Automated Aircraft Recognition via Vision Transformers

Yintong Huo
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Hong Kong, China
yintonghuo98@gmail.com

Yun Peng
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Hong Kong, China
pengyun2016@gmail.com

Michael Lyu
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Hong Kong, China
lyu@cse.cuhk.edu.hk

***Abstract*—Aircraft recognition aims to identify an aircraft type from its external appearance, serving as a vital task in the military field. The development of photography equipment allows technicians to collect images of the aircraft with rich information over a variety of scales and resolutions in a convenient way. However, these images are taken from different light and viewing angles, leading to the aircraft's various shapes, radiations, and colors. Such variance raises challenges for automated aircraft recognition techniques. This paper proposes an accurate and robust automated aircraft recognition technique based on Vision Transformers (ViT) to resist the variation carried by visual images. In particular, the self-attention mechanism in the ViT can better models the long-range dependency of pixels, compared with the existing convolutional neural network (CNN) approaches. We evaluate the effectiveness of our approach on a publicly available benchmark FGVCAircraft over multi-level granularity categories. The suggested ViT model achieves an overall Precision@1 of 0.915, outperforming other baselines, especially in images with complex variations.**

Figure 1: An example of the aircraft AST42

TABLE OF CONTENTS

1. INTRODUCTION

Aircraft recognition is visually recognizing and identifying different kinds of airplanes. The task is regarded as a vital military skill to recognize friendly or hostile aircraft, so as to apply prompt military strategies. Moreover, aviation enthusiasts who usually come to the airport to view aircraft should employ aircraft recognition skills if they want to know the specific aircraft category. An aircraft expert can identify the category based on aircraft characteristics. For example, ATR42 (Figure 1) is characterized by the short landing gear, the large vertical stabilizer, and the curved leading edge equipped with two bents.

However, such human-based aircraft recognition is not flexible enough and thus necessitates an automated recognition approach. Firstly, human-based aircraft recognition relies highly on domain knowledge that only a few experts have. Many experts are required if we want to identify or monitor thousands of aircraft. Secondly, images of an aircraft are taken from different lights and viewing angles, leading to various aircraft sizes, shapes, and colors. Consequently, these variances raise difficulties for human to recognize. An accurate, and intelligent aircraft recognition approach is inspired to reduce the burden of labor and to resist the variations brought by photo shooting.

Although many techniques have been proposed for image-related tasks [1, 2], the specific scenario in the aircraft area raises new challenges and uncertainties for applying these techniques. First, aircraft recognition is more challenging than classical image classification. Instead of the conventional image classification task that asks models to classify images into *coarse-grained* categories according to their content (e.g., dog, cat, airplane), aircraft recognition requires the model to identify the *fine-grained* types (e.g., Boeing 727, Boeing 767, CRJ-200). Intuitively, the discrepancy between "dog" and "airplane" is much more significant than "Boeing 727" and "Boeing 767". Therefore, a promising aircraft recognition model should extract more representative, high-level, descriptive features in one category and distinguish them from other types.

Moreover, aircraft recognition can be treated as a multi-level hierarchical classification problem, as different scenarios may concern different granularity of aircraft types. For instance, certain aerospace industry manufacturers are likely to focus on the difference between aircraft produced by different manufacturers (e.g., Boeing, Airbus, and ATR). Therefore,

978-1-6654-9032-0/23/$31.00 ©2023 IEEE

2023 IEEE Aerospace Conference | 978-1-6654-9032-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/AERO55745.2023.10115895

a manufacturer-level identification is sufficient for usage. Nevertheless, a variant-level (e.g., Boeing 707-320, Boeing 737-200, and Airbus 310) recognition is required for commercial usage to support air control operations, detect landing aircraft type for verification, and adopt proper air surveillance actions promptly. Consequently, it is worth investigating the technique's performance under such a multi-level hierarchical classification task.

A series of studies devise deep learning-based approaches to classify images automatically, which learn features from a large number of images with corresponding annotations (i.e., training phase), followed by predicting the category for an unseen image (i.e., testing phase). Deep learning-based models are usually made up of multiple neural layers connection, whose parameters (i.e., weights) can be learned during the training phase to fit the given labels. LeCun [3] proposes the first influential digit recognition LeNet-5 model using convolutional neural networks. Then, GoogLeNet [4] ensembles inception modules that leverage multi-scale features through multiple convolutional filter sizes to train the first large network efficiently and wins the visual recognition challenges. To mitigate the notorious vanishing gradient problem brought by increasing deep neural networks, ResNet [5] suggests a "shortcut connection" that skips one or more layers in the model, which later becomes one of the most popular architectures in the image classification community. Motivated by the most recent prosperous Transformer [6] model in natural language understanding, several studies attempt to adapt the Transformer for visual understanding tasks, named vision transformer (ViT). However, it is still unclear how effective the ViT model is in the aerospace area and how capable it is in performing multi-level aircraft recognition.

In this paper, we employ a novel vision transformer-based approach (ViT) to recognize multi-level aircraft types from images. As Transformer accepts textual data (e.g., the token sequence in texts), ViT represents a given image by visual tokens formed by a series of small image patches from the original image, then predicts the image category. In particular, ViT firstly divides an image into fixed-size patches, then embeds these patches, and further incorporates their patch positional embedding as an input to the transformer encoder to reconstruct the structure of the image. The designed multi-head self-attention layer in ViT allows it to capture global information over the whole image. It forces the model to pay attention to the informative and helpful part when recognizing objects. Considering aircraft recognition as a fine-grained classification task, such a self-attention mechanism and the patch positional embedding strategy enable our model to learn high-level features in aircraft images, and benefit the challenging task.

We evaluate our proposed ViT model for the aircraft recognition task from two perspectives: (1) how does it perform in overall aircraft recognition tasks; and (2) how is its effectiveness in identifying aircraft of multi-granularity. According to the experimental results, ViT shows its effectiveness by achieving 0.915 and 0.984 on the top-1 and top-5 precision for the overall performance, surpassing the existing baselines by 1% and 0.5%, respectively. Furthermore, the experiments on multi-granularity settings indicate that ViT is able to reach 0.849, 0.937, and 0.960 precision concerning identifying aircraft variants, families, and manufacturers, respectively. Finally, we believe these promising results and findings shed light on the aerospace community of using Vision Transformers in practice, not limited to automated aircraft recognition, rockfall detection, meteorite landing prediction, etc.

The rest of this paper is organized as follows. We introduce the study background and related works in Section 2. We then introduce our overall ViT methodology for aircraft recognition in Section 3. Implementation details are illustrated in Section 4. We display experimental results and discuss them in Section 5. We conclude our work in Section 6.

2. BACKGROUND

With the tremendous amount of accessible data, deep learning methods have obtained success in many fields over the past several decades. Computer vision has become one of the most prominent areas, which aims to teach computers to understand visual signals (e.g., images, videos). Applications in other fields, such as medical image segmentation [7], and material recognition [8], also benefit from the developments in computer vision.

Convolutional Neural Network (CNN) shows competitiveness in visual recognition tasks (i.e., ImageNet) once it was proposed by LeCun. LeNet5 [3] is an initial CNN network for digit recognition, but its simple architecture constrains its performance on a large-scale real-world dataset. Then, AlexNet [9], with a deeper and larger architecture of eight-layer CNNs, is proposed to be a leading architecture for object detection tasks, winning a large-scale visual recognition challenge. Afterward, VGG [10] firstly suggests the idea of the block, which is comprised of a small sequence of convolutional layers and activation functions. Blocks are connected with nonlinear transformations, preventing the network from a multi-layer spacial resolution vanishing problem. As researchers find deeper networks are more expressive and powerful for visual feature learning than shallow ones, a new challenge emerges: adding too many layers causes gradient vanishing when optimizing the neural networks and further deteriorates model performance. To solve this challenge, ResNet [5] is built with an innovative idea of stacking multiple identity mappings and creating a "shortcut connection" to skip one or more layers so that the model is easier to fit data. DenseNet [11] further enriches the connectivity pattern and the concatenation operation ways between layers while preserving features from earlier layers. While CNN shows a solid ability to capture image patterns, the vast success of Transformer [6] in natural language understanding catches people's eyes. Researchers attempt to employ the sequential model Transformer for image processing in order to gain the advantage from its unique self-attention mechanism and efficient parallel computation strategy. The basic idea is to divide the image into small patches and then sequentially feed them into the Transformer architecture. Nevertheless, how the vision transformer architecture brings benefits to the aerospace community has been unexplored. Hence, our paper fills this gap by extensively investigating aircraft recognition tasks via a vision transformer model.

3. METHODOLOGY

In this section, we present how we leverage ViT-based method to recognize multi-level aircraft types from images.

Overview

We show the overview of our method in Fig. 2. Given an image, unlike traditional CNN-based methods that directly input it into the model, we first cut the image into several small fixed-size patches. These small fixed-size patches are like the "words" in the "sentence", which embrace the sequential

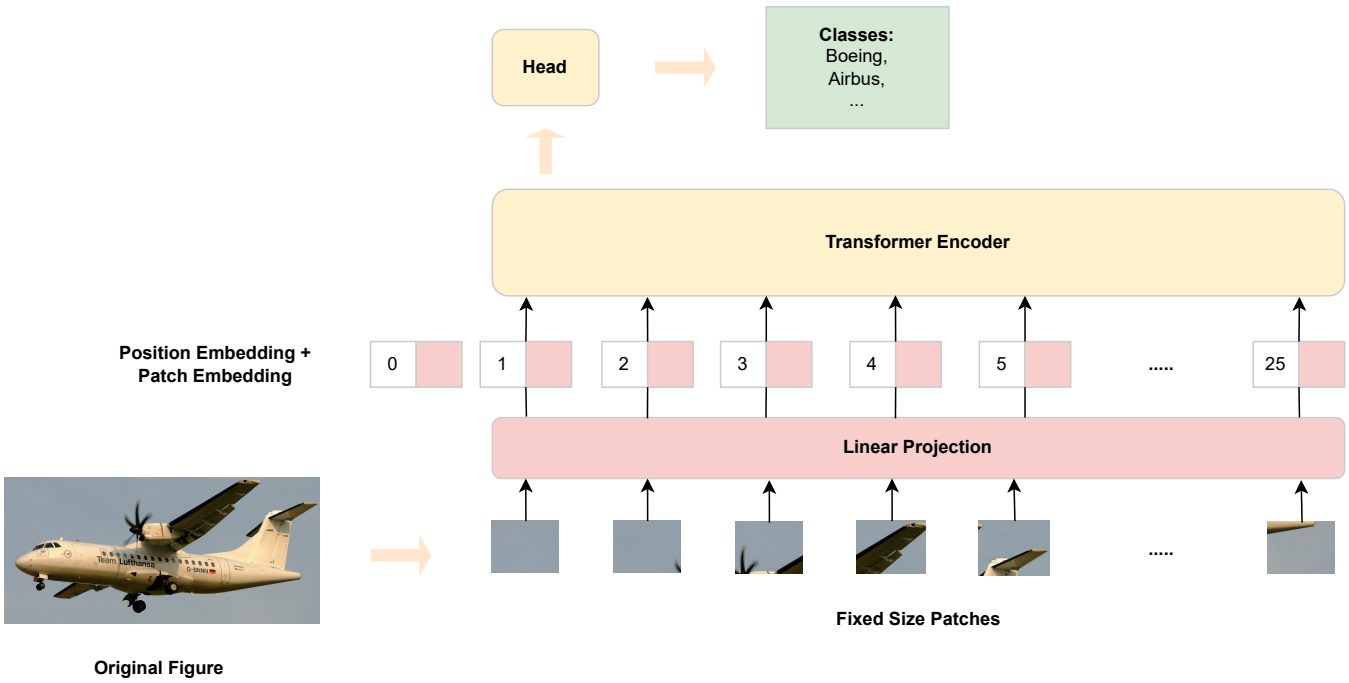


Figure 2: The structure of vision Transformer

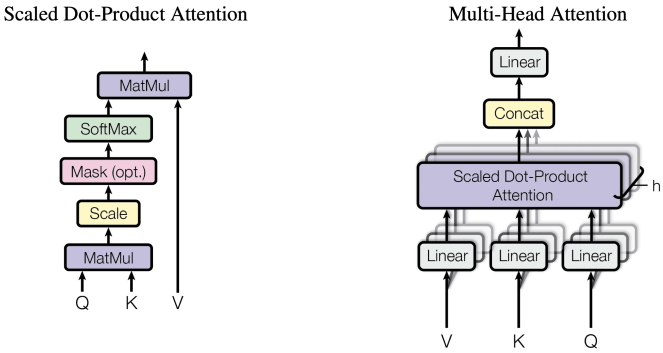


Figure 3: The structure of attention mechanism

nature of Transformers. We then encode each patch into a vector with linear projection. As each patch itself does not contain information about the location in the original image, we add a position embedding along with the patch embedding to represent a patch. After encoding an image with a sequence of vectors, we can leverage the basic Transformer to complete the classification task. A Transformer takes an input sequence and outputs a [CLS] head vector representing the whole input sequence. Finally, we train a linear classifier based on the head vector output by the Transformer to classify an aircraft image into an aircraft category.

Attention Mechanism and Transformer

As our method is based on Transformer, we first describe the technical details of the attention mechanism and basic Transformers. Attention [12] is firstly proposed in the neural machine translation (NMT) task. By integrating attention, many models such as CNN and RNN achieve better performance in this task. However, it is still hard to capture long-term dependencies in sentences. For example, CNN-based methods require a very deep network architecture with many layers to capture long-term dependencies, leading to high computational costs. Transformer [6] is then proposed to address this problem. Without complicated network design, Transformer completely relies on the attention mechanism to encode sentences and shows superior performance on many tasks such as neural machine translation [13–16], language modeling [17–19], etc.

We show the Scaled Dot-Product Attention mechanism [6] used in Transformer in the left of Fig. 3. The input of an attention block is three matrices: query Q, key K, and value V. They can be generated by a simple neural network from the input sequence. The calculation of output in an attention block follows Equation.1. The dot products of Q and V are divided by the square root of d_k, where d_k is the dimension of the queries. Then the result is sent to a softmax function to generate the weights and multiplied with V to get the final weighted values.

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \qquad (1)$$

The calculation of Scaled Dot-Product Attention is a step further than Sot-Product Attention. The dot products of Q and V are divided by the dimension before feeding into the softmax function. This is to prevent a large dimension of queries from generating large dot products, thus making softmax function focus on regions with small gradients.

Since Scaled Dot-Product Attention is found effective in capturing high-level features of input sequences, Transformer employs a multiple attention mechanism [6], which is called

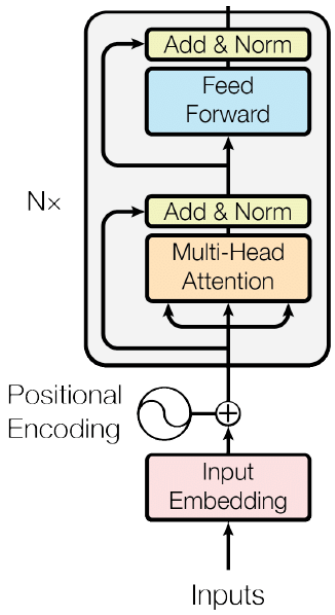


Figure 4: A basic Transformer block

multi-head attention. We show the structure of multi-head attention in the right of Fig. 3. The calculation of multi-head attention follows Equation. 2, 3.

$$\text{MultiHead}(Q, K, V) = \text{Concat}(Head_1, ..., Head_h)W^o \tag{2}$$

$$Head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{3}$$

Compared with single Q, K, V matrices in Scaled Dot-Product Attention, in multi-head attention with h heads, the three matrices are projected h times with different and learned linear layers. The h sets of matrices are then fed into h Scaled Dot-Product Attention blocks to generate h weighted values. These h weighted values are concatenated and go through a linear layer to get the final attention value.

Transformer is built upon multi-head self-attention. We show the architecture of a basic Transformer block in Fig. 4. A Transformer can include N repetitive Transformer blocks, depending on the requirements of specific tasks. As seen in Fig. 4, a basic Transformer block consists of a multi-head self-attention layer and a linear feed-forward layer, along with several normalization layers and residual layers. The calculation of multi-head self-attention layer follows Equation. 2, 3. The forward feed layer consists of two linear layers with RELU activation. Its calculation follows Equation. 4

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{4}$$

Image Encoding with Transformer

Transformer-based models used in natural language processing (NLP) tasks usually accept a word sequence, while in this task, our input is a single image. To generate sequence inputs from images, we follow the processes in ViT paper [20] and cut the image into several fixed-size patches. Specifically, given an image $x \in R^{H\times W\times C}$, where $H \times W$ is the revolution of the image and C is the number of channels, we cut it into N patches with revolution of $P \times P$, where $N = \frac{H\times W}{P\times P}$. We note the patch as $x_p \in R^{P\times P\times C}$ so that $x = [x_p^0; x_p^1; ...; x_p^N]$. Note that like the [CLS] token in sentences, here we add a special patch x_p^0 at the beginning of the sequence. We also record the location of each patch in the original image as the position embedding. Since Transformer requires a constant dimension D for vectors going through different layers, there is a linear projection layer to process the patches before feeding them into the Transformer. Therefore, the embedding of the original image can be calculated as Equation. 5.

$$z_0 = \text{Embedding}(x) = [x_p^0; x_p^1 W; ...; x^N W] + W_{pos} \tag{5}$$

In Equation. 5, $W \in R^{(P\cdot P)\times D}$ is the projection matrix for patches and $W_{pos} \in R^{(N+1)\times D}$ is the projection matrix for position embedding. Given Embedding(x), we feed it into the Transformer. Suppose Transformer contains L blocks as in Fig. 4, it conducts the calculations in Equation. 6, 7 for L times.

$$z_l' = \text{MultiHead}(\text{LayerNorm}(z_{l-1})) + z_{l-1} \tag{6}$$

$$z_l = \text{FFN}(\text{LayerNorm}(z_l')) + z_l' \tag{7}$$

$$y = \text{LayerNorm}(z_L^0) \tag{8}$$

Finally, we take the first element of the hidden vector output by the last Transformer block as the representation of the whole image, as shown in Equation. 8. This is similar to NLP tasks that take the encoding output of the [CLS] token as the representation of the entire input sentence.

With the representation y of the input image x, we then implement a linear classifier to classify the aircraft image into aircraft categories. We show this classification process in Equation. 9.

$$o = \text{argmax}(W_c y + b) \tag{9}$$

Suppose there are M categories of aircraft, we design a simple linear layer with input y and output the probabilities of each category the aircraft may belong to. Last an argmax function is invoked to select the category with the highest probability as the result.

4. IMPLEMENTATION

In this section, we elaborate on the implementation details of our experiments. Our ViT model is realized by Timm [21], one of the most popular PyTorch image libraries for research purposes.

Pretrained Model

Following previous studies, we fine-tune the pre-trained model (with learned parameters) in our aircraft recognition dataset instead of re-training a new one. This is because the evidence suggests that pre-training on large-scale image datasets enables the model to preserve transferrable knowledge for new tasks [22]. In particular, we selected the pre-trained model from Image-21K.

Hyper-Parameters

In our proposed ViT model, we set the patch size to be 16, so an aircraft image will be split into multiple 16*16 patches. The learning rate is 1e-3 with a decay rate of 0.99, so the learning rate decreases along the training epochs to accelerate model convergence. We also employ data augmentation techniques described in Auto Augmentation [23] to enrich the small training dataset. The ViT model is optimized by stochastic gradient descent (SGD) with a batch size of 16. In total, the base ViT model has 85,875,556 neural parameters for learning.

Computational Resources

All experiments are conducted in Intel(R) Xeon(R) CPU E5620 @ 2.40GHz machine with 48GB memory under the system 64bit CentOS 7. The computation is accelerated by 4 Nvidia TITAN V GPU. On average, training an epoch over the whole dataset costs approximately 2 minutes in our machine.

5. EXPERIMENTS

This section introduces our experimental details, including the dataset, baselines, and metrics, then displays our extensive experimental results with findings and analysis.

Experimental Details

(1) Dataset

Similar to previous research, we chose the classical aircraft recognition dataset, FGVCAircraft [24], in our study. The FGVCAircraft dataset includes 10,000 images in total, of which 3,333 are for testing and 6,667 for training and validating.

For each image, it provides three-level granularity annotation illustrated as following, from finer to coarser.

- Variant. A group of aircraft within the same pilot type rating belongs to the same variant. It is the finest distinction level that is visually detectable.
- Family. An aircraft family gathers variants with slight differences. For example, the family of "Boeing 737" contains variants including 737-200 and 737-300.
- Manufacturer. A manufacturer including a group of aircraft families that are fabricated by the same company.

Taking Figure 5 as an example, the aircraft are labeled with A330-200 (variant), A330 (family), and Airbus (manufacturer). In summary, the dataset contains 100 distinct variants, 70 families, and 30 manufacturers. It covers a wide range of typical aircraft, demonstrating the representativeness of this dataset.

(2) Baseline

- ResNet [5]. ResNet is initially proposed for resolving the gradient vanishing problem with the increasing layers in neural networks. Its "shortcut connection" design to bypass layers extended its expressiveness and won the large image recognition challenge at that time.
- DenseNet [11]. DenseNet is a typical design for the convolutional neural network that utilizes (dense) connections between layers. In this way, each layer can directly calculate its gradients from the loss function instead of via propagating.
- Inception [25]. Inception is a milestone in CNN development in the way that it designs dedicated and complex methods to chain the CNN layers instead of simply stacking them. In particular, we select the latest Inception-v4 with the best performance from the Inception family.

Figure 5: An example of an aircraft in A330-200 variant, A330 family, and Airbus manufacturer.

Table 1: **Overall performance of ViT comparing to baselines. The Avg. is calculated by the average performance accross three-level granularities.**

Approach	Avg. Precision@1	Avg. Precision@5
Random	0.019	0.096
ResNet-18	0.852	0.970
DenseNet	0.900	0.979
Inception-v4	0.905	**0.984**
ViT	**0.915**	**0.984**

(3) Metric

Following previous image recognition studies, we use *Precision* as the evaluation metric, which calculates how a model can accurately classify the percentage of the images. We especially report Precision@1 and Precision@5, which refers to top-1 and top-5 prediction accuracy, respectively.

In order to investigate multi-level granularity scenarios, we also report both the *overall* performance and *multi-level* granularity performance, that is, variant-level (e.g., Boeing 737-700), family-level (e.g., Boeing 737), and manufacturer-level (e.g., Boeing), ordered from finest to coarsest.

Experimental Results and Findings

We present our experimental results and discovered findings as follows.

***Finding 1**: ViT outperforms existing baselines concerning the overall performance.*

We are firstly interested in how our ViT model performs general aircraft recognition tasks. To answer the question, we train three ViT models corresponding to the three-level clas-

Table 2: **Experimental results of ViT concerning multi-level granuality comparing with baselines.**

Approach	Variant		Family		Manufacturer	
	Precision@1	Precision@5	Precision@1	Precision@5	Precision@1	Precision@5
Random	0.010	0.050	0.014	0.071	0.033	0.167
ResNet-18	0.756	0.949	0.881	0.973	0.919	0.989
DenseNet	0.834	0.965	0.917	0.981	0.949	0.991
Inception-v4	0.841	**0.977**	0.927	0.982	0.947	0.992
ViT	**0.849**	0.969	**0.937**	**0.987**	**0.960**	**0.995**

Table 3: **The overall aircraft recognition performance from different model size. ViT-base model is equivalent to the ViT model mentioned in previous experiments.**

Approach	# Parameters	Training Time/Epoch	Avg. Precision@1	Avg. Precision@5
Random	-	-	0.019	0.096
ViT-small	21,692,614	30 seconds	0.910	0.982
ViT-base	85,852,486	2 minutes	0.915	0.984
ViT-large	303,373,382	10 minutes	**0.917**	**0.985**

sifier and report the average Precision@1 and Precision@5 over three granularities.

We present the experimental results in Table 1, where the ViT model achieves the average Precision@1 score of 0.915, outperforming all baselines by at least 1%. Concerning average Precision@5, the latest Inception-v4 generates comparable results with our ViT, whereas other baselines are worse than ours. We attribute ViT's promising results to its dedicated self-attention mechanism and positional embedding in its architecture, which enables the strong ability to capture, preserve, and learn distinct features from different classes in recognition. In addition, since Precision@1 is more challenging than Precision@5, we conclude that ViT is better at completing complex tasks, distinguishing more minor differences between different aircraft in fine-grained classification tasks. In a word, the ViT model is shown to achieve effectiveness in recognizing aircraft and is better than other existing approaches.

Finding 2*: ViT outperforms existing baselines in three-level granularities.*

We then investigate the multi-level granularity classification performance as different scenarios and applications may require various granularity labels. Military action may want accurate and finest variant category, whereas a piece of manufacture-level information is enough for a business analytic company.

To do so, we compare the ViT model with other automated aircraft recognition approaches over three-level granularity, with the result shown in Table 2. It is observed that ViT excels all other baselines concerning Precision@1 by at least 0.8%, 1%, and 1.1%, at variant-level, family-level, and manufacturer-level, respectively. Regarding Precision@5, our ViT model outperforms others at the family-level and manufacturer-level. The encouraging experimental results demonstrate the versatility of the ViT model, which benefits from its convolution complementarity. Previous CNN-based architecture is only able to capture the relationship between local pixels. However, Transformer applies a positional embedding operation that can capture the relationship globally (over the whole image). In this way, the features across different granularities can be sufficiently studied by ViT and provide leading results. To sum up, ViT demonstrates its efficacy in identifying multi-level granularity aircraft, showing flexibility over multiple scenarios and situations.

Table 4: **The overall aircraft recognition performance from different pretrained model, where ViT-1K is pretrained on Image-1K.**

Approach	Avg. Precision@1	Avg. Precision@5
Random	0.019	0.096
ViT-1K	0.903	0.980
ViT-21K	**0.915**	**0.984**

Finding 3*: The larger the ViT model is, the longer the required time to train is, and the better the performance is.*

Afterward, we are interested in how the model size will affect the final results. We apply three different sizes of ViT models for our evaluation and report their parameter number (# Parameters), the time required to train an epoch (Traning Time/Epoch), as well as the performance concerning overall Precision@1 and Precision@5 in Table 3.

Our experimental results indicate that the performance enhances when the model size increases, for example, ViT-small achieves the Precision@1 score of 0.910; whereas ViT-base achieves 0.915. However, we also observe that the computational costs grow when the model becomes larger. ViT-small model costs 30 seconds to train an epoch, but ViT-base requires 2 minutes (120 seconds). Hence, there is a balance between the model performance and training costs for any ViT application.

Moreover, we recognize that the performance will not significantly increase when comparing ViT-large to ViT-base. This is mainly because of the limited data size. Since FGVCAircraft only contains 6,667 images for training and validating, a model too large may cause the data over-fitting problem and further damage its generalization ability. In conclusion, choosing the appropriate model size is vital for aircraft recognition. The selection should consider computational costs, training costs, and data size.

Finding 4*: The pretrained ViT model affects the task-specific performance.*

Last but not least, researchers and companies fine-tune pre-trained models with very few samples to resolve a task, instead of training an initial large model from scratch. Since pre-trained models with learned weights are used in our study, we look into the impact of these pre-trained models

on aircraft recognition. To this end, we employ two pre-trained models from different datasets, ImageNet-1K and ImageNet-21K, denoted as ViT-1K and ViT-21K (same as ViT mentioned before). ImageNet-1K includes 1 million images, and ImageNet-21K contains more than 14 million images in total.

We report the overall performance from two pre-trained models in Table 4, and observe that ViT-21K performs much better than ViT-1K. Moreover, we find that ViT-1K generates comparable results with Inception-v4 and DenseNet. Therefore, the larger the pre-trained dataset, the better the ViT can perform in aircraft recognition. We analyze the advantages of large-scale pretraining as follows. Intuitively, the prerequisite knowledge for learning new stuff is helpful for humans. Similarly, the pretraining material is so important because it encourages the model to understand the preliminary and general knowledge so that such knowledge can be transferred to an unseen task with only a few samples. From these experiments, we summarize that large-scale pretraining significantly improves ViT's performance on a specific task.

6. Conclusion

In this study, we employ a novel vision transformer model (ViT) to recognize aircraft over multi-level granularity automatically. ViT accepts patches derived from an image and predicts aircraft category with its superior multi-head self-attention mechanism and positional embedding strategy. Our extensive experiments demonstrate the effectiveness of the ViT model over multiple scenarios. This paper draws four findings from the experiments and provides discussions and suggestions when applying ViT in practical aircraft recognition scenarios.

7. Acknowledgement

The work was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (CUHK 14210920).

References

[1] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene graph generation by iterative message passing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5410–5419.

[2] B. Dai and D. Lin, "Contrastive learning for image captioning," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[3] Y. LeCun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Muller, E. Sackinger *et al.*, "Comparison of learning algorithms for handwritten digit recognition," in *International conference on artificial neural networks*, vol. 60, no. 1. Perth, Australia, 1995, pp. 53–60.

[4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017.

[7] N. Sharma, L. M. Aggarwal *et al.*, "Automated medical image segmentation techniques," *Journal of medical physics*, vol. 35, no. 1, p. 3, 2010.

[8] J. Xue, H. Zhang, K. Dana, and K. Nishino, "Differential angular imaging for material recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 764–773.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[10] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[11] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[12] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014.

[13] Z. Fan, Y. Gong, D. Liu, Z. Wei, S. Wang, J. Jiao, N. Duan, R. Zhang, and X. Huang, "Mask attention networks: Rethinking and strengthen transformer," 2021.

[14] S. Mehta, M. Ghazvininejad, S. Iyer, L. Zettlemoyer, and H. Hajishirzi, "Delight: Deep and light-weight transformer," 2020.

[15] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," 2019.

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018.

[17] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," 2019.

[18] J. W. Rae, A. Potapenko, S. M. Jayakumar, and T. P. Lillicrap, "Compressive transformers for long-range sequence modelling," 2019.

[19] A. Roy, M. Saffar, A. Vaswani, and D. Grangier, "Efficient content-based sparse attention with routing transformers," 2020.

[20] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2020.

[21] R. Wightman, "Pytorch image models," https://github.com/rwightman/pytorch-image-models, 2019.

[22] H. Bao, L. Dong, and F. Wei, "Beit: Bert pre-training of

image transformers," *arXiv preprint arXiv:2106.08254*, 2021.

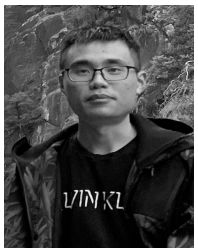
[23] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation policies from data," *arXiv preprint arXiv:1805.09501*, 2018.

[24] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," Tech. Rep., 2013.

[25] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-first AAAI conference on artificial intelligence*, 2017.

BIOGRAPHY

Yintong Huo *received the B.Eng. degree from the University of Electronic Science and Technology of China. She is currentlya Ph.D. student at CSE department, the Chinese University of Hong Kong. Her research interests are on automated log analysis, artificial intelligence for IT operations, and knowledge graph. She have published several papers on top conferences of software engineering such as ICSE and ISSRE.*



Yun Peng *received the B.Eng. degree from the University of Science and Technology of China. He is currently a Ph.D. candidate at the Chinese University of Hong Kong. His research interests are on intelligent code analysis, program analysis and artificial intelligence for software engineering. He published several papers on top conferences of software engineering such as ICSE and ESEC/FSE.*

Michael R. Lyu *received his B.S. in Electrical Engineering from National Taiwan University, Taipei, Taiwan; his M.S. in Computer Science from University of California, Santa Barbara, USA; and his Ph.D. in Computer Science from University of California, Los Angeles, USA. He is currently Choh-Ming Li Professor of Computer Science and Engineering in The Chinese University of Hong Kong. Prof. Lyu's research interests include software engineering, software reliability, machine learning, cloud and mobile computing, and distributed systems. He has published over 600 refereed journal and conference papers in his research areas. His Google Scholar citation is over 46,000, with an h-index of 104. Prof. Lyu initiated the first International Symposium on Software Reliability Engineering (ISSRE) in 1990. He was an Associate Editor of IEEE Transactions on Reliability, IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Services Computing, and Journal of Information Science and Engineering. He is currently on the editorial board of IEEE Access, Wiley Software Testing, Verification and Reliability Journal (STVR), and ACM Transactions on Software Engineering Methodology (TOSEM). Prof. Lyu was elected to IEEE Fellow (2004), AAAS Fellow (2007), ACM Fellow (2015), and named IEEE Reliability Society Engineer of the Year (2010). He was granted with China Computer Federation (CCF) Overseas Outstanding Contributions Award in 2018, and the 13th Guanghua Engineering Science and Technology Award in 2020. He was also named in The AI 2000 Most Influential Scholars Annual List with three appearances in 2020.*