxing Zhu. Automatic data augmentation for 3d medical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 378–387. Springer, 2020.
- Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019.
- Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.
- Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13001–13008, 2020.
- Xinyue Zhu, Yifan Liu, Jiahong Li, Tao Wan, and Zengchang Qin. Emotion classification with data augmentation using generative adversarial networks. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 349–360. Springer, 2018.