3D TransUNet: Advancing Medical Image Segmentation through Vision Transformer

Introduction

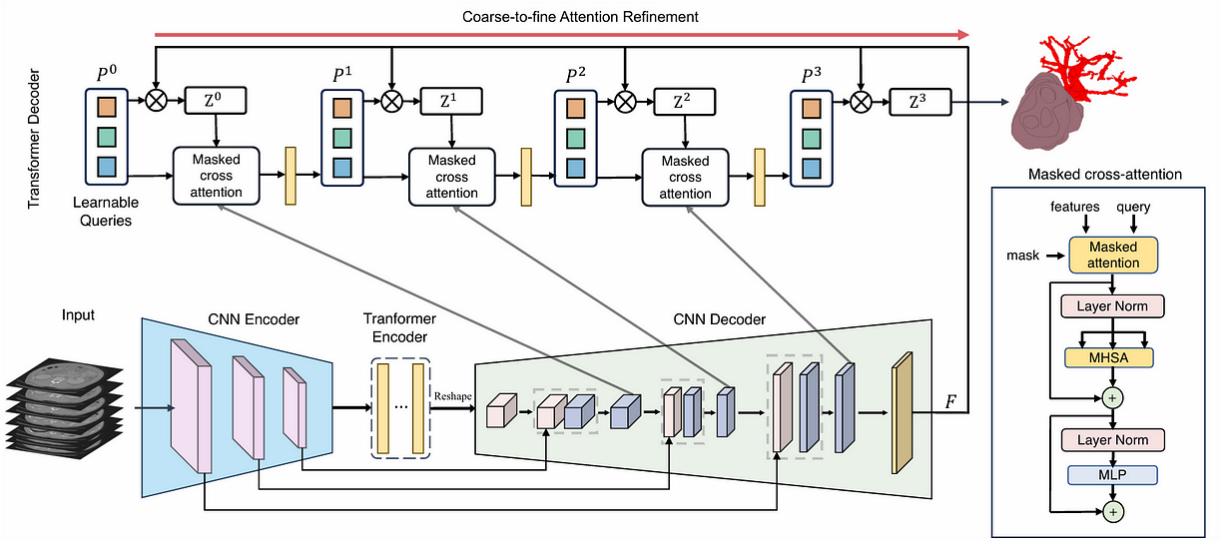
In this blog post, we delve into the realm of medical image segmentation with the **3D TransUNet**, an advanced architecture that extends the capabilities of the previously successful **TransUNet** by incorporating three-dimensional data processing. The 3D TransUNet builds on the foundation of the **nnU-Net framework** renowned for its versatility in handling both 2D and 3D segmentation tasks. We will also present the different configurations of this model and assess their effectiveness and applications in various medical imaging contexts

The core strength of this architecture lies in two main components:

- **Transformer Encoder:** This module leverages the transformer's self-attention capabilities to enhance **global context awareness**, which is crucial for accurate **multi-organ segmentation**.
- **Transformer Decoder:** It focuses on **refining segmentation accuracy**, particularly useful for precisely **identifying smaller and more challenging targets like tumors.**

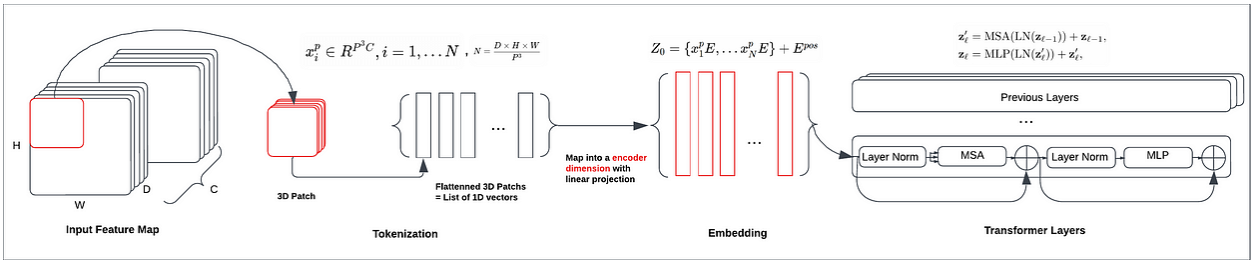
Our discussion will begin by examining the initial **feature map** from a CNN encoder **(D*W*H*C)** and how it transitions into a comprehensive **label map (D*W*H)**. For those new to this subject, I recommend checking out my previous blog on the standard **TransUNet**, where I cover the its architecture which sets the groundwork for understanding the more complex 3D TransUNet structure.

Detailed Exploration of 3D TransUNet:



3D TransUnet Architecture

Transformer Encoder Workflow



3D TransUnet Encoder

. **Tokenization**: The feature map is reshaped into a sequence of flattened 3D patches. Each patch is P×P×P, and the total number of patches N is calculated as $N=(D\times H\times W)/ P^3$

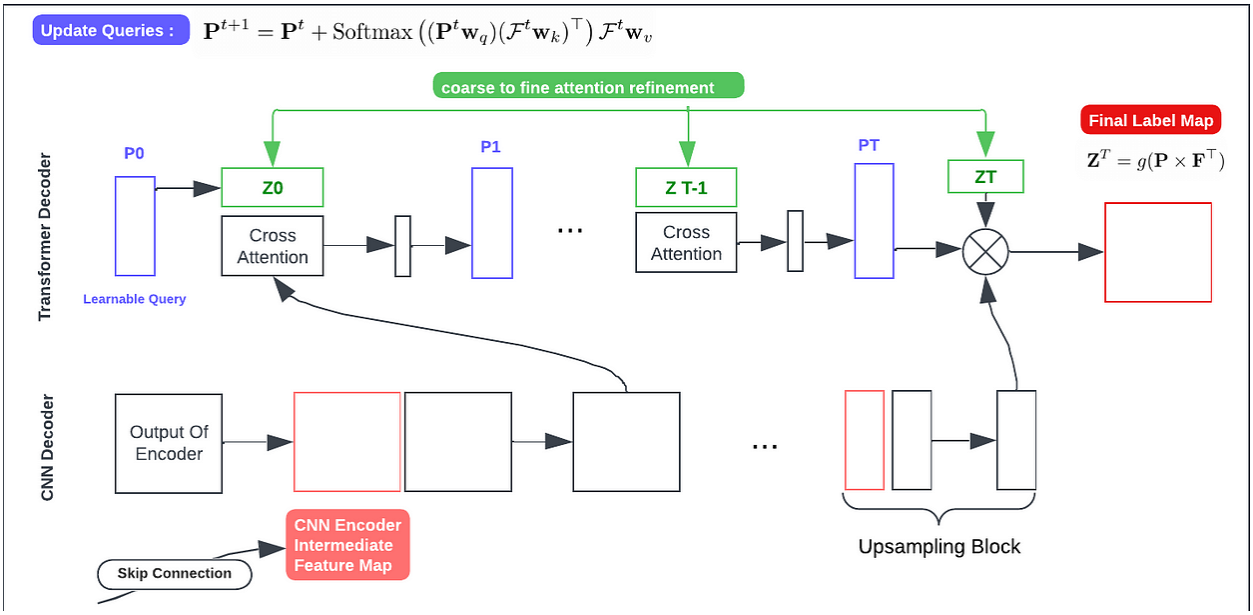
. **Patch Embedding and Positional Encoding**: These vectorized patches are then mapped into a denc-dimensional embedding space using a **linear projection. Positional encodings** are added to imbue spatial relationships within the data.

. **Transformer Layers Execution**: The feature representation is enhanced through multiple layers of self-attention and feed-forward networks. An example of layer operations includes:

- zl'= MSA(LN(zl-1))+zl-1

- zl=MLP(LN(zl′))+zl′

Transformer Decoder Workflow:



3D TransUnet Decoder

Coarse Candidate Estimation:

Transition from Pixel-Wise to Query-Based Segmentation: Instead of classifying each pixel independently based on likelihood and using loss at pixel level, this approach utilizes **organ queries** where each query vector represents a specific organ or region. So in theory number of queries = number of classes but in reality we take it superiorto decrease False negatives.

Initial Coarse Prediction (Z0): The first approximation of the segmentation map Z0 is computed as Z0=g(P0×F′) where:

- g(·) is a sigmoid function followed by a thresholding operation set at 0.5.
- P0 represents the initial organ queries.
- F′ denotes the transposed embedding of the final U-Net block feature, dimensioned as R^(D×H×W×dec) , d_dec is the dimensionality of the object queries.

Transformer decoder:

Refinement of Organ Queries: Each decoder layer refines the organ queries Pt by incorporating cross-attention with **intermediate U-Net features** using **residual path**. These features are mapped into the same dimensional space d_dec to facilitate the computation of cross-attention:

$$\mathbf{P}^{t+1} = \mathbf{P}^{t} + \mathrm{Softmax}\left((\mathbf{P}^{t}\mathbf{w}_q)(\mathcal{F}^{t}\mathbf{w}_k)^{\top}\right)\mathcal{F}^{t}\mathbf{w}_v$$

Where wq, wk, and wv are parametric weight matrices transforming P and F into queries, keys, and values, respectively.

coarse to fine attention refinement:

This component utilizes a coarse mask derived from an initial estimation to guide the refinement process in subsequent iterations. By grounding the cross-attention within the regions identified in the coarse prediction, the model focuses more accurately on regions of interest, iteratively enhancing the segmentation accuracy.

the iterative refinement process: During each iteration ttt, the organ queries are further refined based on the adjusted attention:

$$\mathbf{P}^{t+1} = \mathbf{P}^{t} + \mathrm{Softmax}((\mathbf{P}^{t}\mathbf{w}_q)(\mathcal{F}\mathbf{w}_k)^{\top} + h(\mathbf{Z}^{t})) \times \mathcal{F}\mathbf{w}_v$$

where:

$$h\left(\mathbf{Z}^{t}(i,j,s)\right) = \begin{cases} 0 & \text{if } \mathbf{Z}^{t}(i,j,s) = 1 \\ -\infty & \text{otherwise} \end{cases}$$

The updated organ queries PTP_TPT from the final iteration are decoded into a refined, binarized segmentation map ZTZ_TZT using the equation:

$$Z_T = g(P \times F^T)$$

This step associates each binarized mask with a specific semantic class, leveraging the final features for precise segmentation.

A linear layer with weight matrices wfc ∈ Rd×k projects the refined organ embeddings PT to the output class logits O, where k is the label index. The final class labels associated with the predicted masks ZT are determined by argmaxk(O) , identifying the most likely class for each segmented region.

$$\mathbf{O} = \mathbf{P}^T \mathbf{w}_{fc}$$
$$\hat{\mathbf{y}} = \arg\max_{k=0,1,\ldots,K-1} \mathbf{O}$$

Configuration Variants

1. Encoder Only Configuration:

This configuration combines a CNN with a transformer encoder, followed by a traditional U-Net decoder. The loss is calculated using a combination of **Dice loss** and **pixel-wise cross-entropy loss**. These metrics focus on enhancing both the accuracy of segmentation boundaries and the pixel-level classification performance.

2. Decoder Only Configuration:

Utilizes a standard CNN encoder paired with a transformer-based decoder, emphasizing the refinement of segmentation outputs. The loss function includes **Hungarian matching loss**, which **comprises pixel-wise class classification loss and binary mask loss** for each segmented prediction. The formulation is:

$$\mathcal{L} = \lambda_0(\mathcal{L}_{ce} + \mathcal{L}_{dice}) + \lambda_1 \mathcal{L}_{cls}$$

Additionally, **deep supervision** is implemented by **applying the training loss at the output at each stage of the decoder**, ensuring that each layer contributes effectively to the learning process.

3. Combined Encoder and Decoder:

Integrates both the transformer encoder and decoder for a comprehensive and unified approach to segmentation. The combined configuration utilizes **Hungarian matching loss**, aiming to synergize the strengths of both the encoder and decoder in a single coherent framework.

Effectiveness and Applications

Encoder Only: This setup shows superior performance in **multi-organ segmentation tasks**. The combination of CNN and transformer encoder efficiently captures both local and global features necessary for distinguishing between different organs in a complex anatomical landscape.

Decoder Only: More effective for **tumor segmentation**, where precision in identifying small, often irregularly shaped targets is critical. The transformer-based decoder excels in refining segmentation outputs, making it ideal for detailed and nuanced tasks like tumor detection.

Combined Encoder and Decoder: While **theoretically promising**, the combined approach **does not offer further enhancements for either multi-organ or hepatic vessel segmentation compared to using the configurations separately**. This could be due to an overlap in the functionalities or counteractive effects between the encoder and decoder components.

Query Number: Increasing the number of queries beyond the number of classes generally improves the model's ability to reduce false negatives. **However, further augmentation of queries** shows no benefit, indicating an optimal threshold for the number of queries based on the specific application needs.

Conclusion

The 3D TransUNet presents a flexible architecture with multiple configurations, each suited to specific types of medical image segmentation tasks. While each configuration has its strengths, the optimal choice depends on the specific requirements of the segmentation task at hand.