electronics

MDPI

Article

A Systematic Evaluation: Fine-Grained CNN vs. Traditional CNN Classifiers

Saeed Anwar [1,2,*], Nick Barnes [3] and Lars Petersson [4]

[1] Information and Computer Science Department, King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia
[2] SDAIA-KFUPM Joint Research Center for Artificial Intelligence, King Fahd University of Petroleum and Minerals, Dhahran 31261, Saudi Arabia
[3] School of Computing, The Australian National University (ANU), Canberra 2601, Australia; nick.barnes@anu.edu.au
[4] Data61-CSIRO, Black Mountain, Canberra 2601, Australia; lars.petersson@data61.csiro.au
* Correspondence: saeed.anwar@kfupm.edu.sa

Abstract: Fine-grained classifiers collect information about inter-class variations to best use the underlying minute and subtle differences. The task is challenging due to the minor differences between the colors, viewpoints, and structure in the same class entities. The classification becomes difficult and challenging due to the similarities between the differences in viewpoint with other classes and its own. This work investigates the performance of landmark traditional CNN classifiers, presenting top-notch results on large-scale classification datasets and comparing them against state-of-the-art fine-grained classifiers. This paper poses three specific questions. (i) Do the traditional CNN classifiers achieve comparable results to fine-grained classifiers? (ii) Do traditional CNN classifiers require any specific information to improve fine-grained ones? (iii) Do current traditional state-of-the-art CNN classifiers improve the fine-grained classification while utilized as a backbone? Therefore, we train the general CNN classifiers throughout this work without introducing any aspect specific to fine-grained datasets. We show an extensive evaluation on six datasets to determine whether the fine-grained classifier can elevate the baseline in their experiments. We provide ablation studies regarding efficiency, number of parameters, flops, and performance.

Keywords: fine-grained visual classification; traditional classification; systematic evaluation; deep learning; experimental review; baselines

Citation: Anwar, S.; Barnes, N.; Petersson, L. A Systematic Evaluation: Fine-Grained CNN vs. Traditional CNN Classifiers. *Electronics* **2023**, *12*, 4877. https://doi.org/10.3390/electronics12234877

Academic Editor: Luca Mesin

Received: 8 October 2023
Revised: 21 November 2023
Accepted: 1 December 2023
Published: 4 December 2023

1. Introduction

Fine-grained visual classification (FGVC) refers to the task of distinguishing the categories of the same class. Fine-grained classification differs from traditional classification, as the former models intra-class variance, while the latter is about the inter-class difference. Examples of naturally occurring fine-grained classes include birds [1,2], dogs [3], flowers [4], vegetables [5], plants [6], etc., while human-made categories include airplanes [7], cars [8], food [9], etc. Fine-grained classification is helpful in numerous computer vision and image processing applications such as image captioning [10], machine teaching [11], instance segmentation [12], etc.

Fine-grained visual classification is a challenging problem, as there are minute and subtle differences within the species of the same classes, e.g., a crow and a raven, compared to traditional classification, where the difference between the classes is quite visible, e.g., a lion and an elephant. Fine-grained visual classification of species or objects of any category is a Herculean task for human beings and usually requires extensive domain knowledge to correctly identify the species or objects.

As mentioned earlier, fine-grained classification in image space aims to reduce the high intra-class and low inter-class variance. We provide a few sample images from the

dog and bird datasets in Figure 1 to highlight the problem's difficulty. The examples in the figure show the images with the same viewpoint. The colors are also roughly similar. Although the visual variation is minimal between classes, all of these belong to different dog and bird categories. In Figure 2, we provide more examples of the same mentioned categories. Here, the differences in the viewpoint and colors are prominent. The visual variation is more significant than the images in Figure 1, but these belong to the same class.

Many approaches have been proposed to tackle the problem of fine-grained classification; for example, earlier works converged on part detection to model the intra-class variations. Next, the algorithms exploited three-dimensional representations to hand multiple poses and viewpoints to achieve state-of-the-art results. Recently, with the advent of CNNs, most methods have exploited the modeling capacity of CNNs as a component or as a whole.

This paper aims to investigate the capability of traditional CNN networks compared to specially designed fine-grained CNN classifiers. We strive to answer whether current general CNN classifiers can achieve comparable performance to fine-grained ones. To show competitiveness, we employ several fine-grained datasets and report top-1 accuracy for both classifier types. These experiments provide a proper place for general classifiers in fine-grained performance charts and serve as baselines for future comparisons of FGVC problems.

Figure 1. The difference between classes (inter-class variation) is limited for various classes.

Figure 2. The intra-class variation is usually high due to pose, lighting, and color.

Our Contributions: We claim the following contributions in this article.

- We present an overview of Fine-Grained Visual Classification (FGVC) CNN algorithms, which leverage deep learning for nuanced object classification within closely related categories.
- We provide a comprehensive review of state-of-the-art traditional classification algorithms, highlighting their limitations in addressing the subtleties of fine-grained distinctions.
- We systematically compare, investigate, and evaluate traditional classifiers against FGVC methods across six diverse fine-grained datasets, providing insights into their respective performances across varying complexities and offering a valuable resource for benchmarking and further exploration.
- We further provide a forward-looking perspective suggesting a future direction for FGVC algorithms by exploring the integration of traditional classifiers as the backbone.

This paper is organized as follows. Section 2 presents related work about the fine-grained classification networks. Section 3 introduces the traditional state-of-the-art algorithms, which will be compared against fine-grained classifiers. Section 4 shows the experimental settings and datasets for evaluation. Section 5 offers a comparative evaluation between the traditional classifiers and fine-grained classifiers; finally, Section 7 concludes the paper. Train models and codes are available at https://github.com/saeed-anwar/FGSE (accessed on 20 October 2023).

2. Fine-Grained Classifiers

Fine-grained visual classification is an important and well-studied problem. Fine-grained visual classification aims to differentiate between subclasses of the same category instead of the traditional classification problem, where discriminative features are learned to distinguish between classes. Some of the challenges in this domain are the following. (i) The categories are highly correlated, i.e., small differences and small inter-category variance to discriminate between subcategories. (ii) Similarly, the intra-category variation can be significant due to different viewpoints and poses. Many algorithms, such as [13–19], are presented to achieve the desired results. In this section, we highlight the recent approaches. The FGVC research can be divided into the following main branches, reviewed in the paragraphs below.

Part-Based FGVC Algorithms. The part-based category of algorithms relies on the distinguishing features of the objects to leverage the accuracy of visual recognition, which includes [20–25]. These FGVC methods [26,27] aim to learn the distinct features present in different parts of the object, e.g., the differences present in the beak and tail of the bird species. Similarly, the part-based approaches normalize the variation present due to poses and viewpoints. Many works [1,28,29] assume the availability of bounding boxes at the object level and the part level in all the images during the training as well as testing settings. To achieve higher accuracy, Refs. [22,30,31] employed both object-level and part-level annotations. These assumptions restrict the applicability of the algorithms to larger datasets. A reasonable alternative setting would be the availability of a bounding box around the object of interest. Recently, Ref. [21] applied simultaneous segmentation and detection to enhance the performance of segmentation and object part localization. Similarly, a supervised method is proposed [16], which locates the training images similar to a test image using KNN. The object part locations from the selected training images are regressed to the test image.

Bounding Box-Based Methods. The succeeding supervised methods take advantage of the annotated data during the training phase while requiring no knowledge during the testing phase and learning on both object-level and object-part-level annotation in the training phase. This approach is furnished in [32], where only object-level annotations are given during training, while no supervision is provided at the object part level. Similarly, Spatial Transformer Network (STCNN) [33] handles data representation and outputs vital regions' locations. Furthermore, recent approaches focused on removing the limitation of previous works, aiming for conditions where the information about the object part location

is not required in the training or testing phase. These FGVC methods are suitable for deployment on a large scale and help the advancement of research in this direction.

Attention Models. Recently, attention-based algorithms have been employed in FGVC, which focuses on distinguishing parts via an attention mechanism. Using attention, Ref. [25] presented two attention models to learn appropriate patches for a particular object and determine the discriminative object parts using deep CNN. The fundamental idea is to cluster the last CNN feature maps into groups. The object patches and object parts are obtained from the activations of these clustered feature maps. Ref. [25] needs the model to be trained on the category of interest, while we only require the general trained CNN. Similarly, DTRAM [34] learns to end the attention process for each image after a fixed number of steps. Several methods are proposed to take advantage of object parts. However, the most popular one is the deformable part model (DPM) [35], which learns the constellation relative to the bounding box with Support Vector Machines (SVM). Ref. [36] improved upon [37] and employed DPM to localize the parts using the constellation provided by DPM [35]. Navigator–Teacher–Scrutinizer Network (NTSNet) [38] uses informative regions in images without employing any annotations. Another teacher–student network was proposed recently as Trilinear Attention Sampling Network (TASN) [39], composed of a trilinear attention module, attention-based sampler, and a feature distiller.

No Bounding Box Methods. Contrary to utilizing the bounding box annotations, current state-of-the-art methods of fine-grained visual categorization avoid incorporating the bounding boxes during testing and training phases altogether. Refs. [24,40] used a two-stage network for object and object part detection and classification, employing R-CNN and Bilinear CNN, respectively. Part Stacked CNN [18] adopts the same strategy as [24,40] of a two-stage system; however, the difference lies in the stacking of the object parts at the end for classification. Ref. [41] proposed multiple-scale RACNN to acquire distinguishing attention and region feature representations. Moreover, HIHCA [42] incorporated higher-order hierarchical convolutional activations via a kernel scheme.

Distance metric learning Methods. An alternative approach to part-based algorithms is distance learning algorithms, which aim to cluster the data points/objects into the same category while moving different types away from each other. Ref. [43] trained Siamese networks using deep metrics for signature verification and, in this context, set the trend in this direction. Recently, Ref. [44] employed a multi-stage framework that accepts pre-computed feature maps and learns the distance metric for classification. The pre-computed features can be extracted from DeCAF [45], as these features are discriminative and can be used in many tasks for classification. Ref. [46] employs pairwise confusion (PC) via traditional classifiers.

Feature Representation-Based Methods. These methods utilize the features from CNN methods to capture the global information. Many works, including [24,25,32,47], utilized the feature representations of a CNN and employed them in many tasks, such as object detection [48], understanding [49], and recognition [50]. CNN captures global information directly instead of traditional descriptors that capture local information and require manual engineering to encode global representation. Destruction and Construction Learning (DCL) [51] takes advantage of a standard classification network and emphasizes discriminative local details. The model then reconstructs the semantic correlation among local regions. Ref. [49] illustrated the reconstruction of the original image from the activations of the fifth max-pooling layer. Max-pooling ensures invariance to small-scale translation and rotation; however, global spatial information might achieve robustness to larger-scale deformations. Ref. [52] combined the features from fully connected layers using VLAD pooling to capture global information. Similarly, Ref. [53] pooled the features from convolutional layers instead of fully connected layers for text recognition based on the idea that the convolutional layers are transferable and are not domain-specific. Following in the footsteps of [52,53], PDFR by [17] encoded the CNN filter responses, employing a picking strategy via the combination of Fisher Vectors. However, considering feature encoding as an isolated element is not an optimum choice for convolutional neural networks.

Feature Integration Algorithms. Recently, feature integration methods combine features from different layers of the same CNN model. This technique is becoming popular and is adopted by several approaches. The intuition behind feature integration is to take advantage of global semantic information captured by fully connected layers and instance-level information preserved by convolutional layers [54]. Ref. [55] merged the features from intermediate and high-level convolutional activations in their convolutional network to exploit low-level details and high-level semantics for image segmentation. Similarly, for localization and segmentation, Ref. [56] concatenated the feature maps of convolutional layers at a pixel as a vector to form a descriptor. Likewise, for edge detection, Ref. [57] added several feature maps from the lower convolutional layers to guide CNN and predict edges at different scales.

3. Traditional Networks

To make the paper self-inclusive, we briefly provide the basic building blocks of the modern state-of-the-art traditional CNN architectures. These architectures can be broadly categorized into plain, residual, densely connected, inception, and split-attention networks. We review the most prominent and pioneering traditional networks that fall in each mentioned category and then adapt these models for the fine-grained classification task. The five architectures we investigate are VGG [58], ResNet [59], DenseNet [60], Inception [61], and ResNest [62].

3.1. Plain Network

Pioneering CNN architectures such as VGG [58] and AlexNet follow a single path, i.e., without any skip connections. The success of AlexNet [63] inspired VGG. These networks rely on the smaller convolutional filters because a sequence of smaller ones achieves the same performance compared to a larger convolutional filter. For example, when four convolutional layers of 3×3 are stacked together, it has the same receptive field as two 5×5 convolutional layers in sequence. However, the large receptive field has fewer parameters than the smaller ones. The basic building block of the VGG [58] architecture is shown in Figure 3.

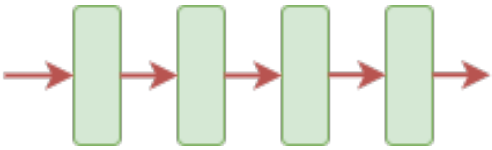

Figure 3. Basic building block of the VGG [58], where no skip connections are used.

VGG [58] has many variants; we use the 19-layer convolutional network, which has shown promising results on ImageNet. As mentioned earlier, the block structure of VGG is planar (without any skip connection), and the number of feature channels is increased from 64 to 512.

3.2. Residual Network

To solve the vanishing gradient problem, the residual network employed network elements with skip connections known as identity shortcuts, as shown in Figure 4a. The pioneering research in this direction is ResNet [59].

The identity shortcuts help to propagate the gradient signal back without being diminished. The identity shortcuts theoretically "skip" over all layers and reach the network's initial layers, learning the task at hand. Because of the summation of features at the end of each module, ResNet [59] learns only an offset, and therefore, it does not require the learning of the full features. The identity shortcuts allow for successful and robust training of much deeper architectures than previously possible. We compare ResNet50 and ResNet152 variants with fine-grained classifiers due to successful classification results.

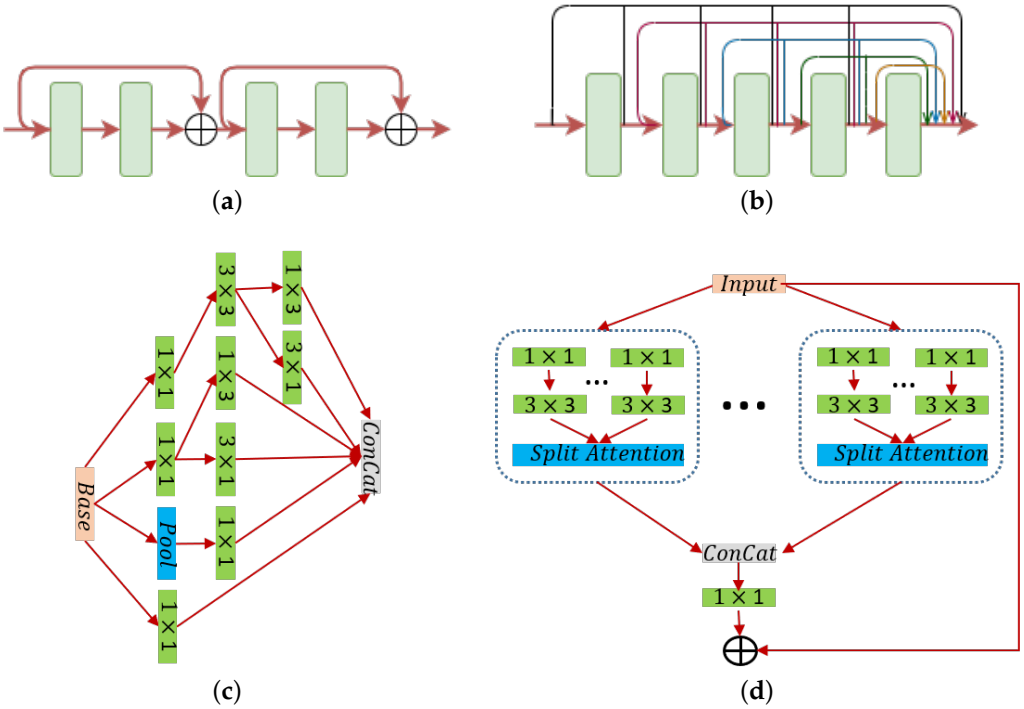

Figure 4. (**a**) ResNet [59] utilize skip connections inside each module. (**b**) Basic block of the DenseNet [60], where each layer gets a connection from previous layers of the block. (**c**) Basic building block of the Inception-v3 [61], where many paths are used for feature extraction and concatenated. (**d**) Basic building block of the ResNest [62], where different paths are used for feature extraction and concatenated.

3.3. Dense Network

Building upon the success of ResNet [59], DenseNet [60] concatenates each convolutional layer in the modules, replacing the expensive element-wise addition and retaining the current features from the previous layers through skipped connections. Furthermore, there is always a path for information from the last layer backward to deal with the gradient diminishing problem. Moreover, to improve computational efficiency, DenseNet [60] utilizes 1×1 convolutional layers to reduce the number of input feature maps before each 3×3 convolutional layer. Transition layers are applied to compress the number of channels that result from the concatenation operations. The building block of DenseNet [60] is shown in Figure 4b.

The performance of DenseNet on ILSVRC is comparable with ResNet. However, it has significantly fewer parameters, thus requiring fewer computations, e.g., DenseNet with 201 convolutional layers with 20 million parameters produces a comparable validation error to a ResNet with 101 convolutional layers having 40 million parameters. Therefore, we consider DenseNet a suitable candidate for fine-grained classification.

3.4. Inception Network

Here, we present Inception-v3 [61], which utilizes label smoothing as a regularization with 7×7 convolution factorization. Similarly, to propagate label information in the deepest parts of the network, Inception-v3 [61] employs an auxiliary classifier along with batch normalization help with sidehead layers. Figure 4c shows the proposed block in the Inception-v3 [61] architecture used on 8×8 grids of the coarsest level to promote high-dimensional representations.

3.5. Split-Attention Network

Lastly, we present the split-attention network in Figure 4d, which employs attention and residual block, called ResNest [62], an extension of the Resnet. The cardinal group

representations are then concatenated along the channel dimension. The final output of other split-attention blocks is produced using a shortcut connection similar to standard residual blocks, considering that the input and output feature maps have the same shape. Moreover, to align the outputs of blocks having a stride, an appropriate transformation is implemented to the shortcut connection, e.g., transformation can be convolution, strided convolution, or convolution with pooling.

4. Experiments

4.1. Experimental Settings

Stochastic Gradient Descent (SGD) [64] optimizer and a decay rate of 10^{-4} are used. We choose a batch size of 32, with an initial learning rate of 0.01 for 200 epochs, where the learning rate is decreased linearly by 0.1 after every 30 epochs for all datasets. The networks are fine-tuned from ImageNet [65] training weights. According to each dataset, the last fully connected layer is also re-mapped from 1k to the number of classes.

4.2. Datasets

This section provides the details of the six most prominent fine-grained datasets used for evaluation and comparison against the current state-of-the-art algorithms.

- Birds: The bird datasets that we compare include Caltech-UCSD Birds-200-2011, abbreviated as CUB [1], composed of 11,788 photographs of 200 categories, which are further divided into 5994 training and 5794 testing images. The second dataset for fine-grained bird classification is North American Birds, generally known as NABirds [2], the largest in this comparison. NABirds [2] has 555 species found in North America, with 48,562 photographs.
- Dogs: The Stanford Dogs [3] is a subset of ImageNet [65] gathered for the task of fine-grained categorization. The dataset is composed of 12k training and 8580 testing images.
- Cars: The cars dataset [8] has 196 classes with different make, model, and year. It has a total number of 16,185 car photographs, where the split is 8144 training images and 8041 testing images, i.e., roughly 50% for both.
- Airplanes: A total of 10,200 images with 102 variants having 100 images for each are present in the fine-grained visual classification of aircraft, i.e., the FGVC-aircraft dataset [7]. Airplanes are an alternative to objects considered for fine-grained categorization, such as birds and pets.
- Flowers: The number of classes in the flower [4] dataset is 102. The training images total 2040, while the testing images total 6149. Furthermore, there are significant variations within categories, while there are similarities to other categories.

Table 1 summarizes the number of classes and the number of images, including the data split for training, testing, and validation (if any) for the fine-grained visualization datasets.

Table 1. Details of six fine-grained visual categorization datasets to evaluate the proposed method.

		No. of Images		
Dataset	**Classes**	**Train**	**Val**	**Test**
NABirds [2]	555	23,929	-	24,633
Dogs [3]	120	12,000	-	8580
CUB [1]	200	5994	-	5794
Aircraft [7]	100	3334	3333	3333
Cars [8]	196	8144	-	8041
Flowers [4]	102	2040	-	6149

5. Evaluations

5.1. Performance on CUB Dataset

We present the comparisons on the CUB dataset [1] in Table 2. The best performer on this dataset is DenseNet, which is unsurprising because the model concatenates the feature maps from preceding layers to preserve details. The worst performing among the traditional classifiers is inception-v3 [61], maybe due to its design, which is more inclined towards a specific dataset (i.e., ImageNet [65]). The ResNet models perform relatively better than NasNet, which shows that networks with shortcut connections surpass in performance those with multi-scale representations for fine-grained classification. DenseNet offers higher accuracy than ResNet because the former does not fuse the feature and carry the details forward, unlike the latter, where the features are combined in each block.

Table 2. Comparison of the state-of-the-art fine-grained classification on CUB [1] dataset.

CNN	Methods	Acc.
Fine-Grained	MGCNN [27]	81.7%
	STCNN [33]	84.1%
	FCAN [66]	84.3%
	PDFR [17]	84.5%
	RACNN [41]	85.3%
	HIHCA [42]	85.3%
	BoostCNN [67]	85.6%
	DTRAM [34]	86.0%
	BilinearCNN [40]	84.1%
	PC-BilinearCNN [46]	85.6%
	PC-DenseCNN [46]	86.7%
	Cui et al. [68]	86.2%
	MACNN [26]	86.5%
	NTSNet [38]	87.5%
	DCL-VGG16 [51]	86.9%
	DCL ResNet50 [51]	87.8%
	TASN [39]	**87.9**%
Traditional	VGG19 [58]	77.8%
	ResNet50 [59]	84.7%
	ResNet152 [59]	85.0%
	Inception-v3 [61]	76.2%
	NasNet [69]	83.0%
	ResNest50 [62]	82.3%
	EfficientNet-B0 [70]	78.0%
	EfficientNet-B4 [70]	84.7%
	EfficientNet-B7 [70]	85.6%
	DenseNet161 [60]	**87.7**%

The fine-grained classification literature considers CUB-200-2011 [1] as a standard benchmark for evaluation; therefore, image-level labels, bounding boxes, and different types of annotations are employed to extract the best results on this dataset. Similarly, multi-branch networks focusing on various parts of images and multiple objective functions are combined for optimization. On the contrary, the traditional classifiers [59,60] use a single loss without any extra information or other annotations. The best performing fine-grained classifiers for CUB [1] are DCL ResNet50 [51], TASN [39], and NTSNet [38], where merely 0.1% and 0.2% gain is recorded over DenseNet [60] for [51] and [39], respectively. Furthermore, NTSNet [38] lags by a margin of 0.2%. The improvement over DenseNet is insignificant, keeping in mind the different computationally expensive tactics employed to learn the distinguishable features by fine-grained classifiers.

5.2. Quantitative Analysis on Aircraft and Cars

Table 3 shows the performances of fine-grained classifiers on car and aircraft datasets. Here, we also observe that the performance of the traditional classifiers is better than the fine-grained classifiers. DenseNet161 has an improvement of about 1.5% and 3% on aircraft [7] compared to best performing NTSNet [38] and MACNN [26], respectively. Similarly, an improvement of 0.6% and 1.4% is recorded against NTSNet [38] and DTRAM [34] on the car dataset, respectively. The fine-grained classifiers such as [34,38,39] fail to achieve the same accuracy as the traditional classifiers, although the former employ more image-specific information for learning.

Table 3. Experimental results on FGVC aircraft [7] and cars [8].

		Datasets	
CNN	**Methods**	**Aircraft**	**Cars**
Fine-Grained	FVCNN [71]	81.5%	-
	FCAN [66]	-	89.1%
	BilinearCNN [40]	84.1%	91.3%
	RACNN [41]	88.2%	92.5%
	HIHCA [42]	88.3%	91.7%
	BoostCNN [67]	88.5%	92.1%
	Cui et al. [68]	88.5%	92.4%
	PC-BilinearCNN [46]	85.8%	92.5%
	PC-ResCNN [46]	83.4%	93.4%
	PC-DenseCNN [46]	89.2%	92.7%
	MACNN [26]	89.9%	92.8%
	DTRAM [34]	-	93.1%
	TASN [39]	-	93.8%
	NTSNet [38]	91.4%	93.9%
Traditional	VGG19 [58]	85.7%	80.5%
	ResNet50 [59]	91.4%	91.7%
	ResNet152 [59]	90.7%	93.2%
	NasNet [69]	88.5%	-
	Inception-v3 [61]	85.4%	85.8%
	ResNest50 [62]	89.9%	89.6%
	EfficientNet-B0 [70]	80.9%	82.8%
	EfficientNet-B4 [70]	86.8%	86.9%
	EfficientNet-B7 [70]	92.0%	90.2%
	DenseNet161 [60]	**92.9**%	**94.5**%

5.3. Comparison on Stanford Dogs

The Stanford dogs [3] is another challenging dataset where the performance is compared in Table 4. Here, we utilize ResNet and DenseNet from the traditional ones. The performance of ResNet, composed of 152 layers, is similar to DenseNet with 161 layers; both achieved 85.2% accuracy, which is 1.4% higher than PC-DenseCNN [46], the best performing method in fine-grained classifiers. This experiment suggests that incorporating traditional classifiers in the fine-grained ones requires more insight than just utilizing them in the framework. It is also worth mentioning that some of the fine-grained classifiers employ a large amount of data from other sources in addition to the Stanford dogs training data.

5.4. Results of Flower Dataset

The accuracy of DenseNet on the flower dataset [4] is 98.1%, which is around 5.5% higher as compared to the second best performing state-of-the-art method (PC-ResCNN [46]) in Table 4. Similarly, the other traditional classifiers outperform the fine-grained ones significantly. It should also be noted that the performance on this dataset is approaching saturation.

Table 4. Comparison of the state-of-the-art fine-grained classification on dogs [3], flowers [4], and NABirds [2] dataset.

		Datasets		
CNN	**Methods**	**Dogs**	**Flowers**	**NABirds**
Fine-Grained	Zhang et al. [72]	80.4%	-	-
	Krause et al. [73]	80.6%	-	-
	Det.+Seg. [74]	-	80.7%	-
	Overfeat [50]	-	86.8%	-
	Branson et al. [47]	-	-	35.7%
	Van et al. [2]	-	-	75.0%
	BilinearCNN [40]	82.1%	92.5%	80.9%
	PC-ResCNN [46]	73.4%	93.5%	68.2%
	PC-BilinearCNN [46]	83.0%	93.7%	82.0%
	PC-DenseCNN [46]	83.8%	91.4%	82.8%
Traditional	VGG19 [58]	76.7%	88.73%	74.9%
	ResNet50 [59]	83.4%	97.2%	79.6%
	ResNet152 [59]	85.2%	97.5%	84.0%
	Inception-v3 [61]	85.8%	93.3%	68.4%
	ResNest50 [62]	87.7%	94.7%	80.4%
	EfficientNet-B0 [70]	84.9%	91.4%	63.7%
	EfficientNet-B4 [70]	92.4%	92.8%	77.0%
	EfficientNet-B7 [70]	**93.6**%	96.2%	-
	DenseNet161 [60]	85.2%	**98.1%**	**86.3**%

5.5. Performance on NABirds

Relatively fewer methods have reported their results on this dataset. However, for the sake of completeness, we provide comparisons on the NABirds [2] dataset. Again, the leading performance on NABirds is achieved by DenseNet161, followed by ResNet152. The third best performer is a fine-grained classifier, i.e., PC-DenseCNN [46], which internally employs DenseNet161, lagging by 3.5%. This shows the superior performance of the traditional CNN classifiers against state-of-the-art fine-grained CNN classifiers.

5.6. Ablation Studies

Fine-tuned vs. Scratch: Here, we present two strategies for training traditional CNN classification networks, i.e., fine-tuning the weights via ImageNet [65] and training from scratch (randomly initializing the weights) for the car dataset. The accuracy presented for each is given in Table 5. The ResNet50 achieves higher accuracy when fine-tuned as compared to the randomly initialized version. Similarly, ResNet152 performed better for the fine-tuned network but failed when trained from scratch. The reason may be due to a large number of parameters and smaller training data.

Table 5. Different strategies for initialing the network weights, i.e., fine-tuning from ImageNet and random initialization (scratch) for car [8] dataset.

Initial	**Methods**	
Weights	**ResNet50**	**ResNet152**
Scratch	83.4%	36.9%
Fine-tune	91.7%	93.2%

Backbone Improvement Over Standalone Classifiers: Some fine-grained state-of-the-art methods use ResNet50 as the backbone and achieve higher accuracy than the standalone ResNet50. To be precise, Table 6 shows the backbones used by state-of-the-art methods in their algorithm. One can observe that many algorithms employ the same backbones more than once, increasing the overhead and doubling or tripling the number of param-

eters. Besides utilizing traditional classifiers as backbones, state-of-the-art fine-grained methods rely on specialized techniques to extract fine-grained features, hence adding more parameters and computation. Therefore, the improvement achieved by the state-of-the-art fine-grained methods comes at the cost of extra considerations and the number of parameters, while the traditional classifier, like DenseNet, does not require such tricks to achieve the same accuracy.

Table 6. The comparison of backbone and number of parameters in fine-grained methods regarding classification accuracy on the CUB dataset. The input to all methods is 448 × 448.

Methods	Backbone	Parameters	Accuracy
MGCNN [27]	3× VGG16	429 M	81.7
STCNN [33]	3× Inception-v2	71.5 M	84.1
RA-CNN [42]	3× VGG19	429 M	85.3
MACNN [26]	3× VGG19	144 M	85.4
TASN [39]	1× VGG19	140 M	87.1
MAMC [75]	1× Resnet50	434 M	86.5
NTSNet [38]	3× Resnet50	25.5 M	87.3
TASN [39]	1× Resnet50	35.2 M	87.9
DenseNet [60]	1× DenseNet161	28.7 M	87.7

Parameters, FLOPs, and Performance: We provide comparisons in terms of the number of parameters, FLOPs, and performance on the ImageNet for the traditional classifiers employed in our experiments in Table 7. The ResNet50 [59] approximately has the same parameters as DenseNet161 [60] numerically, but the performance of DenseNet161 [60] is much higher than ResNet50 [59]. It should also be noted that DenseNet169 and DenseNet201 have fewer parameters but higher performance on imageNet; hence, we argue that backbones in the fine-grained methods should be updated to appropriate ones, as suggested by our experimental analysis.

Table 7. Traditional classifier comparison on ImageNet [65] regarding the number of parameters, FLOPS, and accuracy.

	No. of		Accuracy	
Methods	**Parameters**	**FLOPS**	**Top-1**	**Top-5**
ResNet18 [59]	11.69 M	1819.06 M	69.76%	89.08%
ResNet34 [59]	21.97 M	3671.26 M	73.3%	91.42%
ResNet50 [59]	25.60 M	4111.51 M	76.15%	92.87%
ResNet101 [59]	44.60 M	7833.97 M	77.37%	93.56%
ResNet152 [59]	60.20 M	11,558.83 M	78.31%	94.06%
Densenet121 [60]	7.98 M	2865.67 M	74.65%	92.17%
Densenet161 [60]	28.68 M	7787.01 M	77.65%	93.80%
Densenet169 [60]	14.15 M	3398.07 M	76.00%	93.00%
Densenet201 [60]	20.01 M	4340.97 M	77.20%	93.57%
Inception-v3 [61]	23.83 M	5731.28 M	77.45%	93.56%

6. Discussions

Based on the results obtained in our experiments, we would like to answer the three questions raised in the abstract. We observed that the results obtained by the state-of-the-art traditional CNN classifiers are comparable to the fine-grained classifiers. This is because the fine-grained classifiers employ the basic version of the state-of-the-art traditional CNN classifiers, such as Resnet18, ignoring the higher counterparts that provide sophisticated results. Furthermore, the traditional classifiers can focus on the minute differences between the images and learn from them; however, it is also essential to note that these differences can be highlighted via techniques employed in fine-grained classification. As a final

observation, we concluded that the traditional backbones help improve the fine-grained classification as the number of images for fine-grained datasets is limited; hence, a pre-trained model will help improve the results, as shown in Table 5.

7. Conclusions

In this paper, we compared state-of-the-art traditional CNN classifiers and fine-grained CNN classifiers. It has been shown that conventional models achieve state-of-the-art performance on fine-grained classification datasets and outperform the fine-grained CNN classifiers or achieve similar results on the fine-grained datasets. Therefore, updating the baselines for comparisons in the fine-grained CNN classifiers is necessary to take full advantage of traditional CNN classifiers. Based on our ablation studies, it is also important to note that the performance increase is due to the initial weights trained on the ImageNet [65] datasets. Furthermore, we have established that the DenseNet161 model achieves state-of-the-art or similar results to fine-grained classifiers for all datasets without adding significant overhead; hence, DenseNet161 can be considered a better backbone than those employed in fine-grained classifiers.

Author Contributions: Conceptualization, S.A.; methodology, S.A., N.B. and L.P.; validation, S.A. and N.B.; formal analysis, S.A.; investigation, S.A.; resources, N.B. and L.P.; data curation, S.A.; writing, original draft preparation, S.A.; writing, review and editing—N.B. and L.P.; visualization, S.A.; supervision, N.B. and L.P.; project administration, N.B.; All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: https://github.com/saeed-anwar/FGSE (accessed on 20 October 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wah, C.; Branson, S.; Welinder, P.; Perona, P.; Belongie, S. *The Caltech-UCSD Birds-200-2011 Dataset*; Technical Report; California Institute of Technology: Pasadena, CA, USA, 2011.
2. Van Horn, G.; Branson, S.; Farrell, R.; Haber, S.; Barry, J.; Ipeirotis, P.; Perona, P.; Belongie, S. 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
3. Khosla, A.; Jayadevaprakash, N.; Yao, B.; Li, F.F. Novel dataset for fgvc: Stanford dogs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshop, Colorado Springs, CO, USA, 20–25 June 2011.
4. Nilsback, M.E.; Zisserman, A. Automated flower classification over a large number of classes. In Proceedings of the Indian Conference on Vision Graphics and Image Processing, Bhubaneswar, India, 16–19 December 2008.
5. Hou, S.; Feng, Y.; Wang, Z. Vegfru: A domain-specific dataset for fine-grained visual categorization. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
6. Wegner, J.D.; Branson, S.; Hall, D.; Schindler, K.; Perona, P. Cataloging public objects using aerial and street-level images-urban trees. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
7. Maji, S.; Rahtu, E.; Kannala, J.; Blaschko, M.; Vedaldi, A. Fine-grained visual classification of aircraft. *arXiv* **2013**, arXiv:1306.5151.
8. Krause, J.; Stark, M.; Deng, J.; Fei-Fei, L. 3d object representations for fine-grained categorization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Worshops, Sydney, Australia, 2–8 December 2013.
9. Chen, M.; Dhingra, K.; Wu, W.; Yang, L.; Sukthankar, R.; Yang, J. PFID: Pittsburgh fast-food image dataset. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, 7–10 November 2009.
10. Aafaq, N.; Mian, A.; Liu, W.; Gilani, S.Z.; Shah, M. Video description: A survey of methods, datasets, and evaluation metrics. *ACM Comput. Surv. (CSUR)* **2019**, *52*, 115. [CrossRef]
11. Spivak, G.C. *Outside in the Teaching Machine*; Routledge: London, UK, 2012.
12. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
13. Angelova, A.; Zhu, S.; Lin, Y. Image segmentation for large-scale subcategory flower recognition. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Clearwater Beach, FL, USA, 15–17 January 2013.
14. Chai, Y.; Rahtu, E.; Lempitsky, V.; Van Gool, L.; Zisserman, A. Tricos: A tri-level class-discriminative co-segmentation method for image classification. In Proceedings of the European Conference on Computer Vision (ECCV), Florence, Italy, 7–13 October 2012.

15. Deng, J.; Krause, J.; Fei-Fei, L. Fine-grained crowdsourcing for fine-grained recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013.
16. Gavves, E.; Fernando, B.; Snoek, C.G.; Smeulders, A.W.; Tuytelaars, T. Fine-grained categorization by alignments. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Sydney, Australia, 1–8 December 2013.
17. Zhang, X.; Xiong, H.; Zhou, W.; Lin, W.; Tian, Q. Picking deep filter responses for fine-grained image recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
18. Huang, S.; Xu, Z.; Tao, D.; Zhang, Y. Part-stacked cnn for fine-grained visual categorization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
19. Zhang, X.; Zhou, F.; Lin, Y.; Zhang, S. Embedding label structures for fine-grained feature representation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
20. Yang, S.; Bo, L.; Wang, J.; Shapiro, L.G. Unsupervised template learning for fine-grained object recognition. *Adv. Neural Inf. Process. Syst. (NIPS)* **2012**, *25*, 3122–3130.
21. Chai, Y.; Lempitsky, V.; Zisserman, A. Symbiotic segmentation and part localization for fine-grained categorization. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Sydney, Australia, 1–8 December 2013.
22. Berg, T.; Belhumeur, P. Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013.
23. Zhang, N.; Farrell, R.; Iandola, F.; Darrell, T. Deformable part descriptors for fine-grained recognition and attribute prediction. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Sydney, Australia, 1–8 December 2013.
24. Zhang, N.; Donahue, J.; Girshick, R.; Darrell, T. Part-based R-CNNs for fine-grained category detection. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014.
25. Xiao, T.; Xu, Y.; Yang, K.; Zhang, J.; Peng, Y.; Zhang, Z. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
26. Zheng, H.; Fu, J.; Mei, T.; Luo, J. Learning multi-attention convolutional neural network for fine-grained image recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
27. Wang, D.; Shen, Z.; Shao, J.; Zhang, W.; Xue, X.; Zhang, Z. Multiple granularity descriptors for fine-grained categorization. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
28. Parkhi, O.M.; Vedaldi, A.; Jawahar, C.; Zisserman, A. The truth about cats and dogs. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Barcelona, Spain, 6–13 November 2011.
29. Parkhi, O.M.; Vedaldi, A.; Zisserman, A.; Jawahar, C. Cats and dogs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012.
30. Liu, J.; Kanazawa, A.; Jacobs, D.; Belhumeur, P. Dog breed classification using part localization. In Proceedings of the European Conference on Computer Vision (ECCV), Florence, Italy, 7–13 October 2012.
31. Xie, L.; Tian, Q.; Hong, R.; Yan, S.; Zhang, B. Hierarchical part matching for fine-grained visual categorization. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Sydney, NSW, Australia, 1–8 December 2013.
32. Krause, J.; Jin, H.; Yang, J.; Fei-Fei, L. Fine-grained recognition without part annotations. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
33. Jaderberg, M.; Simonyan, K.; Zisserman, A.; Kavukcuoglu, K. Spatial transformer networks. *Adv. Neural Inf. Process. Syst. (NIPS)* **2015**, *28*, 2017–2025.
34. Li, Z.; Yang, Y.; Liu, X.; Zhou, F.; Wen, S.; Xu, W. Dynamic computational time for visual attention. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
35. Felzenszwalb, P.; McAllester, D.; Ramanan, D. A discriminatively trained, multiscale, deformable part model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Anchorage, AK, USA, 23–28 June 2008.
36. Simon, M.; Rodner, E. Neural activation constellations: Unsupervised part model discovery with convolutional networks. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
37. Simon, M.; Rodner, E.; Denzler, J. Part detector discovery in deep convolutional neural networks. In Proceedings of the Asian Conference on Computer Vision, Singapore, 1–5 November 2014.
38. Yang, Z.; Luo, T.; Wang, D.; Hu, Z.; Gao, J.; Wang, L. Learning to Navigate for Fine-grained Classification. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
39. Zheng, H.; Fu, J.; Zha, Z.J.; Luo, J. Looking for the Devil in the Details: Learning Trilinear Attention Sampling Network for Fine-grained Image Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
40. Lin, T.Y.; RoyChowdhury, A.; Maji, S. Bilinear cnn models for fine-grained visual recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
41. Fu, J.; Zheng, H.; Mei, T. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.

42. Cai, S.; Zuo, W.; Zhang, L. Higher-order integration of hierarchical convolutional activations for fine-grained visual categorization. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
43. Bromley, J.; Guyon, I.; LeCun, Y.; Säckinger, E.; Shah, R. Signature verification using a "siamese" time delay neural network. *Adv. Neural Inf. Process. Syst. (NIPS)* **1994**, *6*, 737–744.
44. Qian, Q.; Jin, R.; Zhu, S.; Lin, Y. Fine-grained visual categorization via multi-stage metric learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
45. Donahue, J.; Jia, Y.; Vinyals, O.; Hoffman, J.; Zhang, N.; Tzeng, E.; Darrell, T. Decaf: A deep convolutional activation feature for generic visual recognition. *Proc. Int. Conf. Mach. Learn.* **2014**, *32*, 647–655.
46. Dubey, A.; Gupta, O.; Guo, P.; Raskar, R.; Farrell, R.; Naik, N. Pairwise confusion for fine-grained visual classification. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.
47. Branson, S.; Van Horn, G.; Belongie, S.; Perona, P. Bird species categorization using pose normalized deep convolutional nets. *arXiv* **2014**, arXiv:1406.2952.
48. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014.
49. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014.
50. Sharif Razavian, A.; Azizpour, H.; Sullivan, J.; Carlsson, S. CNN features off-the-shelf: An astounding baseline for recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Columbus, OH, USA, 23–29 June 2014.
51. Chen, Y.; Bai, Y.; Zhang, W.; Mei, T. Destruction and Construction Learning for Fine-grained Image Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019.
52. Gong, Y.; Wang, L.; Guo, R.; Lazebnik, S. Multi-scale orderless pooling of deep convolutional activation features. In Proceedings of the European Conference on Computer Vision (ECCV), Zurich, Switzerland, 6–12 September 2014.
53. Cimpoi, M.; Maji, S.; Vedaldi, A. Deep filter banks for texture recognition and segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
54. Babenko, A.; Lempitsky, V. Aggregating local deep features for image retrieval. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
55. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
56. Hariharan, B.; Arbeláez, P.; Girshick, R.; Malik, J. Hypercolumns for object segmentation and fine-grained localization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
57. Xie, S.; Tu, Z. Holistically-nested edge detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015.
58. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2015**, arXiv:1409.1556.
59. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
60. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2017.
61. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
62. Zhang, H.; Wu, C.; Zhang, Z.; Zhu, Y.; Zhang, Z.; Lin, H.; Sun, Y.; He, T.; Muller, J.; Manmatha, R.; et al. ResNeSt: Split-Attention Networks. *arXiv* **2020**, arXiv:2004.08955.
63. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst. (NIPS)* **2012**, *25*, 1097–1105. [CrossRef]
64. Bottou, L. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT*; Springer: Berlin/Heidelberg, Germany 2010; pp. 177–186.
65. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 20–25 June 2009.
66. Liu, X.; Xia, T.; Wang, J.; Yang, Y.; Zhou, F.; Lin, Y. Fully convolutional attention networks for fine-grained recognition. *arXiv* **2016**, arXiv:1603.06765.
67. Moghimi, M.; Belongie, S.J.; Saberian, M.J.; Yang, J.; Vasconcelos, N.; Li, L.J. Boosted Convolutional Neural Networks. In Proceedings of the British Machine Vision Conference, York, UK, 19–22 September 2016.
68. Cui, Y.; Zhou, F.; Wang, J.; Liu, X.; Lin, Y.; Belongie, S. Kernel pooling for convolutional neural networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
69. Zoph, B.; Vasudevan, V.; Shlens, J.; Le, Q.V. Learning transferable architectures for scalable image recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018.
70. Tan, M.; Le, Q. Efficientnet: Rethinking model scaling for convolutional neural networks. *Proc. Int. Conf. Mach. Learn.* **2019**, *97*, 6105–6114.

71. Gosselin, P.H.; Murray, N.; Jégou, H.; Perronnin, F. Revisiting the fisher vector for fine-grained classification. *Pattern Recognit. Lett.* **2014**, *49*, 92–98. [CrossRef]
72. Zhang, Y.; Wei, X.S.; Wu, J.; Cai, J.; Lu, J.; Nguyen, V.A.; Do, M.N. Weakly supervised fine-grained categorization with part-based image representation. *IEEE Trans. Image Process.* **2016**, *25*, 1713–1725. [CrossRef] [PubMed]
73. Krause, J.; Sapp, B.; Howard, A.; Zhou, H.; Toshev, A.; Duerig, T.; Philbin, J.; Li, F.-F. The unreasonable effectiveness of noisy data for fine-grained recognition. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016.
74. Angelova, A.; Zhu, S. Efficient object detection and segmentation for fine-grained recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013.
75. Sun, M.; Yuan, Y.; Zhou, F.; Ding, E. Multi-attention multi-class constraint for fine-grained image recognition. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 805–821.