Cost Aggregation with 4D Convolutional Swin Transformer for Few-Shot Segmentation

Sunghwan Hong[1,⋆], Seokju Cho[1,⋆], Jisu Nam[1], Stephen Lin[2], and Seungryong Kim[1]

[1] Korea University, Seoul, Korea
{sung_hwan,seokju_cho,18wltnzzang,seungryong_kim}@korea.ac.kr
[2] Microsoft Research Asia, Beijing, China
stevelin@microsoft.com

Abstract. This paper presents a novel cost aggregation network, called Volumetric Aggregation with Transformers (VAT), for few-shot segmentation. The use of transformers can benefit correlation map aggregation through self-attention over a global receptive field. However, the tokenization of a correlation map for transformer processing can be detrimental, because the discontinuity at token boundaries reduces the local context available near the token edges and decreases inductive bias. To address this problem, we propose a 4D Convolutional Swin Transformer, where a high-dimensional Swin Transformer is preceded by a series of small-kernel convolutions that impart local context to all pixels and introduce convolutional inductive bias. We additionally boost aggregation performance by applying transformers within a pyramidal structure, where aggregation at a coarser level guides aggregation at a finer level. Noise in the transformer output is then filtered in the subsequent decoder with the help of the query's appearance embedding. With this model, a new state-of-the-art is set for all the standard benchmarks in few-shot segmentation. It is shown that VAT attains state-of-the-art performance for semantic correspondence as well, where cost aggregation also plays a central role. Code and trained models are available at https://seokju-cho.github.io/VAT/.

1 Introduction

Semantic segmentation is a fundamental computer vision task that aims to label each pixel in an image with its corresponding class. Substantial progress has been made in this direction with the help of deep neural networks and large-scale datasets containing ground-truth segmentation annotations [37, 46, 3, 4, 60]. Manual labeling of pixel-wise segmentation maps, however, requires considerable labor, making it difficult to add new classes. Towards reducing reliance on labeled data, attention has increasingly focused on *few*-shot segmentation [48, 54], where only a handful of support images and their associated masks are used in predicting the segmentation of a query image.

⋆ Equal contribution

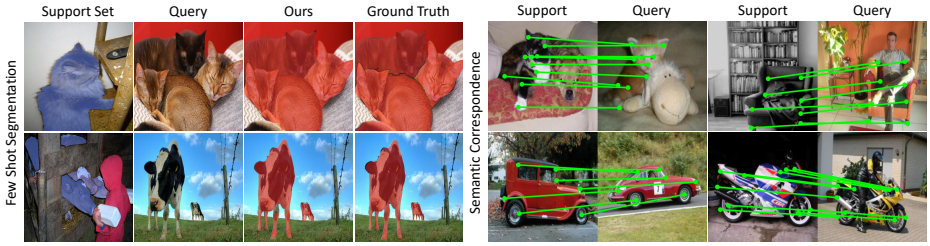

Fig. 1: **Our VAT reformulates few-shot segmentation as semantic correspondence.** VAT sets a new state-of-the-art in few-shot segmentation, and attains state-of-the-art performance for semantic correspondence as well.

The key to few-shot segmentation is in making effective use of the few support samples. Many works attempt this by extracting a prototype model from the samples and using it for feature comparison with the query [57, 10, 35, 76]. However, such approaches disregard pixel-level pairwise relationships between support and query features or the spatial structure of features, which may lead to sub-optimal results.

To account for such relationships, we observe that few-shot segmentation can be reformulated as semantic correspondence, which aims to find pixel-level correspondences across semantically similar images which may contain large intra-class appearance and geometric variations [13, 14, 43]. Recent semantic correspondence models [49, 25, 50, 52, 42, 44, 34, 64, 41] follow the classical matching pipeline [53, 47] of feature extraction, cost aggregation and flow estimation. The cost aggregation stage, where matching scores are refined to produce more reliable correspondence estimates, is of particular importance and has been the focus of much research [52, 42, 51, 22, 34, 29, 41, 6]. Recently, CATs [6] proposed to use vision transformers [11] for cost aggregation, but its quadratic complexity to the number of input tokens limits its applicability. It also disregards the spatial structure of matching costs, which may hurt its performance.

In the area of few-shot segmentation, there also exist methods that attempt to leverage pairwise information by refining features through cross-attention [81] or graph attention [79, 67, 73]. However, they solely rely on raw correlation maps without aggregating the matching scores. As a result, their correspondence may suffer from ambiguities caused by repetitive patterns or background clutters [49, 25, 27, 64, 17]. To address this, HSNet [40] aggregates the matching scores with 4D convolutions, but its limited receptive fields prevent long-range context aggregation and it lacks an ability to adapt to the input content due to the use of fixed kernels.

In this paper, we introduce a novel cost aggregation network, called Volumetric Aggregation with Transformers (VAT), that tackles the few-shot segmentation task through a proposed 4D Convolutional Swin Transformer. Specifically, we first extend Swin Transformer [36] and its patch embedding module to handle a high-dimensional correlation map. The patch embedding module is further extended by incorporating 4D convolutions that alleviate issues caused by patch

embedding, *i.e.,* limited local context near patch boundaries and low inductive bias. The high-dimensional patch embedding module is designed as a series of overlapping small-kernel convolutions, bringing local contextual information to each pixel and imparting convolutional inductive bias. To further boost performance, we compose our architecture with a pyramidal structure that takes the aggregated correlation maps at a coarser level as additional input at a finer level, providing hierarchical guidance. Our affinity-aware decoder then refines the aggregated matching scores in a manner that exploits the higher-resolution spatial structure given by the query's appearance embedding and finally outputs the segmentation mask prediction.

We demonstrate the effectiveness of our method on several benchmarks [54, 31, 30]. Our work attains state-of-the-art performance on all the benchmarks for few-shot segmentation and even for semantic correspondence, highlighting the importance of cost aggregation for both tasks and showing its potential for general matching. We also include ablation studies to justify our design choices.

2 Related Work

Few-shot Segmentation. Inspired by the few-shot learning paradigm [48, 57], which learns to learn a model for a novel task with only a limited number of samples, few-shot segmentation has received considerable attention. Following the success of [54], prototypical networks [57] and numerous other works [10, 45, 55, 68, 35, 76, 33, 74, 77, 59, 82, 28] proposed to extract a prototype from support samples, which is used to identify foreground features in the query. In addition, inspired by [80] which observed that simply adding high-level features in feature processing leads to a performance drop, [62] proposed to instead utilize high-level features to compute a prior map that helps to identify targets in the query image. Many variants [59, 78] extended this idea of utilizing prior maps to act as additional information for aggregating feature maps.

However, as methods based on prototypes or prior maps have apparent limitations, *e.g.*, disregarding pairwise relationships between support and query features or spatial structure of feature maps, numerous recent works [79, 67, 40, 73, 32] utilize a correlation map to leverage the pairwise relationships between source and query features. Specifically, [79, 67, 73] use graph attention, HSNet [40] proposes 4D convolutions to exploit multi-level features, and [32] formulates the task as an optimal transport problem. However, these approaches do not provide a means to aggregate the matching scores, solely utilize convolutions for cost aggregation, or use a handcrafted method that is neither learnable nor robust to severe deformations.

Recently, [81] utilized transformers and proposed to use a cycle-consistent attention mechanism to refine the feature maps to become more discriminative, without considering aggregation of matching scores. [59] propose a global and local enhancement module to refine the features using transformers and convolutions, respectively. [39] focuses solely on the transformer-based classifier by

freezing the encoder and decoder. Unlike these works, we propose a 4D Convolutional Swin Transformer for an enhanced and efficient cost aggregation.

Semantic Correspondence. The objective of semantic correspondence is to find correspondences between semantically similar images with additional challenges posed by large intra-class appearance and geometric variations [34, 6, 41]. This is highly similar to the few-shot segmentation setting in that few-shot segmentation also aims to label objects of the same class with large intra-class variation, and thus recent works on both tasks have taken similar approaches. The latest methods [52, 42, 51, 22, 34, 29, 41, 6] in semantic correspondence focus on the cost aggregation stage to find reliable correspondences and demonstrated its importance. Among them, [41] proposed to use 4D convolutions for cost aggregation, though exhibiting apparent limitations due to the limited receptive fields of convolutions and lack of adaptability. CATs [6] resolves this issue and sets a new state-of-the-art by leveraging transformers [65] to aggregate the cost volume. However, it disregards the spatial structure of correlation maps and imparts less inductive bias, *i.e.,* translation equivariance, which limits its generalization power [36, 7, 8]. Moreover, its quadratic complexity may limit applicability when it is used to aggregate correlation maps on its own. In this paper, we propose to resolve the aforementioned issues.

Vision Transformer. Recently, transformer [65], the standard architecture in Natural Language Processing (NLP), has been widely adopted in Computer Vision. Since the pioneering work on ViT [11], numerous works [39, 81, 59, 23, 71, 6, 36] have adopted transformers to replace CNNs or to be used together with CNNs in a hybrid manner. However, due to quadratic complexity to sequence length, transformers often suffer from large a computational burden. Efficient transformers [69, 24, 75, 70] aim to reduce the computational load via an approximated or simplified self-attention. Swin Transformer [36], a network we extend from, reduces computation by performing self-attention within pre-defined local windows. However, these works inherit the issues caused by patch embedding, which we alleviate by incorporating 4D convolutions.

3 Methodology

3.1 Problem Formulation

The goal of *few*-shot segmentation is to segment objects from unseen classes in a query image given only a few annotated examples [66]. To mitigate the overfitting caused by insufficient training data, we follow the common protocol of *episodic* training [66]. Let us denote the training and test sets as $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$, respectively, where the object classes of both sets do not overlap. Under the K-shot setting, multiple *episodes* are formed from both sets, each consisting of a support set $\mathcal{S} = \{(x_s^k, m_s^k)\}_{k=1}^{\text{K}}$, where (x_s^k, m_s^k) is k-th support image and its corresponding mask pair, and a query sample (x_q, m_q), where x_q is a query image and m_q is its paired mask. During training, our model takes a sampled

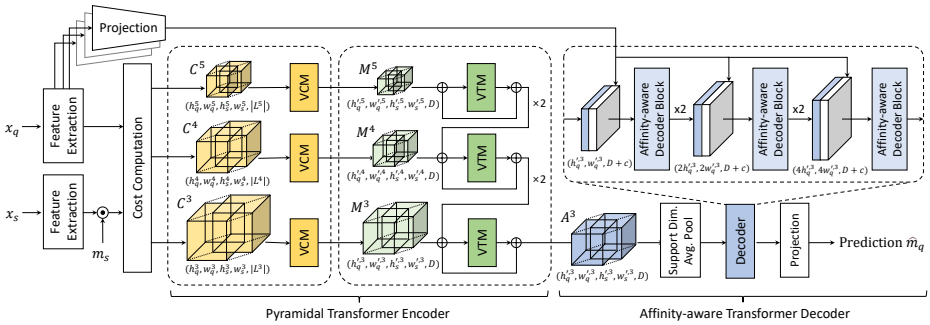

Fig. 2: **Overall network architecture.** Our network consists of feature extraction and cost computation, a pyramidal transformer encoder, and an affinity-aware transformer decoder.

episode from $\mathcal{D}_{\text{train}}$ and learns a mapping from $\mathcal{S}$ and x_q to a prediction m_q. At inference, our model predicts $\hat{m}_q$ given randomly sampled $\mathcal{S}$ and x_q from $\mathcal{D}_{\text{test}}$.

3.2 Motivation and Overview

The key to few-shot segmentation is how to effectively utilize the support samples provided for a query image. While conventional methods [62, 59, 81, 77, 28] utilize global- or part-level prototypes extracted from support features, recent methods [79, 67, 40, 73, 32, 81] instead leverage pairwise matching relationships between query and support. However, exploring such relationships is notoriously challenging due to intra-class variations, background clutters, and repetitive patterns. One of the state-of-the-art methods, HSNet [40], aggregates the matching scores with 4D convolutions. However, solely utilizing convolutions may limit performance due to limited receptive fields or lack of adaptability for convolutional kernels. While there has been no approach to aggregate the matching scores with transformers in few-shot segmentation, CATs [6] proposes cost aggregation with transformers in semantic correspondence, demonstrating the effectiveness of transformers as a cost aggregator. On the other hand, the quadratic complexity of transformers with respect to the number of tokens may limit its utility for segmentation. The absence of operations that impart inductive bias, *i.e.,* translation equivariance, may limit its performance as well. Also, CATs [6] defines the tokens of a correlation map in a way that disregards spatial structure, which is likely to be harmful.

The proposed Volumetric Aggregation with Transformers (VAT) is designed to overcome these problems. In the following, we first describe its feature extraction and cost computation. We then present a general extension of Swin Transformer [36] for cost aggregation. Subsequently, we present 4D Convolutional Swin Transformer for resolving the aforementioned issues. Lastly, we introduce several additional techniques including Guided Pyramidal Processing (GPP) and Affinity-aware Transformer Decoder (ATD) to further boost performance, and combine them to complete the design.

3.3 Feature Extraction and Cost Computation

We extract features from query and support images and compute an initial cost between them following the conventional process [49, 58, 52, 51, 64, 17, 6]. Given query and support images, x_q and x_s, we use a CNN [16, 56] to produce a sequence of L feature maps, $\{(F_q^l, F_s^l)\}_{l=1}^L$, where F_q^l and F_s^l denote query and support feature maps at the l-th level. A support mask, m_s, is used to encode segmentation information and filter out the background information as done in [28, 40, 78]. We obtain a masked support feature as $\hat{F}_s^l = F_s^l \odot \psi^l(m_s)$, where $\odot$ denotes the Hadamard product and $\psi^l(\cdot)$ denotes a function that resizes the given tensor followed by expansion along the channel dimension of the l-th layer.

Given a pair of feature maps, F_q^l and F_s^l, we compute a correlation map using the inner product between l-2 normalized features such that

$$\mathcal{C}^l(i,j) = \mathrm{ReLU}\left(\frac{F_q^l(i) \cdot \hat{F}_s^l(j)}{\|F_q^l(i)\|\|\hat{F}_s^l(j)\|}\right), \tag{1}$$

where i and j denote 2D spatial positions of feature maps. As done in [40], we collect correlation maps computed from all the intermediate features of the same spatial size and stack them to obtain a stacked correlation map $\mathcal{C}^p \in \mathbb{R}^{h_q \times w_q \times h_s \times w_s \times |\mathcal{L}^p|}$, where (h_q, w_q) and (h_s, w_s) are the height and width of the query and support feature maps, respectively, and $\mathcal{L}^p$ is a subset of CNN layer indices $\{1, ..., L\}$ at pyramid layer p, containing correlation maps of identical spatial size.

3.4 Pyramidal Transformer Encoder

In this section, we present 4D Convolutional Swin Transformer for aggregating the correlation maps and then incorporate it into a pyramidal architecture.

Cost Aggregation with Transformers. For a transformer to process a correlation map, a means for token reduction is essential, since it would be infeasible for even an efficient transformer [69, 24, 75, 70, 36] to handle a correlation map otherwise. However, when one employs a transformer for cost aggregation, the problem of how to define the tokens for correlation maps, which differ in shape from images, text or features [65, 11], is non-trivial. The first attempt to process correlation maps is CATs [6], which reshapes the 4D correlation maps into 2D maps and performs self-attention in 2D. This disregards the spatial structure of correlation maps, *i.e.,* over both support and query, which could limit its performance. To address this, one may treat all the spatial entries, *e.g.,* $h_q \times w_q \times h_s \times w_s$, as tokens and treat $\mathcal{L}^p$ as the feature dimension for tokens. However, this results in a substantial computational burden that increases with larger correlation maps. This prevents the use of standard transformers [65, 11] and encourages use of efficient versions as in [69, 24, 75, 70, 36]. However, the use of simplified (or approximated) self-attention may be sub-optimal for performance, as will be discussed in Section 4.4. Furthermore, as proven in the

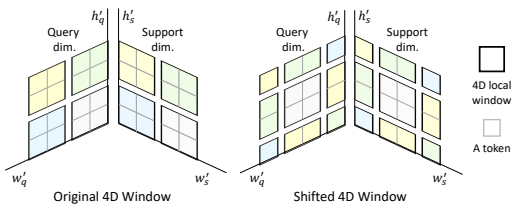

Fig. 3: **Illustration of shifted 4D windows in VTM.** It computes self-attention within the partitioned windows and considers inter-window interactions by shifting the windows.

optical flow and semantic correspondence literature [58, 51], neighboring pixels tend to have similar correspondences. To preserve the spatial structure of correlation maps, we choose to use Swin Transformer [36] as it not only provides efficient self-attention computation, but also maintains the smoothness property of correlation maps while still providing sufficient long-range self-attention.

To employ Swin Transformer [36] for cost aggregation, we need to extend it to process higher dimensional input, specifically a 4D correlation map. We first follow the conventional patch embedding procedure [11] to embed correlation maps, as they cannot be processed by transformers due to the large number of tokens. However, we extend the patch embedding module to a Volumetric Embedding Module (VEM) which handles higher dimensional inputs, such that $\mathcal{M}^p = \text{VEM}(\mathcal{C}^p)$. Following a procedure similar to patch embedding, we reshape the correlation map to a sequence of flattened 4D windows using a large convolutional kernel, *e.g.,* 16×16×16×16. Then, we extend the self-attention computations, as shown in Fig. 3, by evenly partitioning the query and support spatial dimensions of $\mathcal{M}^p$ into non-overlapping sub-correlation maps $\mathcal{M}'^p \in \mathbb{R}^{n\times n\times n\times n\times D}$. We compute self-attention within each partitioned sub-correlation map. Subsequently, we shift the windows by a displacement of $\left(\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor\right)$ pixels from the previously partitioned windows, then perform self-attention within the newly created windows. Then as done in the original Swin Transformer [36], we simply roll the correlation map back to its original form. In computing self-attention, we use relative position bias and take the values from an expanded parameterized bias matrix, following [19, 20, 36]. We leave the other components of Swin Transformer blocks unchanged, e.g., Layer Normalization (LN) [1] and MLP layers. We call this extension the Volumetric Transformer Module (VTM). To summarize, the overall process is defined as:

$$\mathcal{A}^p = \text{VTM}(\mathcal{M}^p). \tag{2}$$

4D Convolutional Swin Transformer. Although the proposed cost aggregation with transformers can solve the aforementioned issues of using CNNs and the high computational burden of using standard transformers, it may not avoid the issue that other transformers share [11, 69, 24, 75, 70]: lack of translation equivariance. This is primarily caused by utilizing non-overlapping operations prior to self-attention computation. Although Swin Transformer alleviates the issue to some extent by using relative positioning bias [36], it provides an insufficient approximation. We argue that the Volumetric Embedding Module is what needs to be addressed as it leads to several issues. First, the use of large

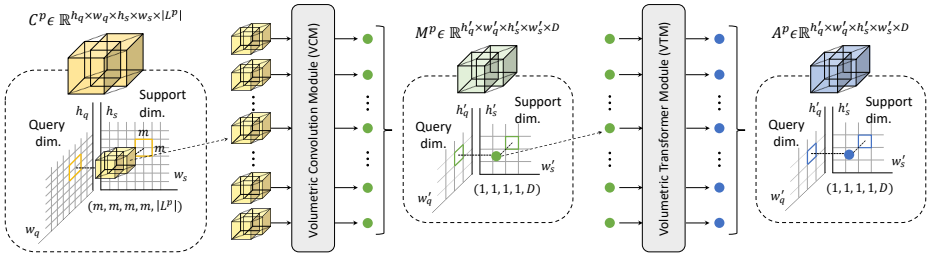

Fig. 4: **Overview of 4D Convolutional Swin Transformer.** We replace the VEM with VCM and the output undergoes VTM for cost aggregation.

non-overlapping convolution kernels only provides limited inductive bias. Relatively lower translation equivariance is achieved from non-overlapping operations compared to that which are overlapping. This limited inductive bias results in relatively lower generalization power and performance [72, 8, 7, 36]. Furthermore, we argue that for dense prediction tasks, disregarding window boundaries due to non-overlapping kernels hurts overall performance due to discontinuity.

To address the above issues, we replace the Volumetric Embedding Module (VEM) with a module consisting of a series of overlapping convolutions, which we call the Volumetric Convolution Module (VCM). Concretely, we sequentially reduce spatial dimensions of the support and query by applying 4D spatial max-pooling, overlapping 4D convolutions, ReLU, and Group Normalization (GN), where we project the multi-level similarity vector at each 4D position, i.e., projecting a vector size of $|\mathcal{L}^p|$, to an arbitrary fixed dimension denoted as D. Considering receptive fields as a 4D window, i.e., $m \times m \times m \times m$, we obtain a tensor $\mathcal{C}^p \in \mathbb{R}^{h_q'^{,p} \times w_q'^{,p} \times h_s'^{,p} \times w_s'^{,p} \times D}$ from $\mathcal{C}^p$, where $h_s'^{,p}$, $w_s'^{,p}$, $h_q'^{,p}$, and $w_q'^{,p}$ are the processed sizes. Note that a different size of m can be chosen for the support and query spatial dimensions. An overview of VCM is illustrated in Fig. 4. Overall, we define such a process as the following:

$$\mathcal{M}^p = \text{VCM}(\mathcal{C}^p). \tag{3}$$

In this way, our model benefits from additional inductive bias as well as better handling at window boundaries.

Moreover, to stabilize the learning, we propose an additional technique to enforce the networks to estimate residual matching scores as complementary details. We add residual connections in order to expedite the learning process [16, 6, 83], accounting for the fact that at the initial phase when the input $\mathcal{M}^p$ is fed, erroneous matching scores are inferred due to randomly-initialized parameters of transformers, which could complicate the learning process as the networks need to learn the complete matching details from random matching scores.

Guided Pyramidal Processing. Following [40, 59], we also employ a coarse-to-fine approach through pyramidal processing as illustrated in Fig. 2. Motivated by numerous recent works [81, 41, 6, 40] in both semantic matching and few-shot

segmentation which have demonstrated that leveraging multi-level features can boost performance by a large margin, we also use a pyramidal architecture.

In our coarse-to-fine approach, which we refer to as Guided Pyramidal Processing (GPP), the aggregation of a finer-level correlation map $\mathcal{A}^p$ is guided by the aggregated correlation map of the previous (coarser) level $\mathcal{A}^{p+1}$. Concretely, an aggregated correlation map $\mathcal{A}^{p+1}$ is up-sampled into a map $\text{up}(\mathcal{A}^{p+1})$ which is added to the next level's correlation map $\mathcal{A}^p$ to serve as guidance. This process is repeated until the finest-level aggregated map is computed and passed to the decoder. As shown in Table 4, GPP leads to appreciable performance gains.

With GPP, the pyramidal transformer encoder is finally defined as:

$$\mathcal{A}^p = \text{VTM}(\text{VCM}(\mathcal{C}^p) + \text{up}(\mathcal{A}^{p+1})), \tag{4}$$

where $\text{up}(\cdot)$ denotes bilinear upsampling.

3.5 Affinity-Aware Transformer Decoder

Given the aggregated correlation map produced by the pyramidal transformer encoder, a transformer-based decoder generates the final segmentation mask. To improve performance, we propose to conduct further aggregation within the decoder with the aid of the appearance embedding obtained from query feature maps. The query's appearance embedding can help in two ways. First, appearance affinity information is an effective guide for filtering noise in matching scores, as proven in the stereo matching literature, e.g., Cost Volume Filtering (CVF) [18, 58]. In addition, the higher-resolution spatial structure provided by an appearance embedding can be exploited to improve up-sampling quality, resulting in a highly accurate prediction mask $\hat{m}^q$ where fine details are preserved.

For the design of our Affinity-aware Transformer Decoder (ATD), we take the average over the support image dimensions of $\mathcal{A}^p$, concatenate it with the appearance embedding from query feature maps, and then aggregate by transformers [65, 69, 70, 36] with subsequent bilinear interpolation. The process is defined as the following:

$$\hat{m}_q = \text{ATD}([\mathcal{A}'^{,p}, \mathcal{P}(F_q)]), \tag{5}$$

where $\mathcal{A}'^{,p} \in \mathbb{R}^{h'^{,p}_q \times w'^{,p}_q \times D}$ is extracted by average pooling on $\mathcal{A}^p$ over the spatial dimensions of the support image, $\mathcal{P}(\cdot)$ is a linear projection, $\mathcal{P}(F_q) \in \mathbb{R}^{h'^{,p}_q \times w'^{,p}_q \times c}$, and $[\,\cdot,\cdot\,]$ denotes concatenation. We sequentially refine the output immediately after bilinear upsampling to recapture fine details and integrate appearance information.

3.6 Extension to *K*-Shot Setting

Given K pairs of support image and mask $\{(x^i_s, m^i_s)\}^K_{i=1}$ and a query image x_q, our model forward-passes K times to obtain K different query masks $\hat{m}^k_q$. We sum up all the K predictions at each spatial location, and if the sum divided by K exceeds a threshold τ, the location is predicted as foreground, and otherwise it is background.

Backbone network	Methods	1-shot							5-shot							# learnable params
		5^0	5^1	5^2	5^3	mIoU	FB-IoU	mBA	5^0	5^1	5^2	5^3	mIoU	FB-IoU	mBA	
ResNet50 [16]	PANet [68]	44.0	57.5	50.8	44.0	49.1	-	-	55.3	67.2	61.3	53.2	59.3	-	-	23.5M
	PFENet [62]	61.7	69.5	55.4	56.3	60.8	73.3	-	63.1	70.7	55.8	57.9	61.9	73.9	-	10.8M
	ASGNet [28]	58.8	67.9	56.8	53.7	59.3	69.2	-	63.4	70.6	64.2	57.4	63.9	74.2	-	10.4M
	CWT [39]	56.3	62.0	59.9	47.2	56.4	-	-	61.3	68.5	68.5	56.6	63.7	-	-	-
	RePRI [2]	59.8	68.3	62.1	48.5	59.7	-	49.0	64.6	71.4	**71.1**	59.3	66.6	-	43.8	-
	HSNet [40]	64.3	70.7	60.3	**60.5**	64.0	76.7	53.9	70.3	73.2	67.4	**67.1**	69.5	80.6	54.5	**2.6M**
	CyCTR [81]	65.7	71.0	59.5	59.7	64.0	-	-	69.3	73.5	63.8	63.5	67.5	-	-	-
	VAT (ours)	**67.6**	**72.0**	**62.3**	60.1	**65.5**	**77.8**	**54.4**	**72.4**	**73.6**	68.6	65.7	**70.1**	**80.9**	**54.8**	3.2M
ResNet101 [16]	FWB [45]	51.3	64.5	56.7	52.2	56.2	-	-	54.8	67.4	62.2	55.3	59.9	-	-	43.0M
	DAN [67]	54.7	68.6	57.8	51.6	58.2	71.9	-	57.9	69.0	60.1	54.9	60.5	72.3	-	-
	PFENet [62]	60.5	69.4	54.4	55.9	60.1	72.9	-	62.8	70.4	54.9	57.6	61.4	73.5	-	10.8M
	ASGNet [28]	59.8	67.4	55.6	54.4	59.3	71.7	-	64.6	71.3	64.2	57.3	64.4	75.2	-	10.4M
	CWT [39]	56.9	65.2	61.2	48.8	58.0	-	-	62.6	70.2	**68.8**	57.2	64.7	-	-	-
	RePRI [2]	59.6	68.6	62.2	47.2	59.4	-	45.1	66.2	71.4	67.0	57.7	65.6	-	42.0	-
	HSNet [40]	67.3	72.3	62.0	63.1	66.2	77.6	53.9	71.8	74.4	67.0	68.3	70.4	80.6	54.4	**2.6M**
	CyCTR [81]	67.2	71.1	57.6	59.0	63.7	73.0	-	71.0	75.0	58.5	65.0	67.4	75.4	-	-
	VAT (ours)	**70.0**	**72.5**	**64.8**	**64.2**	**67.9**	**79.6**	**54.7**	**75.0**	**75.2**	68.4	**69.5**	**72.0**	**83.2**	**54.8**	3.3M

Table 1: **Performance comparison on PASCAL-5^i [54].** Best results in bold, and second best are underlined.

Backbone feature	Methods	1-shot							5-shot						
		20^0	20^1	20^2	20^3	mean	FB-IoU	mBA	20^0	20^1	20^2	20^3	mean	FB-IoU	mBA
ResNet50 [16]	PMM [76]	29.3	34.8	27.1	27.3	29.6	-	-	33.0	40.6	30.3	33.3	34.3	-	-
	RPMM [76]	29.5	36.8	28.9	27.0	30.6	-	-	33.8	42.0	33.0	33.3	35.5	-	-
	PFENet [62]	36.5	38.6	34.5	33.8	35.8	-	-	36.5	43.3	37.8	38.4	39.0	-	-
	ASGNet [28]	-	-	-	-	34.6	60.4	-	-	-	-	-	42.5	67.0	-
	RePRI [2]	32.0	38.7	32.7	33.1	34.1	-	6.31	39.3	45.4	39.7	41.8	41.6	-	4.21
	HSNet [40]	36.3	43.1	38.7	38.7	39.2	68.2	53.0	43.3	**51.3**	48.2	45.0	46.9	70.7	53.8
	CyCTR [81]	38.9	43.0	39.6	**39.8**	40.3	-	-	41.1	48.9	45.2	**47.0**	45.6	-	-
	VAT (ours)	**39.0**	**43.8**	**42.6**	39.7	**41.3**	**68.8**	**54.2**	**44.1**	51.1	**50.2**	46.1	**47.9**	**72.4**	**54.9**

Table 2: **Performance comparison on COCO-20^i [31].**

4 Experiments

4.1 Implementation Details

We use ResNet50 and ResNet101 [16] pre-trained on ImageNet [9] and freeze the weights during training, following [40, 80]. No data augmentation is used for training, as explained in the supplementary material. We set the input image sizes to 417 or 473, following [28, 2]. The window size for Swin Transformer is set to 4. We use AdamW [38] with a learning rate of $5e-4$. Feature maps from conv3_x ($p=3$), conv4_x ($p=4$) and conv5_x ($p=5$) are taken for cost computation. The K-shot threshold τ is set to 0.5 and the embedding dimension D to 128. For appearance affinity, we take the last layers from conv2_x, conv3_x and conv4_x when training on FSS-1000 [30], and conv4_x is excluded when training on PASCAL-5^i [54] and COCO-20^i [31]. We set c to 16, 32, and 64 for conv2_x, conv3_x, and conv4_x.

4.2 Experimental Settings

Datasets. We evaluate our approach on three standard few-shot segmentation datasets, PASCAL-5^i [54], COCO-20^i [31], and FSS-1000 [30]. PASCAL-5^i contains images from PASCAL VOC 2012 [12] with added mask annotations [15].

Backbone feature	Methods	mIoU		FB-IoU		mBA	
		1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ResNet50 [16]	FSOT [32]	82.5	83.8	-	-	-	-
	HSNet [40]	85.5	87.8	91.0	92.5	62.1	63.3
	VAT	**90.1**	**90.7**	**93.8**	**94.2**	**68.3**	**68.4**
ResNet101 [16]	DAN [67]	85.2	88.1	-	-	-	-
	HSNet [40]	86.5	88.5	91.6	92.9	62.4	63.6
	VAT	**90.3**	**90.8**	**94.0**	**94.4**	**68.0**	**68.6**

Table 3: **Mean IoU comparison on FSS-1000 [30].**

It consists of 20 object classes, and as done in OSLSM [54], they are evenly divided into 4 folds $i \in \{0, 1, 2, 3\}$ for cross-validation, where each fold contains 5 classes. COCO-20^i contains 80 object classes, and as done for PASCAL-5^i, the dataset is evenly divided into 4 folds of 20 classes each. FSS-1000 is a more diverse dataset consisting of 1000 object classes. Following [30], we divide the 1000 categories into 3 splits for training, validation and testing, which consist of 520, 240 and 240 classes, respectively. For PASCAL-5^i and COCO-20^i, we follow the common evaluation practice [40, 62, 35] and standard cross-validation protocol, where each fold i is used for evaluation with the other folds used for training.

Evaluation Metric. Following common practice [80, 62, 40, 81], we adopt mean intersection over union (mIoU) and foreground-background IoU (FB-IoU) as our evaluation metrics. The mIoU averages over all IoU values for all object classes such that $\text{mIoU} = \frac{1}{C}\sum_{c=1}^{C} \text{IoU}_c$, where C is the number of classes in each fold, e.g., $C = 20$ for COCO-20^i. FB-IoU disregards the object classes and instead averages over foreground and background IoU (IoU_F and IoU_B) such that $\text{FB} - \text{IoU} = \frac{1}{2}(\text{IoU}_\text{F} + \text{IoU}_\text{B})$. We additionally adopt Mean Boundary Accuracy (mBA) introduced in [5] to evaluate the model's ability to capture fine details. To measure mBA, we first sample 5 radii in $[3, \frac{w+h}{300}]$ at a uniform interval, where w and h are width and height of input image, and average the segmentation accuracy within each radius from the ground-truth boundary.

4.3 Few-shot Segmentation Results

Table 1 summarizes quantitative results on PASCAL-5^i [54]. The tests were conducted on two backbone networks, ResNet50 and ResNet101 [16]. The proposed method outperforms the others on almost all the folds in terms of both mIoU and FB-IoU. It surpasses the others, including HSNet [40], in mBA as well, since our ATD helps to improve up-sampling quality by providing higher-level spatial structure for reference. Consistent with this, VAT also attains state-of-the-art performance on COCO-20^i [31], as shown in Table 2. Interestingly, for the most recent dataset specifically created for few-shot segmentation, FSS-1000 [30], VAT outperforms HSNet [40] and FSOT [32] by a large margin, almost a 4.6% increase in mIoU compared to HSNet with ResNet50 as shown in Table 3. VAT sets a new state-of-the-art for all of these benchmarks. We note that our method outperforms HSNet [40] despite having more learnable parameters, which is known

to have an inverse relation to generalization power [61], a trend seen in Table 1. With the proposed method, *i.e.,* 4D convolutional Swin Transformer, that is designed to address the issues like lack of inductive bias, VAT can have a larger number of learnable parameters than that of HSNet [40], yet VAT has greater generalization power as well.

4.4 Ablation Study

We conducted ablations on FSS-1000 [30], a large-scale dataset specifically constructed for few-shot segmentation.

Effectiveness of each component in VAT. As the baseline model, we take the architecture composed of VEM and the 2D convolution decoder used in HSNet [40]. We then progressively add our components one-by-one as shown in Table 4. Note that we included (**IV**) and (**V**) to show the effectiveness of VCM alone and the performance of a model highly similar to HSNet [40], respectively.

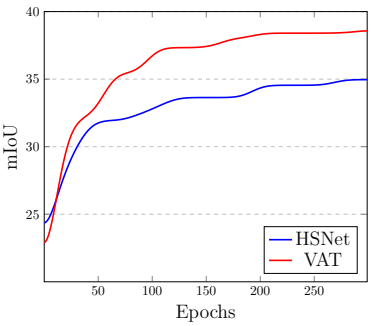

Fig. 5: **Convergence comparison.** Although VAT starts at a lower mIoU, it quickly exceeds HSNet [40].

As summarized in Table 4, each component helps to boost performance. Starting from the baseline (**I**), adding Swin Transformer (**II**) brings a large gain, which indicates that Swin Transformer effectively performs cost aggregation thanks to its approximated inductive bias and ability to consider spatial structure. When the VEM is replaced by VCM (**III**), we also observe a significant improvement, which confirms that the issues due to non-overlap are alleviated. We note that (**IV**) also highlights the importance of inductive bias. As (**V**) is approximately equivalent to HSNet [40], we first compare it with (**III**), which shows the superiority of the proposed 4D Convolutional Swin Transformer. By including the additional components in (**VI**) and (**VII**), the performance is further boosted. Moreover, we observe a large gain in mBA by adding ATD. This shows that the higher-resolution spatial structure provided by appearance embeddings help to refine the fine details. We additionally provide a visualization of convergence in comparison to HSNet [40] in Fig. 5. Thanks to the early convolutions [72], VAT quickly converges and exceeds HSNet [40] even though it starts at a lower mIOU.

Base architecture of VTM. As summarized in Table 5, we provide an ablation study to evaluate the effectiveness of different aggregators for VTM. For cost aggregation, there exists a few learnable aggregators, including MLP-, convolution- and transformer-based aggregators, any of which could be used as a base architecture for VTM. It should be noted that the use of standard transformer [65] and MLP-mixer [63] is not feasible due to memory requirements. Specifically, we calculated the memory consumption of each and found

	Components	FSS-1000 [30] mIoU (%) 1-shot	5-shot	mBA (%) 1-shot	5-shot
(**I**)	Baseline	80.0	81.8	56.7	56.9
(**II**)	+ Swin Trans.	85.4	87.4	58.8	59.5
(**III**)	+ VCM	87.0	88.6	60.1	61.3
(**IV**)	only VCM	86.4	88.0	59.6	60.1
(**V**)	(**IV**) + 4D mix	86.4	87.8	59.9	59.6
(**VI**)	(**III**) + GPP	87.3	88.8	60.7	61.4
(**VII**)	+ ATD	**90.3**	**90.8**	**68.0**	**68.6**

Table 4: **Ablation study for VAT.**

Different aggregators	FSS-1000 [30] mIoU (%)	mBA (%)	Memory (GB)	Run-time (ms)
Standard transformer [65]	OOM	OOM	84	N/A
MLP-Mixer [63]	OOM	OOM	OOM	N/A
Center-pivot 4D convolutions [40]	88.1	66.5	**3.5**	**52.7**
Linear transformer [24]	87.7	66.5	**3.5**	56.8
Fastformer [70]	87.8	66.4	**3.5**	122.9
4D Conv. Swin transformer (Ours)	**90.3**	**68.0**	3.8	57.3

Table 5: **Ablation study for VTM.** OOM: Out of Memory.

that using standard transformer requires approximately 84 GB per batch, while the memory for MLP-Mixer could not be measured as it is much greater than standard transformer. Also, we note that the architecture with center-pivot convolutions is equivalent to a deeper version of the architecture with VCM.

For a fair comparison, we only replace VTM with another aggregator and leave all the other components in our architecture unchanged. We observe that our method outperforms the other aggregators by a large margin. Interestingly, although center-pivot 4D convolution [40] also focuses on locality as in Swin Transformer [36], the performance gap indicates that the ability to adaptively consider pixel-wise interactions is critical. Also, we conjecture that the SW-MSA operation helps to compensate for the lack of global aggregation, which center-pivot convolutions lack. Another interesting point is that Linear Transformer [24] and Fastformer [70], which benefit from the global receptive fields of transformers and approximate the self-attention computation, achieve similar performance.

We additionally provide memory and run-time comparison to other aggregators in Table 5. The results are obtained using a single NVIDIA GeForce RTX 3090 GPU and Intel Core i7-10700 CPU. We observe that VAT is relatively slower and consumes more memory. However, 0.3 GB more memory consumption and 5 ms slower run time is a minor sacrifice for better performance.

Can VAT also perform well on semantic correspondence? To tackle the few-shot segmentation task, we reformulated it as finding semantic correspondences under large intra-class variations and geometric deformations. This suggests that the proposed method could be effective for semantic correspondence as well. Here, we compare VAT to other state-of-the-art methods in semantic correspondence.

In order to ensure a fair comparison, we note whether each method leverages multi-level features and fine-tunes the backbone networks. We additionally denote the types of cost aggregation. Note that the only difference we made for this experiment is the objective function for loss computation. Following the common protocol [42, 44, 83, 21, 41, 6], we use standard benchmarks for this task and our model was trained on the training split of PF-PASCAL [14] when evaluated on the test split of PF-PASCAL [14] and PF-WILLOW [13], and trained on SPair-71k [43] when evaluated on SPair-71k [43]. Experimental setting and implementation details can be found in supplementary material.

As shown in Table 6, VAT either sets a new state-of-the-art [43, 13] or attains the second highest PCK [14], indicating the importance of cost aggregation in

Methods	F.T. Feat.	Data Aug.	Cost Aggregation	SPair-71k [43] PCK @ α_{bbox}				PF-PASCAL [14] PCK @ α_{img}				PF-WILLOW [13] PCK @ α_{bbox}		
				0.03	0.05	0.1	0.15	0.03	0.05	0.1	0.15	0.05	0.1	0.15
NC-Net [52]	✓	✗	4D Conv.	-	-	20.1	-	30.9	54.3	78.9	86.0	33.8	67.0	83.7
SCOT [34]	-	✗	OT-RHM	-	-	35.6	-	-	63.1	85.4	92.7	47.8	76.0	87.1
CHM [41]*	✓	✗	4D Conv.	14.9	27.2	46.3	57.5	67.5	80.1	91.6	94.9	52.7	79.4	87.5
MMNet [83]	✓	✗	-	-	-	40.9	-	-	77.6	89.1	94.3	-	-	-
PMNC [26]	✓	✗	4D Conv.	-	-	50.4	-	**71.6**	**82.4**	90.6	-	-	-	-
DHPF [44]*	✓	✗	RHM	11.0	20.9	37.3	47.5	52.0	75.7	90.7	95.0	49.5	77.6	89.1
	✓	✓	RHM	-	-	39.4	-	-	-	-	-	-	-	-
CATs [6]*	✓	✗	Transformer	10.2	21.6	43.5	55.0	41.6	67.5	89.1	94.9	46.6	75.6	87.5
	✓	✓	Transformer	13.8	27.7	49.9	61.7	49.9	75.4	**92.6**	**96.4**	50.3	79.2	90.3
VAT	✓	✗	Transformer	14.9	28.3	48.4	59.1	54.6	72.9	91.1	95.6	46.0	78.8	91.3
	✓	✓	Transformer	**19.6**	**35.0**	**55.5**	**65.1**	62.7	78.2	92.3	96.2	**52.8**	**81.6**	**91.4**

Table 6: **Quantitative results on SPair-71k [43], PF-PASCAL [14] and PF-WILLOW [13].** *: The results are obtained using pretrained weights provided by authors or taken from papers.

both few-shot segmentation and semantic correspondence. It also has the potential to benefit general-purpose matching networks as well. Furthermore, when data augmentation is used, we observe a relatively large performance gain compared to DHPF [44], showing that augmentation helps to address the heavy need for data and lack of inductive bias in transformers [11, 6]. Although VAT is on par with state-of-the-art on PF-PASCAL [14], we argue that PF-PASCAL [14] is almost saturated, which makes a comparison difficult. Also, it should be noted that for performance on PF-WILLOW [13], VAT outperforms other methods by large margin, which clearly shows superior generalization power of the proposed 4D Convolutional Swin Transformer.

5 Conclusion

In this paper, we presented a novel cost aggregation network for few-shot segmentation. To address issues that arise from tokenization of a correlation map for transformer processing, we proposed a 4D Convolutional Swin Transformer, where a high-dimensional Swin Transformer is preceded by a series of small-kernel convolutions. To boost aggregation performance, we applied transformers within a pyramidal structure, and the output is then filtered and in the subsequent decoder with the help of image's appearance embedding. We have shown that the proposed method attains state-of-the-art performance for all the standard benchmarks for both few-shot segmentation and semantic correspondence, where cost aggregation plays a central role.

Acknowledgements. This research was supported by the MSIT, Korea (IITP-2022-2020-0-01819, ICT Creative Consilience program), and National Research Foundation of Korea (NRF-2021R1C1C1006897).

References

1. Ba, J.L., Kiros, J.R., Hinton, G.E.: Layer normalization. arXiv preprint arXiv:1607.06450 (2016)
2. Boudiaf, M., Kervadec, H., Masud, Z.I., Piantanida, P., Ben Ayed, I., Dolz, J.: Few-shot segmentation without meta-learning: A good transductive inference is all you need? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2021)
3. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence (2017)
4. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV) (2018)
5. Cheng, H.K., Chung, J., Tai, Y.W., Tang, C.K.: CascadePSP: Toward class-agnostic and very high-resolution segmentation via global and local refinement. In: CVPR (2020)
6. Cho, S., Hong, S., Jeon, S., Lee, Y., Sohn, K., Kim, S.: Cats: Cost aggregation transformers for visual correspondence. In: Thirty-Fifth Conference on Neural Information Processing Systems (2021)
7. Dai, Z., Liu, H., Le, Q., Tan, M.: Coatnet: Marrying convolution and attention for all data sizes. Advances in Neural Information Processing Systems **34** (2021)
8. d'Ascoli, S., Touvron, H., Leavitt, M., Morcos, A., Biroli, G., Sagun, L.: Convit: Improving vision transformers with soft convolutional inductive biases. arXiv preprint arXiv:2103.10697 (2021)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. Ieee (2009)
10. Dong, N., Xing, E.P.: Few-shot semantic segmentation with prototype learning. In: BMVC (2018)
11. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
12. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. International journal of computer vision (2010)
13. Ham, B., Cho, M., Schmid, C., Ponce, J.: Proposal flow. In: CVPR (2016)
14. Ham, B., Cho, M., Schmid, C., Ponce, J.: Proposal flow: Semantic correspondences from object proposals. IEEE transactions on pattern analysis and machine intelligence (2017)
15. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Simultaneous detection and segmentation. In: European conference on computer vision. Springer (2014)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2016)
17. Hong, S., Kim, S.: Deep matching prior: Test-time optimization for dense correspondence. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021)

18. Hosni, A., Rhemann, C., Bleyer, M., Rother, C., Gelautz, M.: Fast cost-volume filtering for visual correspondence and beyond. PAMI (2012)
19. Hu, H., Gu, J., Zhang, Z., Dai, J., Wei, Y.: Relation networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3588–3597 (2018)
20. Hu, H., Zhang, Z., Xie, Z., Lin, S.: Local relation networks for image recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3464–3473 (2019)
21. Huang, S., Wang, Q., Zhang, S., Yan, S., He, X.: Dynamic context correspondence network for semantic alignment. In: ICCV (2019)
22. Jeon, S., Min, D., Kim, S., Choe, J., Sohn, K.: Guided semantic flow. In: ECCV. Springer (2020)
23. Jiang, W., Trulls, E., Hosang, J., Tagliasacchi, A., Yi, K.M.: Cotr: Correspondence transformer for matching across images. arXiv preprint arXiv:2103.14167 (2021)
24. Katharopoulos, A., Vyas, A., Pappas, N., Fleuret, F.: Transformers are rnns: Fast autoregressive transformers with linear attention. In: International Conference on Machine Learning. pp. 5156–5165. PMLR (2020)
25. Kim, S., Min, D., Ham, B., Jeon, S., Lin, S., Sohn, K.: Fcss: Fully convolutional self-similarity for dense semantic correspondence. In: CVPR (2017)
26. Lee, J.Y., DeGol, J., Fragoso, V., Sinha, S.N.: Patchmatch-based neighborhood consensus for semantic correspondence. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)
27. Lee, J., Kim, D., Ponce, J., Ham, B.: Sfnet: Learning object-aware semantic correspondence. In: CVPR (2019)
28. Li, G., Jampani, V., Sevilla-Lara, L., Sun, D., Kim, J., Kim, J.: Adaptive prototype learning and allocation for few-shot segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8334–8343 (2021)
29. Li, S., Han, K., Costain, T.W., Howard-Jenkins, H., Prisacariu, V.: Correspondence networks with adaptive neighbourhood consensus. In: CVPR (2020)
30. Li, X., Wei, T., Chen, Y.P., Tai, Y.W., Tang, C.K.: Fss-1000: A 1000-class dataset for few-shot segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2020)
31. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision (2014)
32. Liu, W., Zhang, C., Ding, H., Hung, T.Y., Lin, G.: Few-shot segmentation with optimal transport matching and message flow. arXiv preprint arXiv:2108.08518 (2021)
33. Liu, W., Zhang, C., Lin, G., Liu, F.: Crnet: Cross-reference networks for few-shot segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2020)
34. Liu, Y., Zhu, L., Yamada, M., Yang, Y.: Semantic correspondence as an optimal transport problem. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2020)
35. Liu, Y., Zhang, X., Zhang, S., He, X.: Part-aware prototype network for few-shot semantic segmentation. In: European Conference on Computer Vision. pp. 142–158. Springer (2020)
36. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. arXiv preprint arXiv:2103.14030 (2021)

37. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2015)
38. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
39. Lu, Z., He, S., Zhu, X., Zhang, L., Song, Y.Z., Xiang, T.: Simpler is better: Few-shot semantic segmentation with classifier weight transformer. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)
40. Min, J., Kang, D., Cho, M.: Hypercorrelation squeeze for few-shot segmentation. arXiv preprint arXiv:2104.01538 (2021)
41. Min, J., Kim, S., Cho, M.: Convolutional hough matching networks for robust and efficient visual correspondence. arXiv preprint arXiv:2109.05221 (2021)
42. Min, J., Lee, J., Ponce, J., Cho, M.: Hyperpixel flow: Semantic correspondence with multi-layer neural features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)
43. Min, J., Lee, J., Ponce, J., Cho, M.: Spair-71k: A large-scale benchmark for semantic correspondence. arXiv preprint arXiv:1908.10543 (2019)
44. Min, J., Lee, J., Ponce, J., Cho, M.: Learning to compose hypercolumns for visual correspondence. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XV 16. Springer (2020)
45. Nguyen, K., Todorovic, S.: Feature weighting and boosting for few-shot segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 622–631 (2019)
46. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE international conference on computer vision (2015)
47. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. In: CVPR. IEEE (2007)
48. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning (2016)
49. Rocco, I., Arandjelovic, R., Sivic, J.: Convolutional neural network architecture for geometric matching. In: CVPR (2017)
50. Rocco, I., Arandjelović, R., Sivic, J.: End-to-end weakly-supervised semantic alignment. In: CVPR (2018)
51. Rocco, I., Arandjelović, R., Sivic, J.: Efficient neighbourhood consensus networks via submanifold sparse convolutions. In: ECCV (2020)
52. Rocco, I., Cimpoi, M., Arandjelović, R., Torii, A., Pajdla, T., Sivic, J.: Neighbourhood consensus networks. arXiv preprint arXiv:1810.10510 (2018)
53. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International journal of computer vision (2002)
54. Shaban, A., Bansal, S., Liu, Z., Essa, I., Boots, B.: One-shot learning for semantic segmentation. arXiv preprint arXiv:1709.03410 (2017)
55. Siam, M., Oreshkin, B., Jagersand, M.: Adaptive masked proxies for few-shot segmentation. arXiv preprint arXiv:1902.11123 (2019)
56. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
57. Snell, J., Swersky, K., Zemel, R.S.: Prototypical networks for few-shot learning. arXiv preprint arXiv:1703.05175 (2017)
58. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In: CVPR (2018)
59. Sun, G., Liu, Y., Liang, J., Van Gool, L.: Boosting few-shot semantic segmentation with transformers. arXiv preprint arXiv:2108.02266 (2021)

60. Tao, A., Sapra, K., Catanzaro, B.: Hierarchical multi-scale attention for semantic segmentation. arXiv preprint arXiv:2005.10821 (2020)
61. Tetko, I.V., Livingstone, D.J., Luik, A.I.: Neural network studies, 1. comparison of overfitting and overtraining. J. Chem. Inf. Comput. Sci. **35**, 826–833 (1995)
62. Tian, Z., Zhao, H., Shu, M., Yang, Z., Li, R., Jia, J.: Prior guided feature enrichment network for few-shot segmentation. IEEE Transactions on Pattern Analysis & Machine Intelligence (2020)
63. Tolstikhin, I.O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., et al.: Mlp-mixer: An all-mlp architecture for vision. Advances in Neural Information Processing Systems **34** (2021)
64. Truong, P., Danelljan, M., Timofte, R.: Glu-net: Global-local universal network for dense flow and correspondences. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6258–6268 (2020)
65. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems (2017)
66. Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. Advances in neural information processing systems (2016)
67. Wang, H., Zhang, X., Hu, Y., Yang, Y., Cao, X., Zhen, X.: Few-shot semantic segmentation with democratic attention networks. In: European Conference on Computer Vision (2020)
68. Wang, K., Liew, J.H., Zou, Y., Zhou, D., Feng, J.: Panet: Few-shot image semantic segmentation with prototype alignment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)
69. Wang, S., Li, B.Z., Khabsa, M., Fang, H., Ma, H.: Linformer: Self-attention with linear complexity. arXiv preprint arXiv:2006.04768 (2020)
70. Wu, C., Wu, F., Qi, T., Huang, Y., Xie, X.: Fastformer: Additive attention can be all you need. arXiv preprint arXiv:2108.09084 (2021)
71. Wu, S., Wu, T., Lin, F., Tian, S., Guo, G.: Fully transformer networks for semantic image segmentation. arXiv preprint arXiv:2106.04108 (2021)
72. Xiao, T., Singh, M., Mintun, E., Darrell, T., Dollár, P., Girshick, R.: Early convolutions help transformers see better. arXiv preprint arXiv:2106.14881 (2021)
73. Xie, G.S., Liu, J., Xiong, H., Shao, L.: Scale-aware graph neural network for few-shot semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5475–5484 (2021)
74. Xie, G.S., Xiong, H., Liu, J., Yao, Y., Shao, L.: Few-shot semantic segmentation with cyclic memory network. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)
75. Xiong, Y., Zeng, Z., Chakraborty, R., Tan, M., Fung, G., Li, Y., Singh, V.: Nyströmformer: A nyström-based algorithm for approximating self-attention (2021)
76. Yang, B., Liu, C., Li, B., Jiao, J., Ye, Q.: Prototype mixture models for few-shot semantic segmentation. In: European Conference on Computer Vision. Springer (2020)
77. Yang, L., Zhuo, W., Qi, L., Shi, Y., Gao, Y.: Mining latent classes for few-shot segmentation. arXiv preprint arXiv:2103.15402 (2021)
78. Zhang, B., Xiao, J., Qin, T.: Self-guided and cross-guided learning for few-shot segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2021)

79. Zhang, C., Lin, G., Liu, F., Guo, J., Wu, Q., Yao, R.: Pyramid graph networks with connection attentions for region-based one-shot semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9587–9595 (2019)
80. Zhang, C., Lin, G., Liu, F., Yao, R., Shen, C.: Canet: Class-agnostic segmentation networks with iterative refinement and attentive few-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5217–5226 (2019)
81. Zhang, G., Kang, G., Wei, Y., Yang, Y.: Few-shot segmentation via cycle-consistent transformer. arXiv preprint arXiv:2106.02320 (2021)
82. Zhang, H., Ding, H.: Prototypical matching and open set rejection for zero-shot semantic segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2021)
83. Zhao, D., Song, Z., Ji, Z., Zhao, G., Ge, W., Yu, Y.: Multi-scale matching networks for semantic correspondence. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3354–3364 (2021)