Multi-View Active Fine-Grained Visual Recognition

Ruoyi Du[1], Wenqing Yu[1], Heqing Wang[1], Ting-En Lin[2], Dongliang Chang[1]*, Zhanyu Ma[1]
[1]Beijing University of Posts and Telecommunications, China
[1]{duruoyi, yuwenqing, wangheqing, changdongliang, mazhanyu}@bupt.edu.cn,
[2]sss950123@gmail.com

Abstract

Despite the remarkable progress of Fine-grained visual classification (FGVC) with years of history, it is still limited to recognizing 2D images. Recognizing objects in the physical world (i.e., 3D environment) poses a unique challenge – discriminative information is not only present in visible local regions but also in other unseen views. Therefore, in addition to finding the distinguishable part from the current view, efficient and accurate recognition requires inferring the critical perspective with minimal glances. E.g., a person might recognize a "Ford sedan" with a glance at its side and then know that looking at the front can help tell which model it is. In this paper, towards FGVC in the real physical world, we put forward the problem of multi-view active fine-grained visual recognition (MAFR) and complete this study in three steps: (i) a multi-view, fine-grained vehicle dataset is collected as the testbed, (ii) a pilot experiment is designed to validate the need and research value of MAFR, (iii) a policy-gradient-based framework along with a dynamic exiting strategy is proposed to achieve efficient recognition with active view selection. Our comprehensive experiments demonstrate that the proposed method outperforms previous multi-view recognition works and can extend existing state-of-the-art FGVC methods and advanced neural networks to become "FGVC experts" in the 3D environment. Our code is available at https://github.com/PRIS-CV/MAFR.

1. Introduction

In the past two decades, fine-grained visual classification (FGVC) has made significant progress in recognizing sub-categories of objects belonging to the same class. This progress has been demonstrated in various domains, such as recognizing cars [32, 57], aircraft [36], birds [50, 48], and foods [39], with extensive outstanding works surpassing human experts in many application scenarios [34, 56, 19, 53, 13, 4, 5, 16, 14]. However, the previous efforts on FGVC have remained mainly limited to a single-view-based paradigm, where only the visual content within a single static image is considered. This paradigm may be sufficient for coarse-grained classification where inter-class differences are easily captured, such as distinguishing a coupe from other vehicles by its streamlined body, seductive engine, or headlamps. However, fine-grained classification presents a different challenge where discriminative clues are rare and often found in subtle structural differences that are not easily captured by a single static view. For instance, to distinguish between different Ford sedans, one can only rely on subtle differences in the design of car headlights. Predictably, for single-view-based approaches, an image/view without discriminative clues is completely indistinguishable at the fine-grained level, fundamentally limiting the model's theoretical performance.

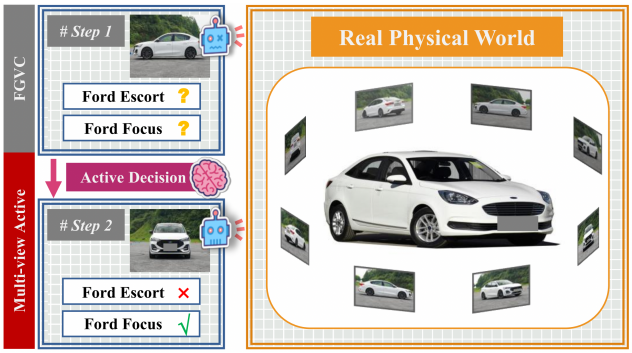

Figure 1. Conventional FGVC *versus* Multi-view Active FGVC. Conventional FGVC takes a single static image for inference, which may omit the discriminative information. In contrast, Multi-view FGVC takes a step further by predicting the potential discriminative view and fusing a sequence of visual information for the final decision.

Factually, visual recognition is never limited to observing 2D environments and processing static images. Vision algorithms equipped by portal devices (*e.g.*, smartphone, smart glasses, etc.) or embodied AI agents [18] (*e.g.*, intelligent robots) play the core roles during machine-

*indicates the corresponding author.

environment interaction and have become one of the focuses of computer vision research. Therefore, to embrace the new trend, a natural extension of ordinary FGVC follows – in addition to locating discriminative parts within an image, we aim to infer the unseen distinguishable perspective within the physical world (*i.e.*, 3D environment). Figure 1 shows an example where, with a single glance from the front, the algorithm may be uncertain about the model of the Ford sedan, but can infer that looking at its front will help to resolve the ambiguity.

Specifically, we re-propose the concept of active vision [1] in the context of FGVC termed multi-view active fine-grained visual recognition (MAFR). MAFR is based on two essential hypotheses. Firstly, we hypothesize that discriminative information for different fine-grained categories is distributed across various object views, making discriminative perspective inference a non-trivial and worth studying. Secondly, we hypothesize that even views that appear indistinguishable at first glance can contain visual clues that can help infer the discriminative perspective, making the problem solvable.

To start with, due to the absence of qualified datasets, we first collect a fine-grained, multi-view vehicle dataset named Multi-view Cars (MvCars) as the testbed. MvCars comprises 233 car models from 35 brands, covering 5 different car types. Each car in the dataset has 8 aligned views, and the dataset contains more than $26,000$ images. In Section 3.2, we conducted a simple pilot experiment and verified our first hypothesis, which suggests that MvCars is well-suited for the problem at hand.

Secondly, our next contribution is an efficient multi-view fine-grained recognition framework building upon active next-view selection and dynamic exiting. In particular, following the general idea of view-based 3D object understanding [47], an extraction-aggregation architecture is designed as the feature encoder, where a convolutional neural network (*e.g.*, ResNet50 [25]) is first applied to extract single-view features independently. Then a recurrent neural network (*e.g.*, GRU [11]) is adopted to aggregate multi-view features and form global descriptions. Afterwards, we formulate the next-view selection as a sequential decision process, where the model is demanded to *decide the next discriminative view* (action) according to *previously observed views* (state). Therefore, we implement a proximal policy optimization (PPO) [46] based multi-stage training strategy. In addition, considering that the difficulty of recognizing different samples varies and the resulting number of perspectives required differs, we design a dynamic exit inference strategy for better effort allocation – inspired by the idea of budgeted batch classification, a series of exciting confidence thresholds are estimated under a given effort expectation. Note that the proposed framework does not rely on specific neural network architectures. It can extend any visual recognition network to an *FGVC experts* in the 3D environment.

Finally, several carefully designed baselines are reproduced on MvCars as benchmark results, including previous multi-view recognition works, advanced neural networks, and popular FGVC methods. The experimental results demonstrate that the proposed method delivers a better performance-efficient trade-off than all competitors. After that, analysis of the upper bound and the selected trajectories reveals the inherent characteristics of MAFR. In addition, comprehensive ablation studies are carried out to verify the necessities of each model component.

2. Related Work

2.1. Fine-Grained Visual Classification

Due to the inherent subtle inter-class variance and the relatively large intra-class variance, fine-grained visual classification is much more challenging than ordinary coarse-grained classification. With vigorous efforts made by researchers, great progress has been made in many directions. Localization-based approaches [61, 31, 56, 19, 53, 6] that explicitly locate discriminative parts for feature extraction to alleviate the intra-class variance. High-order encoding methods [34, 20, 19, 59, 62, 21] that adopt high-order feature interactions for better representation ability that can capture the subtle differences. Chen *et al.* [9] and Du *et al.* [13] train the model with jigsaw patches to implicitly encourage knowledge mining from local regions. Recently, Chang *et al.* [5] leverage the underlying hierarchical structure of fine-grained categories to achieve user-friendly outputs and better performance.

Except for good performances being brought, these works above also reveal that FGVC is never just a more challenging classification problem but a stand-alone field that requires well-directed research. In this paper, to further broaden the horizon of FGVC, we propose the multi-view active fine-grained visual recognition (MAFR) task aiming at effectively recognizing fine-grained categories in the 3D environment along with a targeted dataset. It is worth noting that the CompCars [57] dataset also provides a car dataset with view annotations. However, its multi-view images are taken from different samples, making it less suitable for the raised problem.

2.2. Multi-View Recognition

Elsewhere for common object recognition problems, particular progress has been made to recognizing 3D objects with three trends that can be summarized [7]: point-based methods [42, 44, 2, 41], volume-based methods [37, 55, 43, 38], and view-based methods [10, 47, 29, 30, 60, 58]. Among them, point-based and volume-based approaches demand to perceive the 3D structure of objects via lidar,

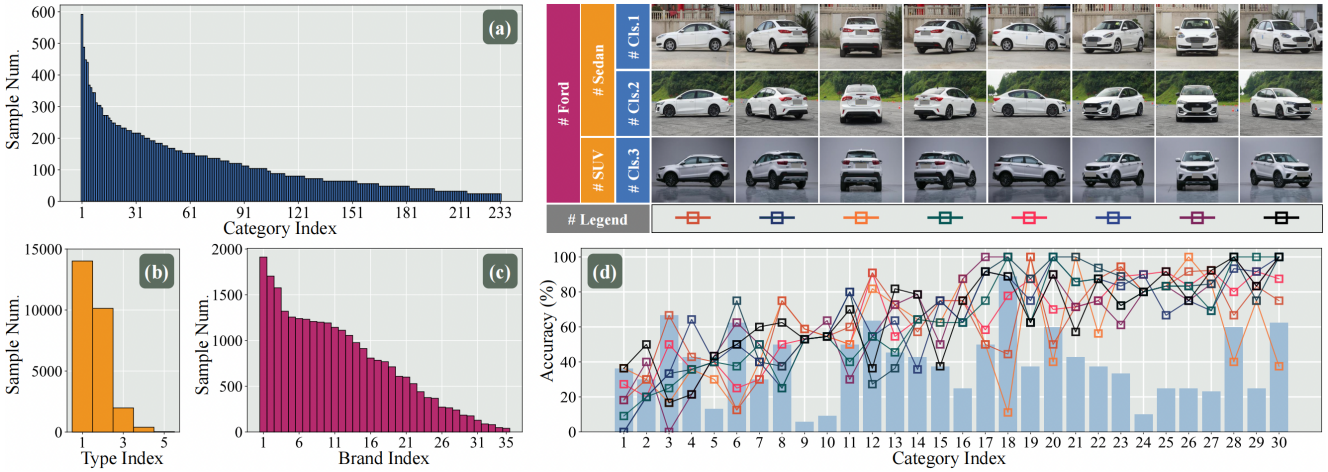

Figure 2. Illustration of samples from MvCars and some data statistics: (a) The number of samples per category. (b) The number of samples per type. (c) The number of samples per brand. (d) Results of our pilot experiment. The broken lines show model accuracy based on 8 individual views. Moreover, the bars represent the differences between each category's maximum and minimum accuracy.

depth sensor or something else, which makes them less practicable in daily applications, *e.g.*, recognizing an unfamiliar car for detailed information simply with a mobile phone. On the contrary, view-based methods that leverage multiple surrounding 2D views as descriptors for 3D objects tend to be an optimal choice.

Specifically, view-based methods share the core idea of encoding single-view features through a vision neural network and then aggregating multi-view features. Su *et al.* [47] first approaches the multi-view recognition problem with CNN for feature extraction and sum-pooling for aggregation. Then, Johns *et al.* [29] decomposes image sequences into image pair sets and then aggregates the pair-based classification in a weighted manner. After that, feature concatenation [51], hierarchical attention [22], and weighted fusion [17] are also adopted for better aggregating sequence features. In addition, sequences models (*e.g.*, LSTM [26], GRU [11], Transformer blocks [49], *etc*) are also widely considered [28, 23, 7] and demonstrate their effectiveness.

In this paper, specifically towards the multi-view active fine-grained visual recognition (MAFR) task, traditional multi-view recognition dataset (*e.g.*, RGB-D [33], ModelNet10, ModelNet40 [55]) is not sufficient any more. Thus, we first collect a fine-grained, multi-view car dataset named MvCars as a suitable testbed. Then, an active fine-grained recognition framework is built upon the general extraction-aggregation scheme. Note that, similar to ours, some approaches also take recognition efficiency into consideration [28, 29] by actively controlling the agent motion within a viewing sphere. However, a strict viewing sphere is not readily available in daily applications, especially for recognition with portable devices. We, therefore, consider the view selection as a discrete decision problem.

3. Dataset

In this work, we first contribute a testbed for MAFR by collecting a multi-view fine-grained car recognition dataset named Multi-view Cars (MvCars). Considering that car recognition is challenging and close to daily life [57], a vehicle dataset can demonstrate well how the MAFR model performs in real applications.

3.1. Data Statistic

MvCars consists of 233 models of cars from 35 brands (*e.g.*, Ford, Volvo, Escort, *etc.*) that cover 5 types (*e.g.*, sedan, SUV, MPV, sports car, and pickup). For each car, we collect and annotate 8 aligned perspectives – front, front-left, front-right, left, right, rear, rear-left, rear-right (as shown in Figure 2), and samples with missing perspectives are discarded. In total, there are $26,552$ images collected from 3319 cars and are then split into $15,960/3,000/7,592$ (at a ratio of $60\%/10\%/30\%$) for train/val/test set, respectively.

In addition to *multi-view* and *fine-grained*, MvCars is also a *hierarchical* and *long-tail* dataset. We can easily build the brand-type-model label hierarchies for categories in MvCars, *e.g.*, "Ford car"-"Ford sedan"-"Ford Mondeo", which may benefit directions like hierarchical structure-based FGVC [15] and human-friendly FGVC [5]. The numbers of samples per category are shown in Figure 2 (a), where a significant long-tail distribution can be observed. Such a long-tail dataset is also closer to realistic application and provides a new scenario to related topic [3]. The sample sizes per type and brand are shown in Figure 2 (b) and (c), respectively, and similar data distributions exist.

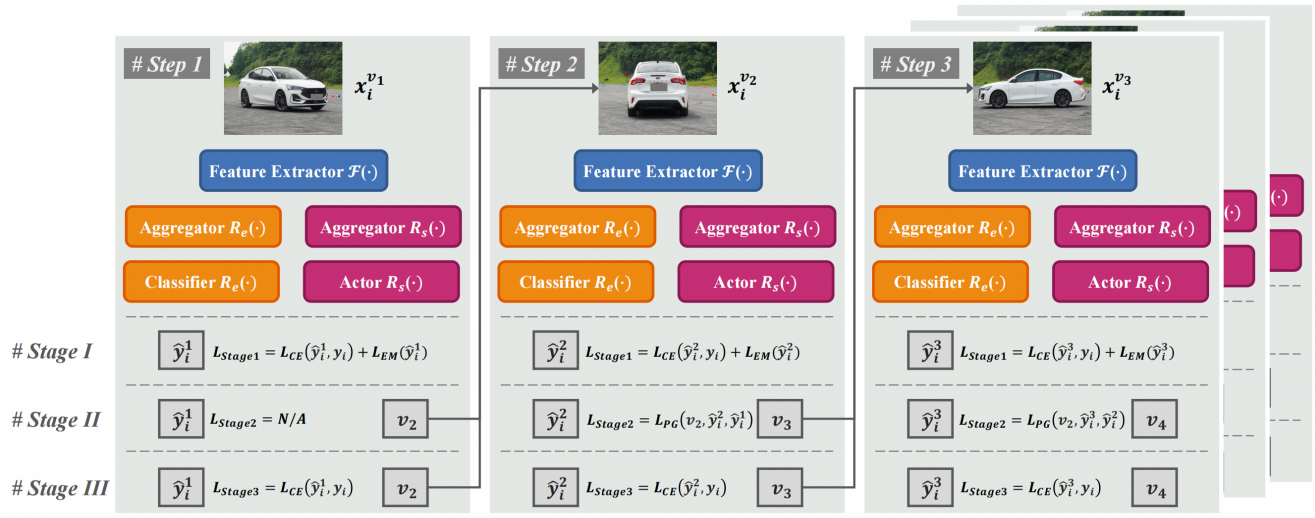

Figure 3. Illustration of the proposed MAFR framework where three training stages are included: **Stage I** for training a multi-view recognition model with smooth predictions, **Stage II** for optimizing the next-view selection component based on the behavior of the classifier, and **Stage III** for fine-tuning the recognition model along with the trajectory decided by the actor. Here we use three training steps for brief illustration.

3.2. Pilot Experiment

With the collected MvCars, here we first experimentally validate our first hypothesis mentioned in Section 1 – the discriminative information hides in various object views for different fine-grained categories. Factually, it is two-fold: (i) different perspectives contribute differently to FGVC; otherwise, actively selecting object view is meaningless, and (ii) different categories own different discriminative perspectives; otherwise, there is a trivial solution existing – consistently seeking the fixed distinguishable view.

In particular, for each perspective, we train a ResNet50 [25] for classification and obtain its accuracy in each category. Therefore, we can tell which perspective is more distinguishable for any specific category by comparing the performances of 8 models based on different views. We show the results of 30 classes in Figure 2 (d). Bars in the graph indicate the differences between the maximum and minimum accuracy of each category, and the average difference of all 233 classes is about 36.40%. It powerfully proves that different perspectives contribute differently in the context of FGVC. On the other hand, broken lines in the graph represent the single-view accuracy changes along with different categories. The interaction of lines indicates that the ranking of view discrimination is inconsistent, demonstrating that different categories have different discriminative perspectives.

In a word, in MvCars, different perspectives provide significantly various meanings for FGVC, which is also hard to pre-defined via prior knowledge. Thus, an active fine-grained recognition method is called for, and the collected MvCars dataset can serve as an eligible testbed.

4. Methodology

4.1. Overview

Here we first give an overview of the data flow of MAFR during inference and introduce the model components needed.

Data structure. For MAFR, a dataset D consists of N samples can be expressed as $D = \{X_i, y_i\}_{i=1}^N$, where $X_i = \{x_i^1, \ldots, x_i^v, \ldots, x_i^V\}$ is a sequence of images depict a specific sample from V perspectives and y_i is their common ground-truth label. Note that, the annotations of views $\{1, \ldots, V\}$ are aligned, *i.e.*, for arbitrary two samples X_i and X_j, x_i^v and x_j^v are taken from the same perspective.

Inference process. For sample X_i, the model will take an image $x_i^{v_1}$ from arbitrary view v_1 as the initial visual input, which simulates the situation that the model may start recognition while facing any views of the target object. After that, the recognition process will carry on step-by-step. In particular, at step t with input $x_i^{v_t}$ from view v_t, the model will utilize all currently perceived information $\{x_i^{v_1}, \ldots, x_i^{v_{t-1}}, x_i^{v_t}\}$ to deliver the category prediction $\hat{y}_i^t$ and the next-view proposal v_{t+1}. Then, an inference cycle is closed, and the process can continue with $x_i^{v_{t+1}}$ as the next input. Generally, $t \in [1, T]$, and T can be set to a maximum V.

Framework component. An extraction-aggregation structure tends to be an intuitive choice to process a sequence of correlated visual inputs. Similar structures are also developed by previous multi-recognition arts [8]. Specifically, for any image $x_i^{v_t}$ input the system, a CNN-based feature extractor $\mathcal{F}(\cdot)$ is first applied to extract single-

view feature as $f_i^{v_t} = \mathcal{F}(x_i^{v_t})$. It is worth noting that the feature extractors for different views share their weights, and this design will not lead to additional parameters. After that, an ideal model should consider all previously acquired information. Thus, a recurrent neural network is introduced as the aggregator $\mathcal{R}(\cdot)$ that aggregates features from all seen perspectives. In particular, we adopt two aggregators with the same structure but individual weights. $\mathcal{R}_e(\cdot)$ and $\mathcal{R}_s(\cdot)$ that, $\mathcal{R}_e(\cdot)$ form global embeddings $e_i^t = \mathcal{R}_e(f_i^{v_1}, \ldots, f_i^{v_t})$ for category prediction, while $\mathcal{R}_s(\cdot)$ depicts the current states $s_i^t = \mathcal{R}_s(f_i^{v_1}, \ldots, f_i^{v_t})$ for next-view selection. Finally, a classifier $\mathcal{P}(\cdot)$ and an actor $\mathcal{A}(\cdot)$ are equipped in parallel with outputs $\hat{y}_i^t = \mathcal{P}(e_i^t)$ and $v_{t+1} = \mathcal{A}(s_i^t)$, respectively.

4.2. Model Training

According to the aforementioned inference process, we can tell that the recognition component and the next-view selection component work separately but not independently. The mission of the recognition component is quite straightforward – conducting category prediction based on acquired information as well as possible. While the optimization goal of the next-view selection component largely depends on the behaviour of recognition – basically, the actor should try to select the next-view that can maximize the prediction probability of the target category. Therefore, a three stages training framework is intuitively designed: **Stage I** aims to train a good recognition model (including $\mathcal{F}(\cdot)$, $\mathcal{R}_e(\cdot)$, and $\mathcal{P}(\cdot)$) that can handle sequence input, **Stage II** aims to optimize the next-view selection module (where $\mathcal{R}_s(\cdot)$ and $\mathcal{A}(\cdot)$ participate) according to the behaviour of the trained recognition model, and **Stages III** aims to refine the recognition model under the trajectories decided by the actor.

Note that the inductive bias behind training the recognition component in the first place is that its behaviour can reveal view discrimination – a more discriminative view will greatly reduce the entropy of category prediction. However, a well-convergent classification model often delivers high-confidence predictions, especially for the small-scale datasets in the FGVC scenario, which will cause little changes in prediction probabilities and limit the information being revealed. For this, we take a very intuitive yet effective solution that splitting the training set D_T into two non-overlapping sub-sets for the first two stages respectively – specifically, a *rehearsal* set D_R is used for training the recognition components in **Stage I**, and a *evaluation* set D_E is used for providing authentic rewards while training the next-view selection module in **Stage II**. We have $D_R \cup D_E = D_T$ and $D_R \cap D_E = \emptyset$. In **Stage III**, the model will be trained with the whole training set D_T.

The whole framework is illustrated in Figure 3, and introductions about the three stages are as follows.

Stage I. To train a recognition model that can handle a sequence of inputs with dynamic length, each training iteration is divided into T steps with input sequence lengths from 1 to T. As the next-view planning module is not able for now, when $t \geq 2$, a new image $x_i^{v_t}$ is randomly selected from unseen views and appended to the input sequence at the $(t-1)$-th step. Here we set $T = V$ to ensure the maximum no-duplicated sequence can be sampled. With cross-entropy for optimization, the loss function for a batch of B samples can be formulated as:

$$L_{CE}(\hat{y}_i^t, y_i) = \frac{-1}{BTC} \sum_{i=1}^{B} \sum_{t=1}^{T} \sum_{c=1}^{C} y_i \times log(\hat{y}_{i,c}^t), \quad (1)$$

where C is the total category number, *i.e.*, the channel number of the last layer.

Additionaly, we also introduce a entropy maximization penalty as

$$L_{EM}(\hat{y}_i^t) = \frac{1}{BTC} \sum_{i=1}^{B} \sum_{t=1}^{T} \sum_{c=1}^{C} \hat{y}_{i,c}^t \times log(\hat{y}_{i,c}^t), \quad (2)$$

to encourage smooth predictions that can better reveal the priority between views. The total loss can be expressed as $L_{Stage1} = L_{CE}(\hat{y}_i^t, y_i) + \alpha \times L_{EM}(\hat{y}_i^t)$, and the degree of entropy maximization constrain can be control by the hyper-parameter α.

Stage II. Here, the recognition components ($\mathcal{F}(\cdot)$, $\mathcal{R}_e(\cdot)$, and $\mathcal{P}(\cdot)$) are frozen, and we only optimize $\mathcal{R}_s(\cdot)$ and $\mathcal{A}(\cdot)$ for next-view selection. As a non-differentiable sequential decision problem, we adopt the policy gradient method for optimization instead of directly optimizing with the classification loss. At the t-th ($t \geq 2$) training step, the model will receive the input $x_i^{v_t}$ with the perspective v_t decided by the actor at the $(t-1)$-th step. Then the view selection components can be updated according to the change of the target category's prediction probability, *i.e.*, the rewards are set as $r_i^t = \hat{y}_{iy_i}^t - \hat{y}_{iy_i}^{t-1}$. And the t-th ($t \geq 2$) step's loss function of **Stage II** can be simply expressed as: $L_{Stage2} = L_{PG}(v_t, \hat{y}_i^{t-1}, \hat{y}_i^t)$. Detailed expansion of the loss function can be found in Section 4.3

Notably, for popular policy gradient algorithms [45, 46], the total reward for the current step's optimization is a (weighted) sum of all feature rewards from now on. This is because these methods are designed for scenarios where an agent should achieve an ultimate goal through a series of actions. However, on the contrary, MAFR aims to use as few steps as possible to achieve as high accuracy as possible, *i.e.*, we care more about achieving the best performance at the current step rather than in the future. Therefore, we slightly modify the policy gradient algorithm by utilizing only r_i^t for the t-th step's optimization.

Stage III. There is nothing new in this stage – all settings are the same as **Stage I** except for (i) the selected view v_t

when $t \geq 2$ is given by the actor, (ii) the whole training set D_T is used. The model is refined under standard classification supervision (*i.e.*, $L_{Stage3} = L_{CE}(\hat{y}_i^t, y_i)$) to especially adjust the trajectories decided by the actor and utilizing all the training data.

4.3. Design Details

Feature extractor $\mathcal{F}(\cdot)$. The feature extractor can be any backbone network for vision tasks, including various CNN architectures and Transformers (*e.g.*, ResNet50 [25], ViT [12], *etc.*). Besides, by replacing $\mathcal{F}(\cdot)$ with other FGVC models, the proposed method can also extend them to work in 3D environments.

Feature aggregator $\mathcal{R}_e(\cdot)$ and $\mathcal{R}_s(\cdot)$. The two feature aggregators should be able to aggregate information from sequences with variable lengths. Here we adopt GRU [11] for best performance. There are alternatives like LSTM [26], self-attention block [49], *etc.*.

Classifier $\mathcal{P}(\cdot)$ and actor $\mathcal{A}(\cdot)$. Both the classifier and the actor are formed by one fully connected layer. For the cases that equip the proposed framework with other FGVC approaches, the structure of the classifier can be modified accordingly.

Policy gradient algorithm. For training the actor $\mathcal{A}(\cdot)$ (the next-view selection module) at **Stgae II**, we adopt the proximal policy optimization (PPO) algorithm [46] with a slight modification. Specifically, given a series of inputs $x_i^{v_1}, \ldots, x_i^{v_t}$ at the t-th step, the extractor $\mathcal{F}(\cdot)$ and the aggregator $\mathcal{R}_s(\cdot)$ are first applied to form the current state:

$$s_i^t = \mathcal{R}_s(\mathcal{F}(x_i^{v_1}), \ldots, \mathcal{F}(x_i^{v_t})). \tag{3}$$

And then, the actor take the state s_i^t as input and decide the next view proposal v_{t+1} as the action (*i.e.*, $v_{t+1} = \mathcal{A}(s_i^t)$). For the general PPO algorithm with the reward r_i^t for t-th step, the advantage estimator $\hat{A}_i^t$ can be expressed as:

$$\hat{A}_t = -V(s_i^t) + r_i^t + \gamma r_i^{t+1} + \cdots + \gamma^{T-t} r_i^T, \tag{4}$$

where $V(s_i^t)$ is the learned state-value function, $\gamma \in (0, 1)$ is a pre-defined discount factor, T is the maximum length of the input sequence. The principle behind it is straightforward – the current action should not only benefit the next step but also contribute to the overall goal. However, in this work, aiming at achieving reliable prediction with the least number of steps, we only focus on the profit at the very next step, *i.e.*, we set $\gamma = 0$. The advantage estimator we use can be formulated by $\hat{A}_i^t = -V(s_i^t) + r_i^t$.

After that, we denote the prediction probability of v_t by $\mathcal{A}(v_t|s_i^t)$. Then the clipped surrogate objective is:

$$\begin{aligned} L_{CLIP} = \frac{1}{B} \sum_{i=1}^{B} \sum_{t=2}^{T} min\{ & \frac{\mathcal{A}(v_t|s_i^t)}{\mathcal{A}_{old}(v_t|s_i^t)} \hat{A}_i^t \\ & clip(\frac{\mathcal{A}(v_t|s_i^t)}{\mathcal{A}_{old}(v_t|s_i^t)}, 1-\epsilon, 1+\epsilon) \hat{A}_i^t \}, \end{aligned} \tag{5}$$

where $\mathcal{A}_{old}(\cdot)$ stands for the actor before update, and $\epsilon \in (0, 1)$ is a hyper-parameter. Note that t starts from $t = 2$ since the first view is randomly selected. Finally, the overall objective of **Stage II** can be expressed as:

$$L_{Stage2} = L_{PG}(v_t, \hat{y}_i^{t-1}) = L_{CLIP} - c_1 L_{VF} + c_2 L_E, \tag{6}$$

where $L_{VF} = \frac{1}{B} \sum_{i=1}^{B} \sum_{t=2}^{T} (V(s_i^t) - V^{target}(s_i^t))^2$ is the squared-error loss suggested by [45], and $L_E = \frac{1}{B} \sum_{i=1}^{B} \sum_{t=2}^{T} S_{\mathcal{A}}(s_i^t)$ is the entropy bonus following [54, 40]. c_1 and c_2 is hyper-parameters to balance the three loss components.

4.4. Dynamic Exiting

For the proposed method, we can achieve the accuracy-efficiency trade-off by setting the maximum number of steps. However, simply assigning the same maximum step number to all samples may be too arbitrary since difficult samples may require information from more views than simple samples do. For that, inspired by previous sequential prediction work [52], we introduce the concept of budget batch classification [27] for dynamic exiting. Specifically, defining the exit probability of a certain step to be p, the probability of model exiting at the t-th step is $p_t = \alpha(1-p)^{t-1}p$, where α is a scaling factor that ensures $\sum_{t=1}^{T_{max}} p_t = 1$. The value of p can be found by solving the function $C = \sum_{t=1}^{T_{max}} t \times p_t$, where C is the expectation of average prediction steps. Then, we can estimate the exiting threshold μ_t for each step on the validation set. Such a strategy can further reveal the model potential under given step expectations by enabling better resource allocation among all test data.

5. Experiment

Implementation. The feature extractor $\mathcal{F}(\cdot)$ is initialized with ImageNet pre-trained weights, while other model components are ramdomly initialized. During **Stage I** and **Stage III**, we use SGD optimizor with a momentum of 0.9 and the cosine learning rate schedule [35] for optimization. The start learning rate is set to be 0.01 for the feature extractor and 0.1 for the rest. The input images are resize to 256×256 and then random-cropped to 224×224. The hyper-parameter α is set to be 0.01. During **Stage II**, we use the Adam optimizor with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and the cosine learning rate schedule [35] for optimization. The start learning rate of the actor $\mathcal{A}(\cdot)$ is set to be 0.005. The input images are resize to 256×256 and then center-cropped to 224×224. The hyper-parameters ϵ, c_1, and c_2 are set to be 0.2, 0.5, and 0.01, respectively. The model is trained for 15 epochs for each stage. During inference, the input images are resize to 256×256 and then center-cropped to 224×224.

	Method	mAcc. (%)	w-mAcc. (%)	Step2-Acc. (%)
ResNet50	RotationNet [30]	79.53 ± 0.4	79.32 ± 0.7	78.72 ± 0.8
	VERAM [8]	79.42 ± 0.5	80.50 ± 1.3	80.19 ± 1.4
	Baseline†	78.21 ± 0.3	78.04 ± 0.5	76.74 ± 0.3
	Baseline‡	79.60 ± 0.7	79.75 ± 1.2	78.28 ± 0.8
	Ours	80.84 ± 0.4	82.20 ± 0.7	81.45 ± 0.9
	Ours*	$\mathbf{81.33 \pm 0.5}$	$\mathbf{83.25 \pm 0.9}$	$\mathbf{82.86 \pm 1.1}$
ViT-B_16	Baseline†	80.05 ± 0.4	80.32 ± 1.3	78.82 ± 0.4
	Baseline‡	80.47 ± 0.7	80.79 ± 1.4	79.93 ± 1.3
	Ours	81.35 ± 1.1	82.60 ± 0.6	81.53 ± 0.8
	Ours*	$\mathbf{82.13 \pm 1.2}$	$\mathbf{83.58 \pm 1.3}$	$\mathbf{82.50 \pm 1.4}$
PMG	Baseline†	80.26 ± 0.9	80.51 ± 1.4	79.92 ± 0.9
	Baseline‡	80.65 ± 0.9	80.98 ± 0.6	80.21 ± 0.4
	Ours	82.69 ± 1.1	83.47 ± 1.4	82.77 ± 1.3
	Ours*	$\mathbf{82.99 \pm 0.3}$	$\mathbf{84.46 \pm 0.5}$	$\mathbf{83.79 \pm 0.6}$
TransFG	Baseline†	81.26 ± 1.5	81.33 ± 0.9	80.76 ± 0.6
	Baseline‡	81.78 ± 1.2	82.15 ± 1.4	81.52 ± 0.6
	Ours	83.76 ± 1.3	84.72 ± 0.5	84.35 ± 1.0
	Ours*	$\mathbf{84.89 \pm 0.3}$	$\mathbf{85.77 \pm 1.3}$	$\mathbf{85.57 \pm 0.6}$

Table 1. Results of the proposed method with different backbones. The best results of each section are marked in **bold**.

Baseline models. For extensive evaluation, three groups of approaches were considered for comparison: **I.** previous multi-view recognition methods, including RotationNet [30] and VERAM [8], **II.** state-of-the-art FGVC methods, including PMG [13] and TransFG [24], and **III.** advanced vision neural networks, including ResNet [25] and ViT [12]. Specifically, for group **I**, we re-implemented them with ResNet50 as the backbone model for a fair comparison. And for group **II** and **III**, as they are designed for single-view recognition, two baseline frameworks, which may be intuitive choices for most people, are designed to adapt them to the MAFR task: (i) Baseline†: a naive model ensemble scheme, *i.e.*, predictions are independently delivered according to each view, and at the t-th step, the average of t inputs' predictions is adopted as the current result, and (ii) Baseline‡: due to their conciseness, these models can directly serve as the feature extractor of the proposed method, therefore, a strong sequential prediction baseline can be implemented by replacing our feature extractors with these approaches and training under the setting of **Stage III**.

Evaluation Metrics. For quantitative evaluation, results based on 3 metrics are reported: (i) **Mean Accuracy (mAcc)** that takes the mean value of all T steps' accuracy, which can be regarded as the area under the accuracy-step line that represents the general performances of models, (ii) **Weighted Mean Accuracy (w-mAcc)** that weights different steps with exponentially decreased weights, since the performance of the first few steps should be more important in the consideration of efficiency[1], and (iii) **Step2 Accuracy (Step2-Acc)** that takes the 2-nd step's accuracy to highlight the profit of the first view selection[2].

[1]Here we take $[0, 0.5039, 0.2520, 0.1260, 0.0630, 0.0315, 0.0157, 0.0079]$ for w-mAcc when $T = 8$. The accuracy of the first step is weighted by 0.0 because it is randomly selected and does not relate to the performance of active selection.

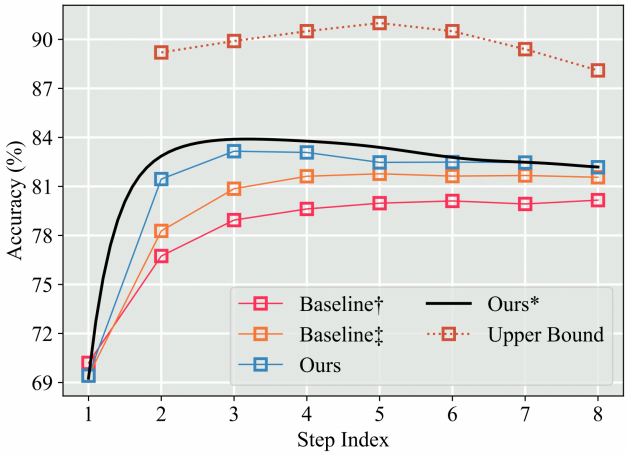

Figure 4. Illustration of model performance per-step and the performance upper bound from the perspective of trajectory decision.

5.1. Main Results

The results of the proposed method against all mentioned baselines are reported in Table 5. The table is organized into 4 sections according to different backbone networks. For ResNet50 [25] as the base model, we can observe that Baseline‡ already achieves impressive performance and delivers quite competitive results to previous multi-view recognition models. On the contrary, the proposed method outperforms it by ~1.2%, ~2.5%, ~3.2% for mAcc, w-mAcc, and Step2-Acc, respectively. The larger margins on w-mAcc and Step2-Acc also demonstrate its superiority in efficiency that benefits from the active next-view selection scheme. In addition, while applying the dynamic exiting strategy, we also calculate these three metrics by obtaining the accuracy when the step expectations equal 1-8, and there is no doubt that it further boosts model performance via a better efforts allocation. An interesting finding is that for models with the next-view selection mechanism, we can generally observe that mAcc $<$ Step2-Acc, while mAcc $>$ Step2-Acc for other models, which indicates a significant performance boosting caused by the active view selection at step 2. In addition, for other baselines implemented with the more advanced network (*i.e.*, ViT [12]) and FGVC models (*i.e.*, PMG [13] and TransFG [24]), the proposed framework can also boost their performance via active view selection and dynamic exiting, which indicates most of the recognition models can be extended to be an *FGVC experts* in the real physical world.

[2]Step2-Acc can be regarded as w-mAcc with weight set $[0.0, 1.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0]$.

Method	mAcc. (%)	w-mAcc. (%)	Step2-Acc. (%)
Random Selection	80.15 ± 0.5	79.86 ± 0.8	78.28 ± 0.6
Selecting the Farthest	80.50 ± 0.4	81.49 ± 0.6	80.68 ± 0.9
Selecting w/ Prior	80.43 ± 0.6	81.32 ± 0.9	80.55 ± 0.7
w/ Duplicate View	80.22 ± 0.9	81.38 ± 0.8	81.18 ± 1.1
w/o Data Split	80.19 ± 0.9	79.98 ± 1.1	78.53 ± 0.9
w/o L_{EM}	80.24 ± 0.4	81.57 ± 0.7	80.96 ± 1.0
w/ Future Rewards	80.56 ± 0.6	81.44 ± 0.8	80.53 ± 0.8
Ours	80.84 ± 0.4	82.20 ± 0.7	81.45 ± 0.9
Ours*	$\mathbf{81.33 \pm 0.5}$	$\mathbf{83.25 \pm 0.9}$	$\mathbf{82.86 \pm 1.1}$

Table 2. Results of ablation studies. The best ones are marked in **bold**.

5.2. Upper Bound Analysis

To better illustrate the change of model accuracy over inference steps, we show the accuracy-step lines of all models with ResNet50 [25] as the backbone in Figure 4. At this point, the audiences may question why these models' performances do not consistently increase. With the same question, we study the upper bound of our model. Due to the finite total view numbers, we are able to visit all possible trajectories for each sample. Therefore, a performance upper bound can be obtained from the perspective of trajectory decision – an arbitrary sample will be considered correctly classified when any trajectory allows this sample to be classified correctly. According to Figure 4, the degradation in the last few steps is also observed on the performance upper bound. A similar phenomenon is also observed in previous multi-view recognition works (*e.g.*, [30]). In this work, we also attribute this to the inherent feature of fine-grained recognition in the 3D environment – the discriminative clues only hide in a few views, and the noises caused by intra-class variance will be more likely to be introduced when full visual information (*i.e.*, all views) is included. This echoes the essential insight in the 2D fine-grained recognition where subtle differences of local regions are discriminative, and the global structures are more likely disturbed.

5.3. Ablation Study

In this section, we evaluate several variants of the proposed method based on ResNet50 to demonstrate the necessities of our designs. First, to verify the effectiveness of the active next-view selection mechanism, we study our model with no-duplicate inputs of random order and the order decided by the priority found in our pilot experiment. Fortunately, the proposed method passes the test with significant w-mAcc margins of ~2.3% and ~0.9%, respectively.

In addition, The farthest viewpoint sampling is also an important baseline – specifically, we implement it by selecting the farthest and unselected view from the second step. In cases where there are two equally distant views, we randomly choose one of them. However, the proposed method consistently outperforms it with a significant margin. This is because the most complementary view may not be the farthest one, as we further illustrate on the right in the next Section (see Figure 5).

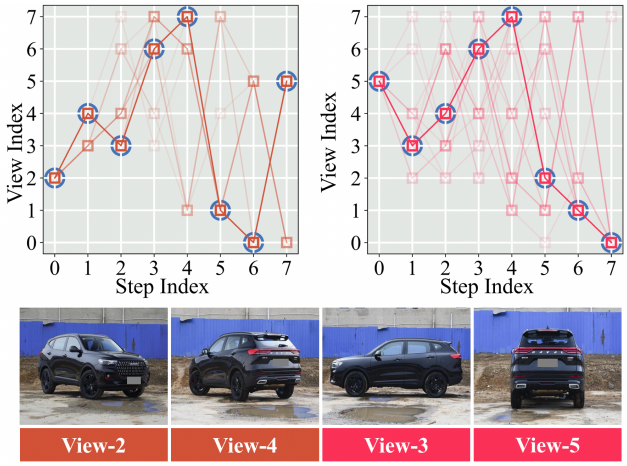

Figure 5. Predicted trajectories of "Haval H6" with "View-2" and "View-5" as the initial views. The darker the colour, the more times the trajectory has been selected. The most selected trajectory is highlighted with blue circles.

Besides, an artificial restriction is added for all evaluations to ensure that new views selected are no-duplicate. It is intuitive since unseen views can offer complementary information, and the information about which views have been selected is easily acquired. Here we also evaluate by allowing duplicate views, and the model performance degrades with no surprise.

After that, our designs for model training are also demonstrated to be effective. Splitting the training data into two non-overlapped subsets and encouraging entropy maximization both significantly boost model performance. It is worth noting that when we include the future rewards for policy optimization, mAcc is not significantly affected (with a slight degradation of 0.28%), but w-mAcc and Step2-Acc decrease by ~0.8% and ~0.9%. This indicates that future rewards may be meaningful for traditional sequential decision problems but not for MAFR, where efficiency is highly required.

5.4. Trajectory Analysis

For an in-depth analysis of the characteristics of the MAFR task, we further analyze the view-selection trajectory decided by our model. Specifically, with ResNet50 [25] as the feature extractor, we visualize the predicted trajectories of "Haval H6" in Figure 5. The two subplots show the results with "View-2" and "View-5" as the initial views, respectively, and the darker the colour, the more times the trajectory has been selected. The most se-

lected trajectory is highlighted with blue circles. According to the visualization, we can first conclude that there is some certainty in the best trajectory of a particular category – a significant broken line can be observed. However, the optimal trajectory is associated with the complementarity between different views – when we enter View-2, the next best choice is View-4; and when we enter View-5, the next best choice is View-3. Therefore, solving MAFR is not simply a ranking of the discriminative power of perspective information, echoing the results in Section 5.3 that view selecting with prior knowledge does not yield comparative performance. Factually, the optimal trajectory may even vary for different samples, as there is not just a single trajectory in each subplot. However, this may also be caused by the fact that our model still has room for improvement, so we do not discuss it further here.

6. Conclusion

This paper extends the fine-grained visual classification to 3D environments and proposes the multi-view active fine-grained visual recognition (MAFR) problem. We first collect a multi-view fine-grained car dataset (MvCars) as a qualified benchmark. Then we re-implement several multi-view recognition methods, FGVC approaches, and vision neural networks as baseline methods. A policy-gradient-based framework with a dynamic exiting strategy is proposed for the problem raised and yields the best performance. We also discuss the upper bound and predicted trajectories of the proposed method.

Acknowledgment

This work was supported in part by National Natural Science Foundation of China (NSFC) No. U19B2036, 62106022, 62225601, in part by Beijing Natural Science Foundation Project No. Z200002, in part by scholarships from China Scholarship Council (CSC) under Grant CSC No. 202206470055, in part by BUPT Excellent Ph.D. Students Foundation No. CX2022152, in part by the Program for Youth Innovative Research Team of BUPT No. 2023QNTD02, and in part by the Supported by High-performance Computing Platform of BUPT.

References

[1] John Aloimonos, Isaac Weiss, and Amit Bandyopadhyay. Active vision. *International Journal of Computer Vision*, 1988. 2

[2] Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey. Pointnetlk: Robust & efficient point cloud registration using pointnet. In *CVPR*, 2019. 2

[3] Samy Bengio. Sharing representations for long tail computer vision problems. In *ACM ICMI*, 2015. 3

[4] Dongliang Chang, Kaiyue Pang, Ruoyi Du, Yujun Tong, Yi-Zhe Song, Zhanyu Ma, and Jun Guo. Making a bird ai expert work for you and me. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1

[5] Dongliang Chang, Kaiyue Pang, Yixiao Zheng, Zhanyu Ma, Yi-Zhe Song, and Jun Guo. Your" flamingo" is my" bird": Fine-grained, or not. In *CVPR*, 2021. 1, 2, 3

[6] Dongliang Chang, Yujun Tong, Ruoyi Du, Timothy Hospedales, Yi-Zhe Song, and Zhanyu Ma. An erudite fine-grained visual classification model. In *CVPR*, 2023. 2

[7] Shuo Chen, Tan Yu, and Ping Li. Mvt: Multi-view vision transformer for 3d object recognition. In *BMVC*, 2021. 2, 3

[8] Songle Chen, Lintao Zheng, Yan Zhang, Zhixin Sun, and Kai Xu. Veram: View-enhanced recurrent attention model for 3d shape classification. *IEEE Transactions on Visualization and Computer Graphics*, 2018. 4, 7

[9] Yue Chen, Yalong Bai, Wei Zhang, and Tao Mei. Destruction and construction learning for fine-grained image recognition. In *CVPR*, 2019. 2

[10] Han-Pang Chiu, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Virtual training for multi-view object class recognition. In *CVPR*, 2007. 2

[11] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NeurIPS Workshops*, 2014. 2, 3, 6

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 6, 7

[13] 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 *ECCV*, 2020. 1, 2, 7

[14] Ruoyi Du, Dongliang Chang, Kongming Liang, Timothy Hospedales, Yi-Zhe Song, and Zhanyu Ma. On-the-fly category discovery. In *CVPR*, 2023. 1

[15] Ruoyi Du, Dongliang Chang, Zhanyu Ma, Yi-Zhe Song, and Jun Guo. Clue me in: Semi-supervised fgvc with out-of-distribution data. *arXiv preprint arXiv:2112.02825*, 2021. 3

[16] Ruoyi Du, Jiyang Xie, Zhanyu Ma, Dongliang Chang, Yi-Zhe Song, and Jun Guo. Progressive learning of category-consistent multi-granularity features for fine-grained visual classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 1

[17] Yifan Feng, Zizhao Zhang, Xibin Zhao, Rongrong Ji, and Yue Gao. Gvcnn: Group-view convolutional neural networks for 3d shape recognition. In *CVPR*, 2018. 3

[18] Stan Franklin. Autonomous agents as embodied ai. *Cybernetics & Systems*, 1997. 1

[19] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *CVPR*, 2017. 1, 2

[20] Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell. Compact bilinear pooling. In *CVPR*, 2016. 2

[21] Yurong Guo, Ruoyi Du, Xiaoxu Li, Jiyang Xie, Zhanyu Ma, and Yuan Dong. Learning calibrated class centers for few-

shot classification by pair-wise similarity. *IEEE Transactions on Image Processing*, 2022. 2

[22] Zhizhong Han, Honglei Lu, Zhenbao Liu, Chi-Man Vong, Yu-Shen Liu, Matthias Zwicker, Junwei Han, and CL Philip Chen. 3d2seqviews: Aggregating sequential views for 3d global feature learning by cnn with hierarchical attention aggregation. *IEEE Transactions on Image Processing*, 2019. 3

[23] Zhizhong Han, Mingyang Shang, Zhenbao Liu, Chi-Man Vong, Yu-Shen Liu, Matthias Zwicker, Junwei Han, and CL Philip Chen. Seqviews2seqlabels: Learning 3d global features via aggregating sequential views by rnn with attention. *IEEE Transactions on Image Processing*, 2018. 3

[24] Ju He, Jie-Neng Chen, Shuai Liu, Adam Kortylewski, Cheng Yang, Yutong Bai, Changhu Wang, and Alan Yuille. Transfg: A transformer architecture for fine-grained recognition. In *AAAI*, 2022. 7

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2, 4, 6, 7, 8

[26] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 1997. 3, 6

[27] Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, Laurens van der Maaten, and Kilian Weinberger. Multi-scale dense networks for resource efficient image classification. In *ICLR*, 2018. 6

[28] Dinesh Jayaraman and Kristen Grauman. Look-ahead before you leap: end-to-end active recognition by forecasting the effect of motion. In *ECCV*, 2016. 3

[29] Edward Johns, Stefan Leutenegger, and Andrew J Davison. Pairwise decomposition of image sequences for active multi-view recognition. In *CVPR*, 2016. 2, 3

[30] Asako Kanezaki, Yasuyuki Matsushita, and Yoshifumi Nishida. Rotationnet: Joint object categorization and pose estimation using multiviews from unsupervised viewpoints. In *CVPR*, 2018. 2, 7, 8

[31] Jonathan Krause, Hailin Jin, Jianchao Yang, and Li Fei-Fei. Fine-grained recognition without part annotations. In *CVPR*, 2015. 2

[32] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV workshops*, 2013. 1

[33] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *ICRA*. IEEE, 2011. 3

[34] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *ICCV*, 2015. 1, 2

[35] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[36] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 1

[37] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *IROS*, 2015. 2

[38] Hsien-Yu Meng, Lin Gao, Yu-Kun Lai, and Dinesh Manocha. Vv-net: Voxel vae net with group convolutions for point cloud segmentation. In *ICCV*, 2019. 2

[39] Weiqing Min, Zhiling Wang, Yuxin Liu, Mengjiang Luo, Liping Kang, Xiaoming Wei, Xiaolin Wei, and Shuqiang Jiang. Large scale visual food recognition. *arXiv preprint arXiv:2103.16107*, 2021. 1

[40] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *ICML*, 2016. 6

[41] Peiyuan Ni, Wenguang Zhang, Xiaoxiao Zhu, and Qixin Cao. Pointnet++ grasping: learning an end-to-end spatial grasp generation algorithm from sparse point clouds. In *ICRA*, 2020. 2

[42] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 2

[43] Charles R Qi, Hao Su, Matthias Nießner, Angela Dai, Mengyuan Yan, and Leonidas J Guibas. Volumetric and multi-view cnns for object classification on 3d data. In *CVPR*, 2016. 2

[44] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 2017. 2

[45] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *ICLR*, 2016. 5, 6

[46] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 2, 5, 6

[47] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *ICCV*, 2015. 2, 3

[48] 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. In *CVPR*, 2018. 1

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 2017. 3, 6

[50] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 1

[51] Chu Wang, Marcello Pelillo, and Kaleem Siddiqi. Dominant set clustering and pooling for multi-view 3d object recognition. In *BMVC*, 2017. 3

[52] Yulin Wang, Kangchen Lv, Rui Huang, Shiji Song, Le Yang, and Gao Huang. Glance and focus: a dynamic approach to reducing spatial redundancy in image classification. In *NeurIPS*, 2020. 6

[53] Yaming Wang, Vlad I Morariu, and Larry S Davis. Learning a discriminative filter bank within a cnn for fine-grained recognition. In *CVPR*, 2018. 1, 2

[54] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 1992. 6

[55] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *CVPR*, 2015. 2, 3

[56] Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaxing Zhang, Yuxin Peng, and Zheng Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *CVPR*, 2015. 1, 2

[57] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *CVPR*, 2015. 1, 2, 3

[58] Ze Yang and Liwei Wang. Learning relationships for multi-view 3d object recognition. In *ICCV*, 2019. 2

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

[60] Tan Yu, Jingjing Meng, and Junsong Yuan. Multi-view harmonized bilinear network for 3d object recognition. In *CVPR*, 2018. 2

[61] Ning Zhang, Jeff Donahue, Ross Girshick, and Trevor Darrell. Part-based r-cnns for fine-grained category detection. In *ECCV*. Springer, 2014. 2

[62] Heliang Zheng, Jianlong Fu, Zheng-Jun Zha, and Jiebo Luo. Learning deep bilinear transformation for fine-grained image representation. In *NeurIPS*, 2019. 2