TransFG: A Transformer Architecture for Fine-grained Recognition

Ju He[1] Jie-Neng Chen[1] Shuai Liu[2] Adam Kortylewski[1] Cheng Yang[2] Yutong Bai[1]
Changhu Wang[2] Alan Yuille[1]
[1]Johns Hopkins University [2]ByteDance Inc.

Abstract

Fine-grained visual classification (FGVC) which aims at recognizing objects from subcategories is a very challenging task due to the inherently subtle inter-class differences. Recent works mainly tackle this problem by focusing on how to locate the most discriminative image regions and rely on them to improve the capability of networks to capture subtle variances. Most of these works achieve this by reusing the backbone network to extract features of selected regions. However, this strategy inevitably complicates the pipeline and pushes the proposed regions to contain most parts of the objects. Recently, vision transformer (ViT) shows its strong performance in the traditional classification task. The self-attention mechanism of the transformer links every patch token to the classification token. The strength of the attention link can be intuitively considered as an indicator of the importance of tokens. In this work, we propose a novel transformer-based framework TransFG where we integrate all raw attention weights of the transformer into an attention map for guiding the network to effectively and accurately select discriminative image patches and compute their relations. A contrastive loss is applied to further enlarge the distance between feature representations of similar sub-classes. We demonstrate the value of TransFG by conducting experiments on five popular fine-grained benchmarks: CUB-200-2011, Stanford Cars, Stanford Dogs, NABirds and iNat2017 where we achieve state-of-the-art performance. Qualitative results are presented for better understanding of our model. Code is available at this https URL.

1. Introduction

Fine-grained visual classification aims at classifying sub-classes of a given object category, *e.g.*, subcategories of birds [42, 40], cars [27], aircrafts [32]. It has long been considered as a very challenging task due to the small inter-class variations and large intra-class variations along with the deficiency of annotated data, especially for the long-tailed classes. Benefiting from the progress of deep neural networks [28, 35, 19], the performance of FGVC obtain a steady progress in recent years. To avoid labor-intensive parts annotation, the community currently focuses on weakly-supervised FGVC with only image-level labels. Methods now can be roughly classified into two categories, i.e., localization methods and feature-encoding methods. Compared to feature-encoding methods, the localization methods have the advantages that they explicitly capture the subtle differences among different sub-classes which is more interpretable and usually yields better results.

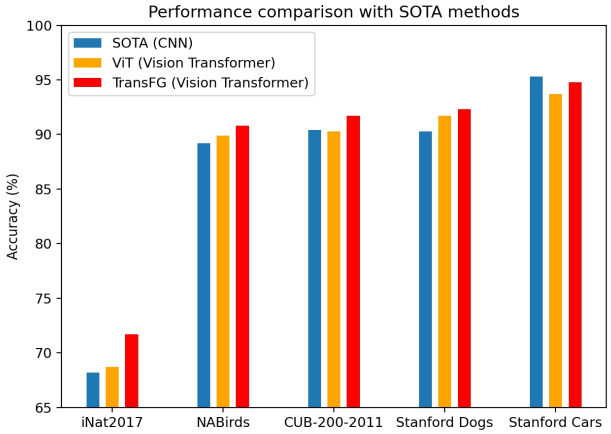

Figure 1: An overview of performance comparison of ViT and TransFG with state-of-the-art methods with CNN backbones on five datasets.

Early works in localization methods rely on the annotations of parts to locate discriminative regions while recent works [16, 29, 10, 54, 49] mainly adopt region proposal networks (RPN) to propose bounding boxes which contain the discriminative regions. After obtaining the selected image regions, they are resized into predefined size and forwarded through the backbone network again to acquire informative local features. A typical strategy is to use these local features for classification individually and adopt a rank loss [5] to maintain the consistency between the quality of bounding boxes and their final probability output. However, this mechanism ignores the relation between selected regions thus inevitably encourages the RPN to propose large bounding boxes that contain most parts of the objects in order to obtain correct classification results. Sometimes these bounding boxes can even contain large areas of backgrounds and lead to confusion. Besides, the RPN module with different optimizing goals compared to the backbone network makes the network harder to train and the re-use of backbone complicates the overall algorithm pipeline.

Recently, vision transformer [11] achieved huge success in traditional classification task which shows that applying a pure transformer directly to a sequence of image patches with its innate attention mechanism can capture the important regions in images thus facilitate the classification. A series of extended work on downstream tasks such as object detection [3], semantic segmentation [55, 46, 4] confirmed the strong ability for vision transformer to capture both global and local features.

In this paper, we present the first study which explores the potential of vision transformers in the context of fine-grained visual classification. We find that directly applying ViT on FGVC already produces satisfactory results while a lot of adaptations according to the characteristics of FGVC can be applied to further boost the performance. To this end, we propose TransFG, a simple yet effective framework based on ViT. To be specific, by leveraging the innate multi-head self-attention mechanism, we propose a part selection module to compute the discriminative regions and remove redundant information. We then concatenate the selected part tokens along with the global classification token as input sequence to the last transformer layer. To further enlarge the distance between feature representations of samples from different categories and decrease that of samples from the same categories and decrease that of samples from same categories, we introduce a contrastive loss that further boosts performance.

We evaluate our model extensively on five popular fine-grained visual classification benchmarks (CUB-200-2011, Stanford Cars, Stanford Dogs, NABirds, iNat2017). An overview of the performance comparison can be seen in Fig 1 where our TransFG outperforms existing SOTA CNN methods with different backbones on most datasets. In summary, we make several important contributions in this work:

1. To the best of our knowledge, we are the first to verify the effectiveness of vision transformer on fine-grained visual classification which offers an alternative to the dominating CNN backbone with RPN model design.

2. We introduce TransFG, a novel neural architecture for fine-grained visual classification that naturally focuses on the most discriminative regions of the objects and achieve SOTA performance on several standard benchmarks.

3. Visualization results are presented which illustrate the ability of our TransFG to accurately capture discriminative image regions and help us to better understand how it makes correct predictions.

2. Related Work

In this section, we briefly review existing works on fine-grained visual classification and transformer.

2.1. Fine-Grained Visual Classification

Many works have been done to tackle the problem of fine-grained visual classification and they can roughly be classified into two categories: localization methods [16, 29, 10, 54, 49, 47] and feature-encoding methods [50, 53, 15, 56, 30]. The former focuses on training a detection network to localize discriminative part regions and reuse them to perform classification. The latter targets at learning more informative features by either computing higher-order information or finding the relationships among contrastive pairs.

2.1.1 Localization FGVC methods

Previously, some works [2, 45] tried to exploit the part annotations to supervise the learning procedure of the localization process. However, since such annotations are expensive and usually unavailable, weakly-supervised parts proposal with only image-level labels draw more attentions nowadays. He et al. [21] proposed a sophisticated reinforcement learning procedure to estimate how to select the discriminative image regions and the number of them. Ge et al. [16] exploited Mask R-CNN and CRF-based segmentation alternatively to extract object instances and discriminative regions. Yang [47] proposed a re-ranking strategy to re-rank the global classification results based on the database constructed with region features. However, these methods all need a special designed module to propose potential regions and these selected regions need to be forwarded through the backbone again for final classification. Besides, some of the proposed regions often contain the whole object which are not discriminative enough.

2.1.2 Feature-encoding methods

The other branch of methods focus on enriching the feature representations to obtain better classification results. Yu et al. [50] proposed a hierarchical framework to do cross-layer bilinear pooling. Zheng et al. [53] adopted the idea of group convolution to first split channels into different groups by their semantic meanings and then do the bilinear pooling within each group without changing the dimension thus it can be integrated into any existed backbones directly.

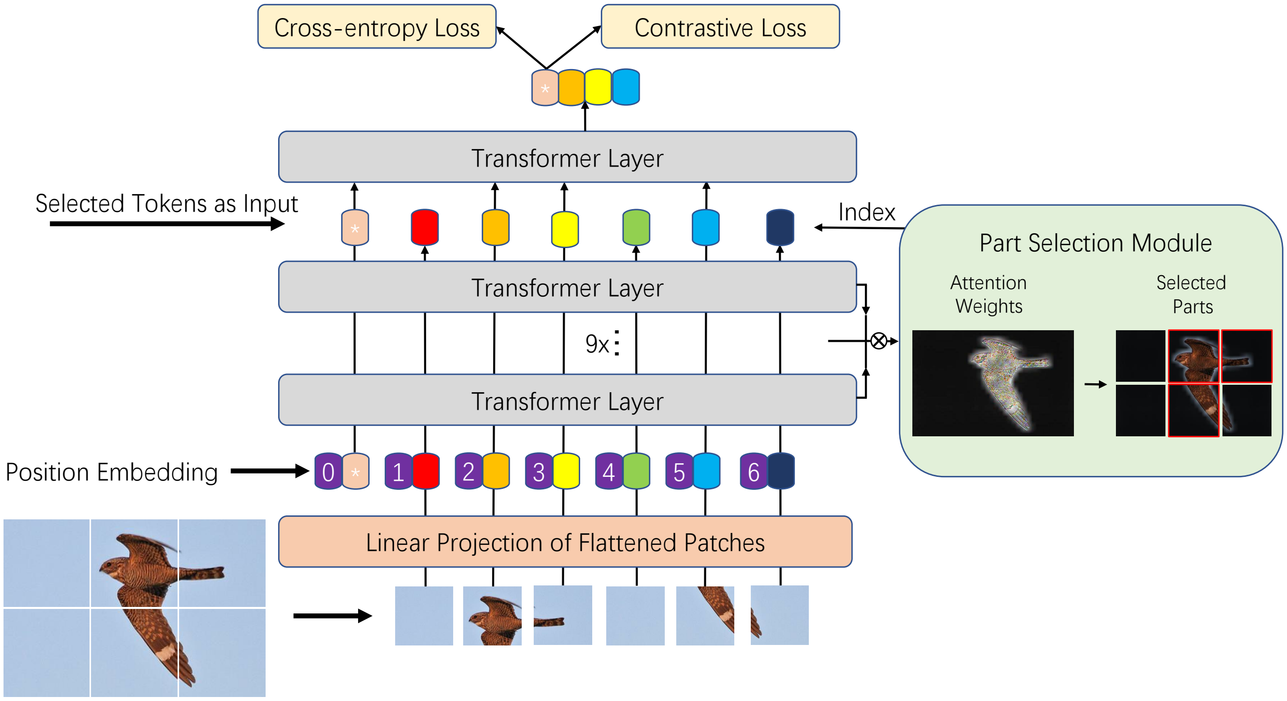

Figure 2: The framework of our proposed TransFG. Images are split into small patches (a non-overlapping split is shown here) and projected into the embedding space. The input to the Transformer Encoder consists of patch embeddings along with learnable position embeddings. Before the last Transformer Layer, a Part Selection Module (PSM) is applied to select tokens that corresponds to the discriminative image patches and only use these selected tokens as input. Cross-entropy loss and contrastive loss on the final classification token contribute to the training of TransFG. Best viewed in color.

Zhuang et al. [56] proposed to construct contrastive input batches and compute the cues between them to force the features contain such discriminative information. However, these methods are usually not interpretable such one does not know what makes the model distinguish sub-categories with subtle differences.

2.2. Transformer

Transformer and self-attention models have greatly facilitated research in natural language processing and machine translation [8, 9, 41, 48]. Inspired by this, many recent studies try to apply transformers in computer vision area. Initially, transformer is used to handle sequential features extracted by CNN backbone for the videos. Girdhar et al. [18] exploited a variant of transformer to aggregate contextual cues related to a specific person in a video. Later, transformer models are further extended to other popular computer vision tasks such as object detection [3], segmentation [46, 4, 51], object tracking [36]. Most recently, pure transformer models are becoming more and more popular. ViT [11] is the first work to show that applying a pure transformer directly to a sequence of image patches can yield state-of-the-art performance on image classification. Based on that, Zheng et al. [55] proposed SETR to exploit ViT as the encoder for segmentation. He et al. [20] proposed TransReID which embedded side information into transformer along with the JPM to boost the performance on object re-identification. In this work, we extend ViT to fine-grained visual classification and show its effectiveness.

3. Method

We briefly review the framework of vision transformer and show how to do some preprocessing steps to extend it into fine-grained recognition in Section 3.1. Then, the overall framework of TransFG will be elaborated in Section 3.2

3.1. Vision transformer as feature extractor

Image Sequentialization. Following ViT, we first preprocess the input image into a sequence of flattened patches x_p. However, the original split method cut the images into non-overlapping patches, which harms the local neighboring structures especially when discriminative regions are split. To alleviate this problem, we propose to generate overlapping patches with sliding window. To be specific, we denote the input image with resolution $H * W$, the size of image patch as P and the step size of sliding window as S. Thus the input images will be split into N patches where

$$N = N_H * N_W = \lfloor \frac{H - P + S}{S} \rfloor * \lfloor \frac{W - P + S}{S} \rfloor \quad (1)$$

In this way, two adjacent patches share an overlapping area of size $(P - S) * P$ which helps to preserve better local region information. Typically speaking, the smaller the step S is, the better the performance will be. But decreasing S will at the same time enlarge the computational cost, so a trade-off needs to be made here.

Patch Embedding. We map the vectorized patches x_p into a latent D-dimensional embedding space using a trainable linear projection. A learnable position embedding is added to the patch embeddings to retain positional information as follows:

$$\mathbf{z}_0 = [x_p^1\mathbf{E}, x_p^2\mathbf{E}, \cdots, x_p^N\mathbf{E}] + \mathbf{E}_{pos} \quad (2)$$

where N is the number of image patches, $\mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) * D}$ is the patch embedding projection, and $\mathbf{E}_{pos} \in \mathbb{R}^{N * D}$ denotes the position embedding.

The Transformer encoder [41] contains L layers of multihead self-attention (MSA) and multi-layer perceptron (MLP) blocks. Thus the output of the l-th layer can be written as follows:

$$\mathbf{z}_l^{'} = MSA(LN(\mathbf{z}_{l-1})) + \mathbf{z}_{l-1} \qquad l \in 1, 2, \cdots, L \quad (3)$$

$$\mathbf{z}_l = MLP(LN(\mathbf{z}_l^{'})) + \mathbf{z}_l^{'} \qquad l \in 1, 2, \cdots, L \quad (4)$$

where $LN(\cdot)$ denotes the layer normalization operation and $\mathbf{z}_l$ is the encoded image representation. ViT exploits the first token of the last encoder layer z_L^0 as the representation of the global feature and forward it to a classifier head to obtain the final classification results without considering the potential information stored in the rest tokens.

3.2. TransFG Architecture

While our experiments in Section 4 show that the pure Vision Transformer can be directly applied into fine-grained visual classification and achieve impressive results. It does not well capture the local information required for FGVC. To this end, we propose the Part Selection Module (PSM) and apply contrastive feature learning to enlarge the distance of representations between similar sub-categories. The framework of our proposed TransFG is illustrated in Fig 2.

3.2.1 Part Selection Module

One of the most important problems in fine-grained visual classification is to accurately locate the discriminative regions that account for subtle differences between similar sub-categories. Take a confusing pair of images from the CUB-200-2011 dataset as shown in Fig 3 for example. The model needs to have the ability to capture the very small differences, i.e., the color of eyes and throat in order to distinguish these two bird species. Region proposal networks and weakly-supervised segmentation strategies are widely introduced to tackle this problem in the traditional CNN-based methods.

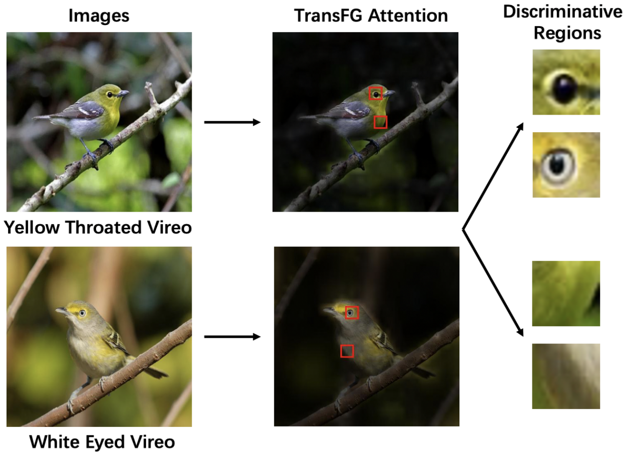

Figure 3: A confusing pair of instances from the CUB-200-2011 dataset. Model needs to has the ability to capture the subtle differences in order to classify them correctly. The second column shows the overall attention maps and two selected tokens of our TransFG method. Best viewed in color.

Vision Transformer model is perfectly suited here with its innate multi-head attention mechanism. To fully exploit the attention information, we change the input to the last Transformer Layer. Suppose the model has K self-attention heads and the hidden features input to the last layer are denoted as $\mathbf{z}_{L-1} = [z_{L-1}^0; z_{L-1}^1, z_{L-1}^2, \cdots, z_{L-1}^N]$. The attention weights of the previous layers can be written as follows:

$$\mathbf{a}_l = [a_l^0, a_l^1, a_l^2, \cdots, a_l^K] \qquad l \in 1, 2, \cdots, L-1 \quad (5)$$

$$a_l^i = [a_l^{i_0}; a_l^{i_1}, a_l^{i_2}, \cdots, a_l^{i_N}] \qquad i \in 0, 1, \cdots, K \quad (6)$$

Previous works [34, 1] suggested that the raw attention weights do not necessarily correspond to the relative importance of input tokens especially for higher layers of a model, due to lack of token identifiability of the embeddings. To this end, we propose to integrate attention weights of all previous layers. To be specific, we recursively apply a matrix multiplication to the raw attention weights in all the layers as

$$\mathbf{a}_{final} = \prod_{l=0}^{L-1} \mathbf{a}_l \quad (7)$$

As $\mathbf{a}_{final}$ captures how information propagates from the input layer to the embeddings in higher layers, it serves as a better choice for selecting discriminative regions compared to the single layer raw attention weights a_{L-1}. We then choose the index of the maximum value $A_1, A_2, \cdots, A_K$ with respect to the K different attention heads in $\mathbf{a}_{final}$. These positions are used as index for our model to extract the corresponding tokens in $\mathbf{z}_{L-1}$. Finally, we concatenate the selected tokens along with the classification token as the input sequence which is denoted as:

$$\mathbf{z}_{local} = [z_{L-1}^{0}; z_{L-1}^{A_1}, z_{L-1}^{A_2}, \cdots, z_{L-1}^{A_K}] \tag{8}$$

By replacing the original entire input sequence with tokens corresponding to informative regions and concatenate the classification token as input to the last Transformer Layer, we not only keep the global information but also force the last Transformer Layer to focus on the subtle differences between different sub-categories while abandoning less discriminative regions such as background or common features among a super class.

3.2.2 Contrastive feature learning

Following ViT, we still adopt the first token z_i of the PSM module for classification. A simple cross-entropy loss is not enough to fully supervise the learning of features since the differences between sub-categories might be very small. To this end, we adopt contrastive loss $\mathcal{L}_{con}$ which minimizes the similarity of classification tokens corresponding to different labels and maximizes the similarity of classification tokens of samples with the same label y. To prevent the loss being dominated by easy negatives (different class samples with little similarity), a constant margin α is introduced that only negative pairs with similarity larger than α contribute to the loss $\mathcal{L}_{con}$. Formally, the contrastive loss over a batch of size N is denoted as:

$$\begin{aligned}\mathcal{L}_{con} = \frac{1}{N^2}\sum_{i}^{N}[\sum_{j:y_i=y_j}^{N}(1 - Sim(z_i, z_j)+ \\ \sum_{j:y_i \neq y_j}^{N}\max((Sim(z_i, z_j) - \alpha), 0)]\end{aligned} \tag{9}$$

where z_i and z_j are pre-processed with $l2$ normalization and $Sim(z_i, z_j)$ is the cosine similarity of z_i and z_j.

In summary, our model is trained with the sum of cross-entropy loss L_{cross} and contrastive L_{con} together which can be expressed as:

$$\mathcal{L} = \mathcal{L}_{cross}(y, y') + \mathcal{L}_{con}(z) \tag{10}$$

where $\mathcal{L}_{cross}(y, y')$ is the cross-entropy loss between the predicted label y' and the ground-truth label y.

4. Experiments

In this section, we first introduce the detailed setup including datasets and training hyper-parameters in Section 4.1. Quantitative analysis is given in Section 4.2 follow by ablation studies in Section 4.3. We further give qualitative analysis and visualization results in Section 4.4.

4.1. Experiments Setup

Datasets. We evaluate our proposed TransFG on five widely used fine-grained benchmarks, i.e., CUB-200-2011 [42], Stanford Cars [27], Stanford Dogs [25], NABirds [40] and iNat2017 [22]. The detailed statistics such as category numbers and data splits are summarized below

Table 1: Statistics of fine-grained datasets used in this paper.

Datasets	Category	Training	Testing
CUB-Birds	200	5994	5794
Stanford Dogs	120	12000	8580
Stanford Cars	196	8144	8041
NABirds	555	23929	24633
iNat2017	5089	579184	95986

Implementation details. Unless stated otherwise, we implement TransFG as follows. First, we resize input images to $448 * 448$ except $304 * 304$ on iNat2017 for fair comparison (random cropping for training and center cropping for testing). We split image to patches of size 16 and the step size of sliding window is set to be 12. Thus the H, W, P, S in Eq 1 is 448, 448, 16, 12 respectively. The margin α in Eq 9 is set to be 0.4. Random horizontal flipping and AutoAugment [7] are adopted for data augmentation. We load intermediate weights from official ViT-B_16 model pretrained on ImageNet21k. The batch size is set to 16. SGD optimizer is employed with a momentum of 0.9. The learning rate is initialized as 0.03 except 0.003 for Stanford Dogs dataset and 0.01 for iNat2017 dataset. We adopt cosine annealing as the scheduler of optimizer.

All the experiments are performed with four Nvidia Tesla V100 GPUs using the PyTorch toolbox and APEX with FP16 training.

4.2. Quantitative Analysis

We compare our proposed method TransFG with state-of-the-art works on above mentioned fine-grained datasets. The experiment results on CUB-200-2011 and Stanford Cars are shown in Table 2. From the results, we find that our method outperforms all previous methods on CUB dataset and achieve competitive performance on Stanford Cars.

To be specific, the third column of Table 2 shows the comparison results on CUB-200-2011. Compared to the best result StackedLSTM [17] up to now, our TransFG achieves a **1.3%** improvement on Top-1 Accuracy metric

Table 2: Comparison of different methods on CUB-200-2011, Stanford Cars.

Method	Backbone	CUB	Cars
ResNet-50 [19]	ResNet-50	84.5	-
RA-CNN [14]	VGG-19	85.3	92.5
GP-256 [44]	VGG-16	85.8	92.8
MaxEnt [13]	DenseNet-161	86.6	93.0
DFL-CNN [43]	ResNet-50	87.4	93.1
NTS-Net [49]	ResNet-50	87.5	93.9
Cross-X [30]	ResNet-50	87.7	94.6
DCL [6]	ResNet-50	87.8	94.5
CIN [15]	ResNet-101	88.1	94.5
DBTNet [53]	ResNet-101	88.1	94.5
ACNet [24]	ResNet-50	88.1	94.6
S3N [10]	ResNet-50	88.5	94.7
FDL [29]	DenseNet-161	89.1	94.2
PMG [12]	ResNet-50	89.6	95.1
API-Net [56]	DenseNet-161	90.0	**95.3**
StackedLSTM [17]	GoogleNet	90.4	-
DeiT-B [38]	DeiT-B	-	93.9
ViT [11]	ViT-B_16	90.3	93.7
TransFG	ViT-B_16	**91.7**	94.8

Table 3: Comparison of different methods on Stanford Dogs.

Method	Backbone	Dogs
MaxEnt [13]	DenseNet-161	83.6
FDL [29]	DenseNet-161	84.9
RA-CNN [14]	VGG-19	87.3
SEF [31]	ResNet-50	88.8
Cross-X [30]	ResNet-50	88.9
API-Net [56]	ResNet-101	90.3
ViT [11]	ViT-B_16	91.7
TransFG	ViT-B_16	**92.3**

Table 4: Comparison of different methods on NABirds.

Method	Backbone	NABirds
MaxEnt [13]	DenseNet-161	83.0
Cross-X [30]	ResNet-50	86.4
API-Net [56]	DenseNet-161	88.1
CS-Parts [26]	ResNet-50	88.5
MGE-CNN [52]	ResNet-50	88.6
FixSENet-154 [39]	SENet-154	89.2
ViT [11]	ViT-B_16	89.9
TransFG	ViT-B_16	**90.8**

Table 5: Comparison of different methods on iNat2017.

Method	Backbone	iNat2017
ResNet152 [19]	ResNet152	59.0
SSN [33]	ResNet101	65.2
Huang et al. [23]	ResNet101	66.8
IncResNetV2 [37]	IncResNetV2	67.3
TASN [54]	ResNet101	68.2
ViT [11]	ViT-B_16	68.7
TransFG	ViT-B_16	**71.7**

and 1.4% improvement compared to our base framework ViT [11]. NTS-Net [49] exploits ranking loss to maintain a consistency for learning region features which ignored the global relationship. Multiple ResNet-50 are adopted as multiple branches in [10] which greatly increases the complexity. It is also worth noting that StackLSTM is a very messy multi-stage training model which hampers the availability in practical use, while our TransFG maintains the simplicity of both framework and training strategy.

The fourth column of Table 2 shows the results on Stanford Cars. Our method outperforms most existing methods while performs worse than PMG [12] and API-Net [56] with small margin. We argue that the reason might be the images of Stanford Cars dataset have much simpler and clearer backgrounds than others thus it requires less work on locating discriminative regions for classifying sub-categories. We can observe that most recent methods achieve quite similar results on this dataset. However, even with this property, our TransFG consistently gets **1.1%** improvement compared to the standard ViT model.

The results of experiments on Stanford Dogs are shown in Table 3. Stanford Dogs is a more challenging dataset compared to Stanford Cars with its the more subtle differences between certain species and the large variances of samples from the same category. Only a few methods have tested on this dataset and our TransFG outperforms all of them. API-Net [56] learns to capture the subtle differences by elaborately constructing batches of data and learning the mutual feature vectors and residuals of them. While ViT [11] outperforms other methods by a large margin, our TransFG achieves 92.3% accuracy which outperforms SOTA by **2.0%** with its discriminative part selection and contrastive loss supervision.

NABirds is a much larger birds dataset not only from the side of images numbers but also with 355 more categories which significantly makes the fine-grained visual classification task more challenging. We show our results on it in Table 4. We observe that most methods achieve good results by either exploiting multiple backbones for different branches or adopting quite deep CNN structures to extract better features. While the pure ViT [11] can directly achieve 89.9% accuracy, our TransFG constantly gets 0.9% performance gain compared to ViT and reaches 90.8% accuracy which outperforms SOTA by **1.6%**.

iNat2017 is a large-scale dataset for fine-grained species recognition. Most previous methods do not report results on iNat2017 because of the computational complexity of the multi-crop, multi-scale and multi-stage optimization. With the simplicity of our model pipeline, we are able to scale

Table 6: Ablation study on split way of image patches on CUB-200-2011 dataset.

Method	Patch Split	Accuracy (%)	Training Time (h)
ViT	Non-Overlap	90.3	1.30
ViT	Overlap	**90.5**	3.38
TransFG	Non-Overlap	91.5	1.98
TransFG	Overlap	**91.7**	5.38

Table 7: Ablation study on Part Selection Module (PSM) on CUB-200-2011 dataset.

Method	Accuracy (%)
ViT	90.3
TransFG	**91.0**

Table 8: Ablation study on contrastive loss on CUB-200-2011 dataset.

Method	Contrastive Loss	Acc (%)
ViT		90.3
ViT	✓	**90.7**
TransFG		91.0
TransFG	✓	**91.5**

Table 9: Ablation study on value of margin α on CUB-200-2011 dataset.

Method	Value of α	Accuracy (%)
TransFG	0	91.1
TransFG	0.2	91.4
TransFG	0.4	**91.7**
TransFG	0.6	91.5

TransFG well to big datasets and evaluate the performance which is shown in Table 5. This dataset is very challenging for mining meaningful object parts and the background is very complicated as well. We find that Vision Transformer structure outperforms ResNet structure a lot in these large challenging datasets. ViT outperformes ResNet152 by nearly 10% and similar phenomenon can also be observed in iNat2018 and iNat2019. Our TransFG is the only method to achieve above 70% accuracy with input size of 304 and outperforms SOTA with a large margin of **3.5%**.

4.3. Ablation Study

We conduct ablation studies on our TransFG pipeline to analyze how its variants affect the fine-grained visual classification result. All ablation studies are done on CUB-200-2011 dataset while the same phenomenon can be observed on other datasets as well. We evaluate the influence of the following designs: overlap patches, part selection module, contrastive loss and the results are analyzed in details below.

Influence of image patch split method. We investigate the influence of our overlapping patch split method through experiments with standard non-overlapping patch split. As shown in Table 6, both on the pure Vision Transformer and our improved TransFG framework, the overlapping split method bring consistently improvement, i.e., 0.2% for both frameworks. The additional computational cost introduced by this is also affordable as shown in the fourth column.

Influence of Part Selection Module. As shown in Table 7, by applying the Part Selection Module (PSM) to select discriminative part tokens as the input for the last Transformer layer, the performance of the model improves from 90.3% to 91.0%. We argue that this is because in this way, we sample the most discriminative tokens as input which explicitly throws away some useless tokens and force the network to learn from the important parts.

Influence of contrastive loss. The comparisons of the performance with and without contrastive loss for both ViT and TransFG frameworks are shown in Table 8 to verify the effectiveness of it. We observe that with contrastive loss, the model obtains a big performance gain. Quantitatively, it increases the accuracy from 90.3% to 90.7% for ViT and 91.0% to 91.5% for TransFG. We argue that this is because contrastive loss can effectively enlarge the distance of representations between similar sub-categories and decrease that between the same categories which can be clearly seen in the comparison of confusion matrix in Fig 4.

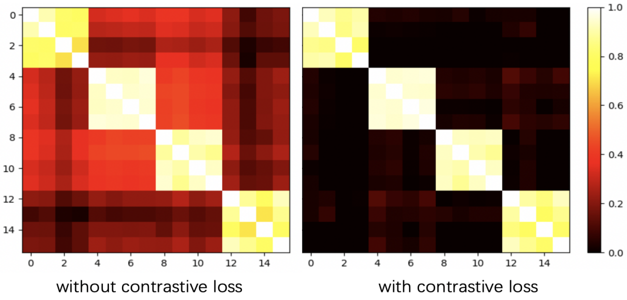

Figure 4: Illustration of contrastive loss. Confusion matrices without and with contrastive loss of a batch with four classes where each contains four samples are shown. The metric of confusion matrix is cosine similarity. Best viewed in color.

Influence of margin α. The results of different setting of the margin α in Eq 9 is shown in Table 9. We find that a small value of α will lead the training signals dominated by easy negatives thus decrease the performance while a high value of α hinder the model to learn sufficient information for increasing the distances of hard negatives. Empirically, we find 0.4 to be the best value of α in our experiments.

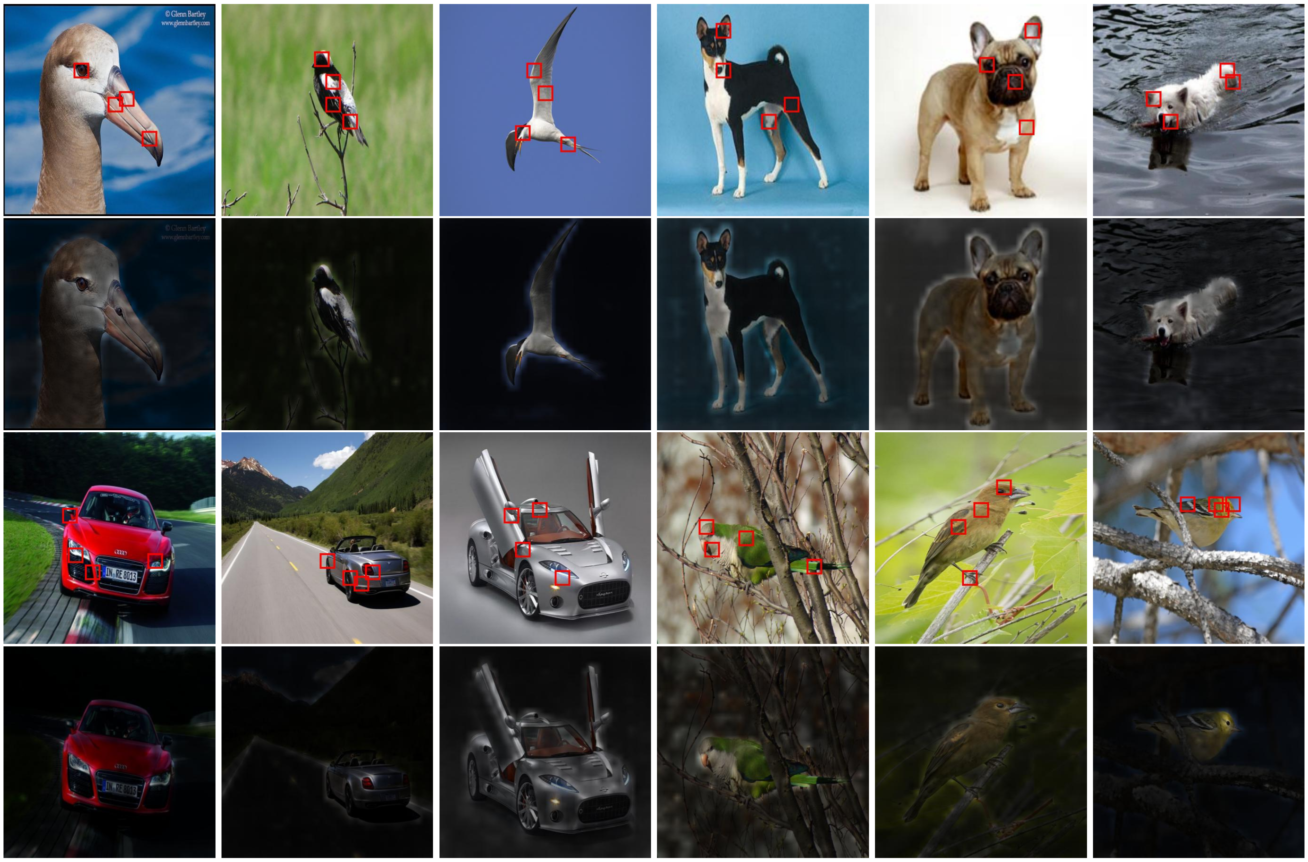

Figure 5: Visualization results of TransFG on CUB-200-2011, Stanford Dogs, Stanford Cars and NABirds datasets. Two kinds of visualization are given, where the first and the third row show the selected Top-4 token positions while the second and fourth rows show the overall global attention maps. Best viewed in color.

4.4. Qualitative Analysis

We show the visualization results of proposed TransFG on the four benchmarks in Fig 5. We randomly sample three images from each dataset. Two kinds of visualizations are presented. The first and the third row of Fig 5 illustrated the selected tokens positions. For better visualization results, we only draw the Top-4 image patches (ranked by the activation value) and enlarge the square of the patches by two times while keep the center positions unchanged. The second and fourth rows show the overall attention map of the whole image where we average the weights of all attention heads to obtain a single attention map. The lighter a region is, the more important it is. From the figure, we can clearly see that our TransFG successfully captures the most important regions for an object, i.e., head, wings, tail for birds; ears, eyes, legs for dogs; lights, doors for cars. At the same time, our overall attention map maps the entire object precisely even in complex backgrounds. See examples from NABirds dataset where birds are sitting on twigs. The bird parts are lighted while the occluder twigs are ignored.

5. Conclusion

In this work, we propose a novel fine-grained visual classification framework TransFG and achieve state-of-the-art results on four common fine-grained benchmarks. We exploit self-attention mechanism to capture the most discriminative regions. Compared to bounding boxes produced by other methods, our selected image patches are much smaller thus becoming more meaningful by showing what regions really contribute to the fine-grained classification. The effectiveness of such small image patches also comes from the Transformer Layer to handle the inner relationships between these regions instead of relying on each of them to produce results separately. Contrastive feature learning is introduced to increase the discriminative ability of the classification tokens. Qualitative visualizations further prove the effectiveness and the interpretability of our method.

With the promising results achieved by TransFG, we believe that the transformer-based models have great potential on fine-grained tasks and our TransFG could be a starting point for future works.

References

[1] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers, 2020. 4

[2] Steve Branson, Grant Van Horn, Serge Belongie, and Pietro Perona. Bird species categorization using pose normalized deep convolutional nets, 2014. 2

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020. 2, 3

[4] Jieneng Chen, Yongyi Lu, Qihang Yu, Xiangde Luo, Ehsan Adeli, Yan Wang, Le Lu, Alan L. Yuille, and Yuyin Zhou. Transunet: Transformers make strong encoders for medical image segmentation, 2021. 2, 3

[5] Wei Chen, Tie-Yan Liu, Yanyan Lan, Zhi-Ming Ma, and Hang Li. Ranking measures and loss functions in learning to rank. *Advances in Neural Information Processing Systems*, 22:315–323, 2009. 1

[6] Yue Chen, Yalong Bai, Wei Zhang, and Tao Mei. Destruction and construction learning for fine-grained image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5157–5166, 2019. 6

[7] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data, 2019. 5

[8] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019. 3

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3

[10] Yao Ding, Yanzhao Zhou, Yi Zhu, Qixiang Ye, and Jianbin Jiao. Selective sparse sampling for fine-grained image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6599–6608, 2019. 1, 2, 6

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. 2, 3, 6

[12] Ruoyi Du, Dongliang Chang, Ayan Kumar Bhunia, Jiyang Xie, Zhanyu Ma, Yi-Zhe Song, and Jun Guo. Fine-grained visual classification via progressive multi-granularity training of jigsaw patches. In *European Conference on Computer Vision*, pages 153–168. Springer, 2020. 6

[13] Abhimanyu Dubey, Otkrist Gupta, Ramesh Raskar, and Nikhil Naik. Maximum-entropy fine grained classification. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 6

[14] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4438–4446, 2017. 6

[15] Yu Gao, Xintong Han, Xun Wang, Weilin Huang, and Matthew Scott. Channel interaction networks for fine-grained image categorization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 10818–10825, 2020. 2, 6

[16] Weifeng Ge, Xiangru Lin, and Yizhou Yu. Weakly supervised complementary parts models for fine-grained image classification from the bottom up. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3034–3043, 2019. 1, 2

[17] Weifeng Ge, Xiangru Lin, and Yizhou Yu. Weakly supervised complementary parts models for fine-grained image classification from the bottom up. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 5, 6

[18] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. Video action transformer network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 244–253, 2019. 3

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. 1, 6

[20] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. Transreid: Transformer-based object re-identification, 2021. 3

[21] Xiangteng He, Yuxin Peng, and Junjie Zhao. Which and how many regions to gaze: Focus discriminative regions for fine-grained visual categorization. *International Journal of Computer Vision*, 127:1–21, 09 2019. 2

[22] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset, 2018. 5

[23] Zixuan Huang and Yin Li. Interpretable and accurate fine-grained recognition via region grouping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8662–8672, 2020. 6

[24] Ruyi Ji, Longyin Wen, Libo Zhang, Dawei Du, Yanjun Wu, Chen Zhao, Xianglong Liu, and Feiyue Huang. Attention convolutional binary neural tree for fine-grained visual categorization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10468–10477, 2020. 6

[25] 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. 5

[26] Dimitri Korsch, Paul Bodesheim, and Joachim Denzler. Classification-specific parts for improving fine-grained visual categorization. In *German Conference on Pattern Recognition*, pages 62–75. Springer, 2019. 6

[27] 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. 1, 5

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 1

[29] Chuanbin Liu, Hongtao Xie, Zheng-Jun Zha, Lingfeng Ma, Lingyun Yu, and Yongdong Zhang. Filtration and distillation: Enhancing region attention for fine-grained visual categorization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11555–11562, 2020. 1, 2, 6

[30] Wei Luo, Xitong Yang, Xianjie Mo, Yuheng Lu, Larry S Davis, Jun Li, Jian Yang, and Ser-Nam Lim. Cross-x learning for fine-grained visual categorization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8242–8251, 2019. 2, 6

[31] Wei Luo, Hengmin Zhang, Jun Li, and Xiu-Shen Wei. Learning semantically enhanced feature for fine-grained image classification. *IEEE Signal Processing Letters*, 27:1545–1549, 2020. 6

[32] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013. 1

[33] Adrià Recasens, Petr Kellnhofer, Simon Stent, Wojciech Matusik, and Antonio Torralba. Learning to zoom: a saliency-based sampling layer for neural networks, 2018. 6

[34] Sofia Serrano and Noah A Smith. Is attention interpretable? *arXiv preprint arXiv:1906.03731*, 2019. 4

[35] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. 1

[36] Peize Sun, Yi Jiang, Rufeng Zhang, Enze Xie, Jinkun Cao, Xinting Hu, Tao Kong, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple-object tracking with transformer, 2020. 3

[37] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017. 6

[38] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention, 2021. 6

[39] Hugo Touvron, Andrea Vedaldi, Matthijs Douze, and Hervé Jégou. Fixing the train-test resolution discrepancy. *arXiv preprint arXiv:1906.06423*, 2019. 6

[40] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 595–604, 2015. 1, 5

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017. 3, 4

[42] 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. 1, 5

[43] Yaming Wang, Vlad I Morariu, and Larry S Davis. Learning a discriminative filter bank within a cnn for fine-grained recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4148–4157, 2018. 6

[44] Xing Wei, Yue Zhang, Yihong Gong, Jiawei Zhang, and Nanning Zheng. Grassmann pooling as compact homogeneous bilinear pooling for fine-grained visual classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 355–370, 2018. 6

[45] Xiu-Shen Wei, Chen-Wei Xie, and Jianxin Wu. Mask-cnn: Localizing parts and selecting descriptors for fine-grained image recognition, 2016. 2

[46] Enze Xie, Wenjia Wang, Wenhai Wang, Peize Sun, Hang Xu, Ding Liang, and Ping Luo. Trans2seg: Transparent object segmentation with transformer, 2021. 2, 3

[47] Shaokang Yang, Shuai Liu, Cheng Yang, and Changhu Wang. Re-rank coarse classification with local region enhanced features for fine-grained image recognition, 2021. 2

[48] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*, 2019. 3

[49] Ze Yang, Tiange Luo, Dong Wang, Zhiqiang Hu, Jun Gao, and Liwei Wang. Learning to navigate for fine-grained classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 420–435, 2018. 1, 2, 6

[50] Chaojian Yu, Xinyi Zhao, Qi Zheng, Peng Zhang, and Xinge You. Hierarchical bilinear pooling for fine-grained visual recognition, 2018. 2

[51] Boxiang Yun, Yan Wang, Jieneng Chen, Huiyu Wang, Wei Shen, and Qingli Li. Spectr: Spectral transformer for hyperspectral pathology image segmentation, 2021. 3

[52] Lianbo Zhang, Shaoli Huang, Wei Liu, and Dacheng Tao. Learning a mixture of granularity-specific experts for fine-grained categorization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8331–8340, 2019. 6

[53] Heliang Zheng, Jianlong Fu, Zheng-Jun Zha, and Jiebo Luo. Learning deep bilinear transformation for fine-grained image representation. *arXiv preprint arXiv:1911.03621*, 2019. 2, 6

[54] Heliang Zheng, Jianlong Fu, Zheng-Jun Zha, and Jiebo Luo. Looking for the devil in the details: Learning trilinear attention sampling network for fine-grained image recognition, 2019. 1, 2, 6

[55] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers, 2020. 2, 3

[56] Peiqin Zhuang, Yali Wang, and Yu Qiao. Learning attentive pairwise interaction for fine-grained classification, 2020. 2, 3, 6