Segmentation and Recognition Model based on SAM and CLIP

Yuchen Xia[1]
[1]School of Information Science and Engineering,
Shandong University, Qingdao 266237, China.
Emails: yuchenxia@mail.sdu.edu.cn

***Abstract*—Image analysis and processing have been a hot topic in current research. With the development of hardware, neural networks have provided methods for image analysis and processing. The Transformer has emerged as a powerful tool in natural language processing (NLP), achieving tremendous success, and the Generative Pre-trained Transformer (GPT) model has even realized a model that can solve all problems. With the development of Transformer models in the image field, Contrastive Language-Image Pre-Training (CLIP) provides bimodal pairing of text and images, while the Segment Anything Model (SAM) enables the segmentation of any object. This paper will mainly focus on using SAM and CLIP to segment and recognize any specified object in any scenario. You can find the relevant code here: https://github.com/MonsterXia/Segmentation-and-Recognition-Model-based-on-SAM-and-CLIP**

***Index Terms*—CLIP, SAM, segment, recognize.**

I. Introduction

Neural networks are a computational model inspired by the network of neurons in the brain. Early neural network models included Hopfield and perceptron, but due to limitations in computing power and data availability, these models were heavily constrained in their performance. Moore's Law [1] states that "The number of transistors incorporated in a chip will approximately double every 18 months." With the development of hardware, computational power of computers has significantly increased, making it possible to process complex neural networks.

The proposals of LeNet-5 [2] , AlexNet [3], and VGG [4] have enabled networks to better extract image features, allowing for classification and other tasks. In traditional deep learning networks, in order to improve model accuracy, it is often necessary to increase the number of network layers. ResNet [5] addresses the problems of network degradation, and gradient vanishing/exploding that occur with large numbers of layers in traditional networks, allowing the network to continue to develop deeper. However, as the network depth increases, it causes a sharp expansion of parameters, which reduces the efficiency of the model in processing information and results in an explosive increase in training time.

This enables networks to perform very well on specific tasks such as classification. With the proposal and application of attention mechanisms , a new solution has emerged [6]. GPT [7] is a NLP model developed by OpenAI, which has been pre-trained on large-scale corpora through unsupervised learning, enabling it to perform tasks such as generating fluent text, understanding text, answering questions, and translating. Noticing the tremendous impact of Transformers in the NLP field, to handle problems in other fields, various variations have been derived such as Sandwich Transformers [8], Universal Transformers [9], Transformer-XL [10], Reformer [11], and Vision Transformer [12]. Among them, ViT is used to solve image-related problems. To improve the ability of ViT to handle images, Transformer in Transformer (TNT) [13] has been proposed. On April 5th, 2023, Meta AI released the SAM [14], a large segmentation model designed to segment any object. However, Meta AI did not disclose the prompt input section, which makes it impossible to specify the segmentation of a particular object using given prompts. Therefore, in order to enable our model to segment specific objects, we still need to identify what is being segmented. Open AI's CLIP [15] model achieves good results by leveraging contrastive learning to pair a wide range of images and text. Although Meta AI provided the multimodal large-scale model ImageBind [16] on May 9th, 2023, this model uses multiple modalities, which makes the entire system relatively large and requires high deployment requirements. Therefore, we still use CLIP. The main contributions of this paper can be concluded as follows:

1) The SAM and CLIP models are combined to detect selectable objects in any image.
2) To improve the model validation, it has been enhanced to allow for overall performance testing on different datasets, rather than just single image operations.
3) The parameter loading process has been optimized to distribute the device load more evenly, enabling the model to adapt to a wider range of device environments and reducing its dependence on high-performance devices.

In this paper, the section II explains the methodology used, while the section III the results are compared and analyzed, leading to the final conclusion.

II. METHODOLOGY

In this section, the principles and implementation processes of the CLIP and SAM models are introduced.

A. CLIP

To avoid having to re-label data and retrain models for each specific scenario, CLIP solves all text-image multimodal

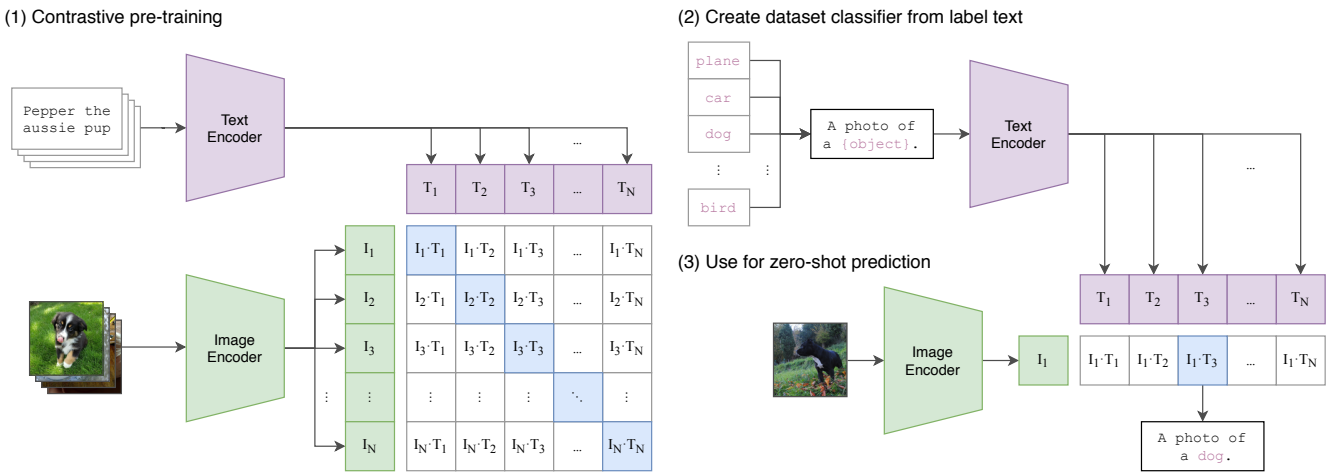

Fig. 1. While standard image models jointly train an image feature extractor and a linear classifier to predict some label, CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the names or descriptions of the target dataset's classes.

matching problems through a single model. It directly attempts to solve all problems through zero-shot learning by pre-training the model, thereby saving deployment costs for downstream tasks.

Similar methods were already present when Transformer was introduced in 2017, but they did not achieve good results. CLIP innovatively uses a large amount of data for pre-training and achieves good results with a small parameter size. It is trained using 400 million paired data and text, without using any data from ImageNet, and crawled without labeling. The results obtained through zero-shot learning are roughly consistent with those of ResNet-50 trained on ImageNet data.

Figure 1 shows an example of the model architecture of CLIP. The model encodes images and text into features using Transformer separately, calculates the similarity between the image feature vector and the text feature vector, and uses contrastive learning to bring the vectors closer to positive samples and farther away from negative samples, thereby achieving text-image matching. After training, during the inference process, we can provide a set of texts, input an image, and the model will return the similarity between the given text and the image, outputting the corresponding probability values. Then, we can use softmax to choose the text that is most likely to match the image.

B. SAM

The SAM model takes the first step towards large-scale models in the field of segmentation and detection. To enable the model to perform segmentation on any object, similar to the CLIP model, the amount of training data is crucial. Meta AI team innovatively proposed a data engine. The existing datasets are inadequate to provide the massive amount of data required to train models. Therefore, using their efficient model in data collection, they built the largest segmentation dataset to date, with over 1 billion masks on 11 million licensed and privacy-respecting images. Similarly, the SAM model can achieve segmentation tasks through zero-shot learning.

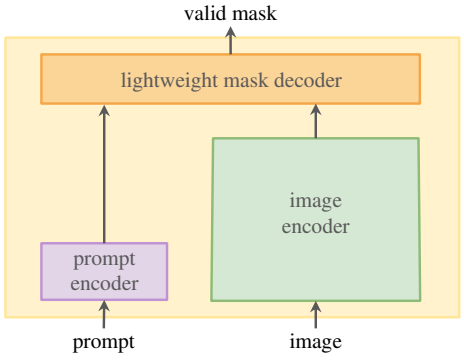

Fig. 2. SAM.

The macro structure of the SAM model is shown in Figure 2. The detailed structure of the lightweight mask decoder is illustrated in Figure 3. Although Meta AI mentioned in the original paper that various prompts such as points, boxes can be used as input, no other modality of prompt words such as text was publicly disclosed. We will utilize CLIP to simulate these in our implementation.

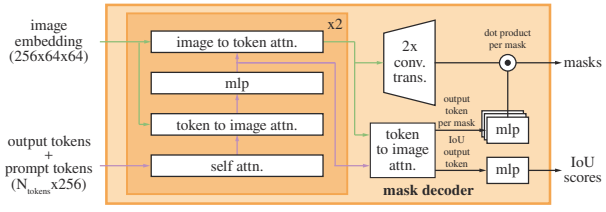

Fig. 3. Lightweight mask decoder.

To reduce the training cost, the SAM model utilizes a series

of pre-trained models such as MAE [17] for training. To enable the SAM model to be deployed in a wider range of scenarios, not only does it make the output mask closer to the prompt, but it also adjusts the prompt in reverse based on the output mask, making the feature vector extraction more detailed. This allows for a reduction in the number of Transformer layers, greatly reducing the computational cost of the model. However, to achieve the same level of accuracy, more data is required, which is addressed by Meta AI's data engine.

III. SIMULATION

We have selected Python version 3.9.16, PyTorch version $2.0.0+cu118$, and Torchvision version $0.15.1+cu118$ for our environment. In order to achieve a better performance, we have chosen the ViT-H SAM model ($sam\_vit\_h\_4b8939.pth$). We have chosen the $openai/clip-vit-base-patch32$ model for CLIP.

Taking the recognition and segmentation of sports cars as an example, we randomly set about ten other categories as the distinguishing categories, with sports cars being the target. We input a sports car image to the model, as shown in Figure 4a. The image is segmented, assigning a mask to each object and different colors to different masks, as shown in Figure 4b. Using the segmented masks, we multiply them with the original image to obtain images of the segmented objects. These images are then fed into CLIP for recognition and analysis, in order to select the target masks that we want, as shown in Figure 4c. Finally, by multiplying the obtained target masks with the original image, we can highlight the object that we want to segment, as shown in Figure 4d.

(a) Original image.

(b) All masks.

(c) Selected masks

(d) Result

Fig. 4. Model input, incremental validation, output results.

From Figure 4, we can see that the model can achieve relatively accurate results in specific scenes. In order to verify the generalization of the model, we conducted a simple test on CIFAR-10 using the entire model. The test result of CLIP alone on the dataset is 86.67%, while the test result of the complete model on the dataset is 63.72%, as shown in Table I.

TABLE I
MODEL ACCURACY ON CIFAR-10

Model	CLIP	SAM+CLIP
Accuracy	86.67%	63.72%

Although the accuracy has decreased compared to CLIP alone, segmentation has been achieved, and the extra cost is acceptable considering the functionality gained. Moreover, following the suggestion in the CLIP paper, by modifying the prompt words and providing a more precise description of the target in a specific environment, the accuracy of the entire model can be significantly improved with clearer images, and the segmentation accuracy can reach over 90%.

IV. CONCLUSION

By combining and improving the SAM and CLIP models, we have achieved detection and segmentation of specific targets in natural scenes. The model has achieved good results on some datasets, and by changing the target settings, it can directly adapt to a wide range of targets in natural scenes through zero-shot learning.

REFERENCES

[1] G. E. Moore, "Cramming more components onto integrated circuits," Proceedings of the IEEE, vol. 86, no. 1, pp. 82-85, 1998.
[2] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.
[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Communications of the ACM, vol. 60, no. 6, pp. 84-90, 2017.
[4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.
[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770-778.
[6] A. Vaswani et al., "Attention is all you need," Advances in neural information processing systems, vol. 30, 2017.
[7] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.
[8] O. Press, N. A. Smith, and O. Levy, "Improving Transformer Models by Reordering their Sublayers," ed, 2019.
[9] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and L. Kaiser, "Universal Transformers," ArXiv, vol. abs/1807.03819, 2018.
[10] Z. Dai, Z. Yang, Y. Yang, J. G. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: Attentive Language Models beyond a Fixed-Length Context," ArXiv, vol. abs/1901.02860, 2019.
[11] N. Kitaev, L. Kaiser, and A. Levskaya, "Reformer: The Efficient Transformer," ArXiv, vol. abs/2001.04451, 2020.
[12] A. Dosovitskiy et al., "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," ArXiv, vol. abs/2010.11929, 2020.
[13] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in Transformer," ArXiv, vol. abs/2103.00112, 2021.
[14] A. Kirillov et al., "Segment anything," arXiv preprint arXiv:2304.02643, 2023.
[15] A. Radford et al., "Learning transferable visual models from natural language supervision," in International conference on machine learning, 2021: PMLR, pp. 8748-8763.
[16] R. Girdhar et al., "Imagebind: One embedding space to bind them all," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 15180-15190.
[17] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 16000-16009.