Medical Image Analysis 64 (2020) 101746

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Rubik's Cube+: A self-supervised feature learning framework for 3D medical image analysis

Jiuwen Zhu[a,1], Yuexiang Li[b,*], Yifan Hu[b], Kai Ma[b], S. Kevin Zhou[a], Yefeng Zheng[b]

[a] Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[b] Tencent Jarvis Lab, Shenzhen, China

ARTICLE INFO

Article history:
Received 21 January 2020
Revised 7 May 2020
Accepted 1 June 2020
Available online 6 June 2020

Keywords:
Self-supervised learning
3D Medical imaging data
Rubik's cube recovery

ABSTRACT

Due to the development of deep learning, an increasing number of research works have been proposed to establish automated analysis systems for 3D volumetric medical data to improve the quality of patient care. However, it is challenging to obtain a large number of annotated 3D medical data needed to train a neural network well, as such manual annotation by physicians is time consuming and laborious. Self-supervised learning is one of the potential solutions to mitigate the strong requirement of data annotation by deeply exploiting raw data information. In this paper, we propose a novel self-supervised learning framework for volumetric medical data. Specifically, we propose a pretext task, i.e., Rubik's cube+, to pre-train 3D neural networks. The pretext task involves three operations, namely cube ordering, cube rotating and cube masking, forcing networks to learn translation and rotation invariant features from the original 3D medical data, and tolerate the noise of the data at the same time. Compared to the strategy of training from scratch, fine-tuning from the Rubik's cube+ pre-trained weights can remarkablely boost the accuracy of 3D neural networks on various tasks, such as cerebral hemorrhage classification and brain tumor segmentation, without the use of extra data.

1. Introduction

Medical imaging techniques, e.g., computed tomograph (CT) and magnetic resonance imaging (MRI), produce volumetric data that naturally appears in a 3D form. Experienced physicians usually need to repetitively browse and analyze the 3D volume for abnormality detection and disease diagnosis, which is laborious and the accuracy suffers from inter-observer variation. This dilemma motivates the development of computer-aided diagnosis (CAD) systems to effectively increase the efficiency of physicians. A convolutional neural network (CNN), proving its advanced capability on many challenging tasks in natural image analysis such as image classification (Kumar et al., 2015; Graeter et al., 2016; Gao et al., 2017) and semantic segmentation (Havaei et al., 2017), is a viable option for the CAD systems. The success of CNNs benefits from the information explosion—the amount of annotated data rapidly increases in the past few years. Crowd sourcing can be used to obtain ground-truth labels for natural images, as most humans are able to recognize the objects in natural images. However, crowd sourcing is not applicable for medical images, due to the expert knowledge required for medical image annotation (Chen et al., 2019). Furthermore, acquiring annotations of 3D medical data is extremely laborious, i.e., each 3D volume requires experienced physicians to spend a couple of hours or even days for investigation. As a result, although researchers have proposed various 3D network architectures (Cicek et al., 2016) to increase model accuracy, the performance is still confined by the limited amount of annotated volumetric data.

To deal with the deficiency of annotated data, researchers attempted to exploit useful information from the unlabeled data with unsupervised approaches (Zhang et al., 2017; Spitzer et al., 2018). More recently, self-supervised learning, as a new paradigm of unsupervised learning, attracts increasing attentions from the community. The pipeline usually consists of two steps: 1) pre-train a convolutional neural network (CNN) on a pretext task with a large non-annotated dataset, and 2) fine-tune the pre-trained network for the specific target task with a small set of annotated data. The pretext task enforces neural networks to deeply mine useful information from the unlabeled raw data, which can boost the accuracy of the subsequent target task with limited training data. Various pretext tasks had been proposed, which include grayscale image colorization (Larsson et al., 2017), jigsaw puzzles

* Corresponding author.
E-mail address: vicyxli@tencent.com (Y. Li).
[1] This work was mostly done when Jiuwen Zhu was an intern at Tencent Jarvis Lab.

https://doi.org/10.1016/j.media.2020.101746
1361-8415/

Table 1
Examplar self-supervised learning approaches proposed to process different types of data.

Researches	Data Type	Network Type (2D/3D)	Pretext Task
Doersch et al. (2015)	RGB images	2D	Patch relative position prediction
Noroozi and Favaro (2016)			Jigsaw puzzles
Larsson et al. (2017)			Colorization
Pathak et al. (2016)			Local context prediction
Fernando et al. (2017)	Videos	2D	Temporal order verification
Lee et al. (2017)			Object motion prediction
Zhang et al. (2017)	Medical data	2D	Slice relative position prediction
Spitzer et al. (2018)			Distance prediction
Chen et al. (2019)			Content restoration

(Noroozi et al., 2018) and object motion estimation (Lee et al., 2017).

For the applications with medical data, researchers took some prior-knowledge into account when formulating the pretext task. Zhang et al. (2017) defined a pretext task that sorted the 2D axial slices extracted from the conventional 3D CT and MR volumes to pre-train neural networks for the fine-grained body part recognition. Spitzer et al. (2018) proposed to pre-train neural networks on a self-supervised learning task of predicting the 3D distance between two patches sampled from the same brain for the better segmentation of brain areas. However, all of the aforementioned self-supervised learning frameworks (Spitzer et al., 2018; Zhang et al., 2017), including those for natural images (Larsson et al., 2017; Noroozi et al., 2018; Lee et al., 2017), were proposed for 2D networks. As the 3D neural networks integrating the 3D spatial information usually outperform the 2D networks on volumetric medical data, a 3D-based self-supervised learning approach is worthwhile to develop. This paper aims to bridge this gap.

An early version (Zhuang et al., 2019) of this work was presented at a conference. This paper extends the previous work substantially with the following improvements: First, we propose a pretext task, namely Rubik's cube+,[2] consisting of three sub-tasks, i.e., cube ordering, cube orientation and masking identification. The previous two tasks (cube ordering and orientation)—same as Rubik's cube recovery (Zhuang et al., 2019)—enforce the network to learn the features invariant to translation and rotation from the raw data, while the last task (masking identification) increases the diversity and difficulty of Rubik's cube recovery by randomly blocking the content of cubes, which is inspired by the observation—learning from a harder task often leads to a more robust feature representation (Deng et al., 2010; Wei et al., 2019). Second, several additional experiments on the two target tasks, i.e., cerebral hemorrhage classification and brain tumor segmentation, are conducted to demonstrate the effectiveness of our Rubik's cube+. Experimental results show that the proposed approach can significantly improve the accuracy of 3D neural networks on the target tasks, although the model is never explicitly pre-trained to exploit the knowledge of cerebral hemorrhage and brain tumor. Last but not least, comprehensive discussions on the limitation and potential applications of our study are included.

2. Related work

2.1. Medical image processing using 3D networks

Computed tomography (CT) and magnetic resonance imaging (MRI)—widely used for the screening and diagnosis of various diseases—produce volumetric data. Lots of CAD systems have been proposed to improve the accuracy of physicians and most of the recent works exploit deep learning techniques. The existing deep-learning-based CAD systems for the 3D data can be separated to two categories—frameworks using the 2D network (Ji et al., 2016) and 3D network (Kamnitsas et al., 2016b; Xie et al., 2019), respectively.

Compared to the 2D network, the 3D deep learning network integrating the spatial information has shown its superiorities for the 3D medical image processing (Dou et al., 2017). Mutimodal brain tumor segmentation (BraTS) is a famous challenge, attracting hundreds of teams to participate since 2012. It focuses on the evaluation of state-of-the-art methods for the segmentation of brain tumors in multimodal MRI scans. Due to the limitation of computation capacity, the challenge winners were mainly 2D approaches—extracting 2D slices from a 3D volume for processing—in the early BraTS challenges (Menze et al., 2015). As the graphics processing unit (GPU) capacity increased, researchers began to make their efforts to directly process 3D volumes. Due to the rich 3D spatial information, the 3D-based approach (DeepMedic) (Kamnitsas et al., 2016a) finally outperformed all the 2D networks and won the first-prize of BraTS 2016. Thereafter, the winners of BraTS in the following years were all based on 3D deep learning networks. However, the power of 3D deep learning networks is still not fully explored, due to the deficient annotated 3D data. Though BraTS provides one of the biggest publicly available multimodal MRI datasets, it consists of only about 300 sets of data.

2.2. Self-supervised learning

Self-supervised learning is an extensively-studied area. Many researches have been proposed, as shown in Table 1. For 2D natural images, for examples, Doersch et al. (2015) proposed a framework, learning the visual features by predicting the relative positions of two patches from the same image. Another representative approach of relative position prediction is the Jigsaw puzzles proposed by Noroozi and Favaro (2016). This work required deep learning networks to rearrange the positions of nine patches cropped from the same image. Based on the Jigsaw puzzles, some variants were developed (Kim et al., 2018; Noroozi et al., 2018; Wei et al., 2019). Moreover, colorization can also be formulated as a pretext task to pre-train neural networks (Larsson et al., 2017). More recently, studies have proposed to adopt rotation prediction (Feng et al., 2019; Gidaris et al., 2018), transformation estimation (Zhang et al., 2019), and optical flow prediction (Zhan et al., 2019) as the pretext task. The self-supervised learning has also been widely used for video processing too (Fernando et al., 2017; Lee et al., 2017).

For the applications with medical data, apart from the aforementioned pretext tasks (Zhang et al., 2017; Spitzer et al., 2018), several recent studies (Chen et al., 2019; Zhou et al., 2019b) proposed to pre-train neural networks by restorating the content of medical images. However, most of the existing approaches were proposed with 2D networks, which may not be optimal on 3D

[2] The symbol '+' represents the improvement compared to the existing Rubik's cube (Zhuang et al., 2019)—masking identification module.

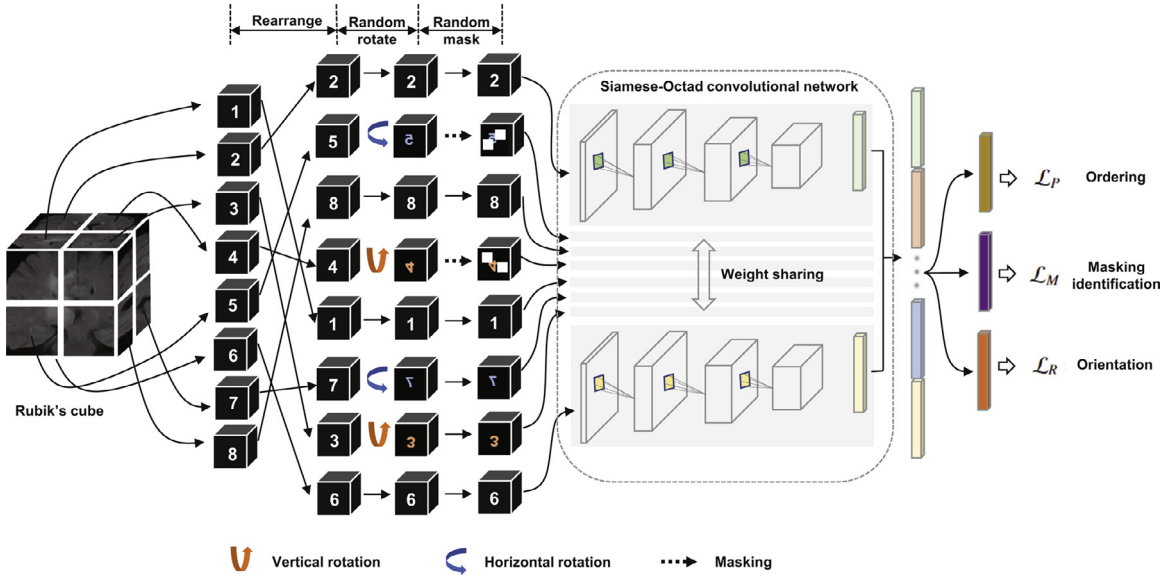

Fig. 1. The proposed Rubik's cube+ self-supervised learning framework. The Rubik's cube+ involves three sub-tasks, i.e., cube ordering, cube orientation and cube masking identification.

medical data due to the neglect of the spatial information existing in adjacent slices.

2.3. Jigsaw puzzles in deep learning

The proposed Rubik's cube+ is inspired by the existing self-supervised learning approach—Jigsaw puzzles (Noroozi and Favaro, 2016). Standard Jigsaw puzzles are created by separating pictures into a grid of tiles. The main challenge is to re-assemble the separated and randomly mixed tiles back into the original picture. Researches (Ritter et al., 2002) have verified that the Jigsaw puzzles and its implicit guidance can facilitate and advance learner's understanding of spatial-functional correlations. Based on the observation that solving Jigsaw puzzles can help the learners to better capture the concept of objects, recent researches began to apply the idea of Jigsaw puzzles to improve the specific capacity (e.g., generalization and feature robustness) of machines. For example, Carlucci et al. (2019) combined Jigsaw puzzles with domain generalization and adaptation in the task of object recognition.

Jigsaw puzzles were firstly introduced to the area of self-supervised learning by Noroozi and Favaro (2016). Following the work, Son et al. (2016) investigated the performance improvement produced by reconstructing a challenging puzzle with small tiles. Paikin and Tal (2015) enforced the network to learn from Jigsaw puzzles with missing tiles. Kim et al. (2018) developed an approach to solve grayscale Jigsaw puzzles with one missing tile. These approaches strongly proved that solving Jigsaw puzzles is an effective pretext task for self-supervised learning.

Compared to the Jigsaw puzzles, our Rubik's cube+ pretext task has two main differences: 1) The Rubik's cube+ works on 3D volumetric data, while the Jigsaw puzzle is proposed for 2D natural images; 2) In addition to the translation transformation adopted in the Jigsaw puzzle, the difficulty of recovering Rubik's cube is increased by adding the cube rotation and masking operations, which encourage deep learning networks to leverage more spatial information and yield more robust feature representation.

3. Method

To address the problem of deficient annotated 3D medical data, we propose a novel self-supervised learning approach for 3D medical imaging data, namely Rubik's cube+. The pipeline of our self-supervised learning approach is illustrated in Fig. 1. The 3D medical data (original state) is seen as a $2 \times 2 \times 2$ Rubik's cube and accordingly partitioned into eight cubes. The cubes are then processed by a series of operations—rearrangement, rotation and masking, which formulates a disarranged state of Rubik's cube. The aim of our Rubik's cube+ pretext task is to recover the original state of medical data from the disarranged Rubik's cube. To complete the pretext task, the 3D neural network requires to deeply mine the 3D anatomical information from the raw data, which significantly benefits the following target tasks.

3.1. Siamese network architecture

As Fig. 1 shows, a Siamese network with M (which is the number of cubes) sharing-weight branches, namely Siamese-Octad, is introduced to tackle Rubik's cube+ pretext task. The backbone network of each branch can be any widely-used 3D CNN, e.g., 3D ResNet (He et al., 2016) and 3D VGG (Simonyan and Zisserman, 2015). The feature maps from the last fully-connected or convolution layer of all branches are concatenated and given as input to the three separate fully-connected layers of three sub-tasks, i.e., cube ordering, orientation and masking identification, which are supervised by losses $\mathcal{L}_P$, $\mathcal{L}_R$ and $\mathcal{L}_M$, respectively.

3.2. Data pre-processing

The neural networks are encouraged to learn the high-level semantic features for Rubik's cube recovery rather than the texture information close to the cube boundaries. Therefore, we leave a gap, i.e., about 10 voxels, between two adjacent cubes during volume participation. The intensities of the whole volume are normalized to [-1, 1] by using the minimum and maximum intensity.

3.3. Cube ordering

The first step of our Rubik's cube+ is the cube rearrangement. Taking a second-order Rubik's cube, i.e., $2 \times 2 \times 2$ shown in Fig. 1, as an example, we first yield all the permutations ($\mathcal{P}$) of cubes, i.e., $\mathcal{P} = (\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_{8!})$. The permutations control the ambiguity of the task; if two permutations are too close to each other, the

Rubik's cube recovery task becomes challenging and ambiguous for networks to learn. Therefore, we iteratively select K permutations with the largest Hamming distance from $\mathcal{P}$. Then, for each time of Rubik's cube recovery, the eight cubes are rearranged according to one of the K permutations, e.g., (2, 5, 8, 4, 1, 7, 3, 6) in Fig. 1. The number of permutations for cube ordering is empirically set to 100 in our experiments. To properly reorder the cubes, the network is trained to identify the selected permutation from the K options, which can be seen as a classification task with K categories. Assuming the $1 \times K$ network prediction as p and the one-hot ground-truth as l, the permutation loss ($\mathcal{L}_P$) in this step can be defined as:

$$\mathcal{L}_P = -\sum_{j=1}^{K} l_j \log p_j. \tag{1}$$

3.4. Cube orientation

Following the cube rearrangement, the operation of cube rotation is performed. Compared to the Jigsaw-puzzles-based self-supervised learning approaches, which only involve the translation variation of image tiles on 2D plane, our 3D Rubik's cube+ pretext task adopts the random cube rotation to encourage the network to extract rotation-invariant features.

As the cubes often have a cuboid shape, free rotations result in 3 $(axes) \times 4$ $(angles) = 12$ configurations. To reduce the complexity of the task, we limit the angles for cube rotation, i.e., only allowing 180° horizontal and vertical rotations. As Fig. 1 shows, the cubes (5, 7) and (4, 3) are horizontally and vertically rotated, respectively. To orientate the cubes, the network is required to recognize whether each of the input cubes has been rotated. It can be seen as a multi-label classification task using the $1 \times M$ (M is the number of cubes) ground truth (g) with 1 on the positions of rotated cubes and 0 vice versa. Hence, the predictions of this task are two $1 \times M$ vectors (r) indicating the possiblities of horizontal (hor) and vertical (ver) rotations for each cube. The rotation loss ($\mathcal{L}_R$) can be written as:

$$\mathcal{L}_R = -\sum_{i=1}^{M} (g_i^{hor} \log r_i^{hor} + g_i^{ver} \log r_i^{ver}). \tag{2}$$

3.5. Masking identification

Inspired by the observation—learning from a harder task often leads to a more robust feature representation (Deng et al., 2010; Wei et al., 2019), an additional operation, i.e., cube masking, is adopted in our Rubik's cube+ to increase the difficulty of Rubik's cube recovery. Such a strategy can be seen as adding noise to the training data to avoid network overfitting. Many researches (Zur et al., 2009; You et al., 2019) have proven the effectiveness of this strategy, but few of them integrate it into a self-supervised learning framework. The proposed Rubik's cube+ tries to employ the random masking strategy to increase the diversity and difficulty of pretext task and boost the improvement yielded by self-supervised learning.

The process of random masking strategy can be summarized as: First, a random possibility ($pos \in [0, 1]$) is generated for each cube to determine whether it should be masked or not. If the cube is to be masked (i.e., $pos \geq 0.5$), we generate a 3D matrix R of the same shape of a cube to randomly block the content. The value of each voxel of R is a probability ($prob$) randomly captured from a uniform distribution [0, 1]. To obtain the mask, a thresholding operation ($th_R = 0.5$) is performed to R, which leads the voxel value of (x, y, z) to be 1 for $prob(x, y, z) \geq th_R$ and 0 versa vice, where (x, y, z) is the 3D coordinate of a voxel. Assuming a cube after rearrangement and rotation as C^o, its masking state (C^m) can be generated by multiplying the C^o and mask R in pixel-wise manner:

$$C^m(x, y, z) = C^o(x, y, z) \cdot R(x, y, z). \tag{3}$$

As the appearance of masked cubes is different from the others, a network deeply mining the anatomical information is expected to have the capacity for identifying them. Thus, we formulate the mask identification as a multi-label classification task, which is similar to cube orientation. Let g^b represent a $1 \times M$ ground truth for cube masking, where 1 on the position of masked cubes and 0 vice versa, and b is the $1 \times M$ prediction. Then, the masking identification loss ($\mathcal{L}_M$) can be defined as:

$$\mathcal{L}_M = -\sum_{i=1}^{M} g_i^b \log b_i. \tag{4}$$

3.6. Objective

Overall, obtaining the previously defined permutation loss $\mathcal{L}_P$, rotation loss $\mathcal{L}_R$ and masking identification loss $\mathcal{L}_M$, the total loss function $\mathcal{L}$ for our 3D self-supervised CNN can be defined as follows:

$$\mathcal{L} = \alpha \mathcal{L}_P + \beta \mathcal{L}_R + \gamma \mathcal{L}_M \tag{5}$$

where α, β and γ are loss weights, adjusting the relative influence of three different tasks. We empirically set the equal weights (i.e., $\alpha : \beta : \gamma = 1 : 1 : 1$) in the experiments.

3.7. Adapting pre-trained weights for the target task

The 3D neural networks pre-trained on Rubik's cube+ task can achieve a robust feature representation, which can then be transferred to the target tasks. For the classification task, the pre-trained 3D CNN can be directly used for finetuning. For the segmentation of 3D medical data, the pre-trained weights can only be adapted to the encoder part of a fully convolutional network (FCN), e.g. U-Net (Cicek et al., 2016). The decoder of FCN still needs random initialization, which may wreck the pre-trained feature representation and neutralize the improvement generated by the pre-training. We conduct experiments to analyse the influence caused by the randomly initialized decoder to different architectures of FCN and present the results in the next section.

4. Experiments

4.1. Datasets

4.1.1. Cerebral hemorrhage dataset

The dataset contains 1486 brain CT volumes, which were collected from a collaborative hospital. The volumes are used to analyse the pathological cause of cerebral hemorrhage. The dataset involves four categories, which are aneurysm, arteriovenous malformation, moyamoya disease and hypertension (see Fig. 2). Each 3D CT volume is organized with a shape of $270 \times 230 \times 30$ voxels. We partition the CT volume to $2 \times 2 \times 2$ cubes of $64 \times 64 \times 8$ voxels for the proposed Rubik's cube+ pretext task. After pre-training, the network is then directly transferred to the target task (i.e., cerebral hemorrhage classification). The average classication accuracy (ACC) and area under curve (AUC) are utilized as metric for the performance evaluation.

4.1.2. BraTS-2018

The Brain Tumor Segmentation (BraTS) 2018 dataset (Menze et al., 2015) provides 285 (210 low-grade gliomas and 75 high-grade gliomas) training subjects with four different modal magnetic resonance (MR) volumes—native T1-weighted

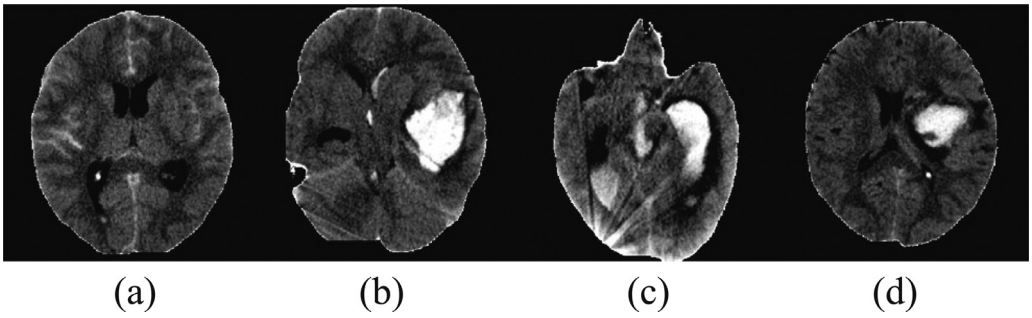

Fig. 2. Examples of cerebral hemorrhage dataset: (a) aneurysm, (b) arteriovenous malformation, (c) moyamoya disease and (d) hypertension.

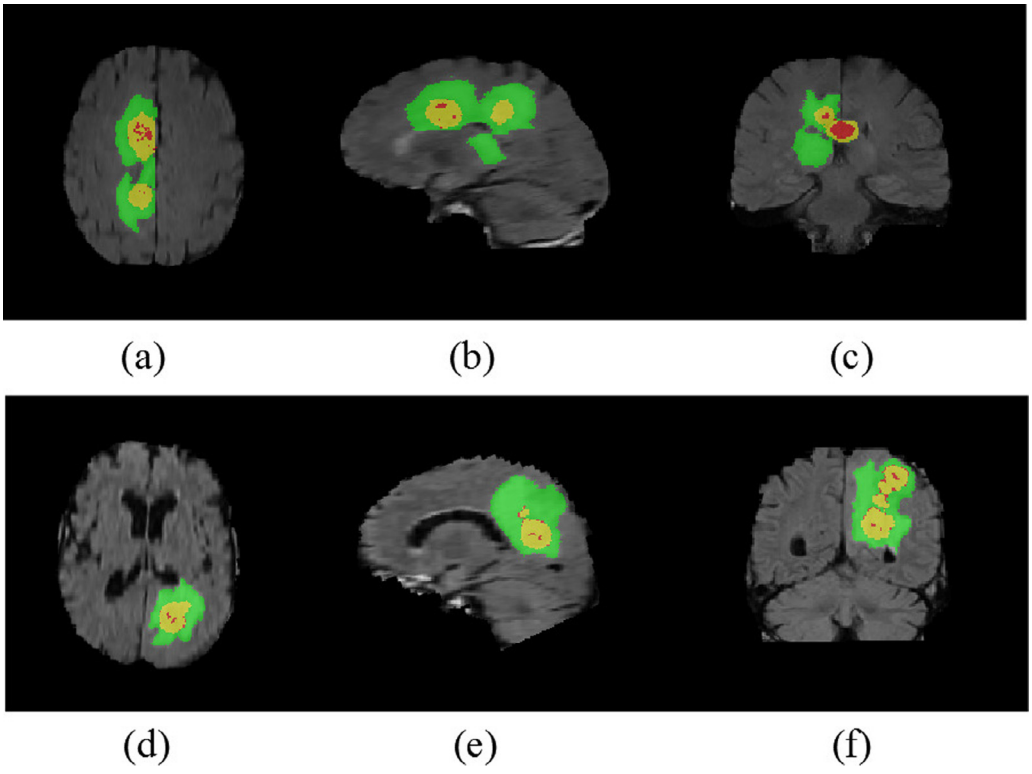

Fig. 3. Examples of BraTS annotation. Each row presents one patient and columns from left to right show the axial, sagittal and coronal views of a 3D volume, respectively. The green, yellow and red areas represent the whole tumor (WT), tumor core (TC) and enhancing tumor (ET), respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

(T1), post-contrast T1-weighted (T1Gd), T2-weighted (T2), and T2 fluid attenuated inversion recovery (FLAIR). The data were collected from 19 institutions by using various MRI scanners. The MR volumes are co-registered to the same anatomical template, interpolated to the same resolution and skull-stripped. The shape of MR volumes is 240 $\times$ 240 $\times$ 155 voxels. Experts are invited to outline the abnormal area. Examples of labels produced by experts are shown in Fig. 3. The colors in Fig. 3 represent different areas of tumors—the green, yellow and red stand for whole tumor (WT), tumor core (TC) and enhancing tumor (ET), respectively. This dataset is widely-used to evaluate the accuracy of segmentation methods for brain tumors. In our Rubik's cube+ pretext task, the MR volume is partitioned to eight (2 $\times$ 2 $\times$ 2) cubes of 64 $\times$ 64 $\times$ 64 voxels. As the BraTS-2018 has four modalities, we concatenate the cubes from different modalities and send them as input to each branch of the Siamese network. Consistent to BraTS-2018 challenge, several metrics, including the Dice coefficient, Hausdorff distance, sensitivity, and specificity, are adopted to assess the segmentation accuracy.

4.2. Experiment setting

4.2.1. Backbone network

For the cerebral hemorrhage classification, the backbone of our Rubik's cube+ network (Siamese-Octad) is either 3D VGG (Simonyan and Zisserman, 2015) or 3D ResNet-18 (He et al., 2016). Both are widely-used in self-supervised studies (Noroozi et al., 2018; Wei et al., 2019) and 3D medical image processing (Cicek et al., 2016; Milletari et al., 2016). For the brain tumor segmentation, two U-Nets with different encoders are adopted as the backbone—the larger U-Net (Kao et al., 2018), denoted as L-U-Net, consists of an encoder with 15 convolutional layers, while the smaller one, denoted as S-U-Net, uses 3D VGG as the encoder.

4.2.2. Baseline overview

The train-from-scratch strategy is adopted as the baseline. Furthermore, similar to the ImageNet pre-trained weights widely-used for 2D image processing, the action recognition video dataset, i.e.,

Table 2
The validation accuracies (ACC %) of solving 2 × 2 × 2 Rubik's cube+ task on two datasets. (Cer. hem.–Cerebral hemorrhage, O. ACC–Ordering ACC, R. ACC–Rotating ACC, M. ACC–Masking ACC).

Dataset	Model	Ordering	Rotating	Masking	Pretext task			Target task	
					O. ACC	R. ACC	M. ACC	ACC	Mean Dice
Cer. hem. dataset	3D VGG	√			95.80	–	–	73.31	–
			√		–	86.00	–	74.32	
				√	–	–	99.77	75.33	
			√	√	–	85.90	99.99	77.02	
		√		√	89.17	–	80.73	75.67	
		√	√		83.28	81.80	–	77.36	
		√	√	√	80.74	81.08	81.08	78.68	
	3D ResNet-18	√			95.29	–	–	85.81	
			√		–	91.97	–	82.77	
				√	–	–	98.81	80.74	
			√	√	–	84.58	99.94	84.80	
		√		√	86.11	–	85.71	86.15	
		√	√		84.22	82.12	–	87.50	
		√	√	√	85.81	83.20	78.08	87.84	
BraTS-2018	S-U-Net	√			99.99	–	–	–	78.10
			√		–	96.55	–		74.18
				√	–	–	99.99		73.94
			√	√	–	93.10	96.55		73.99
		√		√	93.75	–	82.50		75.52
		√	√		89.65	86.21	–		78.20
		√	√	√	86.21	86.21	75.86		79.62
	L-U-Net	√			99.98	–	–		77.17
			√		–	98.12	–		73.28
				√	–	–	99.99		71.90
			√	√	–	89.65	99.99		77.98
		√		√	95.00	–	86.25		80.49
		√	√		99.98	93.10	–		80.51
		√	√	√	96.56	89.65	96.55		81.70

UCF101 (Khurram et al., 2012), is adopted to pre-train our 3D CNNs. The UCF101 consists of 13,320 videos, which can be classified to 101 action categories. We extract frames from videos to form a cube of 112 × 112 × 16 to pre-train 3D networks. The pre-trained models are then transferred to the two target tasks for performance comparison. It is worthwhile to mention that our Rubik's cube+ pre-trained weights are generated by deeply exploiting useful information from limited training data without using any extra dataset. The 3D Jigsaw puzzles (Noroozi and Favaro, 2016) and orignal Rubik's cube (Zhuang et al., 2019) are also involved for comparison.

4.2.3. Implementation

The proposed self-supervised learning framework is implemented using PyTorch. The network is trained with a mini-batch size of 16. The initial learning rate is set to 0.001 and 0.0001 for the pretext and target tasks, respectively. The Adam solver (Kingma and Ba, 2014) is used for network optimization. For the classification task, since the input sizes of pretext and target tasks may be different, which are 64 × 64 × 8 and 270 × 230 × 30, respectively, on brain hermorrhage dataset for instance, we use adaptive average pooling layer by the end of 3D neural network to align the output shape of different training stages (pre-training and finetuning). Different to the classification task, we use FCN for brain tumor segmentation, which can handle input volumes with various sizes; therefore, no adaptive average pooling is required for the target task.

There are slight differences between the results reporeted in this paper and the early conference version (Zhuang et al., 2019), since we change the implementation toolbox from TensorFlow to PyTorch. The datasets are separated to training and validation sets according to the ratio of 80:20 for network training and performance evaluation. All the baseline approaches adopt the same training protocol for fair comparison.

4.3. Performance on solving Rubik's cube+

We evaluate the performance of the Siamese-Octad network on Rubik's cube recovery to verify whether the network can deal with the pretext task. The accuracies on the validation set of cerebral hemorrhage dataset and BraTS-2018 produced by different backbone networks for the three pretext sub-tasks are listed in Table 2. By comparing the validation accuracies, we have three observations:

First, the cube ordering accuracy significantly decreases as the complexity of the pretext task increases, i.e., the ordering accuracy of 3D VGG and ResNet-18 drop from 95.29% and 95.80% to 84.22% and 80.24%, respectively, by adding the cube rotation and masking, on brain hermorrhage dataset for instance. The underlying reason is the cube rotation and masking increases the difficulty of cube ordering. The network requires to exploit more useful 3D structural information to complete the pretext task in this situation, which results in a more robust feature representation—the accuracy of target task (ACC and mean DICE) increases due to the increasement of diversity of the pretext task.

Second, the 3D neural networks achieve statisfactory accuracies for the three sub-tasks—the 3D ResNet-18 achieves 84.22%, 82.12% and 78.08% for the cube ordering, rotating, and masking, respectively, while accuracies of 96.56%, 89.65% and 96.55% are achieved by the L-U-Net encoder on BraTS-2018. The experimental results demonstrate that the auxiliary tasks (cube rotation and masking) enable the network to develop the concept of rotated and blocked contents, which means more anatomical information is captured by the 3D neural networks compared to the ordering-only approach.

Last but not least, since the accuracy of pretext task represents the robustness of feature representation learned via self-supervised learning, the 3D ResNet-18 and L-U-Net, which achieve higher accuracy on the pretext tasks, surpass the 3D VGG and S-U-Net on the target tasks, respectively.

Table 3
The average classification accuracy (ACC %) and the area under curve (AUC) of 3D networks trained with different strategies on the validation set of cerebral hemorrhage dataset.

	3D VGG		3D ResNet-18	
	ACC	AUC	ACC	AUC
Train-from-scratch	72.30	0.889	79.73	0.902
UCF101 pre-trained	74.66	0.890	80.07	0.910
3D Jigsaw puzzles (Noroozi and Favaro, 2016)	73.31	0.891	85.81	0.951
Rubik's cube (Zhuang et al., 2019)	77.36	0.902	87.50	0.960
Rubik's cube+	**78.68**	**0.905**	**87.84**	**0.962**

Table 4
The segmentation accuracy of different approaches on the BraTS-2018 validation set. ↓ represents the lower value the better, and vice versa for ↑. (H. D.—Hausdorff distance, Sen.—Sensitivity, Spe.—Specificity).

Method	Dice (%) ↑			H. D. ↓			Sen. (%) ↑			Spe. (%) ↑		
	WT	TC	ET	WT	TC	ET	WT	TC	ET	WT	TC	ET
State-of-the-art												
DeepSCAN	75.38	56.71	46.79	52.28	61.68	38.80	67.32	62.02	62.66	**99.81**	99.57	99.60
No New-Net	88.33	75.58	**75.59**	23.42	15.26	16.35	**91.17**	75.98	81.92	99.50	99.81	99.80
VAE-CNN	87.37	80.08	74.80	34.61	18.02	16.60	87.93	83.81	79.16	99.61	99.77	99.88
C-A-Net	83.77	75.62	74.58	29.92	14.07	13.68	81.61	78.00	76.79	99.73	99.79	99.90
S-U-Net												
Train-from-scratch	85.10	74.47	73.21	38.17	30.39	28.02	82.38	79.05	78.51	99.69	99.77	99.87
UCF101 pre-trained	85.25	73.45	73.23	45.85	37.38	33.78	82.54	76.47	75.24	99.69	99.79	99.90
3D Jigsaw puzzles	85.63	74.69	73.99	38.84	32.09	31.13	85.20	76.63	75.22	99.56	99.83	**99.91**
Rubik's cube	85.91	74.62	74.07	47.39	46.57	43.87	84.06	81.93	81.30	99.65	99.72	99.85
Rubik's cube+	86.07	78.25	74.53	36.32	24.04	22.06	84.79	79.24	78.47	99.69	**99.86**	99.88
L-U-Net												
Train-from-scratch	85.47	72.26	74.39	30.31	21.02	19.88	83.09	72.46	77.68	99.72	99.73	99.86
UCF101 pre-trained	89.05	77.65	70.63	23.09	18.78	16.72	88.44	**87.04**	**83.78**	99.69	99.62	99.80
3D Jigsaw puzzles	86.54	72.61	72.38	33.10	19.18	19.43	84.84	74.03	78.60	99.67	99.78	99.85
Rubik's cube	89.26	77.35	74.94	23.67	16.19	15.56	88.04	78.89	78.99	99.70	99.82	99.87
Rubik's cube+	**89.60**	**80.42**	75.10	**22.32**	**13.88**	**13.26**	88.53	82.61	81.85	99.70	99.84	99.85

4.4. Performance on cerebral hemorrhage dataset

We fine-tune the networks pre-trained on the Rubik's cube+ pretext task for the cerebral hemorrhage classification. The average classification accuracy (ACC) and area under curve (AUC) of models trained with different training strategies are presented in Table 3.

It can be observed that the self-supervised approaches significantly improve the ACC, compared to the train-from-scratch. Specifically, the Rubik's cube+ pre-trained 3D VGG and ResNet-18 achieve ACCs of 78.68% and 87.84%, which are 6.38% and 8.11% higher than the train-from-scratch networks, respectively. Due to the gap between natural video and volumetric medical data, the improvement yielded by UCF101 pre-trained weights is limited, i.e., +2.36% and +0.34% for 3D VGG and ResNet-18, respectively. Similar trends of improvement are observed in terms of AUC. The Rubik's cube+ pre-trained 3D VGG and ResNet-18 achieve the highest AUC of 0.905 and 0.962, respectively. As the deeper network architecture often leads to better capacity of feature extraction, the 3D ResNet-18 achieves consistent better classification performance under different training strategies than the 3D VGG.

We also notice that the performance improvement is highly related to the diversity of the pretext task. The cube-ordering-only approach, i.e., 3D Jigsaw puzzles, yields +1.01% and +6.08% for 3D VGG and ResNet-18, respectively. In contrast, the networks pre-trained on more complicated pretext tasks, e.g., Rubik's cube and Rubik's cube+, achieve higher ACCs on cerebral hemorrhage dataset. The underlying reason is that the pretext tasks of Rubik's cube and Rubik's cube+ are more diverse so the network can learn more robust features under various transformations (translation, rotation and masking).

4.5. Performance on BraTS-2018

To further evaluate the effectiveness of the proposed approach, the Rubik's cube+ pre-trained networks are finetuned on the BraTS-2018 with pixel-wise manual-annotations for the brain tumor segmentation task. Apart from the L-U-Net and S-U-Net pre-trained with different self-supervised approaches, the state-of-the-art frameworks on BraTS dataset are also included for comparison, i.e., the DeepSCAN (McKinley et al., 2019), No New Net (Isensee et al., 2019), VAE-CNN (Myronenko, 2019), and C-A-Net Zhou et al. (2019a). The segmentation accuracy of different brain tumor areas yielded by different approaches are presented in Table 4.

As shown in Table 4, the pre-training yields consistent improvements to the segmentation accuracy of 3D FCNs on the BraTS-2018 validation set. A strong relationship is found between the performance improvement and the diversity of pretext task. The 3D neural networks do not need to fully exploited the intrinsic 3D anatomical information, while dealing with the easy-to-address pretext task, e.g., 3D Jigsaw puzzles, which only involves the translation transformation. As a result, such an easy-to-address pretext task yields marginal improvements for the target segmentation task. As the diversity of pretext task increases by adding rotation (i.e., Rubik's cube) and masking (i.e., Rubik's cube+), larger improvements are gained. Specifically, the Rubik's cube+ pre-trained L-U-Net achieves the relatively high Dice for whole tumor (89.60%), tumor core (80.42%) and enhancing tumor (75.10%), which are +4.13%, +8.16% and +0.71% higher than the train-from-scratch. We notice that although our Rubik's cube+ pre-trained L-U-Net surpasses the benchmarking algorithms under most metrics, it

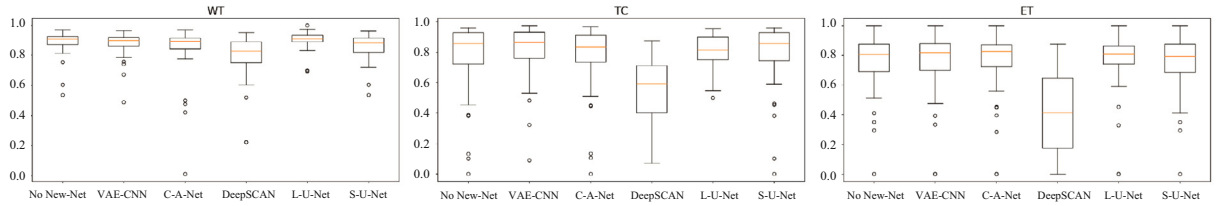

Fig. 4. The box plots of Dice yielded by state-of-the-art approaches and our Rubik's cube+ pre-trained L-U-Net and S-U-Net on BraTS-2018 validation set.

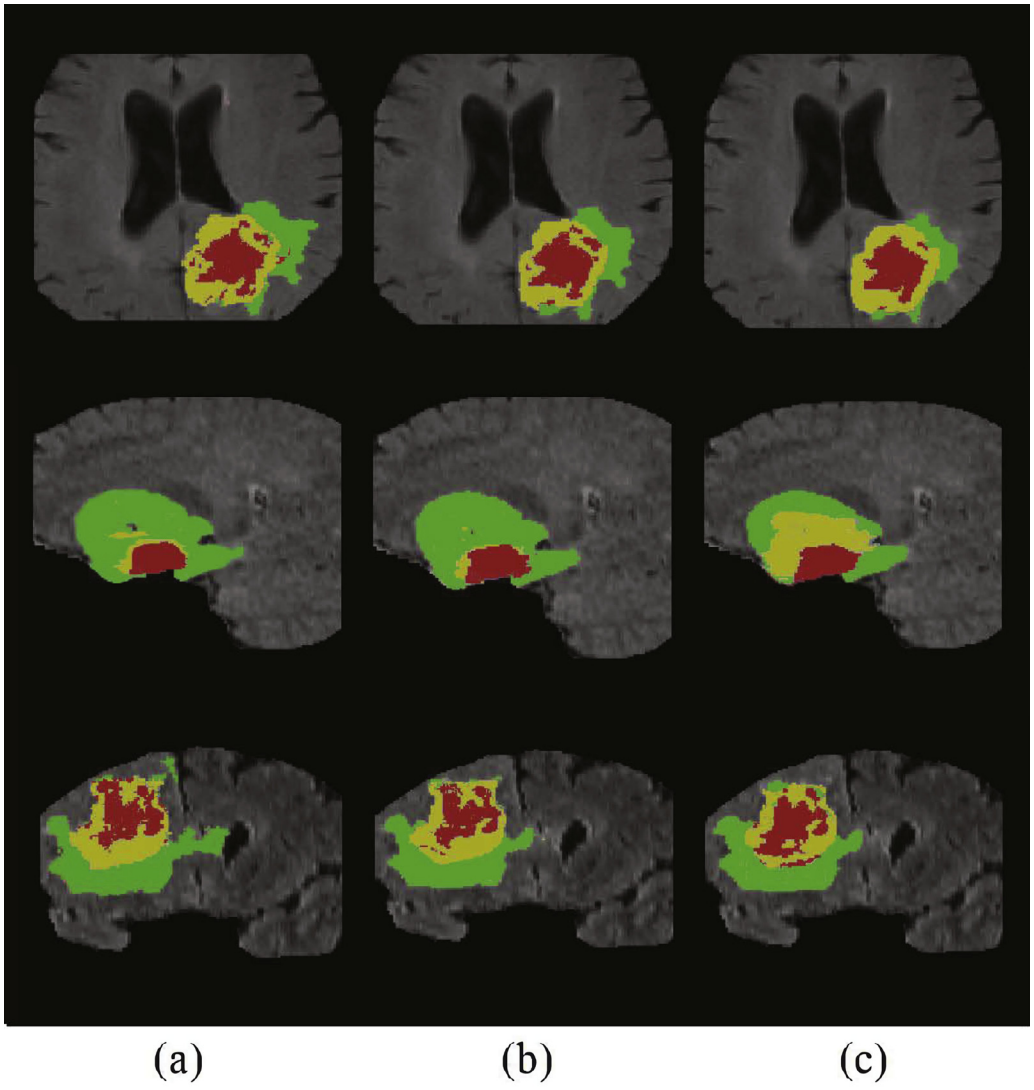

Fig. 5. Visualization of brain tumor segmentation results. The rows from top to bottom present the segmentation results viewed from *axial, sagittal* and *coronal* axes, respectively. (a) Ground truth, (b) segmentation result produced by L-U-Net pre-trained on our Rubik's cube+ pretext task, (c) segmentation result produced by the train-from-scratch L-U-Net. Colors represent different areas of tumor—the green, yellow and red stand for the whole tumor, tumor core and enhancing tumor, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

achieves lower sensitivity or specificity compared to some baselines. The underlying reason may be that our Rubik's cube+ pre-trained L-U-Net aims to maintain the relatively high values for both sensitivity and specificity, alleviating the problem of oversegmentation (high sensitivity and low specificity) and undersegmentation (low sensitivity and high specificity). The box plots of Dice yielded by state-of-the-art approaches and our Rubik's cube+ pre-trained L-U-Net and S-U-Net are presented in Fig. 4 for statistical analysis.

The segmentation results yielded by our Rubik's cube+ pre-trained L-U-Net and the train-from-scratch one are shown in Fig. 5. Compared to train-from-scratch, the L-U-Net pre-trained on our Rubik's cube+ pretext task produces more plausible segmentation results, especially for the tumor core and enhancing tumor.

4.5.1. Application of Rubik's cube+ to state-of-the-art approaches

To further demonstrate the effectiveness of our self-supervised learning approach, we pre-train the backbones of state-of-the-art approaches on our Rubik's cube+ pretext task and then finetune on the target brain tumor segmentation task. Note that the DeepSCAN is excluded as its backbone is 2D DenseNet. The evaluation results are shown in Table 5. It can be oberved that our Rubik's cube+ pre-

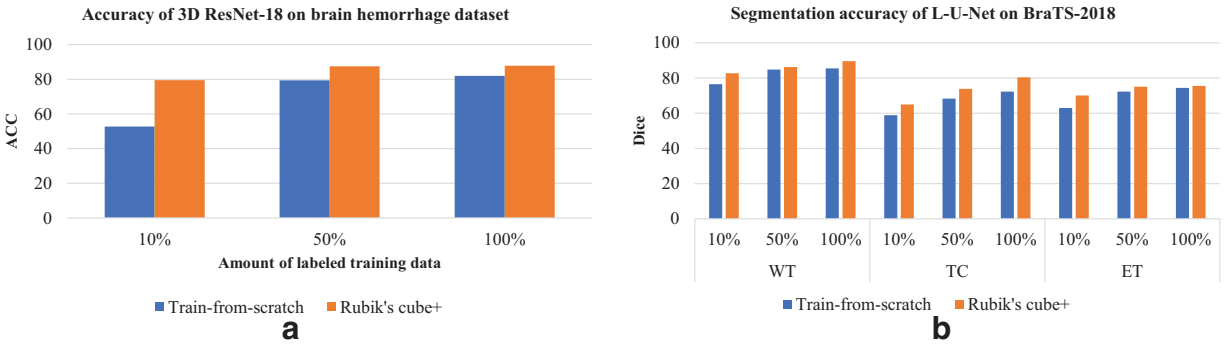

Fig. 6. The performances of networks trained with different amounts of labeled data on the two validation sets. (a) The ACCs of 3D ResNet-18 on brain hemorrhage dataset, and (b) the Dice coefficients yielded by L-U-Net on BraTS-2018.

Table 5
The segmentation accuracy of state-of-the-art frameworks trained with different strategies on the BraTS-2018 validation set. ↓ represents the lower value the better, and vice versa for ↑. (H. D.—Hausdorff distance, Sen.—Sensitivity, Spe.—Specificity).

Method	Dice (%) ↑			H. D. ↓			Sen. (%) ↑			Spe. (%) ↑		
	WT	TC	ET	WT	TC	ET	WT	TC	ET	WT	TC	ET
Train-from-scratch												
No New-Net	88.33	75.58	75.59	23.42	15.26	16.35	91.17	75.98	**81.92**	99.50	99.81	99.80
VAE-CNN	87.37	80.08	74.80	34.61	18.02	16.60	87.93	**83.81**	79.16	99.61	99.77	99.88
C-A-Net	83.77	75.62	74.58	29.92	**14.07**	**13.68**	81.61	78.00	76.79	**99.73**	99.79	**99.90**
With Rubik's cube+ pre-trained weights												
No New-Net	**88.76**	76.14	76.04	**22.87**	14.62	15.35	**91.87**	78.54	80.88	99.45	99.77	99.82
VAE-CNN	88.32	80.46	**77.74**	25.55	16.24	16.18	87.76	81.27	79.53	99.67	99.82	99.87
C-A-Net	88.73	**82.36**	76.94	32.88	16.83	13.74	88.22	79.55	79.99	99.67	**99.91**	99.89

text task consistently boosts the segmentation accuracy of state-of-the-art approaches. The Rubik's cube+ pre-trained No New Net, C-A-Net, and VAE-CNN achieve the best Dice for WT (88.76%), TC (82.36%), and ET (77.74%), respectively.

4.6. Performance with different amounts of training set

The proposed Rubik's cube+ self-supervised learning approach mainly aims to deal with the deficient training data. Hence, to better demonstrate the superiority of our Rubik's cube+ approach, an experiment is conducted to evaluate the segmentation performance of Rubik's cube+ pre-trained networks with different amounts of labeled data used for finetuning (i.e., 10% and 50%). The 3D ResNet-18 and L-U-Net are adopted as the backbone for the brain hermorrhage classification and brain tumor segmentation tasks, respectively, due to their excellent performance. The evaluation results are shown in Fig. 6.

It can be observed from Fig. 6 that our Rubik's cube+ can effectively deal with the situation with few labeled training samples. The Rubik's cube+ pre-trained 3D ResNet-18 and L-U-Net achieve an ACC of 87.50% and Dices of 86.26% (WT), 73.81% (TC) and 75.10% (ET) for brain hermorrhage classification and brain tumor segmentation, respectively, using 50% labeled data, which is comparable to the networks trained from scratch with 100% training data. For the case with extremely few labeled data (10%), the proposed Rubik's cube+ pretext task can still effectively improve the network performance—+26.80% for ACC and +6.13%, +6.02%, +6.99% for the Dice of WT, TC, and ET are observed, compared to the train-from-scratch.

4.7. How does the order of Rubik's cube matter?

In the previous experiments, we mainly explore the performance of a second-order ($2 \times 2 \times 2$) Rubik's cube+. According to the observation—the harder pretext task may lead to robust feature representation. Hence, in this section, we investigate the influence of increasing the order of Rubik's cube+ (i.e., the difficulty of pretext task) brought to the self-supervised feature representation learning.

Table 6
Detailed information of volume partition for Rubik's cube+ of different orders.

Dataset		Cerebral hemorrhage	BraTS-2018
Volume		$270 \times 230 \times 30$	$240 \times 240 \times 155$
Cube	$2 \times 2 \times 2$	$64 \times 64 \times 12$	$64 \times 64 \times 64$
	$3 \times 3 \times 3$	$40 \times 40 \times 8$	$40 \times 40 \times 40$

The volumetric data from brain hermorrhage dataset and BraTS-2018 is partitioned to third-order Rubik's cube+ ($3 \times 3 \times 3$) for comparison. Table 6 shows the cube shape of Rubik's cube+ in different orders. To evaluate the benefit produced by high-order Rubik's cube+, the 3D networks (i.e., 3D ResNet-18 and L-U-Net) are pre-trained to recover the third-order Rubik's cube+ and finetuned with 100% labeled training data on the two target tasks. The evaluation results are presented in Table 7.

Surprisingly, compared to the original Rubik's cube+ ($2 \times 2 \times 2$), the performance improvements yielded by pre-training Rubik's cube+ using the $3 \times 3 \times 3$ Rubik's cube slightly degrade, which is −0.68%, −1.79%, −5.48% and −0.32% for brain hermorrhage classification and the segmentation of whole tumor, tumor core, and enhancing tumor, respectively. The underlying reason may be that partitioning a volumetric data to a $3 \times 3 \times 3$ Rubik's cube severely wrecks the intrinsic anatomical information. In other words, the cube shape of $3 \times 3 \times 3$ Rubik's cube+ (i.e., $40 \times 40 \times 8$ and $40 \times 40 \times 40$ voxels for cerebral hemorrhage classification and brain tumor segmentation, respectively) is too small for 3D neural networks to extract useful information, which neutralizes the performance improvement generated by the pre-training.

Table 7
The classification accuracy (ACC %), the area under curve (AUC) and segmentation accuracy (Dice %) of transferring the 3 × 3 × 3 Rubik's cube+ to the target tasks.

	Cube size	3D ResNet-18		L-U-Net		
		ACC	AUC	WT	TC	ET
Train-from-scratch	-	79.73	0.902	85.47	68.35	72.26
3D Jigsaw puzzles (Noroozi and Favaro, 2016)	3 × 3 × 3	83.11	0.935	86.49	71.53	73.83
Rubik's cube (Zhuang et al., 2019)		84.12	0.941	87.25	75.81	73.40
Rubik's cube+		87.16	0.959	87.81	74.94	74.78
Rubik's cube+	2 × 2 × 2	**87.84**	**0.962**	**89.60**	**80.42**	**75.10**

Hence, we raise an observation—simply increasing the difficulty of the pretext task does not always help, but increasing the diversity of the pretext task is beneficial. The combination of diverse transformations (i.e., cube ordering, rotating and masking) can remarkablely improve the robustness of feature representation learned by the self-supervised pretext task—our Rubik's cube+ invariably outperforms the Rubik's cube and 3D Jigsaw puzzles, while simply increasing the difficulty (e.g., high-order Rubik's cube) may destruct the 3D structure, decrease the anatomical information and consequently degrade the improvements yielded by self-supervised learning.

5. Conclusion

In this paper, we proposed a novel self-supervised learning pretext task (i.e., Rubik's cube+), involving three transformations—cube ordering, rotating and masking, to pre-train 3D neural networks for volumetric medical images. The former two transformations forced the 3D neural networks to learn the translation and rotation invariant features from the original 3D medical data, while the latter one improved the model generalization. The proposed Rubik's cube+ was tested on two datasets for different target tasks, i.e., cerebral hemorrhage classification and brain tumor segmentation. The experimental results showed that our self-supervised learning method remarkablely outperformed the train-from-scratch method without using extra data.

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.

CRediT authorship contribution statement

Jiuwen Zhu: Software, Investigation, Writing - original draft. **Yuexiang Li:** Conceptualization, Methodology, Formal analysis, Supervision. **Yifan Hu:** Conceptualization, Data curation. **Kai Ma:** Resources, Writing - review & editing. **S. Kevin Zhou:** Supervision, Writing - review & editing. **Yefeng Zheng:** Supervision, Writing - review & editing.

Acknowledgements

The work was supported by the Natural Science Foundation of China (No. 61702339), and the Key Area Research and Development Program of Guangdong Province, China (No. 2018B010111001).

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.media.2020.101746.

References

Carlucci, F.M., D'Innocente, A., Bucci, S., Caputo, B., Tommasi, T., 2019. Domain generalization by solving jigsaw puzzles. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2229–2238.
Chen, L., Bentley, P., Mori, K., Misawa, K., Fujiwara, M., Rueckert, D., 2019. Self-supervised learning for medical image analysis using image context restoration. Med. Image Anal. 58, 101539.
Cicek, O., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In: International Conference on Medical Image Computing & Computer Assisted Intervention, pp. 424–432.
Deng, J., Berg, A.C., Li, K., Li, F.F., 2010. What does classifying more than 10,000 image categories tell us? In: European Conference on Computer Vision, pp. 71–84.
Doersch, C., Gupta, A., Efros, A.A., 2015. Unsupervised visual representation learning by context prediction. In: IEEE International Conference on Computer Vision, pp. 1422–1430.
Dou, Q., Yu, L., Chen, H., Jin, Y., Yang, X., Qin, J., Heng, P.-A., 2017. 3D deeply supervised network for automated segmentation of volumetric medical images. Med. Image Anal. 41, 40–54.
Feng, Z., Xu, C., Tao, D., 2019. Self-supervised representation learning by rotation feature decoupling. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 10364–10374.
Fernando, B., Bilen, H., Gavves, E., Gould, S., 2017. Self-supervised video representation learning with odd-one-out networks. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5729–5738.
Gao, X.W., Hui, R., Tian, Z., 2017. Classification of CT brain images based on deep learning networks. Comput. Methods Programs Biomed. 138, 49–56.
Gidaris, S., Singh, P., Komodakis, N., 2018. Unsupervised representation learning by predicting image rotations. In: International Conference on Learning Representations.
Graeter, T., Kratzer, W., Oeztuerk, S., Haenle, M.M., Mason, R.A., Hillenbrand, A., Kull, T., Barth, T.F., Kern, P., Gruener, B., 2016. Proposal of a computed tomography classification for hepatic alveolar echinococcosis. World J. Gastroenterol. 22 (13), 3621–3631.
Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.M., Larochelle, H., 2017. Brain tumor segmentation with deep neural networks. Med. Image Anal. 35, 18–31.
He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.
Isensee, F., Kickingereder, P., Wick, W., Bendszus, M., Maier-Hein, K., 2019. No new-net. In: Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, pp. 234–244.
Ji, X., Zheng, G., Belavy, D., Ni, D., 2016. Automated intervertebral disc segmentation using deep convolutional neural networks. In: Computational Methods and Clinical Applications for Spine Imaging, pp. 38–48.
Kamnitsas, K., Ferrante, E., Parisot, S., Ledig, C., Nori, A.V., Criminisi, A., Rueckert, D., Glocker, B., 2016. DeepMedic for brain tumor segmentation. In: International Workshop on Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, pp. 138–149.
Kamnitsas, K., Ledig, C., Newcombe, V.F.J., Simpson, J.P., Kane, A.D., Menon, D.K., et al., 2016. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Med. Image Anal. 36, 61–78.
Kao, P. Y., Ngo, T., Zhang, A., Chen, J., Manjunath, B. S., 2018. Brain tumor segmentation and tractographic feature extraction from structural MR images for overall survival prediction. arXiv:1807.07716.
Khurram, S., Zamir, A. R., Shah, M., 2012. UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv:1212.0402.
Kim, D., Cho, D., Yoo, D., Kweon, I.S., 2018. Learning image representations by completing damaged jigsaw puzzles. In: IEEE Winter Conference on Applications of Computer Vision, pp. 793–802.
Kingma, D. P., Ba, J., 2014. Adam: a method for stochastic optimization. arXiv:1412.6980.
Kumar, D., Wong, A., Clausi, D.A., 2015. Lung nodule classification using deep features in CT images. In: IEEE Conference on Computer and Robot Vision, pp. 133–138.
Larsson, G., Maire, M., Shakhnarovich, G., 2017. Colorization as a proxy task for visual understanding. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 840–849.

Lee, H.Y., Huang, J.B., Singh, M., Yang, M.-H., 2017. Unsupervised representation learning by sorting sequences. In: IEEE International Conference on Computer Vision, pp. 667–676.
McKinley, R., Meier, R., Wiest, R., 2019. Ensembles of densely-connected CNNs with label-uncertainty for brain tumor segmentation. In: Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, pp. 456–465.
Menze, B.H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al., 2015. The multimodal brain tumor image segmentation benchmark (BraTS). IEEE Trans. Med. Imaging 34 (10), 1993–2024.
Milletari, F., Navab, N., Ahmadi, S.-A., 2016. V-Net: fully convolutional neural networks for volumetric medical image segmentation. In: International Conference on 3D Vision, pp. 565–571.
Myronenko, A., 2019. 3D MRI brain tumor segmentation using autoencoder regularization. In: Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, pp. 311–320.
Noroozi, M., Favaro, P., 2016. Unsupervised learning of visual representations by solving jigsaw puzzles. In: European Conference on Computer Vision, pp. 69–84.
Noroozi, M., Vinjimoor, A., Favaro, P., Pirsiavash, H., 2018. Boosting self-supervised learning via knowledge transfer. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 9359–9367.
Paikin, G., Tal, A., 2015. Solving multiple square jigsaw puzzles with missing pieces. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 4832–4839.
Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A., 2016. Context encoders: feature learning by inpainting. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2536–2544.
Ritter, F., Berendt, B., Fischer, B., Richter, R., Preim, B., 2002. Virtual 3D jigsaw puzzles: Studying the effect of exploring spatial relations with implicit guidance. In: Mensch & Computer, pp. 363–372.
Simonyan, K., Zisserman, A., 2015. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.
Son, K., Hays, J., Cooper, D.B., et al., 2016. Solving small-piece jigsaw puzzles by growing consensus. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1193–1201.
Spitzer, H., Kiwitz, K., Amunts, K., Harmeling, S., Dickscheid, T., 2018. Improving cytoarchitectonic segmentation of human brain areas with self-supervised Siamese networks. In: International Conference on Medical Image Computing & Computer Assisted Intervention, pp. 663–671.
Wei, C., Xie, L., Ren, X., Xia, Y., Su, C., Liu, J., Tian, Q., Yuille, A.L., 2019. Iterative reorganization with weak spatial constraints: Solving arbitrary jigsaw puzzles for unsupervised representation learning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1910–1919.
Xie, Y., Xia, Y., Zhang, J., Song, Y., Feng, D., Fulham, M., et al., 2019. Knowledge-based collaborative deep learning for benign-malignant lung nodule classification on chest CT. IEEE Trans. Med. Imaging 38 (4), 991–1004.
You, Z., Ye, J., Li, K., Xu, Z., Wang, P., 2019. Adversarial noise layer: regularize neural network by adding noise. In: IEEE International Conference on Image Processing, pp. 909–913.
Zhan, X., Pan, X., Liu, Z., Lin, D., Loy, C.C., 2019. Self-supervised learning via conditional motion propagation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1881–1889.
Zhang, L., Qi, G.J., Wang, L., Luo, J., 2019. AET vs. AED: unsupervised representation learning by auto-encoding transformations rather than data. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2547–2555.
Zhang, P., Wang, F., Zheng, Y., 2017. Self supervised deep representation learning for fine-grained body part recognition. In: IEEE International Symposium on Biomedical Imaging, pp. 578–582.
Zhou, C., Chen, S., Ding, C., D., T., 2019. Learning contextual and attentive information for brain tumor segmentation. In: Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries, pp. 497–507.
Zhou, Z., Sodha, V., Siddiquee, M.M.R., Feng, R., Tajbakhsh, N., Gotway, M.B., Liang, J., 2019. Models genesis: generic autodidactic models for 3D medical image analysis. In: International Conference on Medical Image Computing & Computer Assisted Intervention, pp. 384–393.
Zhuang, X., Li, Y., Hu, Y., Ma, K., Yang, Y., Zheng, Y., 2019. Self-supervised feature learning for 3D medical images by playing a Rubik's cube. In: International Conference on Medical Image Computing & Computer Assisted Intervention, pp. 420–428.
Zur, R.M., Jiang, Y., Pesce, L.L., Drukker, K., 2009. Noise injection for training artificial neural networks: a comparison with weight decay and early stopping. Med. Phys. 36, 4810–4818.