Fine-grained Few-shot Recognition by Deep Object Parsing

Ruizhao Zhu
rzhu@bu.edu
Pengkai Zhu
zpk@bu.edu
Samarth Mishra
samarthm@bu.edu
Venkatesh Saligrama
srv@bu.edu

Boston University
Boston, MA, US

Abstract

We propose a new method for fine-grained few-shot recognition via deep object parsing. In our framework, an object is made up of K distinct parts and for each part, we learn a dictionary of templates, which is shared across all instances and categories. An object is parsed by estimating the locations of these K parts and a set of active templates that can reconstruct the part features. We recognize test instances by comparing its active templates and the relative geometry of its part locations against those of the presented few-shot instances. Our method is end-to-end trainable to learn part templates on-top of a convolutional backbone. To combat visual distortions such as orientation, pose and size, we learn templates at multiple scales, and at test-time parse and match instances across these scales. We show that our method is competitive with the state-of-the-art, and by virtue of parsing enjoys interpretability as well.

1 Introduction

Deep neural networks (DNN) can be trained to solve visual recognition tasks with large annotated datasets. In contrast, training DNNs for few-shot recognition [37, 44], and its fine-grained variant [38], where only a few examples are provided for each class by way of supervision at test-time, is challenging. Fundamentally, the issue is that few-shots of data is often inadequate to learn an object model among all of its myriad of variations, which do not impact an object's category. For our solution, we propose to draw upon two key observations from the literature.

(A) There are specific locations bearing distinctive patterns/signatures in the feature space of a convolution neural network (CNN), which correspond to salient visual characteristics of an image instance [4, 58].

(B) Attention on only a few specific locations in the feature space, leads to good recognition accuracy [29, 40, 60].

© 2023. The copyright of this document resides with its authors.
It may be distributed unchanged freely in print or electronic forms.

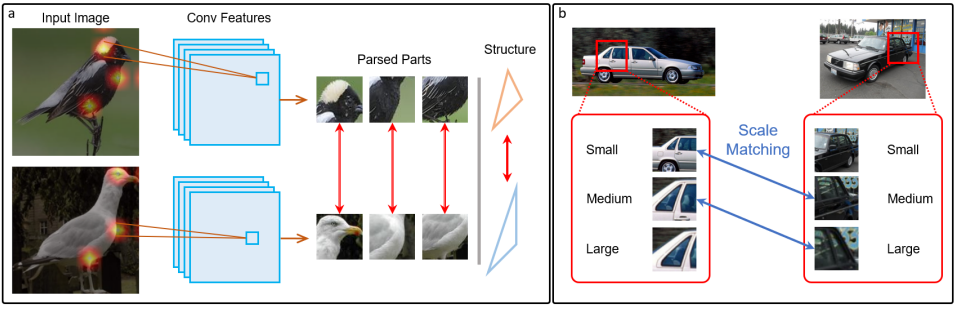

Figure 1: **Motivation:** a) In fine-grained few-shot learning, the most discriminating information is embedded in the salient parts (e.g. head and breast of a bird) and the geometry of the parts (relative part locations). Our method parses the object into a structured combination of a finite set of dictionaries, such that both finer details and the shape of the object are captured and used in recognition. b) In few shot learning, the same part may be distorted or absent in the support samples due to the perspective and pose changes. We propose to extract features and compare across multiple scales for each part to overcome this.

How can we leverage these observations?

Duplication of Traits. In fine-grained classification tasks, we posit that the visual characteristics found in one instance of an object are widely duplicated among other instances, and even among those belonging to other classes. It follows from our proposition that it is the particular collection of visual characteristics arranged in a specific geometric pattern that uniquely determines an object belonging to a particular class.

Parsing. These assumptions, along with (A) and (B), imply that these shared visual traits can be found in the feature maps of CNNs and only a few locations on the feature map suffice for object recognition. We call these finitely many latent locations on the feature maps which correspond to salient traits, *parts*. These parts manifest as patterns, where each pattern belongs to a finite (but potentially large) dictionary of templates. This dictionary embodies both the shared vocabulary and the diversity of patterns found across object instances. Our goal is to learn the dictionary of templates for different parts using training data, and at test-time, we seek to *parse*[1] new instances by identifying part locations and the sub-collection of templates that are expressed for the few-shot task. While CNN features distill essential information from images, parsing helps further suppress noisy information, in situations of high-intra class variance such as in few-shot learning. For classification, few-shot instances are parsed and then compared against the parsed query. The best matching class is then predicted as the output. As an example see Fig 1 (a), where the recognized part locations using the learned dictionary correspond to the head, breast and the knee of the birds in their images with corresponding locations in the convolutional feature maps. In matching the images, both the constituent templates and the geometric structure of the parts are utilized.

Inferring part locations based on part-specific dictionaries is a low complexity task, and is analogous to the problem of detection of signals in noise in radar applications [43], a problem solved by matching the received signal against a known dictionary of transmitted signals.

Challenges. Nevertheless, our situation is somewhat more challenging. Unlike the radar situation, we do not a-priori have a dictionary, and to learn one, we are only provided class-

[1] we view our dictionary as a collection of words, parts as phrases that are a collection of words from the dictionary, and the geometric relationship between different parts as relationship between phrases.

level annotations by way of supervision. In addition, we require that these learnt dictionaries are compact (because we must be able to reliably parse any input), and yet sufficiently expressive to account for diversity of visual traits found in different objects and classes.

Multi-Scale Dictionaries. Variations in position and orientation relative to the camera lead to different appearances of the same object by perspective projections, which means there is variation in the sizes of visual characteristics of parts. To overcome this, we train dictionaries at multiple scales, which leads us to a parsing scheme that parses input instances at multiple scales (see Fig. 1 (b)).

Goodness of fit. Besides part sizes, few-shot instances even within the same class may exhibit significant variations in poses, which can in-turn induce variations in parsed outputs. To mitigate their effects we propose a novel instance-dependent re-weighting method, for comparison, based on goodness-of-fit to the dictionary.

Contributions. (i) We propose a deep object parsing (DOP) method that parses objects into its constituent parts, and each part as a collection of activated templates from a dictionary, while using the representational power of deep CNNs. Via suitable objectives, we derive a simple end-to-end trainable formulation for this method. (ii) We evaluate DOP on the challenging task of fine-grained few shot recognition, where DOP outperforms prior art on multiple benchmarks. Notably, it is better by about 2.5% on Stanford-Car and 10% on the Aircraft dataset. (iii) We provide an analysis of how different components of our method help final performance. We also visualize the part locations recognized by our method, lending interpretability to its decisions in supplementary (Sec.C).

2 Related Work

Few-Shot Classification (FSC). Modern FSC methods can be classified into three categories: metric-learning based, optimization-based, or data-augmentation methods. Methods in the first category focus on learning effective metrics to match query examples to support. Prototypical Network [37] utilizes euclidean distance on feature space for this purpose. Subsequent approaches built on this by improving the image embedding space [1, 2, 33, 53, 59] or focusing on the metric [3, 10, 24, 36, 39, 47, 49, 54, 56]. Some recent methods have also found use of graph based methods, especially in transductive few shot classification [6, 52]. Optimization based methods train for fast adaptation using a few parameter updates with the support examples [2, 13, 22, 28, 30, 32]. Data-augmentation methods learn a generative model to synthesize additional training data for the novel classes to alleviate the issue of insufficient data [25, 35, 48, 51].

Fine-grained FSC. In fine-grained few-shot classification, different classes differ only in finer visual details. An example of this is to tease apart different species of birds in images. The approaches mentioned above have been applied in this context as well [26, 27, 38, 51]. [26] proposes to learn a local descriptor and an image-to-class measure to capture the similarity between objects. [46] uses a foreground object extractor to exclude the noise from background and synthesize foreground features to remedy the data insufficiency. BSNet [27] leverages a bi-similarity module to learn feature maps of diverse characteristics to improve the model's generalization ability. Variational feature disentangling (VFD) [51], a data-augmentation method, is complementary to ours. It disentangles the feature representation into intra-class variance and class-discriminating information, and generates additional features for novel classes at test-time. TDM [23] applies channel-wise attention to represent different classes with sparse vectors.

Recognition using Object Parts. Our method is closely related to recognition based on identifying object components, an approach motivated by how humans learn to recognize

object [5]. It draws inspiration from [42], who showed that information maximization with respect to classes of images resulted in visual features eyes, mouth, etc. in facial images and tyres, bumper, windows, etc. in images of cars. Along these lines, Deformable Part Models (DPM) [11, 12] proposed to learn object models by composing part features and geometries, and utilize it for object detection. Neural Network models for DPMs were proposed in [14, 34]. Multi-attention based models, which can be viewed as implicitly incorporating parts, have been proposed [57] in the context of fine-grained recognition problems. Although related, a principle difference is our few-shot setting, where new classes emerge, and we need to generate new object models on-the-fly.

Prior works on FSC [16, 40, 41, 50, 56] have also focused on combining parts, albeit with different notions of the concept. As such, the term part is overloaded and is unrelated to our notion. We differ in our use of a finite dictionary of templates for learning a compact representation of parts. Also, we use reconstruction as supervision for accurately localizing salient object parts, and impose a meaningful prior on the geometry of parts, which keeps us from degenerate solutions for part locations. For a more detailed comparison with prior related work in fine grained FSC, please refer to the supplementary (Sec.B).

3 Method

3.1 Deep Object Parsing

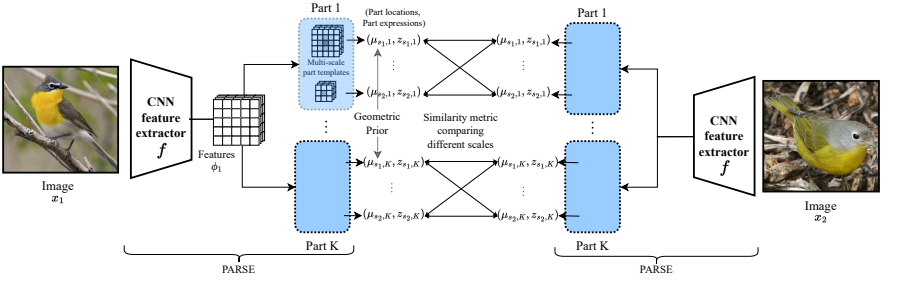

Figure 2: **Deep Object Parsing.** An image x is parsed as a collection of salient parts (K in number). Each part is represented by a 2D location μ and part expression vector z. We notate this operation PARSE and describe it in detail in Algorithm 1. In our method, we estimate locations and expressions at multiple scales for each part (hence there are more than one μ and z per part) and using these, determine image similarity for few-shot recognition.

Parsing Instances. Each input instance to our method is first parsed using learned templates into a higher-level syntax, in the form of parts. While this term, "parts", is overloaded in prior works, our notion of a part is a tuple, consisting of part-location and part-expression at that location.

Introducing notation, let $x \in \mathcal{X}$ be an input instance (in our case, an image), $f : \mathcal{X} \to \mathbb{R}^{G\times G\times C}$ a convolutional neural network (CNN) backbone and $\phi = f(x)$ features of x, with C channels supported on a 2D $G \times G$ grid. We parse x into K distinct part-locations $\mu_p \in [G] \times [G]$ and part expressions $z_p \in \mathbb{R}^C$ for $p \in [K]$. In our method, we also learn a dictionary of feature-space templates $D_{p,c} \in \mathbb{R}^{s\times s}, p \in [K], c \in [C]$ that are used to represent part features of different instances across different categories.

Given an $s \times s$ mask $M(\mu_p)$ centered at μ_p (with $s < G$), the learned templates reconstruct part features ϕ with the z_p acting as codes : $\phi_{c,M_{\mu_p}} \approx z_{p,c} D_{p,c}$, where the subscript M_{μ_p}

denotes a projection onto the support of $M(\mu)$ (or simply an $s \times s$ window cut-out of ϕ_c centered at μ_p). Note that instead of using multiple, we used one dictionary atom per part per channel. While more atoms can reconstruct features better, we found experimentally that they did not benefit few-shot learning performance.

Part Expression as LASSO Regression. Given an instance x, its feature output, ϕ, and a candidate part-location, μ, we can estimate sparse part-expression coefficients $z_p(\mu) \in \mathbb{R}^C$ by optimizing the ℓ_1 regularized reconstruction error, at the location $\mu = \mu_p$ (λ being the regularization constant).

$$z_p(\mu) = \arg\min_{\beta} \sum_{c \in C} \|\phi_{c,M(\mu)} - D_{p,c}\beta_c\|^2 + \lambda \|\beta\|_1. \tag{1}$$

Non-negativity. Part expressions $z_{p,c}$ signify presence or absence of part templates in the observed feature vectors, and as such can be expected to take on non-negative values. This fact turns out to be useful later for DNN implementation.

Part Location Estimation. Note that part expression z_p is a function of location μ, while the part location μ_p can be estimated by plugging in the optimal part-expressions for each candidate location value, namely,

$$\mu_p = \arg\min_{\mu \in [G] \times [G]} \sum_{c \in C} \|\phi_{c,M(\mu)} - D_{p,c} z_{p,c}(\mu)\|^2 + \lambda \|z_p(\mu)\|_1 \tag{2}$$

This couples the two estimation problems, and is difficult to implement with DNNs, motivating our approach below.

Feedforward DNNs for Parsing. To make the proposed approach amenable to DNN implementation, we approximate the solution to Equation (1) by optimizing the reconstruction error followed by thresholding, namely, we compute $z'_p(\mu) = \arg\min_\beta \sum_{c \in C} \|\phi_{c,M(\mu)} - D_{p,c}\beta_c\|^2$, and we threshold the resulting output by deleting entries smaller than ζ: $S_\zeta(u) = u\mathbf{1}_{|u| \geq \zeta}$. This is closely related to thresholding methods employed in LASSO [17].

The quadratic component of the loss allows for an explicit solution, and the solution reduces to template matching per channel, which can further be expressed as a convolution [15]. Using this insight, we derive our estimate of μ_p as

$$\mu_p = \arg\max_{\mu \in [G] \times [G]} \sum_{c \in C} ((\theta_{p,c} * \phi_c)(\mu) - \lambda_c)^2 \tag{3}$$

where $*$ is convolution, $\theta_{p,c} = D_{p,c}/\|D_{p,c}\|$, and $\lambda_c = \lambda/2\|D_{p,c}\|$ becomes a channel dependent constant. With the above estimate of μ_p, we get the estimate of z_p as (recall $S_\zeta(u) = u\mathbf{1}_{|u| \geq \zeta}$):

$$z'_{p,c} = \frac{(D_{p,c} * \phi_c)(\mu_p)}{\|D_{p,c}\|^2};\ z_{p,c}(\mu) = S_\zeta(z'_{p,c}) \tag{4}$$

For a full derivation of the above estimates, please refer to Supplementary (Sec. A).

Estimates differentiable in parameters. Since argmax is a non-differentiable function, using Equation (3) for estimating part-locations does not allow us to use gradient based learning for the parameters of the DNN. We can circumvent this by approximating the argmax as the expectation of a softmax distribution ν_p over $[G] \times [G]$ with a low temperature T.

$$\nu_p(\mu) \triangleq \text{softmax}\left(\frac{1}{T} \sum_{c \in C} ((\theta_{p,c} * \phi_c)(\mu) - \lambda_c)^2\right);\ \mu_p = \mathbb{E}_{\mu \sim \nu_p} \mu \tag{5}$$

Algorithm 1 PARSE (Object Parsing using DNNs)

Given: Backbone f, templates $\{D_{s,p,c}\}$, threshold ζ, ℓ_1 constant λ, temperature T
Input: Image x
Compute convolutional features $\phi = f(x)$
for $p \in [K], s \in \mathcal{S}$ **do**
  Estimate $\mu_{s,p}$ (through $\nu_{s,p}$) using Equation (5)
  Estimate $z_{s,p} = [z_{s,p,c}]_{c\in[C]}$ using Equation (6),
end for
Output: Part locations and expressions $(\{\mu_{s,p}\}_{p\in[K],s\in\mathcal{S}}, \{z_{s,p}\}_{p\in[K],s\in\mathcal{S}})$

$$z'_{p,c} = \left[\frac{(D_{p,c} * \widehat{\delta}_{\mu_p}) : \phi_c}{\|D_{p,c}\|^2}\right]; \; z_{p,c} = S_\zeta(z'_{p,c}) \tag{6}$$

where $\widehat{\delta}_{\mu_p}$ is a differentiable approximation of a dirac delta centered at μ_p using a narrow normal distribution and ':' is the double-dot product or the sum of all elements of an element-wise/Hadamard product. Our derivation (Sec. A1) hence leads to very simple expressions, where part-locations μ_p come from template matching (or convolving the templates) with the CNN features and pooling the product of location indices and ν_p. Part-expressions are then found via a simple convolution and dot product (Equation (6)).

Multi-Scale Extension. We extend our approach to incorporate parsing parts at multiple scales. This is often required because of significant difference in orientation and pose between query and support examples. To do so we simply consider masks $M(\mu)$ and templates D at varying mask sizes $s \in \mathcal{S}$, each leading to independent part location and expression estimates ($\mu_{s,p}$, $z_{s,p}$) for part p. Algorithm 1 specifies the parse of an input instance and Figure 2 shows an overview of object parsing.

3.2 Few-Shot Recognition

At test-time we are given a query instance, q, and by way of supervision, M support examples each for N classes, and the goal is to predict the query class label $y^{(q)} \in [N]$. We first run PARSE (Algorithm 1) on each of these. $\texttt{PARSE}(q) = (\{\mu^{(q)}_{s,p}\}, \{z^{(q)}_{s,p}\})$ and for the i^{th} support example of class y, $\texttt{PARSE}(x^{(i,y)}) = (\{\mu^{(i,y)}_{s,p}\}, \{z^{(i,y)}_{s,p}\})$. For comparing query and support examples we need a notion of distance/similarity, which we define next.

Goodness-of-fit reweighting. The entropy of the distribution $\nu_{s,p}$ is an important indicator of the goodness-of-fit of the dictionary templates (lower entropy meaning a more precise and confident part-location prediction as a result of a better fit). Let $h^{(q)}_{s,p}$ and $h^{(i,y)}_{s,p}$ be the entropies of $\nu^{(q)}_{s,p}$ and $\nu^{(i,y)}_{s,p}$ respectively. To use these as weights for computing distance (as below), we learn a simple parametric function $\alpha : \mathbb{R}^{M+1} \to \mathbb{R}$.

Additionally, with $z^{(y)}_{s,p} = \frac{1}{M}\sum_{i\in[M]} z^{(i,y)}_{s,p}, s \in \mathcal{S}, p \in [K]$ we represent the mean part expression over all support examples in class y. With these, we define the total distance measure $d(q,y)$ between the query example q and the support examples of class y as the combination

of expression distance $d_e(q,y)$ and geometric distance $d_g(q,y)$,

$$d(q,y) = d_e(q,y) + \gamma d_g(q,y), \tag{7}$$
$$d_e(q,y) = \sum_{p\in[K]} \sum_{s_1,s_2\in\mathcal{S}} \alpha(h^{(q)}_{s_2,p}, [h^{(i,y)}_{s_1,p}]_{i\in[M]}) \left\| z^{(y)}_{s_1,p} - z^{(q)}_{s_2,p} \right\|^2,$$
$$d_g(q,y) = \sum_{i\in[M]} \sum_{s_1,s_2\in\mathcal{S}} \left\| \psi([\mu^{(i,y)}_{s_1,p}]_{p\in[K]}) - \psi([\mu^{(q)}_{s_2,p}]_{p\in[K]}) \right\|^2.$$

where $\psi([\mu_{s,p}]_{p\in[K]})$ is a vector of pairwise distances between all part locations at scale s, normalized to unit sum. The distance function consists of an expression term, and a geometric term with γ acting as a tunable weight to control the proportion of the two. Each term is a sum over all combinations of part scales over query and support. Note that the geometric term simply attempts to find if two polygons with vertices at part locations are similar (i.e. have proportional sides), with the distance being 0 if they are. Finally, the class prediction is made as $\widehat{y}^{(q)} = \arg\min_{y\in[N]} d(q,y)$.

Training. We train in episodes following convention. For each episode, we sample N classes at random, and additionally sample support and query examples belonging to these classes from training data (details in Section 4). Using a softmax over the negative distance function above as the class distribution of query q, we define the cross-entropy loss as

$$\ell_{CE}(q) = -\log \frac{\exp(-d(q,y^{(q)}))}{\sum_{y\in[N]} \exp(-d(q,y))} \tag{8}$$

Additionally, while training, we impose a geometric prior to get diverse instance parts in PARSE by maximizing the Hellinger distance [9] $\mathbb{H}(\cdot,\cdot)$ between part distributions. The corresponding criterion for minimization is

$$\ell_{div}(x) = -\sum_{s\in\mathcal{S}} \sum_{\substack{p,p'\in[K] \\ p\neq p'}} \mathbb{H}(\nu_{s,p}, \nu_{s,p'}) \tag{9}$$

We show the overall training process in the Algorithm in supplementary (Sec. A2).

4 Experiments

4.1 Fine-grained Few-Shot Classification

Datasets. We compare DOP on four fine-grained datasets: Caltech-UCSD-Birds (CUB) [45], Stanford-Dog (Dog) [20] Stanford-Car (Car) [21] and Aircraft [31] against state-of-the-art methods. Following convention [18, 23, 26, 49], we split each dataset with non-overlapping base, validation and novel classes for training, validation and testing purposes.

Experiments Setup. We conducted 5-way (5 classes episode) 1-shot and 5-way 5-shot classification tasks on all datasets. Following the episodic evaluation protocol in [44], at test time, we sample 600 episodes and report the averaged Top-1 accuracy. In each episode, 5 classes from the test set are randomly selected. 1 or 5 samples for each class are sampled as support data, and another 15 examples are sampled for each class as the query data. The model is trained on train split and the validation split is used to select the hyper-parameters.

Table 1: Few-shot classification accuracy in % on CUB and Stanford-Dog benchmarks (along with 95% confidence intervals). †: results are obtained by running the codes released.

Methods	Backbones	CUB		Dog	
		1-shot	*5-shot*	*1-shot*	*5-shot*
ProtoNet†[37]	ResNet18	71.88±0.91	87.42±0.48	61.06±0.67	74.31±0.51
MetaOptNet†[22]	ResNet18	75.15±0.46	87.09±0.30	65.48±0.56	79.39±0.43
DeepEMD†[56]	ResNet12	75.56±0.77	88.69±0.50	69.34 ± 0.65	83.45±0.27
BSNet[27]	ResNet12	69.61±0.92	83.24±0.60	69.09±0.90	82.45±0.58
DSN[36]	ResNet12	80.47±0.20	89.92±0.12	61.51±0.22	80.21±0.15
CTX[8]	ResNet12	80.95 ±0.21	91.54±0.11	65.67±0.22	84.48±0.13
VFD†[51]	ResNet12	79.12±0.83	91.11±0.24	**70.60±0.91**	**85.74±0.53**
TOAN[19]	ResNet12	67.17±0.81	82.09±0.56	51.83±0.80	69.83±0.66
FRN[49]	ResNet12	83.16±0.19	92.59±0.23	62.07±0.22	83.18±0.14
TDM[23]	ResNet12	83.36±0.22	92.08 ±0.13	57.64±0.22	75.77±00.16
HelixFormer[53]	ResNet12	81.66±0.30	91.83±0.17	65.92±0.49	80.65±0.36
DOP	ResNet18	**82.62±0.65**	**92.61±0.38**	**70.56±0.75**	**84.75±0.41**
DOP	ResNet12	**83.39±0.82**	**93.01±0.43**	70.10±0.79	85.12±0.55

We compare our DOP to state-of-the-art FSC and fine-grained FSC methods in Table 1 and 2. More details for experiment setting are provided in the supplementary (Sec. C1, C2).

DOP is competitive with or outperforms recent works on fine-grained FSC. On CUB (Table 1), DOP outperforms all compared approaches with 83.39% 1-shot accuracy and 93.01% for 5-shot with Resnet-12 backbone. Same is the case for Car and Aircraft (Table 2), where DOP outperforms a recent method in TDM [23] by a large margin using ResNet-12 backbone. On Stanford-Dog, we outperform all methods but VFD. We note here that VFD generates additional features at test-time for novel classes, which is complementary to DOP. Interpretable visualizations and additional results are provided in supplementary (Sec.C3-C5).

4.2 Analysis

As an overview, DOP combines object parsing, dictionaries at multiple template sizes, use of part geometry for distance computation and instance-dependent distance reweighting based on goodness-of-fit. These can be seen as methodological developments over ProtoNet [37], a simple CNN feature-space distance based few shot classification approach.

ProtoNet to single part DOP. The simplified DOP method is with a single part and parsing done at a single scale $\mathcal{S} = \{5\}$. There is one template $D \in \mathbb{R}^{5\times5\times C}$ consisting of learned parameters. In Eq.5, we convolve the template over the CNN features and perform some additional operations (Eq.6) with no other learnable parameters to find part expressions. A ProtoNet with the same number of learnable parameters can use D as a final conv layer and perform global pooling over its outputs. This ProtoNet reaches 88.38% accuracy on 5-way 5-shot classification on CUB, while the simplified DOP has an accuracy of 90.36%. Thus, this improvement in performance can be attributed to learning a template shareable across parts that can be used for reconstructing features. This reconstruction objective allows the part expressions z to have less noise and thus lower intra-class variance.

Using multiple parts. Table 3 shows the effect of adding more parts on 5-way 5-shot accuracy on CUB. We see more parts up to a certain point ($K = 4$) allows DOP to learn better representations consequently improving performance, but with even more parts performance drops as the model can start learning irrelevant or background signatures.

Table 2: Few-shot classification accuracy in % on Stanford-Car and Aircraft benchmarks (along with 95% confidence intervals). †: results are obtained by running the codes released by authors.

Methods	Backbones	Car		Aircraft	
		1-shot	*5-shot*	*1-shot*	*5-shot*
ProtoNet†[37]	ResNet18	60.67±0.87	75.56±0.45	67.28±0.25	83.21±0.41
MetaOptNet†[23]	ResNet18	60.56±0.78	76.35±0.52	70.34±0.27	83.87±0.23
DeepEMD † [56]	ResNet12	79.12 ± 0.50	92.14±0.43	75.84 ± 0.45	85.33±0.24
BSNet[27]	ResNet12	60.36±0.98	85.28±0.64	-	-
DSN[36]	ResNet12	54.74±0.22	69.63±0.17	70.23±0.21	83.05±0.25
CTX[8]	ResNet12	55.66±0.22	73.78±0.16	65.53±0.22	79.31±0.13
TOAN[19]	ResNet12	76.62±0.70	89.57±0.40	-	-
FRN[49]	ResNet12	55.49±0.21	74.54±0.16	69.58±0.22	82.98±0.14
TDM[25]	ResNet12	68.36±0.22	86.14 ±0.13	70.89±0.22	84.54±0.16
HelixFormer[55]	ResNet12	79.40±0.43	92.26±0.15	74.01±0.54	83.11±0.41
DOP	ResNet18	**81.41±0.71**	**93.48±0.38**	**83.26±0.24**	**92.41±0.45**
DOP	ResNet12	**81.83±0.78**	**93.84±0.45**	84.50±0.25	93.35±0.48

Table 3: Effect of using different number of parts on 5-way 5-shot accuracy on CUB.

Num parts	1	3	4	5	6
Accuracy	90.56	92.10	**92.61**	92.21	92.06

Table 4: Effect of using templates at different scales on 5-way 5-shot accuracy on Dog.

Scales	[3]	[5]	[3,5]	[1,3,5]
Accuracy	81.56	81.38	83.04	**84.75**

Using templates at multiple scales. In Table 4 using the Stanford-Dog dataset, we studied the effect of parsing parts at multiple scales. Learning dictionaries at multiple scales improves performance, since this allows DOP to parse the object parts even when their scale may vary (due to different positions and orientations relative to the camera).

Instance-dependent reweighting based on goodness-of-fit. We use a parametric reweighting function α that reweights the distances between part expressions based on the how well the learned templates fit the part features (see Eq. 7 from the main paper). In Table 5, we show the effect of removing this reweighting, and simply using an average of all pairs of distances between the query and support. As we see, the reweighting function does help few shot classification accuracy.

Effect of using part-geometry for comparison. In Eq. 7 from the main paper, we use part geometries besides part expressions for computing distances. Table 5 also shows scenarios where we remove this component in the distance (equivalent to setting $\gamma = 0$). We see that using a distance between part geometries helps the final few shot classification performances.

Table 5: 5-way 5-shot accuracy on ablating components in distance computation: re-weighting function α and using part-geometry (Eq. 7). Both help FSL accuracy.

Part-geometry	Re-weighting	CUB	Dog	Car
		91.83	82.07	92.78
	✓	92.44	83.90	93.31
✓		91.95	83.33	93.21
✓	✓	**92.61**	**84.75**	**93.48**

5 Conclusions

We presented DOP , a deep object-parsing method for fine-grained few-shot recognition. Our fundamental concept is that, while different object classes exhibit novel visual appearance, at a sufficiently small scale, visual patterns are duplicated. Hence, by leveraging training data to learn a dictionary of templates distributed across different relative locations, an object can be recognized simply by identifying which of the templates in the dictionary are expressed, and how these patterns are geometrically distributed. We build a statistical model for parsing that takes the output of a convolutional backbone as input to produce a parsed output. We then post-hoc learn to re-weight query and support instances to identify the best matching class, and as such this procedure allows for mitigating visual distortions. Our proposed method is an end-to-end deep neural network training method, and we show that our performance is not only competitive but also the outputs generated are interpretable.

ACKNOWLEDGMENTS

This research was supported by Army Research Office Grant W911NF2110246, the National Science Foundation grants CCF-2007350 and CCF-1955981, AFRL Contract no. FA8650-22-C-1039.

References

[1] Arman Afrasiyabi, Jean-Francois Lalonde, and Christian Gagne. Mixture-based feature space learning for few-shot image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9041–9051, 2021.

[2] Sungyong Baik, Janghoon Choi, Heewon Kim, Dohee Cho, Jaesik Min, and Kyoung Mu Lee. Meta-learning with task-adaptive loss function for few-shot learning. In *ICCV*, 2021.

[3] Peyman Bateni, Raghav Goyal, Vaden Masrani, Frank Wood, and Leonid Sigal. Improved few-shot visual classification. In *CVPR*, 2020.

[4] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6541–6549, 2017.

[5] Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987.

[6] Chaofan Chen, Xiaoshan Yang, Changsheng Xu, Xuhui Huang, and Zhe Ma. Eckpn: Explicit class knowledge propagation network for transductive few-shot learning. In *CVPR*, 2021.

[7] Rajshekhar Das, Yu-Xiong Wang, and Jose MF Moura. On the importance of distractors for few-shot classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9030–9040, 2021.

[8] Carl Doersch, Ankush Gupta, and Andrew Zisserman. Crosstransformers: spatially-aware few-shot transfer. *Advances in Neural Information Processing Systems*, 33: 21981–21993, 2020.

[9] Brian Everitt. Cambridge dictionary of statistics. 1998.

[10] Nanyi Fei, Yizhao Gao, Zhiwu Lu, and Tao Xiang. Z-score normalization, hubness, and few-shot learning. In *ICCV*, 2021.

[11] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2009.

[12] Pedro F Felzenszwalb, Ross B Girshick, and David McAllester. Cascade object detection with deformable part models. In *2010 IEEE Computer society conference on computer vision and pattern recognition*, pages 2241–2248. Ieee, 2010.

[13] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

[14] Ross Girshick, Forrest Iandola, Trevor Darrell, and Jitendra Malik. Deformable part models are convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 437–446, 2015.

[15] Rafael C. Gonzalez and Richard E. Woods. *Digital image processing*. Prentice Hall, Upper Saddle River, N.J., 2008.

[16] Fusheng Hao, Fengxiang He, Jun Cheng, Lei Wang, Jianzhong Cao, and Dacheng Tao. Collect and select: Semantic alignment metric learning for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8460–8469, 2019.

[17] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[18] Nathan Hilliard, Lawrence Phillips, Scott Howland, Artëm Yankov, Courtney D Corley, and Nathan O Hodas. Few-shot learning with metric-agnostic conditional embeddings. *arXiv preprint arXiv:1802.04376*, 2018.

[19] Huaxi Huang, Junjie Zhang, Litao Yu, Jian Zhang, Qiang Wu, and Chang Xu. Toan: Target-oriented alignment network for fine-grained image categorization with few labeled samples. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(2): 853–866, 2021.

[20] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, June 2011.

[21] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013.

[22] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10657–10665, 2019.

[23] SuBeen Lee, WonJun Moon, and Jae-Pil Heo. Task discrepancy maximization for fine-grained few-shot classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5331–5340, 2022.

[24] Aoxue Li, Weiran Huang, Xu Lan, Jiashi Feng, Zhenguo Li, and Liwei Wang. Boosting few-shot learning with adaptive margin loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12576–12584, 2020.

[25] Kai Li, Yulun Zhang, Kunpeng Li, and Yun Fu. Adversarial feature hallucination networks for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13470–13479, 2020.

[26] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo. Revisiting local descriptor based image-to-class measure for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7260–7268, 2019.

[27] Xiaoxu Li, Jijie Wu, Zhuo Sun, Zhanyu Ma, Jie Cao, and Jing-Hao Xue. Bsnet: Bi-similarity network for few-shot fine-grained image classification. *IEEE Transactions on Image Processing*, 30:1318–1331, 2020.

[28] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*, 2017.

[29] Yann Lifchitz, Yannis Avrithis, and Sylvaine Picard. Few-shot few-shot learning and the role of spatial attention. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2693–2700. IEEE, 2021.

[30] Q Sun Y Liu, TS Chua, and B Schiele. Meta-transfer learning for few-shot learning. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[31] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[32] Aravind Rajeswaran, Chelsea Finn, Sham Kakade, and Sergey Levine. Meta-learning with implicit gradients. 2019.

[33] Mamshad Nayeem Rizve, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Exploring complementary strengths of invariant and equivariant representations for few-shot learning. In *CVPR*, 2021.

[34] Pierre-André Savalle, Stavros Tsogkas, George Papandreou, and Iasonas Kokkinos. Deformable part models with cnn features. In *European Conference on Computer Vision, Parts and Attributes Workshop*, 2014.

[35] Eli Schwartz, Leonid Karlinsky, Joseph Shtok, Sivan Harary, Mattias Marder, Rogerio Feris, Abhishek Kumar, Raja Giryes, and Alex M Bronstein. Delta-encoder: an effective sample synthesis method for few-shot object recognition. *arXiv preprint arXiv:1806.04734*, 2018.

[36] Christian Simon, Piotr Koniusz, Richard Nock, and Mehrtash Harandi. Adaptive subspaces for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4136–4145, 2020.

[37] Jake Snell, Kevin Swersky, and Richard S Zemel. Prototypical networks for few-shot learning. *arXiv preprint arXiv:1703.05175*, 2017.

[38] Xin Sun, Hongwei Xv, Junyu Dong, Huiyu Zhou, Changrui Chen, and Qiong Li. Few-shot learning for domain-specific fine-grained image classification. *IEEE Transactions on Industrial Electronics*, 68(4):3588–3598, 2020.

[39] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1199–1208, 2018.

[40] Luming Tang, Davis Wertheimer, and Bharath Hariharan. Revisiting pose-normalization for fine-grained few-shot recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14352–14361, 2020.

[41] Pavel Tokmakov, Yu-Xiong Wang, and Martial Hebert. Learning compositional representations for few-shot recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6372–6381, 2019.

[42] Shimon Ullman, Michel Vidal-Naquet, and Erez Sali. Visual features of intermediate complexity and their use in classification. *Nature neuroscience*, 5(7):682–687, 2002.

[43] Harry L Van Trees. *Detection, estimation, and modulation theory, part I: detection, estimation, and linear modulation theory*. John Wiley & Sons, 2004.

[44] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29: 3630–3638, 2016.

[45] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[46] Chaofei Wang, Shiji Song, Qisen Yang, Xiang Li, and Gao Huang. Fine-grained few shot learning with foreground object transformation. *Neurocomputing*, 466:16–26, 2021.

[47] Yan Wang, Wei-Lun Chao, Kilian Q Weinberger, and Laurens van der Maaten. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning. *arXiv preprint arXiv:1911.04623*, 2019.

[48] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7278–7286, 2018.

[49] Davis Wertheimer, Luming Tang, and Bharath Hariharan. Few-shot classification with feature map reconstruction networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8012–8021, 2021.

[50] Jiamin Wu, Tianzhu Zhang, Yongdong Zhang, and Feng Wu. Task-aware part mining network for few-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8433–8442, 2021.

[51] Jingyi Xu, Hieu Le, Mingzhen Huang, ShahRukh Athar, and Dimitris Samaras. Variational feature disentangling for fine-grained few-shot classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8812–8821, 2021.

[52] Ling Yang, Liangliang Li, Zilun Zhang, Xinyu Zhou, Erjin Zhou, and Yu Liu. Dpgn: Distribution propagation graph network for few-shot learning. In *CVPR*, 2020.

[53] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *CVPR*, 2020.

[54] Baoquan Zhang, Xutao Li, Yunming Ye, Zhichao Huang, and Lisai Zhang. Prototype completion with primitive knowledge for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3754–3762, 2021.

[55] Bo Zhang, Jiakang Yuan, Baopu Li, Tao Chen, Jiayuan Fan, and Botian Shi. Learning cross-image object semantic relation in transformer for few-shot fine-grained image classification. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 2135–2144, 2022.

[56] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. In *CVPR*, 2020.

[57] Heliang Zheng, Jianlong Fu, Tao Mei, and Jiebo Luo. Learning multi-attention convolutional neural network for fine-grained image recognition. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5209–5217, 2017. doi: 10.1109/ICCV.2017.557.

[58] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014.

[59] Ziqi Zhou, Xi Qiu, Jiangtao Xie, Jianan Wu, and Chi Zhang. Binocular mutual learning for improving few-shot classification. In *ICCV*, 2021.

[60] Yaohui Zhu, Chenlong Liu, and Shuqiang Jiang. Multi-attention meta learning for few-shot fine-grained image recognition. In *IJCAI*, pages 1090–1096, 2020.