Kidney and Kidney Tumor Segmentation via Transfer Learning

Nozadze Giorgi[1]

Ludwig Maximilian University of Munich
G.Nozadze@campus.lmu.de

Abstract. Recently Segment anything model (SAM) has shown great promise for natural image segmentation. This model was trained on by far the largest segmentation dataset, consisting of over 11 million diverse images and 1 billion corresponding masks. The dataset's impressive size and high quality, combined with the powerful Transformer-based architecture, enabled the model to grasp a general understanding of objects and achieve exceptional zero-shot performances, sometimes even outperforming fully supervised models.
However, despite the significant advancements within the zero-shot framework, there are challenges when applying it to more specialized domains like medical and satellite imaging. Due to the scarcity of images from those domains in the training corpus, the model is not as accurate as it could be. Additionally in the fields where the segmentation of only certain, critical areas is desired using the SAM model can be overwhelming. In this paper, We aim to make use of different Transfer Learning techniques, such as Feature Extraction and Fine-tuning, and investigate different slight adaptations of the architecture to improve the performance of the SAM model and achieve high performance on a given medical image segmentation task.

Keywords: Segment Anything[2] · Semantic Segmentation · Transfer Learning

1 Introduction

Kidney cancer is a prevalent disease affecting a substantial number of individuals worldwide, with more than 430,000 new diagnoses and approximately 180,000 deaths reported annually. Detecting and accurately characterizing kidney tumors presents a significant challenge in clinical practice, as radiographically distinguishing between malignant and benign tumors remains a complex task. For this reason, active surveillance of small renal masses using modern computer vision techniques is becoming increasingly popular by proving its effectiveness.
This paper aims to address the automated semantic segmentation method on the dataset provided by the KiTS23 challenge[1] and existing research to contribute to the improvement of tumor segmentation tasks by utilizing the latest advancements in the field of computer vision.

2 Method

For our project, we have decided to work with the smallest pre-trained model from the Segment Anything Series, specifically the "vit-b" model. The model checkpoint is publicly available in Facebook research's official GitHub repository, allowing us to access its pre-trained weights.

2.1 Model Architecture

The model has the following network architecture:[F1]

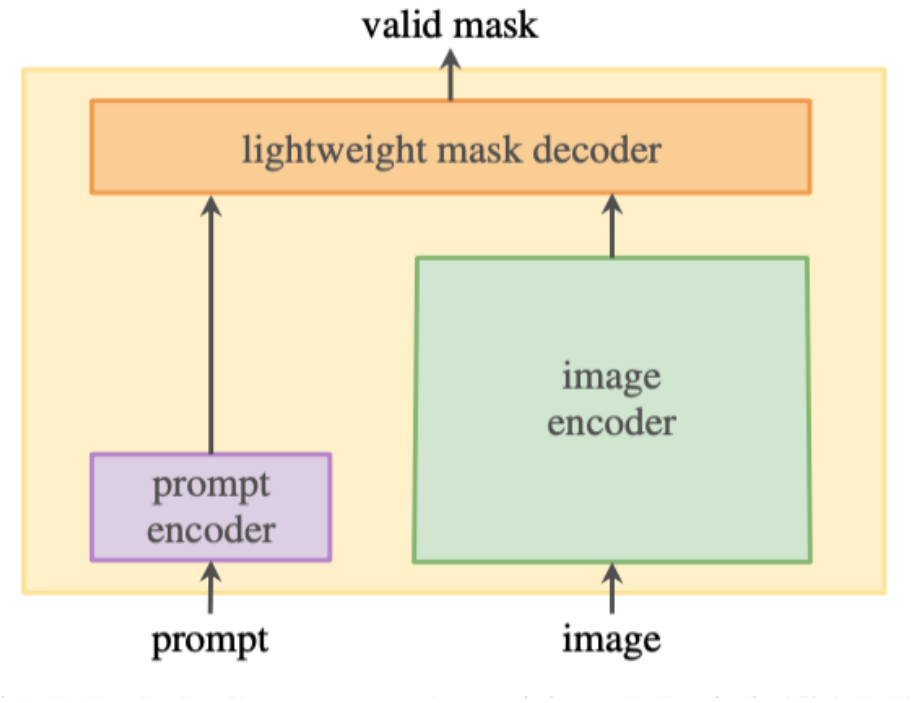


Fig. 1: Model Architecture[2]

In the first stage, it uses an MAE pre-trained Vision Transformer to produce high-quality image embeddings. In the second stage, The embeddings together with the output of the prompt encoder, which encodes the positional information of the desired area on the image are fed to the lightweight mask decoder, which is a modified Transformer decoder block followed by a prediction head, generates the mask prediction.

The SAM model gives an option to incorporate different types of prompts to provide positional and contextual information. It can be prompted using sparse

prompts in the form of point coordinates (x, y) or bounding box coordinates (x1, x2, y1, y2) as well as dense mask prompts. In our study, we opted for bounding box prompts because of their flexibility and the clarity of emphasis on areas of interest compared to the potentially ambiguous nature of point prompts.

In a nutshell, given a 3-channel RGB image of shape $3 \times 1024 \times 1024$, the MAE Vision Transformer generates image embeddings of size $256 \times 64 \times 64$. Subsequently, the importance of the regions in the image (and in the feature space) is emphasized and injected using prompt encoder-generated sparse prompt embeddings of size 2×256. These two combined yield a low-resolution prediction mask of size 256×256, which in the final step is post-processed to match the initial input image. After reviewing the zero-shot performance illustrated in Figure $\mathbf{F}^2$, which was generated using the "Automatic Mask Generation" framework described in the official segment anything paper[2], we believe, that the embeddings generated by the image encoder are of high quality. As a result, we have made the decision to freeze its parameters. This strategic move will prove advantageous as it significantly reduces computational overhead, especially considering that the image encoder alone accounts for about 85 % of the total parameters in the architecture, enabling us to allocate our resources and time efficiently, focusing on re-training or fine-tuning the mask decoder for optimal results.

Fig. 2: Zero-shot performance of SAM vit-b

2.2 Training and Validation Data

Our submission made use of the official KiTS23 training set alone. We will use 90% of the available data for training purposes and the remaining 10% will be reserved for validation purposes, allowing us to assess the effectiveness

and generalization capabilities of the resulting trained models across different experiments.

2.3 Understanding and Cleaning Data

Before beginning to process the data, as a first step, we analyzed the dataset and found 3 outliers in terms of image resolution (case_00160; case_00419; case_00425) that did not correspond to the 512x512 image size. To keep the consistency in the dataset, we reshaped them using bilinear interpolation.
Moreover, we visualized the average pixel distributions of each case and investigated the positive and negative sample ratios across different cases. By conducting this analysis, we gained valuable insights into the class imbalances present in the dataset. These findings played a crucial role in determining the appropriate resampling strategies and selecting suitable loss functions to tackle the class imbalance problem.

2.4 Preprocessing

Our approach consists of two main stages. In the first stage, we generate image embeddings using the image encoder described above. To ensure compatibility with the Vision Transformer, we perform several preprocessing steps on the images: Firstly, we convert the images into RGB format, after that we change the resolution to 1024x1024 pixels and finally, we normalize the pixel values of the images and feed them to the image encoder network. The resulting embeddings are then saved on the disk and we move on to the second stage and start training the mask decoder.

2.5 Training

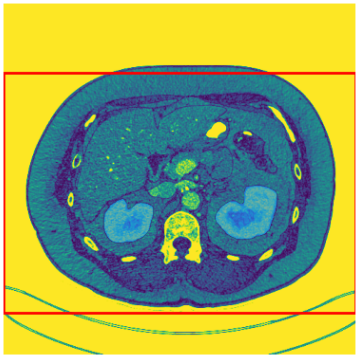
Resampling In our training process, we handle the batch size as a HyperParameter and are experimenting with different values, whereas the values of 64 or 128 slices have shown very promising results. Every slice within the batch is treated as an independent 2D image. This assumption, although restrictive, is imperative due to the inherent limitation of the SAM model, which can only handle 2D images as input. Additionally, efforts to explore alternative approaches for integrating sagittal or coronal views during training were unsuccessful. Hereby, we will sample batches by shuffling all cases, nevertheless maintaining the positive-negative sample ratio in the batch according to our analysis mentioned in the previous subsection. Moreover, we incorporate several data augmentation techniques to improve the generalization performance of the model. These techniques include rotation, scaling, and contrast adjustment.

Optimizer and Loss Function We train the mask decoder using the AdamW optimizer with the beta coefficients set to [0.9, 0.999], a weight decay of 0.1, and a dynamic learning rate. Initially, we set a high learning rate of 5e-5 for a

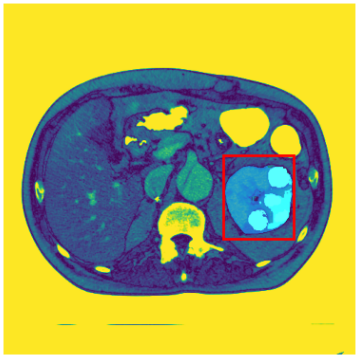
specific number of warm-up steps. After the warm-up period, the learning rate is decayed over time using cosine annealing to facilitate effective learning and convergence of the mask decoder. Regarding the Loss Function, we explored various loss functions to address the class imbalance issue. However, as our primary criterion, we selected the weighted sum of Dice and categorical Cross-Entropy loss with class weights, which were set according to our analysis of average pixel distributions mentioned in the subsection Understanding and Cleaning Data.

Models In total 3 different mask decoders were trained.

1. **ROI Decoder** was trained using a fixed, large box prompt[F3a] to identify regions of interest (Kidneys) in the given 2D slice. The binary masks produced by this model aid in the creation of prompts for the second phase of training, geared towards identifying cysts and tumors. To ensure comprehensive coverage of the complete kidney area, the bounding boxes drawn around the binary mask instances are intentionally expanded by incorporating a slight random factor.[F3b]
2. **Tumor/Cyst Decoders** were trained separately due to SAM's difficulty in identifying multiple objects within a single prompt framework. This issue will be further discussed, along with other relevant challenges and limitations, within the upcoming "Discussion and Conclusion" section.



(a) Prompt used in training of ROI Decoder

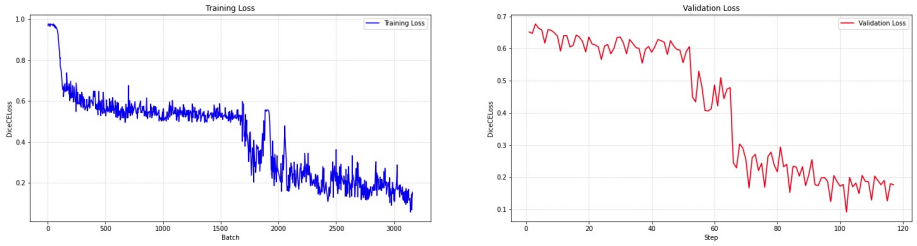


(b) Prompt used in training of Tumor/Cyst Decoders

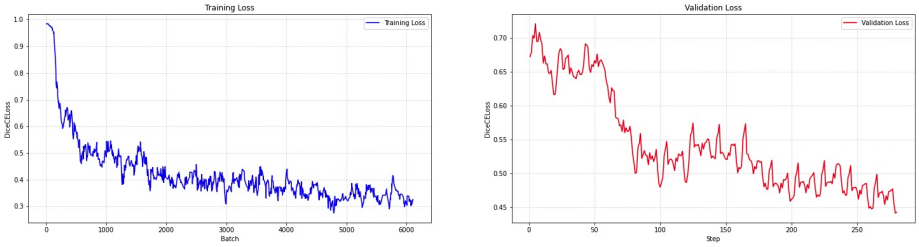
Fig. 3: Overview of Prompts in different decoders.

Resources The training process of a single mask-decoder was carried out on a single GPU with 40GB of GPU memory, spanning a total duration of two days.

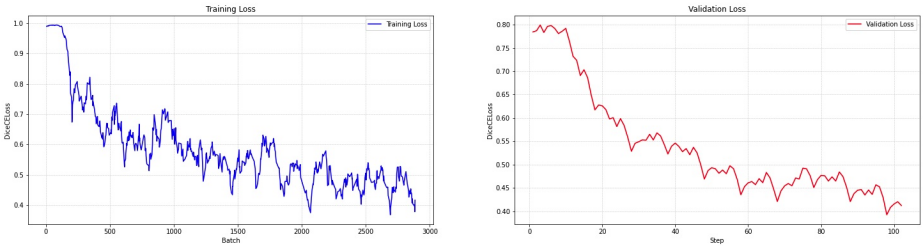
3 Results



(a) Training ROI Decoder



(b) Training Tumor Decoder



(c) Training Cyst Decoder

Fig. 4: Training Overview

Dice	Surface Dice	Tumor Dice	Kidney+Masses Dice	Masses Dice
0.432	0.239	0.222	0.807	0.268

Table 1: Results on the official test set

For a detailed overview of the training results, including the model checkpoint and an example notebook, please refer to the "TransferSAM" repository[4].

4 Discussion and Conclusion

Back in April, when Meta AI introduced the SAM model, discussions began to circulate about its exceptional performance in segmenting natural images. This paper and the method it proposes serve as an experiment to test the model's effectiveness in a scenario where the goal is to solely identify certain objects (in our case kidneys and their masses) in a fully automated fashion without any human intervention.

Adapting the model for this particular objective posed a formidable challenge, primarily due to the difficulty of automatically choosing suitable box prompts. We could not employ "Automatic Mask Generation" framework as shown in Figure $\mathbf{F^2}$, since we were only interested in certain objects and not all of them. To tackle this hurdle, it was necessary to divide the segmentation into two phases. In the initial phase, we used fixed-box prompts which enclosed the complete 2D slice to identify the kidney regions. It is noticeable, that the model has performed comparably well in this part, considering the fact, that it handled every slice of a 3D scan as an independent image. This success might be attributed to kidneys often occupying similar regions as large dense objects, which allowed the model to generalize prompt encoded information during training much better as opposed to the second stage where kidney masses were widely scattered into multiple instances.

From my perspective, the current state of the SAM model is not well-suited for tackling these kinds of segmentation challenges. The model stands out the most when employed in combination with human interaction, where the user provides a precise prompt for a specific object. A comprehensive exploration of this framework is presented in the MedSAM publication[3] and is definitely worth checking out.

References

1. Heller, N., Isensee, F., Trofimova, D., Tejpaul, R., Zhao, Z., Chen, H., Wang, L., Golts, A., Khapun, D., Shats, D., Shoshan, Y., Gilboa-Solomon, F., George, Y., Yang, X., Zhang, J., Zhang, J., Xia, Y., Wu, M., Liu, Z., Walczak, E., McSweeney, S., Vasdev, R., Hornung, C., Solaiman, R., Schoephoerster, J., Abernathy, B., Wu, D., Abdulkadir, S., Byun, B., Spriggs, J., Struyk, G., Austin, A., Simpson, B., Hagstrom, M., Virnig, S., French, J., Venkatesh, N., Chan, S., Moore, K., Jacobsen, A., Austin, S., Austin, M., Regmi, S., Papanikolopoulos, N., Weight, C.: The kits21 challenge: Automatic segmentation of kidneys, renal tumors, and renal cysts in corticomedullary-phase ct (2023)
2. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything. arXiv:2304.02643 (2023)
3. Ma, J., He, Y., Li, F., Han, L., You, C., Wang, B.: Segment anything in medical images (2023)
4. Nozadze, G.: Transfersam. https://github.com/Noza23/TransferSAM.git (2023)