CATrans: Context and Affinity Transformer for Few-Shot Segmentation

Shan Zhang [1*] , **Tianyi Wu**[2,3] , **Sitong Wu**[2,3] , **Guodong Guo**[2,3 †]
[1]Australian National University, Canberra, Australia
[2]Institute of Deep Learning, Baidu Research, Beijing, China
[3]National Engineering Laboratory for Deep Learning Technology and Application, Beijing, China
Shan.Zhang@anu.edu.au, wusitong98@gmail.com, {wutianyi01, guoguodong01}@baidu.com

Abstract

Few-shot segmentation (FSS) aims to segment novel categories given scarce annotated support images. The crux of FSS is how to aggregate dense correlations between support and query images for query segmentation while being robust to the large variations in appearance and context. To this end, previous Transformer-based methods explore global consensus either on context similarity or affinity map between support-query pairs. In this work, we effectively integrate the context and affinity information via the proposed novel Context and Affinity Transformer (CATrans) in a hierarchical architecture. Specifically, the Relation-guided Context Transformer (RCT) propagates context information from support to query images conditioned on more informative support features. Based on the observation that a huge feature distinction between support and query pairs brings barriers for context knowledge transfer, the Relation-guided Affinity Transformer (RAT) measures attention-aware affinity as auxiliary information for FSS, in which the self-affinity is responsible for more reliable cross-affinity. We conduct experiments to demonstrate the effectiveness of the proposed model, outperforming the state-of-the-art methods.

1 Introduction

Fully supervised semantic segmentation has made tremendous progress in recent years [Long *et al.*, 2015; Lin *et al.*, 2017]. However, these methods rely heavily on a large amount of pixel-wise annotations which requires intensive manual labor, and are incapable of generalizing to new classes with a handful of annotations. In contrast, humans can recognize a new category even with little guidance. Inspired by this, Few-shot segmentation (FSS) has recently received a growing interest in the computer vision community [Wang *et al.*, 2019; Zhang *et al.*, 2019b; Zhang *et al.*, 2020].

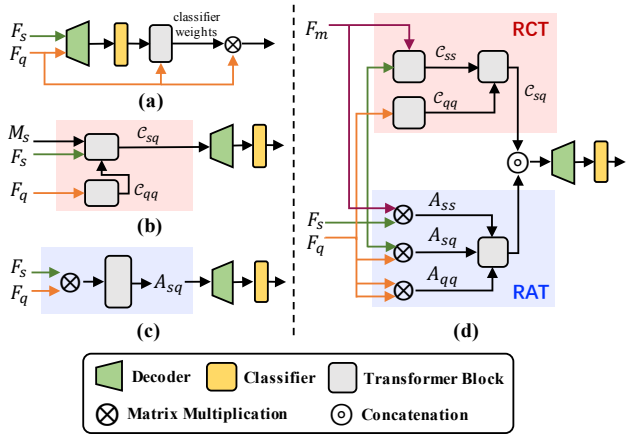


Figure 1: Comparisons with different Transformer-based few-shot segmentation methods. The red and blue square shadow denote context ($\mathcal{C}$) and affinity ($\mathcal{A}$) aggregation, respectively. (a) Classifier Weight Transformer [Lu *et al.*, 2021]. (b) Cycle-Consistent Transformer [Zhang *et al.*, 2021]. (c) Cost Aggregation Transformer [Cho *et al.*, 2021]. (d) Our Context and Affinity Transformer with Relation-guided Context Transformer (RCT) and Relation-guided Affinity Transformer (RAT).

The goal of FSS is to segment the novel class in the query image, conditioned on the given support set which contains only a few support images and the corresponding ground-truth masks. A fundamental challenge is the large intra-class appearance and geometric variations between support-query pairs, so the key issue is how to effectively reason the relationships of paired samples.

Most FSS methods [Cho *et al.*, 2021; Zhang *et al.*, 2021; Lu *et al.*, 2021] follow a learning-to-learn paradigm. Specifically, features are extracted from both query and support images, and then passed through a feature matching procedure to transfer the support mask to the query image. The convolutional neural network (CNN)-based approaches [Wang *et al.*, 2019; Yang *et al.*, 2020; Liu *et al.*, 2020] condense the masked object features in the support image into a single or few prototypes. Recently, some approaches introduce Transformer-based architecture to establish pixel-wise matching scores between support-query pairs, containing two main technical routes, *ie*., the context and affinity.

[*]Interns at the Institute of Deep Learning, Baidu Research
[†]Corresponding author

As the dense **context** information is beneficial to FSS task, especially when the large intra-class variances exist in the support and query set, [Lu *et al.*, 2021] propose the Classifier Weight Transformer (CWT) to dynamically adapt the classifier's weights trained on support set to each query image as shown in Figure 1 (a). [Zhang *et al.*, 2021] aggregates the context information within query images and between support-query pairs via transformer blocks for self- and cross-alignment, as shown in Figure 1 (b). This method, however, suffers from unrepresentative of support feature which stimulates us to propose the Relation-guided Context Transformer (RCT), in which the global context information of support can be considered with a mask encoder [Johnander *et al.*, 2021] and a transformer block. The RCT takes as input discriminative self-feature to build more accurate cross-correlation of context, as briefly shown in Figure 1 (d).

The attention-aware **affinity** is globally constructed between support and query features as another guidance for query segmentation. The Cost Aggregation with Transformers (CATs) [Cho *et al.*, 2021] builds the cross-affinity between support and query features followed by transformer blocks, as shown in Figure 1 (c). However, this method does not incorporate individual self-affinity for support object or query image to disambiguate noisy correlations, which measures pixel-wise correspondences within itself, enabling each spatial fiber to match itself and other tokens. We, thus, design a Relation-guided Affinity Transformer (RAT) for generating a reliable cross-affinity map inherited from the self-affinity. The illustration is schematically depicted in the Figure 1 (d).

Additionally, we explore how to utilize both context and affinity guidance simultaneously. To be specific, we develop a hierarchical CATrans: Context and Affinity Transformer, where we leverage a stack of multi-level correlation maps related to both context and affinity. Moreover, following by [Johnander *et al.*, 2021], we also concatenate the query embedding with high resolution with those low-resolution correlation maps to guide the decoder.

Overall, our contributions are : i) we design a Relation-guided Context Transformer (RCT) with the enhanced support features to propagate informative semantic information from support to query images. ii) we develop a Relation-guided Affinity Transformer (RAT) to measure the reliable cross correspondences by considering the auxiliary self-affinity of both support object and query images. iii) we propose Context and Affinity Transformer, dubbed as CATrans, in a hierarchical architecture to aggregate the context and affinity together, resulting in discriminative representations from support to query mask, enhancing robustness to intra-class variations between support and query images. Our CATrans outperforms the state-of-the-art methods on two benchmarks, Pascal-5^i and COCO-20^i.

2 Related Work

2.1 Semantic Segmentation

Semantic segmentation is a fundamental problem in computer vision, which aims to classify each pixel of an image into predefined categories. Most existing semantic segmentation methods are based on fully convolutional networks (FCNs) [Long *et al.*, 2015], that replaces the fully connected layer with fully convolutional ones for pixel-level prediction. Recent breakthroughs in semantic segmentation have mainly come from multi-scale feature aggregation or attention mechanisms. However, the traditional fully supervised segmentation methods require large amounts of image-mask pairs for training, which are very expensive and time consuming. Additionally, it cannot extend model's generalizability to unseen classes with only a few well-labeled samples.

2.2 Few-shot Segmentation

Few-shot semantic segmentation has attracted lots of research attentions after that [Shaban *et al.*, 2017], which first dealt with this issue by proposing to adapt the classifier for each class, conditioned on the support set. Recent approaches formulate few-shot segmentation from the view of metric learning. [Dong and Xing, 2018] learned prototypes for different classes and the segmentation results are made by cosine similarity between the features and the prototypes. [Wang *et al.*, 2019] developed an efficient prototype learning framework to build consistent prototypes. PFENet [Tian *et al.*, 2020] made progress by further designing an effective feature pyramid module and leveraged a prior map to achieve a better segmentation performance. Recently, [Liu *et al.*, 2020; Yang *et al.*, 2020] found that it is insufficient to represent a category with a single support prototype. Therefore, they used multiple prototypes to represent the support objects via the EM algorithm or K-means clustering. However, these methods disregard the pixel-wise relationships of spatial structure in feature maps.

Recent works [Zhang *et al.*, 2019a; Zhang *et al.*, 2021; Cho *et al.*, 2021; Lu *et al.*, 2021] attempted to fully utilize a correlation map to leverage the pixel-wise relationships between support and query features. Specially, [Zhang *et al.*, 2019a] used graph attention networks to propagate information from the support image to query images, and [Zhang *et al.*, 2021] utilized cycle-consistent transformer to aggregate the pixel-wise support features into the query. [Lu *et al.*, 2021] proposed a classifier weight transformer where the transformer is applied to adapt the classifier solely by freezing the encoder and decoder. However, all these dense matching methods focus on either context correspondences or affinity maps only. This is no study about whether these two measures are complementary, and can be integrate to achieve a better performance.

2.3 Transformers in Vision

Recently, transformers, first introduced in natural language processing [Vaswani *et al.*, 2017], and are receiving increasing interests in the computer community. Since the pioneer works such as ViT [Dosovitskiy *et al.*, 2021], it demonstrates the pure transformer architecture can achieve the state-of-the-art for image recognition. On the other hand, DETR [Carion *et al.*, 2020] built up an end-to-end framework with a transformer encoder-decoder on top of backbone networks for object segmentation. And its deformable variants [Zhu *et al.*, 2021] improved the performance and training efficiency. However, there are few studies that compute both context and affinity.

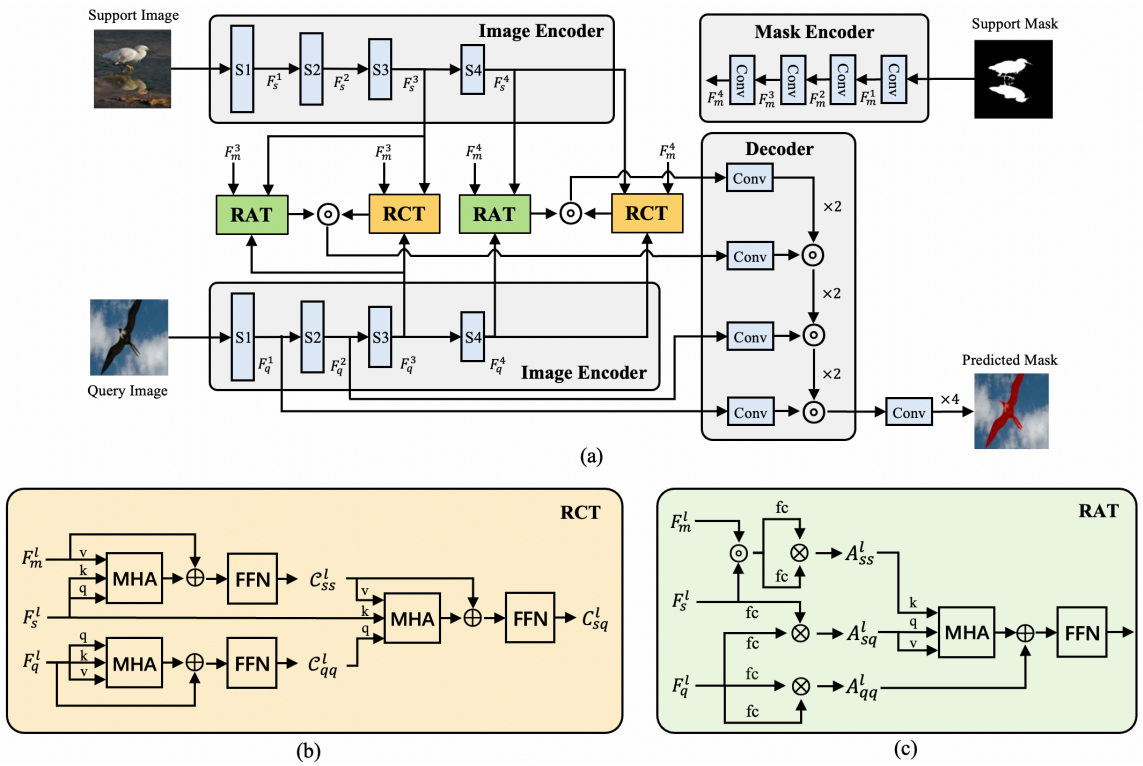


Figure 2: (a) The overall framework of our Context and Affinity Transformer (CATrans). The detailed architecture of our Relation-guided Context Transformer (RCT) and Relation-guided Affinity Transformer (RAT) are shown in (b) and (c), respectively.

3 Preliminaries

3.1 Problem Formulation

The goal of few-shot segmentation is to segment novel objects with very few annotated samples. Specifically, all classes are divided into two disjointed class set C_{train} and C_{test}. To mitigate the overfitting caused by insufficient training data, we follow the common protocol called episodic training. Under K-shot setting, each episode is composed of a support set $\mathcal{S} = \{(I_s, M_s)\}^K$, where I_s, M_s are support image and its corresponding mask, and a query sample $\mathcal{Q} = (I_q, M_q)$, where I_q, M_q are the query image and mask, respectively. In particular, given dataset $D_{train} = \{\mathcal{S}, \mathcal{Q}\}^{N_{train}}$ and $D_{test} = \{\mathcal{S}, \mathcal{Q}\}^{N_{test}}$ with category set C_{train} and C_{test}, respectively, where N_{train} and N_{test} is the number of episodes for training and test sets. During training, our model takes a sampled episode from both support masks M_s and query masks M_q, and only use support masks for predicting the query segmentation map $\hat{M}_q$ during testing.

3.2 Revisiting of Transformer

Transformer block [Vaswani *et al.*, 2017] consists of multi-head attention (MHA) and multi-layer perception (MLP) with inputs of a set of Query (Q), Key (K) and Value (V) elements. In addition, LayerNorm (LN) and residual connection are available at the end of each block. Specially, an attention layer is formulated as Architecture Overview:

$$\text{Atten}(Q, K, V) = softmax\Big(\frac{Q \cdot K^T}{\sqrt{d}}\Big) \cdot V, \qquad (1)$$

where $[Q; K; V] = [W_q F_q; W_k F_k; W_v F_v]$, in which F_q is the input query sequence, F_k/F_v is the input key/value sequence, $W_q, W_k, W_v \in \mathcal{R}^{c\times c}$ are learnable weights, c is the channel dimension of the input sequences.

The multi-head attention layer is an enhancement of attention layer, where h attention units are applied and then concatenated together. Concretely, this operation splits input sequences along the channel dimension c into h groups:

$$\text{MHA}(Q, K, V) = [head_1, \ldots, head_h], \qquad (2)$$

where $head_m = \text{Atten}(Q_m, K_m, V_m)$ and the inputs $[Q_m, K_m, V_m]$ are the m^{th} group from $[Q, K, V]$ with dimension c/h.

4 Methodology

Below we present the overall architecture of CATrans followed by a description of its individual components.

4.1 Architecture Overview

The overall architecture of our Context and Affinity Transformer (CATrans) is illustrated in Figure 2, which consists of Image Encoder, Mask Encoder, Relation-guided Context Transformer (RCT), Relation-guided Affinity Transformer (RAT) and the Decoder. Specifically, the input support-query images $\{I_s, I_q\}$ are passed through the Image Encoder to extract multi-scale features $\{F_s^l, F_q^l \in \mathbb{R}^{H_l \times W_l \times C_l}\}_{l=1}^4$, where l denotes the scale-level, and pyramid mask features

$\{F_m^l \in \mathbb{R}^{H_m^l \times W_m^l \times C_m^l}\}_{l=1}^4$ are extracted via the Mask Encoder with the input of support binary mask M_s. The resulting triple $\{F_s^l, F_q^l, F_m^l\}_{l=3}^4$ is passed into the RCT and RAT, respectively. In practice, the pixel-wise context $\mathcal{C}^l \in \mathbb{R}^{H_l \times W_l \times C_m^l}$ provided by the RCT and dense affinity map $\mathcal{A} \in \mathbb{R}^{H_l W_l \times H_l W_l}$ generated by RAT are concatenated for information aggregation. In the decoder, the fused representations associate with the high-resolution features of query image $\{F_s^l, F_q^l\}_{l=1}^2$ for predicting the query mask $\hat{M}_q$.

4.2 Relation-guided Context Transformer

Inspirations: Current CNN-based prototypical learning approaches, *ie*. [Wang *et al.*, 2019], condense the support features into a single or few context-wise prototypes. However, prototypical features inevitably drop spatial information and fail to discriminate incorrect matches due to limited receptive fields of CNNs. So recent research applied the Transformer-based architecture, such as [Zhang *et al.*, 2021], to establish long-range and pixel-wise context relationships between paired support-query samples, outperforming previous CNN-based methods by a large margin. We conjecture that the representative features within the individual support and query image would help to aggregate more precise cross-relationships, being robust to large intra-class differences of paired support-query samples.

To this end, the RCT is designed and shown in Figure 2 (b). We flatten the given triple features (F_s^l, F_q^l, F_m^l) into 1D tokens as inputs for the following items.

Self-context. Self-context is separately employed for support objects and query features by aggregating its relevant context information, leading to more informative support and query features to be connected via the cross-context. The resulting contexts $\mathcal{C}_{ss}^l$ and $\mathcal{C}_{qq}^l$ are designed as:

$$\mathcal{C}_{ss}^l = \text{MLP}(\text{LN}(\text{MHA}(F_s^l, F_s^l, F_m^l))), \tag{3}$$

$$\mathcal{C}_{qq}^l = \text{MLP}(\text{LN}(\text{MHA}(F_q^l, F_q^l, F_q^l))), \tag{4}$$

where MHA $(\cdot)$, LN$(\cdot)$ and MLP$(\cdot)$ are operations introduced in the section 3.

Cross-context. Considering that $\mathcal{C}_{ss}^l$, guided by mask features F_m^l, mainly focuses on the foreground, but background support pixels are beneficial for building the semantic relationships, we collaborate the enhanced self-context features of support and query with F_s^l to establish the pixel-wise cross-context. The process of cross-context is formed as:

$$\mathcal{C}_{sq}^l = \text{MLP}(\text{LN}(\text{MHA}(\mathcal{C}_{qq}^l, F_s^l, \mathcal{C}_{ss}^l))), \tag{5}$$

where $\mathcal{C}_{sq}^l \in \mathcal{R}^{H_l W_l \times C_m^l}$ is spatially rearranged to the shape of $H_l \times W_l \times C_m^l$.

4.3 Relation-guided Affinity Transformer

A huge feature distinction between the support and query images brings barriers for context knowledge transfer, which cripples the segmentation performance. We explore several attention-aware affinity maps that measure pixel-wise correspondences to facilitate the FSS task, as shown in Figure 2 (c). Overall, this module provides affinity guidance stemming from attention-aware features instead of semantics.

Why needs self-affinity? The training samples belonging to the same class always have features with large variations in appearance as these objects are taken in unconstrained settings. Take aeroplane as an example, all aeroplanes are made by metal and have wings. These features can be seen as intrinsic features. As the differences of shooting angle and lighting conditions, the shape and color of aeroplanes can be different. In few-shot segmentation, we need to enable each pixel-wise feature belonging to itself to match pixel-wise feature at the same position, making it robust to large variations in object appearance between the support and query images.

What is Self-affinity? For a support image, we utilize high-dimension support mask features F_m^l concatenated with support image features F_s^l for estimating its affinity map by scaled-dot product followed by softmax function. The process for l-th features can be defined as:

$$\mathcal{A}_{ss}^l = \text{softmax}\Big(\frac{(f_m^l || f_s^l)W_q \cdot ((f_m^l || f_s^l)W_k)^T}{\sqrt{C_l + C_m^l}}\Big), \tag{6}$$

where $\{\cdot||\cdot\}$ denotes the concatenation operation, softmax $(\cdot)$ is a row-wise softmax function for attention normalization and two individual fc layers are applied to learn discriminative features by learnable parameters. Analogy to support feature, we formulate the self-affinity for query as:

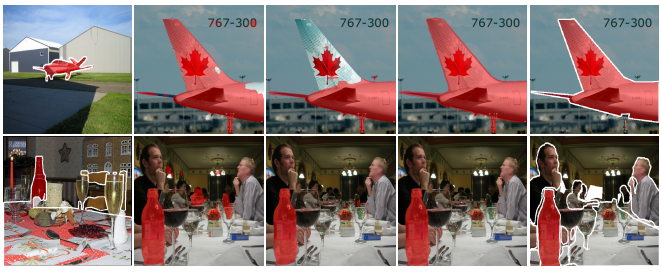
$$\mathcal{A}_{qq}^l = \text{softmax}\Big(\frac{f_q^l W_q \cdot (f_q^l W_k)^T}{\sqrt{C_l}}\Big). \tag{7}$$

Cross-affinity. The cross-affinity between support and query features F_{ca}^l is computed by:

$$\mathcal{A}_{sq}^l = \text{softmax}\Big(\frac{f_q^l W_q \cdot (f_s^l W_k)^T}{\sqrt{C_l}}\Big). \tag{8}$$

However, solely relying on the cross-affinity between features often suffers from the challenges caused by large intra-class variations. We then embed the self-affinity of query $\mathcal{A}_{qq}^l$ into cross-affinity. The final cross-affinity is formulated as:

$$\text{MLP}(\text{LN}(\text{MHA}(\mathcal{A}_{sq}^l, \mathcal{A}_{ss}^l, \mathcal{A}_{sq}^l) + \mathcal{A}_{qq}^l)). \tag{9}$$



Support set RAT RCT CATrans GT

Figure 3: Visualize the results predicted by the CATrans and its variants on PASCAL-5^i, 1-shot setting. GT means Ground Truth of the query image.

Method	Venue	Backbone	1-shot					5-shot				
			5^0	5^1	5^2	5^3	Mean	5^0	5^1	5^2	5^3	Mean
CANet [Zhang *et al.*, 2019b]	CVPR19	ResNet-50	52.5	65.9	51.3	51.9	55.4	55.5	67.8	51.9	53.2	57.1
PGNet [Zhang *et al.*, 2019a]	ICCV19		56.0	66.9	50.6	50.4	56.0	57.7	68.7	52.9	54.6	58.5
RPMMs [Yang *et al.*, 2020]	ECCV20		55.2	66.9	52.6	50.7	56.3	56.3	67.3	54.5	51.0	57.3
PPNet [Liu *et al.*, 2020]	ECCV20		47.8	58.8	53.8	45.6	51.5	58.4	67.8	64.9	56.7	62.0
PFENet [Tian *et al.*, 2020]	TPAMI20		61.7	69.5	55.4	56.3	60.8	63.1	70.7	55.8	57.9	61.9
CWT [Lu *et al.*, 2021]	ICCV21		56.3	62.0	59.9	47.2	56.4	61.3	68.5	68.5	56.6	63.7
CyCTR [Zhang *et al.*, 2021]	NeurIPS21		**67.8**	72.8	58.0	58.0	64.2	71.1	73.2	60.5	57.5	65.6
DGPNet [Johnander *et al.*, 2021]	arXiv21		63.5	71.1	58.2	61.2	63.5	72.4	76.9	73.2	71.7	73.5
CATrans	Ours		67.6	**73.2**	**61.3**	**63.2**	**66.3**	**75.1**	**78.5**	**75.1**	**72.5**	**75.3**
FWB [Nguyen and Todorovic, 2019]	ICCV19	ResNet-101	51.3	64.5	56.7	52.2	56.2	54.9	67.4	62.2	55.3	59.9
DAN [Wang *et al.*, 2020]	ECCV20		54.7	68.6	57.8	51.6	58.2	57.9	69.0	60.1	54.9	60.5
PFENet [Tian *et al.*, 2020]	TPAMI20		60.5	69.4	54.4	55.9	60.1	62.8	70.4	54.9	57.6	61.4
CWT [Lu *et al.*, 2021]	ICCV21		56.9	65.2	61.2	48.8	58.0	62.6	70.2	68.8	57.2	64.7
CyCTR [Zhang *et al.*, 2021]	NeurIPS21		**69.3**	72.7	56.5	58.6	64.3	73.5	74.0	58.6	60.2	66.6
DGPNet [Johnander *et al.*, 2021]	arXiv21		63.9	71.0	63.0	61.4	64.8	74.1	77.4	76.7	73.4	75.4
CATrans	Ours		67.8	**73.2**	**64.7**	**63.2**	**67.2**	**75.2**	**78.4**	**77.7**	**74.8**	**76.5**
CATrans	Ours	Swin-T	**68.0**	**73.5**	**64.9**	**63.7**	**67.6**	**75.9**	**79.1**	**78.3**	**75.6**	**77.3**

Table 1: Comparison with the state-of-the-art in 1-shot and 5-shot segmentation on PASCAL-5^i dataset using the mIoU (%) as evaluation metric. Best results in bold.

RAT	RCT	RCT$^\circ$	PASCAL-5^i 1-shot	PASCAL-5^i 5-shot	COCO-20^i 1-shot	COCO-20^i 5-shot
			58.3	62.7	33.4	42.6
✓			65.1	70.6	41.3	54.9
	✓		66.0	72.4	43.9	56.4
		✓	65.3	71.3	42.1	55.3
✓	✓		**66.3**	**75.3**	**46.6**	**58.2**

(a)

#l 1	#l 2	#l 3	#l 4	PASCAL-5^i 1-shot	PASCAL-5^i 5-shot	COCO-20^i 1-shot	COCO-20^i 5-shot
✓		✓		62.4	72.5	42.4	55.8
	✓			64.8	73.3	44.8	57.9
				65.1	74.2	45.3	58.1
✓			✓	64.9	74.1	44.8	56.5
	✓		✓	61.2	71.7	40.6	53.3
		✓	✓	**66.3**	**75.3**	**46.6**	**58.2**

(b)

#h	PASCAL-5^i 1-shot	PASCAL-5^i 5-shot	COCO-20^i 1-shot	COCO-20^i 5-shot
1	**66.3**	**75.3**	**46.6**	**58.2**
2	65.0	73.9	45.1	56.8
3	65.1	73.6	45.4	56.5

(c)

Table 2: Ablation studies on the effectiveness of RCT and RAT in (a), multi-level context and affinity utilization in (b), and the head number of attention in (c). Best results are shown in bold.

5 Experiments

In this section, we conduct extensive experiments for our CATrans on two widely-used few-shot segmentation benchmarks, PASCAL-5^i and COCO-20^i, to demonstrate the effectiveness of our method.

5.1 Datasets

PASCAL-5^i [Shaban *et al.*, 2017] is composed of PASCAL VOC 2012 with additional SBD [Hariharan *et al.*, 2011] annotations, which contains 20 categories split into 4 folds (15/5 categories as base/novel classes).

COCO-20^i [Lin *et al.*, 2014] is created from MS COCO where the 80 object categories are divided into four splits (60/20 categories as base/novel classes).

5.2 Implementation Details

We conduct all experiments on 1 NVIDIA V100 GPU. The models are trained for 20k and 40k iterations on PASCAL-5^i and COCO-20^i, respectively, with AdamW as the optimizer. The initial learning rate is set to 5e-5 and decays at 10k iteration with a factor of 0.1. During training, we first resize the input images to 384×384 and 512×512 for PASCAL-5^i and COCO-20^i, respectively, and then perform the horizontal flip operation randomly. We simply use cross-entropy loss with a weight of 1 and 4 for background and foreground pixels, respectively. The BN layers of image encoder are frozen. For a fair comparision, we employ the widely-used ResNet-50, ResNet-101 and Swin-Transformer as the image encoder. The mask encoder includes four light-weight layers, each of which is composed of 3×3 convolution, BatchNorm and ReLU.

During evaluation, the results are averaged on the randomly sampled 5k and 20k episodes for each fold and 5 runs with different seeds. we report the mean-IoU (mIoU) under both 1-shot (given a single support example) and 5-shot (given five support examples).

5.3 Comparisons with the State-of-the-art

Results on PASCAL-5^i. As shown in Table 1, our CATrans outperforms the previous best DGPNet by +2.8/+1.8% and +2.4/+1.1% for 1-shot/5-shot mIoU using ResNet-50 and ResNet-101 as the backbone, repectively. Rendering the Swin-T as the image encoder, our CATrans further achieves 67.6% and 77.3% mIoU for 1-shot and 5-shot segmentation.

Results on COCO-20^i. Table 3 reports the comparisons on the more challenging COCO-20^i dataset. Compared with the previous best DGPNet, our CATrans surpasses it by +1.6/+2.0% 1-shot/5-shot mIoU using the ResNet-50. When

Method		Backbone	1-shot					5-shot				
			20^0	20^1	20^2	20^3	Mean	20^0	20^1	20^2	20^3	Mean
PANet [Wang *et al.*, 2019]	ICCV19	ResNet-50	31.5	22.6	21.5	16.2	23.0	45.9	29.2	30.6	29.6	33.8
RPMMs [Yang *et al.*, 2020]	ECCV20		29.5	36.8	29.0	27.0	30.6	33.8	42.0	33.0	33.3	35.5
PPNet [Liu *et al.*, 2020]	ECCV20		34.5	25.4	24.3	18.6	25.7	48.3	30.9	35.7	30.2	36.2
CWT [Lu *et al.*, 2021]	ICCV21		32.2	36.0	31.6	31.6	32.9	40.1	43.8	39.0	42.4	41.3
CyCTR [Zhang *et al.*, 2021]	NeurIPS21		38.9	43.0	39.6	39.8	40.3	41.1	48.9	45.2	47.0	45.6
DGPNet [Johnander *et al.*, 2021]	arXiv21		43.6	47.8	44.5	44.2	45.0	54.7	59.1	56.8	54.4	56.2
CATrans	Ours		**46.5**	**49.3**	**45.6**	**45.1**	**46.6**	**56.3**	**60.7**	**59.2**	**56.3**	**58.2**
FWB [Nguyen and Todorovic, 2019]	ICCV19	ResNet-101	19.9	18.0	21.0	28.9	21.2	19.1	21.5	23.9	30.1	23.7
PFENet [Tian *et al.*, 2020]	TPAMI20		34.3	33.0	32.3	30.1	32.4	38.5	38.6	38.2	34.3	37.4
CWT [Lu *et al.*, 2021]	ICCV21		30.3	36.6	30.5	32.2	32.4	38.5	46.7	39.4	43.2	42.0
DGPNet [Johnander *et al.*, 2021]	arXiv21		45.1	49.5	46.6	45.6	46.7	56.8	60.4	58.4	55.9	57.9
CATrans	Ours		**47.2**	**51.7**	**48.6**	**47.8**	**48.8**	**58.5**	**63.4**	**59.6**	**57.2**	**59.7**
CATrans	Ours	Swin-T	**47.9**	**52.3**	**49.2**	**48.0**	**49.4**	**59.3**	**64.1**	**59.6**	**57.3**	**60.1**

Table 3: Comparison with the state-of-the-art in 1-shot and 5-shot segmentation on COCO-20^i dataset using the mIoU (%) as evaluation metric. Best results in bold.

using ResNet-101, our CATrans is +2.1% and +1.8% higher than DGPNet for 1-shot and 5-shot protocals, repectively. Equipped with the Swin-T, our CATrans achieves 49.4/60.1% 1-shot/5-shot mIoU.

5.4 Ablation Study

Here, we conduct extensive ablation studies with ResNet-50 on PASCAL-5^i and COCO-20^i to analyze the effect of the key components in our CATrans.

Effectiveness of RAT and RCT. We ablate our CATrans to observe the effectiveness of RAT and RCT modules, as shown in Table 2a. We define the baseline, without RAT and RCT modules, that is simply concatenating the support and query features along the channel mode. The variant with either RAT or RCT boosts baseline by up to +6.8/+7.7 on 1-shot protocol. Moreover, it can be seen that the support self-context branch of RCT provides additional $\sim$1% mIoU on 1-shot setting (RCT *vs*. RCT$^\circ$), endowing RCT with a powerful context aggregator. Considering that the cross-affinity between support and query images can serve as the additional guidance for FSS task, when the large feature distinctions impede context knowledge transfer, we verify how much benefit comes from the RAT. Table 2a shows the model equips with both RCT and RAT has $\sim$5% increase for FSS results.

Multi-scale Representations. We first verify the influence of fusing the high-resolution features of query image ($l=1/2$) on the top panel of Table 2b. Then on the bottom panel of Table 2b, it shows a comparison experiment between single-scale and multi-scale representations of CATrans ($l = 3/4$). The more levels of information used, the better the performance. The performance reaches the highest when these two levels are both leveraged, especially in the 1-shot setting where multi-scale guidance can extract more guiding information for query segmentation with 1.4%/1.8% mIoU gain on PASCAL-5^i/COCO-20^i.

Effect of Model Capacity. We stack more numbers of heads of attention layer to increase capacity of our CATrans and validate its effectiveness. It shows that our model performance is stable across different choices, particularly for $\#h \geq 2$.

Memory and Run-time. CATrans, with trivial computational overhead, performs the best over closely related transformer-based methods. CyCTR and CATs stack two successive transformer blocks whereas CATrans consists of RCT and RAT, each with one transformer block. The memory and run-time comparisons are below:

Memory (GB)			Run-time (ms)		
CyCTR	CATrans	CATs	CyCTR	CATrans	CATs
1.78	1.85	1.90	31.7	33.2	34.5

Qualitative results. To show the performance of CATrans intuitively, we visualize some final prediction masks produced by our method and its variants in Figure 3. The first column is support image and its ground truth, and the next three columns are query mask produced by RAT, RCT and CATrans, respectively. The last column is ground truth of query image. On the top row of Figure 3, as the large intra-class appearance variations between support and query images hidden context knowledge transferring, RCT fails to precisely segment one of its aerofoils, where RCT performs better by use of the self- and cross- affinity for query segmentation. On the bottom of Figure 3, the RAT mistakenly regards some plates as part of the target cup because of the similarity in the shape and color, measured by the attention-aware affinity. But RAT reduces the area of wrong segmentation significantly by successfully aggregating the context information. Overall, adopting both RAT and RCT performs the best.

6 Conclusion

We have proposed the novel Context and Affinity Transformer (CATrans) with RCT and RAT in a hierarchical architecture to deal with the large intra-class appearance and geometric variations in few-shot segmentation, so that CATrans can effectively incorporate both context similarity and affinity map for query segmentation. In addition, we consider pixel-wise correspondences for individual support and query features to disambiguate noisy correlations.

References

[Carion *et al.*, 2020] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV 2020*, volume 12346 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2020.

[Cho *et al.*, 2021] Seokju Cho, Sunghwan Hong, Sangryul Jeon, Yunsung Lee, Kwanghoon Sohn, and Seungryong Kim. Semantic correspondence with transformers. *CoRR*, abs/2106.02520, 2021.

[Dong and Xing, 2018] Nanqing Dong and Eric P. Xing. Few-shot semantic segmentation with prototype learning. In *BMVC 2018*, page 79. BMVA Press, 2018.

[Dosovitskiy *et al.*, 2021] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR 2021*. OpenReview.net, 2021.

[Hariharan *et al.*, 2011] Bharath Hariharan, Pablo Arbelaez, Lubomir D. Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV 2011*, pages 991–998. IEEE Computer Society, 2011.

[Johnander *et al.*, 2021] Joakim Johnander, Johan Edstedt, Michael Felsberg, Fahad Shahbaz Khan, and Martin Danelljan. Dense gaussian processes for few-shot segmentation. *CoRR*, abs/2110.03674, 2021.

[Lin *et al.*, 2014] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *ECCV 2014*, volume 8693 of *Lecture Notes in Computer Science*, pages 740–755. Springer, 2014.

[Lin *et al.*, 2017] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian D. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *CVPR 2017*, pages 5168–5177. IEEE Computer Society, 2017.

[Liu *et al.*, 2020] Yongfei Liu, Xiangyi Zhang, Songyang Zhang, and Xuming He. Part-aware prototype network for few-shot semantic segmentation. In *ECCV 2020*, volume 12354 of *Lecture Notes in Computer Science*, pages 142–158. Springer, 2020.

[Long *et al.*, 2015] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR 2015*, pages 3431–3440. IEEE Computer Society, 2015.

[Lu *et al.*, 2021] Zhihe Lu, Sen He, Xiatian Zhu, Li Zhang, Yi-Zhe Song, and Tao Xiang. Simpler is better: Few-shot semantic segmentation with classifier weight transformer. *CoRR*, abs/2108.03032, 2021.

[Nguyen and Todorovic, 2019] Khoi Nguyen and Sinisa Todorovic. Feature weighting and boosting for few-shot segmentation. In *ICCV 2019*, pages 622–631. IEEE, 2019.

[Shaban *et al.*, 2017] Amirreza Shaban, Shray Bansal, Zhen Liu, Irfan Essa, and Byron Boots. One-shot learning for semantic segmentation. In *BMVC 2017*. BMVA Press, 2017.

[Tian *et al.*, 2020] Zhuotao Tian, Hengshuang Zhao, Michelle Shu, Zhicheng Yang, Ruiyu Li, and Jiaya Jia. Prior guided feature enrichment network for few-shot segmentation. *CoRR*, abs/2008.01449, 2020.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems 30: December 4-9, 2017*, pages 5998–6008, 2017.

[Wang *et al.*, 2019] Kaixin Wang, Jun Hao Liew, Yingtian Zou, Daquan Zhou, and Jiashi Feng. Panet: Few-shot image semantic segmentation with prototype alignment. In *ICCV 2019*, pages 9196–9205. IEEE, 2019.

[Wang *et al.*, 2020] Haochen Wang, Xudong Zhang, Yutao Hu, Yandan Yang, Xianbin Cao, and Xiantong Zhen. Few-shot semantic segmentation with democratic attention networks. In *ECCV 2020*, volume 12358 of *Lecture Notes in Computer Science*, pages 730–746. Springer, 2020.

[Yang *et al.*, 2020] Boyu Yang, Chang Liu, Bohao Li, Jianbin Jiao, and Qixiang Ye. Prototype mixture models for few-shot semantic segmentation. In *ECCV 2020*, volume 12353 of *Lecture Notes in Computer Science*, pages 763–778. Springer, 2020.

[Zhang *et al.*, 2019a] Chi Zhang, Guosheng Lin, Fayao Liu, Jiushuang Guo, Qingyao Wu, and Rui Yao. Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In *ICCV 2019*, pages 9586–9594. IEEE, 2019.

[Zhang *et al.*, 2019b] Chi Zhang, Guosheng Lin, Fayao Liu, Rui Yao, and Chunhua Shen. Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In *CVPR 2019*, pages 5217–5226. Computer Vision Foundation / IEEE, 2019.

[Zhang *et al.*, 2020] Xiaolin Zhang, Yunchao Wei, Yi Yang, and Thomas S. Huang. Sg-one: Similarity guidance network for one-shot semantic segmentation. *IEEE Trans. Cybern.*, 50(9):3855–3865, 2020.

[Zhang *et al.*, 2021] Gengwei Zhang, Guoliang Kang, Yunchao Wei, and Yi Yang. Few-shot segmentation via cycle-consistent transformer. *CoRR*, abs/2106.02320, 2021.

[Zhu *et al.*, 2021] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *ICLR 2021*. OpenReview.net, 2021.