Learning From Partially Labeled Data for Multi-Organ and Tumor Segmentation

Yutong Xie, Jianpeng Zhang, Yong Xia, Member, IEEE, and Chunhua Shen

Abstract—Medical image benchmarks for the segmentation of organs and tumors suffer from the partially labeling issue due to its intensive cost of labor and expertise. Current mainstream approaches follow the practice of one network solving one task. With this pipeline, not only the performance is limited by the typically small dataset of a single task, but also the computation cost linearly increases with the number of tasks. To address this issue, we propose a Transformer based dynamic on-demand network (TransDoDNet) that learns to segment organs and tumors on multiple partially labeled datasets. Specifically, TransDoDNet has a hybrid backbone that is composed of the convolutional neural network and Transformer. A dynamic head enables the network to accomplish multiple segmentation tasks flexibly. Unlike existing approaches that fix kernels after training, the kernels in the dynamic head are generated adaptively by the Transformer, which employs the self-attention mechanism to model long-range organ-wise dependencies and decodes the organ embedding that can represent each organ. We create a large-scale partially labeled Multi-Organ and Tumor Segmentation benchmark, termed MOTS, and demonstrate the superior performance of our TransDoDNet over other competitors on seven organ and tumor segmentation tasks. This study also provides a general 3D medical image segmentation model, which has been pre-trained on the large-scale MOTS benchmark and has demonstrated advanced performance over current predominant self-supervised learning methods.

Index Terms—Dynamic convolution, medical image segmentation, partial labeling, transformer.

I. INTRODUCTION

AUTOMATED segmentation of abdominal organs and tumors is one of the most fundamental yet challenging tasks in medical image analysis. It plays a pivotal role in a variety of computer-aided diagnosis jobs, including lesion contouring, surgical planning, and 3D reconstruction. Most recently, it has witnessed rapid progress on this segmentation task, driven by both deep learning algorithms and public benchmark datasets [2], [3]. Constrained by the labor cost and expertise, 3D medical image segmentation benchmarks, however, severely suffer from the limited annotations on every single dataset. In this work, we attempt to collect multiple organs and tumor segmentation datasets to form a large-scale but more challenging multi-organ and tumor segmentation benchmark, termed MOTS.

Unfortunately, most medical datasets were collected for the segmentation of only one type of organ and/or tumors, and all task-irrelevant objects were treated as the background (see Fig. 1). For instance, the LiTS dataset [4] only has the annotations of the liver and its tumors, and the KiTS dataset [5] only provides the annotations of kidneys and its tumors. As shown in Fig. 1, these partially labeled medical image datasets are different from fully labeled benchmarks in other computer vision areas, such as Cityscapes [1], where multiple types of objects were annotated on each image. Therefore, one of the most significant challenges facing multi-organ and tumor segmentation is the so-called partially labeling issue, i.e., learning the representation of multiple organs and tumors under the supervision of these partial annotations.

Mainstream approaches address this issue by separating the partially labeled dataset into several subsets and training a network on each subset for a specific segmentation task [6], [7], [8], [9], [10], shown in Fig. 2(a). Following the one net for one task pipeline, each subset is fully labeled for the corresponding segmentation task. Such a strategy is intuitive but increases the computational complexity dramatically. Another commonly-used solution is to design a multi-head network (see Fig. 2(b)), which is composed of a shared encoder and multiple task-specific decoders (heads) [11], [12], [13]. In the training stage, feeding each partially labeled data to the network triggers the update of only one head, while other heads are frozen. An obvious drawback of a multi-head network is that the number of segmentation heads increases with the number of tasks, leading to higher computational and spatial complexity. Besides, the inflexible multi-head architecture is not easy to be extended to new tasks.
Fig. 1. Comparison of fully labeled and partially labeled segmentation. In the fully labeled annotation, taking Cityscapes [1] for example (a), each pixel is assigned a unique and true category label. As for the partially labeling, only the objects of interest have true category labels, other objects are regarded as the background.

In this work, we aim to segment multiple organs and tumors by learning from several partially labeled datasets (c), each of which is originally specialized for the segmentation of a particular abdominal organ and/or related tumors. Here each color represents a partially labeled dataset.

Fig. 2. Three types of methods to perform M partially labeled segmentation tasks. (a) One net for one task: training M networks on M partially labeled subsets, respectively; (b) one net with multi-heads: training one network that consists of a shared encoder and M task-specific decoders (heads), each head, together with the shared encoder, performing a partially labeled segmentation task; and (c) the proposed one net with a dynamic head: it has an encoder, an organ embedding module, a kernel generator, and a dynamic segmentation head. The kernels in the dynamic head are conditioned on the input images and assigned organ embedding.

In our pilot study [14], we proposed a dynamic on-demand network (DoDNet), which can be trained on partially labeled datasets for multi-organ and tumor segmentation. DoDNet has a simple but efficient encoder-decoder architecture. Different from previous solutions, its decoder is followed by a single but dynamic segmentation head (see Fig. 2(c)). We designed a convolution-based kernel generator to generate adaptively the kernels of the dynamic head for the segmentation of each organ, as shown in Fig. 3(a). Consequently, the dynamic segmentation head enables DoDNet to segment multiple organs and tumors as done by multiple networks or a multi-head network. However, these task-specific kernels are generated individually, ignoring the organ-wise dependencies.

In this paper, we address this issue by introducing a novel Transformer based kernel generator, resulting in TransDoDNet. With the self-attention mechanism, TransDoDNet is superior to its predecessor DoDNet in modeling the long-range organ-wise dependencies, as shown in Fig. 3(b). The kernels generated by the Transformer are not only conditioned on its own task but also mutually influenced by the dependencies on other tasks. We evaluate the effectiveness of TransDoDNet on MOTS, which is an ensemble of multiple organs and tumors segmentation benchmarks, involving the liver and tumors, kidneys and tumors, hepatic vessels and tumors, pancreas and tumors, colon tumors, and spleen. We also transfer the weights of TransDoDNet pre-trained on MOTS to downstream annotation-limited medical image segmentation tasks. The results show that our model has an excellent generalization ability, beating the current predominant self-supervised learning method [15]. Our contributions are four-fold.
• We create a large-scale multi-organ and tumor segmentation dataset by combining multiple partially labeled ones, resulting in a new and challenging medical image segmentation benchmark called MOTS.

• We address the partially labeling issue from a new perspective, i.e., proposing a single network that has a dynamic segmentation head to segment multiple organs and tumors as done by multiple networks or a multi-head network.

• We are the first to employ the Transformer, which has a comprehensive perception of the global task-contextual information, as a kernel generator to achieve dynamic convolutions, contributing to better performance.

• A byproduct of this work, i.e., the model weights pretrained on MOTS, can be transferred to downstream 3D image segmentation tasks even with different image modalities, and hence is beneficial for the community.

II. RELATED WORK

A. Partially Labeled Medical Image Segmentation

Segmentation of multiple organs and tumors is a generally recognized difficulty in medical image analysis [16], [17], [18], [19], particularly when there are no large-scale fully labeled datasets. Although several partially labeled datasets are available, each of them is specialized for the segmentation of one particular organ and/or tumors. Accordingly, a segmentation model is usually trained on a partially labeled dataset, and hence is only able to segment one particular organ and tumors, such as the liver and liver tumors [8], [20], [21], [22], kidneys and kidney tumors [9], [23]. Training multiple networks, however, requires high computational resources and has poor scalability.

To address this issue, many attempts have been made to explore multiple partially labeled datasets in a more efficient manner. Chen et al. [11] collected multiple partially labeled datasets from different medical domains, and co-trained a heterogeneous 3D network on them, which is specially designed with a task-shared encoder and task-specific decoders for eight segmentation tasks. Huang et al. [24] proposed to co-train a pair of weight-averaged models for unified multi-organ segmentation on few-organ datasets. Zhou et al. [25] first approximated anatomical priors of the size of abdominal organs on a fully labeled dataset, and then regularized the organ size distributions on several partially labeled datasets. Fang and Yan [12] treated the voxels with unknown labels as the background, and then proposed the target adaptive loss (TAL) for a segmentation network that is trained on multiple partially labeled datasets. Shi et al. [13] merged unlabeled organs with the background and imposed an exclusive constraint on each voxel (i.e., each voxel belongs to either one organ or the background) to learn a segmentation model jointly on a fully labeled dataset and several partially labeled datasets. To learn multi-class segmentation from single-class datasets, Dmitriev et al. [26] utilized the segmentation task as a prior and incorporated it into the intermediate activation signal.

The proposed TransDoDNet is different from these methods in four main aspects: (1) In [12], [13], the partially labeling issue is formulated into a multi-class segmentation task, and unlabeled organs are treated as the background. This formulation may be misleading since the unlabeled organ in one dataset is indeed the foreground in another dataset. To amend this error, we formulate the partially labeling issue as a single-class segmentation task, aiming to segment each organ respectively; (2) Most related work adopts a multi-head architecture, which is composed of a shared backbone network and multiple segmentation heads for different tasks. Each head is either a decoder [11] or the last segmentation layer [12], [13]. In contrast, the proposed TransDoDNet is a single-head network, in which the head is dynamic and can be generated adaptively; (3) Our TransDoDNet uses the dynamic segmentation head to address the partially labeling issue, instead of embedding the task prior into the encoder and/or decoder; (4) Most existing methods focus on multi-organ segmentation, while our TransDoDNet is designed for the segmentation of both organs and tumors, which is more challenging.

B. Dynamic Filter Learning

Dynamic filter learning has drawn considerable research attention in the computer vision community due to its adaptive nature [27], [28], [29], [30], [31], [32]. Jia et al. [27] designed a dynamic filter network, in which the filters are generated dynamically conditioned on the input. This design is more flexible than traditional convolutional networks, where the learned filters are fixed during the inference. Yang et al. [28] introduced the conditionally parameterized convolution, which learns specialized convolutional kernels for each input and effectively increases the size and capacity of a convolutional neural network (CNN). Chen et al. [29] presented another dynamic network, which dynamically generates attention weights for multiple parallel convolution kernels and assembles these kernels to strengthen the representation capability. Pang et al. [32] integrated the features of RGB images and depth images to generate dynamic filters for better use of cross-modal fusion information in RGB-D salient object detection. Tian et al. [30] applied the dynamic convolution to instance segmentation, where the filters in the mask head are dynamically generated for each target instance. These methods successfully employ dynamic filter learning toward certain ends, such as increasing the network flexibility [27], enhancing the representation capacity [28], [29], integrating cross-modal fusion information [32], or abandoning the use of instance-wise ROIs [30]. Comparing with these works, our work here differs as follows. 1) We employ the dynamic filter learning to address the partially labeling issue for 3D medical image segmentation; and 2) the dynamic filters generated by Transformer, lead to better organ-wise dependency modeling.

C. Vision Transformers

The transformer is a popular architecture commonly used in natural language processing [33], owing to its superior ability to model long-range dependencies. Recently, Transformer has also been widely used in the computer vision community as an alternative to CNN and achieves comparable or even state-of-the-art performance on various tasks, including image recognition [34], [35], semantic segmentation [36], [37], object...
A. Problem Definition

Multiple organs and tumor segmentation aim to predict the pixel-level masks of each organ and corresponding tumors in the image. Given a medical image \( \mathbf{X} \) and its pixel-level annotations \( \mathbf{Y} \) sampled from the dataset \( \mathcal{D} \), this task can be easily optimized by minimizing a loss, usually, Dice loss or cross-entropy loss, as following

\[
\min_{\Theta} \mathbb{E}_{(\mathbf{X}, \mathbf{Y}) \sim \mathcal{D}}[\mathcal{L}(f(\mathbf{X}; \Theta), \mathbf{Y})]
\]

In this case, \( \mathbf{Y} \) should be a full annotation that covers the pixel-level mask of multiple organs and tumors.

Let’s consider \( M \) partially labeled datasets \( \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_M\} \), which were collected from \( M \) organ and tumor segmentation tasks:

\[
\{1: \text{organ}; 2: \text{tumor}; 3: \text{organ & tumor}\}.
\]

Here, \( \mathcal{D}_m = \{\mathbf{X}_i, \mathbf{Y}_i\}_{i=1}^{n_m} \) represents the \( m \)-th partially labeled dataset that contains \( n_m \) labeled images. Given an image sampled from \( \mathcal{D}_m \), we denote it as \( \mathbf{X} \in \mathbb{R}^{W \times H} \), where \( W \times H \) is the spatial size of each slice and \( D \) is number of slices. The corresponding segmentation ground truth is \( \mathbf{Y} \in \{0, 1, 2\}^{D \times W \times H} \), where \( D \) is the number of slices and \( D \leq H \).

Straightforwardly, this partially labeled multi-organ and tumor segmentation problem can be solved by training \( M \) segmentation networks \( \{f_1, f_2, \ldots, f_M\} \) on \( M \) datasets, respectively, shown as follows

\[
\begin{aligned}
\min_{\Theta_1} & \mathbb{E}_{(\mathbf{X}, \mathbf{Y}) \sim \mathcal{D}_1}[\mathcal{L}(f_1(\mathbf{X}; \Theta_1), \mathbf{Y})] \\
\vdots \\
\min_{\Theta_M} & \mathbb{E}_{(\mathbf{X}, \mathbf{Y}) \sim \mathcal{D}_M}[\mathcal{L}(f_M(\mathbf{X}; \Theta_M), \mathbf{Y})]
\end{aligned}
\]

where \( \mathcal{L} \) represents the loss function of each network, \( \{\Theta_1, \Theta_2, \ldots, \Theta_M\} \) represent the parameters of these \( M \) networks. In this work, we attempt to address this problem using only one single network \( f \) with the parameter \( \Theta \), which can be formally expressed as

\[
\min_{\Theta} \mathbb{E}_{(\mathbf{X}, \mathbf{Y}) \sim \mathcal{D}}[\mathcal{L}(f(\mathbf{X}; \Theta), \mathbf{Y})]
\]

The TransDoDNet proposed here for this purpose consists of a shared CNN backbone, a Transformer, a filter prediction head, and a dynamic segmentation head (see Fig. 4). We now delve into the details of each part.

B. Architecture

The main component of TransDoDNet is a shared CNN backbone that has a U-like encoder-decoder architecture [43]. The encoder is composed of repeated applications of 3D residual blocks [44], which has 50 learnable layers. Different from the vanilla ResNet, we employ the instance normalization [45] to replace the batch normalization as done in [7]. Such a normalization strategy enables our model to be trained with a very small batch size. As for the input image \( \mathbf{X} \), its feature map is calculated by

\[
\mathbf{F} = f_E(\mathbf{X}; \Theta_E)
\]

where \( \mathbf{F} \in \mathbb{R}^{C_1 \times D \times W \times H} \), and \( \Theta_E \) represents the CNN encoder parameters. Besides, we employ the Transformer at the bottleneck to model the long-range feature dependencies, expressed by

\[
\mathbf{Z} = f_{TE}(\mathbf{F}; \Theta_{TE})
\]

where \( \Theta_{TE} \) represents the parameters of Transformer encoder. The details of the Transformer are presented in Section III-C.

To decode the image feature map to the segmentation mask, we upsample the feature map to improve its resolution and halve its channel number step by step. At each step, the upsampled feature map is first summed with the corresponding low-level feature map from the encoder and then refined by a residual block. After the feed-forward of the CNN decoder, we have

\[
\mathbf{G} = f_D(\mathbf{F} + \mathbf{Z}; \Theta_D)
\]

where \( \mathbf{G} \in \mathbb{R}^{C_2 \times D \times W \times H} \) is the pre-segmentation feature map, \( \Theta_D \) represents all decoder parameters, and the channel number \( C_2 \) is set to 8 (see ablation study in Section IV-D1).

The CNN encoder-decoder aims to generate \( \mathbf{G} \), which is supposed to be rich-semantic and not subject to a specific task, i.e., containing the semantic information of multiple organs and tumors.

C. Transformer Based Kernel Generator

1) Volume Positional Encoding: To feed the 3D feature map \( \mathbf{F} \) into the Transformer, we first reduce the channel dimension from \( C_1 \) to \( d \) by using a convolution layer, i.e., \( \phi(\mathbf{F}) : \mathbf{F} \in \mathbb{R}^{C_1 \times D \times W \times H} \rightarrow \mathbf{S} \in \mathbb{R}^{d \times D \times W \times H} \), where \( d << C_1 \). Then we flatten the inter-slice and spatial dimensions and transpose it to a 1D sequence, \( \mathbf{S} \in \mathbb{R}^{(D \times W \times H) \times d} \). However, such a 1D sequence is a collection of pixel-level feature maps, suffering a lack of spatial structure information.

To address this issue, we augment the pixel-wise feature maps with the positional encoding that contains three dimensional positional information, including inter-slice, spatial width, and spatial height. We employ the fixed positional encoding strategy as done in [33], and extend it to three dimensions to work on the 3D scenario. For each dimension \( \# \in \{D, W, H\} \), its fixed
The positional encoding is calculated by
\[
PE^#_{(pos, 2k)} = \sin(pos/10000^{2k}/2) \\
PE^#_{(pos, 2k+1)} = \cos(pos/10000^{2k}/2)
\]
(7)

We concatenate the positional encodings of three dimensions and then flatten them as a sequence
\[
E_{pos} = \text{Flat}(PE^D||PE^W||PE^H)
\]
(8)

where || represents the concatenation operation, Flat(·) is the flatten operation, and \(E_{pos} \in \mathbb{R}^{(D \times W \times H) \times d}\).

2) Attention Mechanism: Self-attention: The self-attention mechanism is one of the key components in the Transformer [33]. It models the similarity information of pixel-wise feature maps to all positions. Given a feature map sequence with \(N\) tokens, \(S \in \mathbb{R}^{N \times d}\), we define the self-attention process at \(i\)-th head as
\[
[q, k, v] = SU_{qkv}
\]
\[
Attention_i = \text{softmax} \left( \frac{qk^\top}{\sqrt{dk}} \right) v
\]
(9)

Parallel execution of multiple attention heads can extend the capabilities of attention mechanisms, formulated as
\[
SA(q, k, v) = \rho^s(Attention_1||...||Attention_k)
\]
(10)

where \(U_{qkv} : \mathbb{R}^d \rightarrow \mathbb{R}^{3d_h}, \rho^o : \mathbb{R}^{hd_k} \rightarrow \mathbb{R}^d\) are learnable feature projections, and \(dk = \frac{d}{2}\). The Transformer with the self-attention mechanism would look over all possible locations in the feature map to explore the pixel-to-pixel level attention relationship, suffering from slow convergence and high computational complexity.

Deformable attention: Inspired by [39], [46], we adopt the deformable multi-head attention in the Transformer to facilitate efficient training. The deformable multi-head attention enables the model to efficiently and flexibly attend to information from different representation subspaces at a few sparse positions, i.e., a small set of key sampling locations automatically selected by the model itself. Given the feature map sequence \(S\), the deformable multi-head attention of query \(q\) to a set of key elements \(K\) is calculated as
\[
DA(q, S) = \rho^s(Attention_1||...||Attention_k)
\]
\[
Attention_i = \sum_{k=1}^K A(q)_k \cdot \rho^s(S)(p + \Delta_{pk})
\]
(11)

where \(A(q)_k\) is the learnable attention weights, \(\rho^s\) and \(\rho^o\) are learnable feature projections, \(p\) is a 3D reference point, and \(\Delta_{pk}\) denotes the learnable sampling offsets.

Multi-scale deformable attention: We denote \(S' = \{S_1, S_2, ..., S_L\}\) as the multi-scale feature maps extracted from the different CNN encoder stages. Here \(L\) is the number of multi-scale levels. We extend the single-scale attention mechanism to a multi-scale version, expressed as
\[
MSDA(q, S') = \rho^s(Attention_1||...||Attention_k)
\]
\[
Attention_i = \sum_{l=1}^L \sum_{k=1}^K A(q)_lk \cdot \rho^s(S'_l)(p_l + \Delta_{pk})
\]
(12)

where \(p_l\) represents the reference points at the \(l\)-th level feature map. Compared to single-scale attention, multi-scale deformable attention is more flexible and powerful in observing all possible locations in the multi-scale feature maps.

3) Transformer Encoder: The Transformer encoder is composed of the alternating layers of multi-scale deformable attention layer and fully connected feed-forward network (FFN). To speed up the training process, residual connections [33], [44] are performed across each of the multi-head attention layer and MLP layer, followed by the layer normalization [47]. Specifically, we follow the deep norm strategy [48] to stabilize the deep Transformers. Given a \(L_{enc}\)-layer encoder, the feed-forward process of each Transformer layer can be formulated as
\[
q_0 = S' \\
q_i = z_{i-1} + E_{pos}, x_i = z_{i-1} \\
z_i' = N(MSDA(q_i, x_i) + z_{i-1} \ast \alpha) \\
z_i = N(FFN(z_i') + z_i' \ast \alpha)
\]
(13)
where \( i = 1, \ldots, I_{\text{enc}} \), \( \mathbb{N} \) represents the layer normalization, and \( \alpha \) is a scaling factor of residual connections that is conditioned on the number of Transformer encoder and decoder layers.

4) Transformer Decoder: The Transformer decoder follows a similar architecture as the encoder, including two multi-head attention layers and one MLP layer. Besides, we introduce the organ embeddings as the input of the decoder, denoted as \( E_q \in \mathbb{R}^{M \times d} \). Here \( M \) is the total number of queries, which is fixed during the training. The organ embeddings are learned to decode features for each organ segmentation task. The Transformer decoder receives the organ queries and memory from the output of the Transformer encoder and produces a sequence of output embeddings, each of which can represent one organ in the input volumes. The calculation of the Transformer decoder can be formulated as

\[
\begin{align*}
t_0 &= \emptyset \\
v_i &= t_{i-1}, q_i = k_i = t_{i-1} + E_q \\
t'_i &= \text{N}(\text{SA}(q_i, k_i, v_i) + t_{i-1} \star \alpha) \\
q_i &= t'_i + E_q, x'_i = z_i_{\text{enc}} \\
t_i &= \text{N}(\text{MSDA}(q_i, x'_i) + t'_i \star \alpha) \\
t_i &= \text{N}(\text{FFN}(t_i) + t_i \star \alpha)
\end{align*}
\]

where \( i = 1, \ldots, I_{\text{dec}} \), and \( \emptyset \) is the random initialization of organ embeddings.

D. Filters Prediction Head

Transformer encodes the 3D feature maps and then decodes the organ embeddings into a sequence of output embeddings, i.e., \( t_{I_{\text{dec}}} \), each of which can represent a specific organ. A single MLP is assigned to decode them into the task-specific filters for the dynamic segmentation head.

\[
\omega = \text{MLP}(t_{I_{\text{dec}}}) \in \mathbb{R}^{M \times d_F}
\]

where \( d_F \) is the dimension of generated filters.

E. Dynamic Head

During the partially supervised training, it is worthless to predict the organs and tumors whose annotations are not available. Therefore, a lightweight dynamic head is designed to enable specific kernels to be assigned to each task for the segmentation of specific organs and tumors. The dynamic head contains three stacked convolutional layers with \( 1 \times 1 \times 1 \) kernels. The kernel set \( h_d \) layers, denoted by \( \omega = \{\omega_1, \omega_2, \ldots, \omega_{h_d}\} \), are dynamically generated by the filters prediction head according to the organ embeddings (see (15)). The dynamic width is set to \( h_w \) channels in front of \( h_d - 1 \) layers, while the last layer is fixed to 2 channels, i.e., one channel for organ pixel-wise prediction and the other for tumor pixel-wise prediction. The total number of dynamic parameters is computed by \( d_F = (h_w \times h_w + h_w) \times (h_d - 1) + (h_w \times 2 + 2) \). The prediction mask of the image \( X \) sampled from the partially labeled dataset \( \mathcal{D}_m \) is denoted as

\[
P \in \mathbb{R}^{M \times 2 \times D \times W \times H}, \text{ where}
\]

\[
P_i = \begin{cases} 
(G \ast \omega_{i1}) \ast \omega_{2i} \ast \omega_{3i} & \text{if } i = m \\
\emptyset & \text{otherwise}
\end{cases}
\]

Here, \( i = 1, 2, \ldots, M \), and \( \ast \) represents the convolution. Although each image requires a group of specific kernels for each task, the computation and memory cost of our lightweight dynamic head is negligible compared to the encoder-decoder (see Section IV-B).

F. Training and Testing

For simplicity, we treat the segmentation of an organ and related tumors as two binary segmentation tasks, and jointly use the Dice loss and binary cross-entropy loss as the objective for each task. Given the predictions \( P \) and partially labeled ground truth \( Y \in \mathbb{R}^{2 \times D \times W \times H} \) sampled from \( \mathcal{D}_m \), the loss function is formulated as

\[
\mathcal{L}(P, Y) = -\sum_{k=1}^{2} \frac{2 \sum_{p_i \in P_{m,k}, y_i \in Y_{k}} p_i y_i}{\sum_{p_i \in P_{m,k}, y_i \in Y_{k}} (p_i + y_i + \epsilon)} + \sum_{p_i \in P_{m,k}, y_i \in Y_{k}} (y_i \log(p_i) + (1 - y_i) \log(1 - p_i))
\]

where \( \epsilon \) is a smoothing factor for Dice calculation. A simple strategy is used to process the tasks that only provide organ or tumor annotations, i.e., ignoring the predictions corresponding to unlabeled targets. Taking colon tumor segmentation for example, the result of organ prediction, i.e., when \( k = 1 \), is ignored during the loss computation and error back-propagation since the annotations of organs are unavailable.

During inference, the proposed TransDoDNet is flexible to all segmentation tasks. Given a test image, the pre-segmentation feature \( G \) is first extracted from the CNN encoder-decoder network. Then, the Transformer generates the organ embeddings for the filter prediction head to generate segmentation kernels. Assigned with a task, its specific kernels are extracted and input to the dynamic segmentation head for the specific segmentation purpose. In addition, if \( M \) tasks are all required, our TransDoDNet is able to simultaneously infer the dynamic head by using different kernels in a parallel manner. Compared to the CNN encoder-decoder, the dynamic head is so light that the repeated inference cost of dynamic heads is almost negligible.

When merging multiple segmentation results, we follow the two basic guidelines: (1) to address conflicts between organs, we use the maximum probability to select the corresponding organ label, and (2) to address the fusion between organs and tumors, we prioritize tumors over their corresponding organs, primarily because tumors are considered part of the anatomy of the organ. For the sake of simplicity, we formulate the label fusion process for each voxel \( P' \in \mathbb{R}^{M 	imes 2} \) as follows

\[
y_{\text{fusion}} = \begin{cases} 
0 & P'_{(j,1)} < 0.5 \\
\max(j - 2, 0) & P'_{(j,1)} \geq 0.5, P'_{(j,2)} < 0.5 \\
\max(j, 0) & P'_{(j,1)} \geq 0.5, P'_{(j,2)} \geq 0.5
\end{cases}
\]
Three metrics are used to evaluate the Tmm is adopted as the op-

A. Experiment Setup

Dataset: We build a large-scale partially labeled Multi-Organ and Tumor Segmentation (MOTS) benchmark using multiple medical image segmentation datasets, including LiTS [4], KiTS [5], and Medical Decathlon (MSD) [49]. MOTS is composed of seven partially labeled sub-datasets, involving seven organ and tumor segmentation tasks. There are 1155 3D abdominal CT scans collected from various clinical sites around the world, including 920 scans for training and 235 scans for test. More details are given in Table 1. Each scan is re-sliced to the same voxel size of $1.5 \times 0.8 \times 0.8 \text{ mm}^3$.

Two public datasets are also used for the downstream tasks, including BCV [50] and BraTS [51]. BCV was collected by the MICCAI 2015 Multi Atlas Labeling Beyond the Cranial Vault Challenge. BCV is composed of 50 abdominal CT scans, including 30 scans for training and 20 scans for test. Each training scan is paired with voxel-wise annotations of 13 organs, including the liver (Li), spleen (Sp), pancreas (Pa), right kidney (RK), left kidney (LK), gallbladder (Ga), esophagus (Es), stomach (St), aorta (Ao), inferior vena cava (IVC), portal vein and splenic vein (PSV), right adrenal gland (RAG), and left adrenal gland (LAG). BraTS was provided by the 2018 Brain Tumor Segmentation Challenge. The aim of this challenge is to develop automated segmentation algorithms to delineate intrinsically heterogeneous brain tumors, i.e., (1) enhancing tumor (ET), (2) tumor core (TC) that consists of ET, necrotic and non-enhancing tumor core, and (3) whole tumor (WT) that contains TC and the peritumoral edema. This dataset provides the different imaging modalities, i.e., magnetic resonance imaging (MRI). Each case contains four MRI sequences, including the T1, T1c, T2, and FLAIR. All sequences were registered to the same anatomical template and interpolated to the same dimension of $155 \times 240 \times 240$ voxels and the same voxel size of $1.0 \times 1.0 \times 1.0 \text{ mm}^3$. BraTS is composed of 285 training cases, including 210 cases of high-grade gliomas and 75 cases of low-grade gliomas, and 66 cases for online evaluation. We evaluate the pre-trained weights of MOTS on these downstream segmentation tasks.

Evaluation metric: Three metrics are used to evaluate the segmentation performance, including the Dice similarity coefficient (Dice), Hausdorff distance (HD), and surface distance (SD). Dice measures the overlapping between a segmentation prediction and ground truth. Both HD and SD are used to evaluate the quality of segmentation boundaries. HD is a measure that captures the greatest of all the distances from a point in one set to the closest point in the other set, while SD is a measure of the distance between corresponding points on two surfaces, which provides additional information about the overall agreement between the two segmentation boundaries. We assess all segmentation methods based on the mean Dice (mDice), mean HD (mHD), and mean SD (mSD) across all organ and tumor categories.

Implementation details: To filter irrelevant regions and simplify subsequent processing, we truncate the HU values in each scan to the range of $[-325, +325]$ and linearly normalize them to $[-1, +1]$. Owing to the benefits of instance normalization [45], our model adopts the micro-batch training strategy with a small batch size of 2. The AdamW algorithm [52] is adopted as the optimizer. The learning rate is initialized to $2 \times 10^{-4}$ and decayed according to a polynomial policy $\text{lr} = \text{lr}_{\text{init}} \times (1 - k/K)^{0.9}$, where the maximum epoch $K$ is set to 1,000. In the training stage, we randomly extract sub-volumes with the size of $64 \times 192 \times 192$ from CT scans as the input. To avoid the overfitting problem, we use the batchgenerator library [53] to perform a wide variety of data augmentation, including random scaling, flipping, rotation, Gaussian noise, Gaussian blur, contrast, and brightness. In the test stage, we employ the sliding window-based strategy. To ensure a fair comparison, the same training strategies, including data augmentation, learning rate, optimizer, and other settings, are applied to all competing models.
B. Comparing to State-of-the-Art Methods

In this section, we compare the proposed TransDoDNet to the state-of-the-art competitors, which also attempt to address the partially labeling issue, on seven partially labeled tasks using the MOTS test set. These methods include (1) seven individual networks, each being trained on a partially dataset (denoted by Multi-Nets), (2) two multi-head networks (i.e., Multi-Head [11] and TAL [12]), (3) two single-network methods with the task condition (i.e., Cond-Input [54], Cond-Dec [26]), and DoDNet with convolution as kernel generator [14]. To ensure a fair comparison, we keep the same encoder-decoder architecture for all methods, except that the channels of CNN decoder layers in Multi-Head are halved due to the GPU memory limitation.

Table II shows the performance metrics for the segmentation of each organ/tumor and the average scores over 11 categories. It reveals that (1) almost all the methods (TAL, Multi-Head, Cond-Input, Cond-dec, DoDNet, TransDoDNet) achieve better performance than seven individual networks (Multi-Nets), suggesting that training with more data (even partially labeled) is beneficial to the performance; (2) the dynamic filter generation strategy is superior to directly embedding the task condition into the input or decoder (used in Cond-Input and Cond-Dec); (3) Compared to the DoDNet with the convolution-based kernel generator, our TransDoDNet with the Transformer based kernel generator further improves the Dice by 1.3% and reduces the HD/SD by 1.6/0.5; and (4) the proposed TransDoDNet achieves the highest overall performance with a mDice of 78.2%, a mHD of 11.4, and a mSD of 3.9, beating all competitors. To make a qualitative comparison, we visualize the segmentation results obtained by seven methods on seven tasks in Fig. 6. It shows that our TransDoDNet outperforms other methods, especially in segmenting small tumors.

In Fig. 7, we also compare the accuracy-complexity trade-off of seven methods. Most of the methods, including TAL, Cond-Dec, Cond-Input, DoDNet, and TransDoDNet, share the encoder and decoder for all tasks, and hence have a similar number of parameters as one single network. Although our TransDoDNet has an extra Transformer module, the number of parameters assigned for this part is negligible compared to the CNN encoder. The Multi-Head network has a few more parameters due to the use of multiple task-specific decoders. Multi-Nets has to train seven single networks to address these partially labeled tasks individually, resulting in seven times more parameters than a single network. As for the inference complexity, Cond-Input, Multi-Nets, Multi-Head, and Cond-Dec suffer from the repeated inference processes, and hence need more time to perform seven partial segmentation tasks than other methods. Besides, embedding task information in the input layer leads to the additional increased complexity in the Cond-Input. In contrast, TAL is much more efficient in segmenting all targets, since the encoder-decoder (except for the last segmentation layer) is shared by all tasks. Both DoDNet and TransDoDNet share the encoder-decoder architecture and specialize the dynamic head for each partially labeled task. The inference of the dynamic head is very fast due to its lightweight architecture. Although the Transformer module in TransDoDNet results in the extra computation complexity, it achieves the best performance with an acceptable inference complexity.

C. MOTS Pre-Training for Downstream Tasks

It has been generally recognized that training a deep model with more data contributes to a better generalization ability [55], [56], [57], [58]. However, deep learning remains trapped by the limited annotations, especially in 3D medical image segmentation. Although self-supervised learning [15], [59], [60], [61], [62], [63], [64], [65] has shown great potential to address this issue, it still suffers from the lack of a big dataset with millions or trillions of free data for self-supervised learning, especially for 3D volumes. With one thousand strong labels, we believe that MOTS is able to produce a more effective pre-trained model for annotation-limited downstream tasks. To demonstrate this, we first compare the MOTS pre-training to SinglePT, i.e.,...
pre-training on a single dataset. We choose the largest Hepatic Vessel segmentation benchmark as the single pre-trained dataset, considering it contains up to 242 training cases. Then, we compare MOTS to three advanced self-supervised learning solutions, including BYOL [15], PGL [66], and PCRL [67]. It is worth noting that PGL is a self-supervised method specifically designed for 3D medical image segmentation tasks, which builds upon BYOL to perform dense-level representation learning. PCRL introduces a reconstruction objective to enhance the performance of downstream segmentation tasks. For the self-supervised learning purpose, we make use of MOTS and FLARE [68] datasets, resulting in an unlabeled dataset with more than three thousand volumes.

Transferring to BCV: To compare different pre-training methods, we initialize the segmentation network using pre-trained weights obtained by SinglePT, BYLO, PGL, PCRL, and our MOTS. We split half of the cases from the BCV training set for validation since the annotations of the BCV test set are withheld for online assessment, which is inconvenient. We compare the validation performance of four different initialization methods during the training, shown in Fig. 8. It reveals that, compared to training from scratch, all pre-training methods help the model converge quickly and perform better, particularly in the initial stage. However, pre-training on a single dataset (i.e., #3 Hepatic Vessel) only slightly outperforms training from scratch. Although the unlabeled data is twice as much as MOTS, it is still challenging for self-supervised learning to explore strong representation ability without strong supervision. Not surprisingly, pre-training on MOTS achieves not only the fastest convergence but also a remarkable performance boost. The quantitative results are compared in Table III. It reveals that pre-training on MOTS has a strong generalization potential that achieves 83.2% mDice, outperforming the baseline model (TFS) by 1.7%, outperforming the self-supervised learning methods by at least 0.5%. As for a qualitative comparison, we also visualize the 3D segmentation results obtained by using different initialization strategies in Fig. 9. It shows that initializing the segmentation network with the MOTS pre-trained weights enables the model...
Fig. 7. Accuracy versus complexity. The accuracy refers to the overall Dice score on the MOTS test set. The complexity refers to the inference time cost. Both inference time and model parameters ('#P') are quantified based on a single network that can only process a single partial segmentation task.

Fig. 8. Comparison of validation performance of four different initialization strategies, including training from scratch, pre-training on a single dataset #3 Hepatic Vessel, self-supervised learning (BYOL), and our MOTS partially supervised pre-training. Here the validation performance refers to the averaged Dice score over 13 categories.

Table III

Comparison of Different Pre-trained Methods on the BCV Validation Set

| Method       | mDice | mHD  |
|--------------|-------|------|
| TFS          | 81.5  | 8.1  |
| SinglePT     | 81.7  | 8.0  |
| BYOL [51]    | 82.3  | 7.9  |
| PCRL [66]    | 82.5  | 7.9  |
| MOTS         | 83.2  | 7.4  |

TFS: training network from scratch, SinglePT: pre-training on a single dataset; BYOL, PCRL: three self-supervised learning methods; MOTS: TransDolNet pre-trained on MOTS.

Fig. 9. Visualization of results obtained by applying the segmentation network with four initialization strategies. (a) Ground truth; (b) random initialization; (c) pre-trained weights on the single dataset #3 Hepatic Vessel; (d) self-supervised learning on the unlabeled dataset; and (e) pre-trained weights on MOTS.

Table IV

Comparison of State-of-the-Art Methods on the BCV Test Set

| Methods          | mDice | mHD  | mSD  |
|------------------|-------|------|------|
| Auto Context     | 78.2  | 26.1 | 1.9  |
| deedsJointCL     | 79.0  | 25.5 | 2.3  |
| DLTK             | 81.5  | 62.9 | 1.9  |
| PaNN             | 85.0  | 18.5 | 1.5  |
| nnUnet           | 88.1  | 17.3 | 1.4  |
| TFS              | 87.3  | 14.6 | 1.5  |
| MOTS*            | 88.2  | 13.1 | 0.9  |
| MOTS             | 88.6  | 12.8 | 0.9  |

SD: mean surface distance (lower is better); TFS: training network from scratch; MOTS*: DolNet pre-trained on MOTS; MOTS: TransDolNet pre-trained on MOTS. The values of three metrics are averaged over 13 categories.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
It reveals that compared to the training network from the scratch method, our MOTS pre-trained model improves the average Dice from 85.7% to 86.2% and reduces the average HD from 4.7 to 4.4, resulting in a competitive performance. More importantly, such a competitive performance is obtained by using a single model, instead of the ensembles used by most of the competitors.

D. Ablation Study

We split 20% of training cases as validation data to perform the ablation study. We report the mean Dice and HD of 11 organs and tumors (listed in Table I) as two evaluation indicators. We set the maximum epoch to 500 for the efficient and fair ablation study.

1) Dynamic Head: We first investigate the effectiveness of the dynamic head, including the depth and width. Specifically, we keep the depth of the Transformer encoder and decoder to be 1, and the multi-scale level to be 1, i.e., single scale. We report the segmentation performance and total dynamic parameters of each setting for a clear comparison.

   Head depth: In Table VI, we compare the performance of the dynamic head with different depths, varying from 2 to 4. The width is fixed to 8 channels, except for the last layer, which has 2 channels. It shows that the best performance, including both mDice and mHD, is achieved when the head depth equals 3. Besides, the performance fluctuation is very small when the depth increases from 2 to 4. The results indicate the robustness of the dynamic head to the varying depth, which is empirically set to 3 for this study.

   Head width: In Table VII, we compare the performance of the dynamic head with different widths, varying from 2 to 4. Here the depth is fixed to 3. It shows that the performance improves substantially when increasing the width from 4 to 8, but drops slightly when further increasing the width from 8 to 16. It suggests that the performance tends to become stable when the width of the dynamic head falls within a reasonable range. Considering both performance and complexity, we empirically set the width of the dynamic head to 8.

2) Kernel Generator: In this section, we first compare the effectiveness of two different kernel generators, i.e., convolution and Transformer, and then investigate the key components of the Transformer based kernel generator, including Transformer depth, width, and multi-scale attention mechanism. Note that the depth and width of the dynamic head are uniformly set to 3-layer and 8-channel respectively in the rest of the experiments.

   CNN versus Transformer based kernel generator: We compare the Transformer based kernel generator with the convolution-based solution that was presented in our conference version. The convolution-based method generates kernels conditioned on the independent task encoding, ignoring organ-wise dependencies. Differently, the Transformer based kernel generator is superior to modeling the global organ-wise contextual information, and hence segments the multi-organs and tumors more accurately. As shown in Table VIII, both methods utilize the one-layer architecture in the kernel generator. The proposed
Transform DoDNet is assigned a basic setting that has one Transformer encoder-decoder layer in the kernel generator, and the feature channel is set to 384. For a fair comparison, we scale up the convolution-based DoDNet by increasing the convolution layers in the last backbone stage to achieve comparable parameters with TransDoDNet. We can see that the Transformer generator outperforms the convolution generator by a clear margin in all metrics (Transformer 72.3 mDice versus convolution 71.0 mDice) with comparable parameters (Transformer 40.5 M versus convolution 42.5 M).

**Transformer depth:** We compare the Transformer with different layers of encoder and decoder in Table IX. We observe better segmentation performance when increasing the Transformer encoder/decoder layers. However, this trend is weakened when the layer goes up to 6, which suffers from a little performance saturation. As the Transformer layer increases, the model is mired in the fast-growing parameters. We thus empirically set the layers of the Transformer encoder and decoder to be 3, balancing the model accuracy and complexity.

**Transformer width:** Transformer width, i.e., feature channel, is also an important factor that determines the scale of Transformers. In Table X, we increase progressively the Transformer width from 96-channel to 768-channel, leading to a dramatic increase of parameters from 36.5 to 82.7 million. We observe that the performance deteriorates as the increase of width, and obtain the best performance when the width is 192. The massively increased parameters may bring great difficulty to model optimization, thus failing to observe a performance improvement.

**Multi-scale attention versus single-scale attention:** The proposed TransDoDNet is able to model the long-range dependencies across multiple scales, which is beneficial for segmenting organs and tumors with diverse object scales. Compared to the single-scale attention, as shown in Table XI, the multi-scale attention with 3 levels improves the mDice from 72.8% to 73.8% and reduces the mHD from 20.3 to 19.0.

3) **Comparison of Three Variants of TransDoDNet:** In this section, we design three variants of TransDoDNet that are different from the usage mode of Transformer encoder features, as shown in Fig. 10. Mode-A jointly fuses the CNN features and Transformer features as the input of the CNN decoder; mode-B only transfers the CNN features into the CNN decoder; and mode-C only transfers the Transformer features into the CNN decoder. The three variants are compared in Table XII. It demonstrates that the complementarity of both features contributes to better performance than any of them.

**E. Attention Visualization**

In Fig. 11, we visualize the sampling key points of different reference points in the deformable self-attention module of the last Transformer encoder layer. For the sake of convenience in visualization, we display the sampling points at multi-scale levels on a CT slice. The sampling points and reference points is refers to the circle dots and cross signs, respectively. And the different colors of sampling points denote the different levels of attention weights. It shows that the deformable self-attention can adaptively adjust the attention key points to the locations nearby or associated with query objects.

**F. Broad Benefits of MOTS Benchmark**

MOTS highlights an important challenge in the field of medical image segmentation, e.g., the problem of partially labeling due to the high cost of annotation. It aims to encourage researchers to develop advanced segmentation algorithms to address this issue. The broader benefits for the community include two aspects. From the point of view of application, due to the constraints of labor costs and required expertise, existing 3D medical image segmentation benchmarks have been critically hindered by limited annotations within individual datasets. To alleviate this limitation, MOTS was formulated as a large-scale benchmark by amalgamating multiple organ and tumor segmentation datasets, thereby providing a platform for machine learning models to learn broader and more potent representations from an...
extensive scope of annotated medical data. Besides, the creation of the MOTS benchmark provides new ideas for 3D medical image pre-training, and the integration of large amounts of existing annotated data resources can result in the development of powerful universal segmentation pre-training models. From the point of view of research and development, MOTS involves more complex segmentation tasks, including more than ten partially labeled organs and tumors, which are more challenging than a single fully labeled dataset. This increased task difficulty provides greater challenges and research opportunities for segmentation algorithm research. MOTS encourages a rethinking of the medical image segmentation paradigm, transitioning from a “single task, single dataset” mindset to a “multi-task, multi-dataset” approach, which aligns with the current research vision for general segmentation models. Moreover, partially labeling exists in many fields beyond medical imaging, and this dataset deeply characterizes this issue, inspiring cost-effective research on universal segmentation algorithms with significant potential to drive progress.

V. CONCLUSION

In this paper, we create a large-scale multi-organ and tumor segmentation benchmark (MOTS) that integrates multiple partially labeled medical image segmentation datasets. To address the partial labeling issue of MOTS, we propose the TransDoDNet model that employs the dynamic segmentation head to flexibly process multiple segmentation tasks. Moreover, we introduce a Transformer based kernel generator that models the organ-wise dependencies by using the self-attention mechanism. Our experiment results on MOTS indicate that the TransDoDNet model achieves the best overall performance on seven organ and tumor segmentation tasks. We also demonstrate the value of the TransDoDNet and MOTS dataset by successfully transferring the weights pre-trained on MOTS to downstream tasks that only limited annotations are available. It suggests that the byproduct of this work (i.e., a pre-trained 3D network, including both encoder and decoder) is conducive to other small-sample 3D medical image segmentation tasks.

REFERENCES

[1] M. Cordts et al., “The cityscapes dataset for semantic urban scene understanding,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 3213–3223.
[2] S. Pang et al., “CTumorGAN: A unified framework for automatic computed tomography tumor segmentation,” Eur. J. Nucl. Mol. Imag., vol. 47, no. 10, pp. 2248–2268, 2020.
[3] A. E. Kavur et al., “Chaos challenge–combined (CT-MR) healthy abdominal organ segmentation,” 2020, arXiv:2001.06535.
[4] P. Bilic et al., “The liver tumor segmentation benchmark (LiTS),” 2019, arXiv:1901.04056.
[5] N. Helder et al., “The kits19 challenge data: 300 kidney tumor cases with clinical context, CT semantic segmentations, and surgical outcomes,” 2019, arXiv:1904.00445.
[6] Q. Yu, Y. Shi, J. Sun, Y. Gao, J. Zhu, and Y. Dai, “Crossbar-net: A novel convolutional neural network for kidney tumor segmentation in CT images,” IEEE Trans. Image Process., vol. 28, no. 8, pp. 4060–4074, Aug. 2019.
[7] F. Isensee, P. F. Jaeger, S. A. Kohl, J. Petersen, and K. H. Maier-Hein, “nnU-Net: A self-configuring method for deep learning-based biomedical image segmentation,” Nature Methods, vol. 18, no. 2, pp. 203–211, 2021.
[8] J. Zhang, X. Xie, P. Zhang, H. Chen, Y. Xia, and C. Shen, “Light-weight hybrid convolutional network for liver tumor segmentation,” in Proc. Int. Joint Conf. Artif. Intell., 2019, pp. 4271–4277.
[9] A. Myronenko and A. Hatamizadeh, “3D Kidneys and kidney tumor semantic segmentation using boundary-aware networks,” 2019, arXiv:1909.06684.
[10] Z. Zhu, Y. Xie, L. Xie, E. K. Fishman, and A. L. Yuille, “Multi-scale coarse-to-fine segmentation for screening pancreatic ductal adenocarcinoma,” in Proc. Int. Conf. Med. Image Comput. Assist. Intervention, Springer, 2019, pp. 3–12.
[11] S. Chen, K. Ma, and Y. Zheng, “Med3D: Transfer learning for 3D medical image analysis,” 2019, arXiv:1904.00625.
[12] X. Fang and P. Yan, “Multi-organ segmentation over partially labeled datasets with multi-scale feature abstraction,” IEEE Trans. Med. Imag., vol. 39, no. 11, pp. 3619–3629, Nov. 2020.
[13] G. Shi, L. Xiao, Y. Chen, and S. K. Zhou, “Marginal loss and exclusion loss for partially supervised multi-organ segmentation,” 2020, arXiv:2007.03868.
[14] J. Zhang, Y. Xie, Y. Xia, and C. Shen, “DoDNet: Learning to segment multi-organ and tumors from multiple partially labeled datasets,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 1195–1204.
[15] J.-B. Grill et al., “Bootstrap your own latent: A new approach to self-supervised learning,” Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 21271–21284.
[16] L. Xie, Q. Yu, Y. Zhou, Y. Wang, E. K. Fishman, and A. L. Yuille, “Recurrent saliency transformation network for tiny target segmentation in abdominal CT scans,” IEEE Trans. Med. Imag., vol. 39, no. 2, pp. 514–525, Feb. 2020.
[17] L. Zhang et al., “Block level skip connections across cascaded V-net for multi-organ segmentation,” IEEE Trans. Med. Imag., vol. 39, no. 9, pp. 2782–2793, Sep. 2020.
[18] Y. Wang, Y. Zhou, W. Shen, P. E. Fishman, and A. L. Yuille, “Abdominal multi-organ segmentation with organ-attention networks and statistical fusion,” Med. Image Anal., vol. 55, pp. 88–102, 2019.
[19] O. Schoppe et al., “Deep learning-enabled multi-organ segmentation in whole-body mouse scans,” Nat. Commun., vol. 11, no. 1, pp. 1–14, 2020.
[20] X. Li, H. Chen, X. Qi, Q. Dou, C.-W. Fu, and P.-A. Heng, “IH-DenseUNet: Hybrid densely connected U-Net for liver and tumor segmentation from CT volumes,” IEEE Trans. Med. Imag., vol. 37, no. 12, pp. 2663–2674, Dec. 2018.
A. Dosovitskiy et al., “An image is worth 16x16 words: Transformers for image classification at scale,” 2020, arXiv:2010.04159.

B. Yang, G. Bender, Q. V. Le, and J. Ngiam, “CondConv: Conditionally parameterized convolutions for efficient inference,” in Proc. Adv. Neural Inf. Process. Syst., 2016, pp. 667–675.

B. Yang, Z. Hu, S. Zhang, and H. Li, “Multi-organ segmentation via co-training weight-averaged models from few-organ datasets,” in Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention, Springer, 2020, pp. 512–522.

Z. Tian, C. Shen, and H. Chen, “Conditional convolutions for instance segmentation,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 282–298.

J. Zhou et al., “Prior-aware neural network for partially-supervised multi-organ segmentation,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 10672–10681.

K. Dimitrīev and A. E. Kaufman, “Learning multi-class segmentations from single-class datasets,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2019, pp. 9501–9511.

J. He, Z. Deng, and Y. Qiao, “Dynamic filter networks,” in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 2761–2769.

J. Chen, T. Xia, Z. Tian, and J. Xiao, “E2N: Edge enhanced network for accurate liver and tumor segmentation on CT scans,” in Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention, Springer, 2020, pp. 512–522.

J. He, S. Gao, Z. Tian, R. Tong, H. Wang, W. Wang, and H. Jin, “ResNeXt: Deep Residual Networks with Identity Mappings,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, arXiv:1603.01229.

J. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 770–778.

J. He et al., “Deformable convolutional networks,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2017, pp. 764–773.

J. L. Ba, J. R. Kiros, and G. E. Hinton, “Layer normalization,” 2016, arXiv:1607.06450.

H. Wang, S. Ma, L. Dong, S. Huang, D. Zhang, and F. Wei, “DeepNet: Scaling transformers to 1,000 layers,” 2022, arXiv:2203.06555.

A. L. Simpson et al., “A large annotated medical image database for the development and evaluation of segmentation algorithms,” 2019, arXiv:1902.09063.

A. Landman et al., “MICCAI multi-atlas labeling beyond the cranial vault workshop and challenge,” 2015. [Online]. Available: https://doi.org/10.7303/syn1938005

S. M. Bakas et al., “Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge,” 2018, arXiv:1811.02629.

I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in Proc. Int. Conf. Learn. Representations, 2019, pp. 1–10.

F. Isensee et al., “BatchGenerators—A Python framework for data augmentation,” Zenodo, vol. 3632567, 2020. [Online]. Available: https://doi.org/10.5281/zenodo.4368800.

Q. Chen, J. Xu, and V. Koltun, “Fast image processing with fully-convolutional networks,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2017, pp. 2497–2506.

B. Zoph et al., “Rethinking pre-training and self-training,” in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 3833–3841.

K. He, R. Girshick, and P. Dollár, “Rethinking ImageNet pre-training,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 4918–4927.

D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, “A closer look at spatiotemporal convolutions for action recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2018, pp. 6450–6459.

A. Esteve et al., “Dermatologist-level classification of skin cancer with deep neural networks,” Nature, vol. 542, no. 7639, pp. 115–118, 2017.

L. Jing and Y. Tian, “Self-supervised feature learning with deep neural networks: A survey,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 43, no. 11, pp. 4037–4058, Nov. 2021.

X. Liu et al., “Self-supervised learning: Generative or contrastive,” 2020, arXiv:2006.08218.

I. Misra and L. v. d. Maaten, “Self-supervised learning of pretext-invariant representations,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 6707–6717.

H. Lee, S. J. Hwang, and J. Shin, “Self-supervised label augmentation via input transformations,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 5714–5724.

Y. Tian, D. Krishnan, and P. Isola, “Contrastive multiview coding,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 776–792.

K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, “Momentum contrast for unsupervised visual representation learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2020, pp. 9792–9798.

T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, “A simple framework for contrastive learning of visual representations,” in Proc. Int. Conf. Mach. Learn., 2020, pp. 1597–1607.

Y. Xie, J. Zhang, Z. Liao, Y. Xiao, and C. Shen, “PGL: Prior-guided self-supervised learning for 3D medical image segmentation,” 2020, arXiv:2011.12640.

H.-Y. Zhou, C. Lu, S. Yang, X. Han, and Y. Yu, “Preservational learning improves self-supervised medical image models by reconstructing diverse contexts,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 3499–3509.

J. Ma and B. Wang, “Fast and low-resource semi-supervised abdominal organ segmentation,” in Proc. Int. Conf. Med. Image Comput. Comput.- Assisted Intervention, Springer Nature, vol. 13816, 2022.

H. R. Roth et al., “A multi-scale pyramid of 3D fully convolutional networks for abdominal and pelvic organ segmentation,” in Proc. Int. Conf. Med. Image Comput. Comput.- Assisted Intervention, Springer, 2018, pp. 417–425.

M. P. Heinrich, “Multi-organ segmentation using deeds, self-similarity context and joint fusion,” in Proc. Int. Conf. Med. Image Comput. Comput.- Assisted Intervention BCV Workshop, 2015, pp. 1–12.

N. Pawlowski et al., “UTK: State of the art reference implementations for deep learning on medical images,” 2017, arXiv:1711.06853.

G. Wang, W. Li, S. Ourselin, and T. Vercauteren, “Automatic brain tumor segmentation using convolutional neural networks with test-time augmentation,” in Proc. Int. Conf. MICCAI Brainlesion Workshop, 2018, pp. 61–72.

C. Chen, X. Liu, M. Ding, J. Zheng, and J. Li, “3D dilated multifiber network for real-time brain tumor segmentation in MRI,” in Proc. Int. Conf. Med. Image Comput. Comput.- Assisted Intervention, 2018, pp. 184–192.
[74] C. Zhou, C. Ding, X. Wang, Z. Lu, and D. Tao, “One-pass multi-task networks with cross-task guided attention for brain tumor segmentation,” *IEEE Trans. Image Process.*, vol. 29, pp. 4516–4529, 2020.

[75] R. McKinley, R. Meier, and R. Wiest, “Ensembles of densely-connected CNNs with label-uncertainty for brain tumor segmentation,” in *Proc. Int. MICCAI Brainlesion Workshop*, 2018, pp. 456–465.

[76] A. Myronenko, “3D MRI brain tumor segmentation using autoencoder regularization,” in *Proc. Int. MICCAI Brainlesion Workshop*, 2018, pp. 311–320.

[77] R. Brügger et al., “A partially reversible U-net for memory-efficient volumetric image segmentation,” in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention*, Springer, 2019, pp. 429–437.

[78] H. Xu et al., “Deep cascaded attention network for multi-task brain tumor segmentation,” in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Intervention*, Springer, 2019, pp. 420–428.

[79] J. Zhang, Y. Xie, Y. Wang, and Y. Xia, “Inter-slice context residual learning for 3D medical image segmentation,” *IEEE Trans. Med. Imag.*, vol. 40, no. 2, pp. 661–672, Feb. 2021.

Yutong Xie received the BE degree from the School of Astronautics from Northwestern Polytechnical University (NPU), Xi’an, China, in 2016, and the PhD degree from the School of Computer Science and Engineering, NPU, China, in 2021. She is currently a senior research associate with the University of Adelaide, Adelaide, Australia. Her research mainly focuses on computer vision and machine learning in medical applications.

Jianpeng Zhang received the BE, ME, and PhD degrees from Northwestern Polytechnical University, in 2016, 2019, and 2022, respectively. His research interests mainly focus on deep learning technologies for annotation-limited medical image analysis, especially self-supervised learning, partial label learning, and weakly supervised learning.

Yong Xia (Member, IEEE) received the BE, ME, and PhD degrees in computer science and technology from Northwestern Polytechnical University (NPU), Xi’an, China, in 2001, 2004, and 2007, respectively. He is currently a professor with the School of Computer Science and Engineering, NPU. His research interests include medical image analysis, computer-aided diagnosis, pattern recognition, machine learning, and data mining.

Chunhua Shen is a University chair professor with Zhejiang University, China. Prior to that, he was a professor with the University of Adelaide, Australia and a researcher with Australian National University and National ICT Australia.