IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 23 November 2022, accepted 17 December 2022, date of publication 21 December 2022,
date of current version 29 December 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3231327

RESEARCH ARTICLE

Optimizing and Evaluating Swin Transformer for Aircraft Classification: Analysis and Generalizability of the MTARSI Dataset

KYLE GAO[1], (Graduate Student Member, IEEE), HONGJIE HE[2], DENING LU[1], LINLIN XU[1], (Member, IEEE), LINGFEI MA[3], (Member, IEEE), AND JONATHAN LI[1,2], (Senior Member, IEEE)

[1]Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada
[2]Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON N2L 3G1, Canada
[3]School of Statistics and Mathematics, Central University of Finance and Economics, Beijing 102206, China

Corresponding author: Lingfei Ma (l53ma@cufe.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 41871380 and Grant 42101451, and in part by the Emerging Interdisciplinary Project of Central University of Finance and Economics in China.

ABSTRACT Aircraft classification via remote sensing images has many commercial and military applications. The Swin-Transformer has shown great promise, recently dominating general-purpose image classification benchmarks such as ImageNet. In this paper, we test whether the performance of the Swin-Transformer on general-purpose image classification translates to domain-specific aircraft classification using the Multi-Type Aircraft from the Remote Sensing Images dataset. We also investigate the effect of training procedure vs. model selection on the validation score. Our carefully trained Swin-Transformer model achieved an impressive 99.4 % validation set accuracy without super-resolution, and 99.5 % with super-resolution. Moreover, the generalization of models trained on the MTARSI dataset to real-world and synthetic aircraft classification is evaluated with some out-of-distribution samples. Our results demonstrate that the lack of complexity and heterogeneity of the MTARSI dataset, and the labeling errors resulted in models which struggle to achieve high accuracy on the adopted test samples despite near perfect validation scores.

INDEX TERMS Aircraft classification, deep learning, MTARSI dataset, out-of-distribution, remote sensing, self-attention, Swin transformer, vision transformer.

I. INTRODUCTION

Image classification is one of the most researched tasks in Computer Vision. In remote sensing, one application of the aforementioned is the classification of aircraft from aerial images, which finds uses in air traffic control, surveillance, and military intelligence. The Multi-Type Aircraft of Remote Sensing Images (MTARSI) classification dataset was built for the training and testing of aircraft classification algorithms. This paper details the use of Hierarchical Vision Transformers with Shifted Windows (Swin) models, as well as models of similar complexity, on the MTARSI dataset for aircraft classification. Our results are benchmarked against previous published state-of-the-art works on this dataset.

The associate editor coordinating the review of this manuscript and approving it for publication was Fan Zhang.

A. LITERATURE REVIEW

1) DATASETS

There are a wide variety of well-known general-purpose image classification datasets used to benchmark new deep learning models, the most popular of which are ImageNet [1] and CIFAR10 [2]. These general-purpose datasets contain object classes commonly found in daily life, and were used to develop foundational and highly impactful deep learning models such as ResNet [3] and VGG [4]. Compared to the general-purpose ones, remote sensing datasets tend to be domain specific. Many popular supervised remote sensing datasets exist for tasks such as building footprint

extraction [5] and land-use/land-cover mapping [6], which are commonly approached using segmentation methods. These datasets contain segmentation masks/labels created by geographic information system experts from unsupervised image datasets of satellite or aerial images. The MTARSI aircraft classification dataset [7] used in this study differs from the aforementioned segmentation datasets. The images were taken at different heights from various sources. The images were cropped and zoomed into such that the landed aircraft of interest was centered regardless of size; as such, there was no single spatial resolution for the images. These images were then labeled into distinct aircraft models/classes for classification. Recent researches have achieved good results on the dataset using a variety of classical models, including Convolutional Neural Networks (CNNs) and mixed classical/deep learning methods [7], [8], [9], [10], [11]. The results of Azam et al. [9] were worth highlighting. They achieved an accuracy of 96.8 % using an SVM classifier trained on Principal Components of CNN-extracted features, greatly exceeding the results from the other aforementioned works.

The FGVC-Aircraft dataset [12] was another aircraft classification dataset with some key differences from the MTARSI dataset. The FGVC dataset's labels were organized into a hierarchy of Manufacturer-Family-Variant-Model and are more suited than MTARSI for fine-grained classification of similar variants and models. However, the FGVC aircraft images were not remote sensing images from a top-down view. They were instead mostly ground-level images of landed and low-flying aircrafts with only some images being from a top-down view. As such, the FGVC-trained models are not directly applicable to MTARSI results and vice-versa. The RarePlanes dataset [13] encompassed both real and synthetic airplanes from a remote-sensing top-down view. However, the dataset was organized and labeled as an object detection and instance segmentation dataset and was not directly comparable to the MTARSI dataset. The large scenes in the RarePlanes dataset contained multiple airplanes each. The Aircraft Context Dataset [14] on the other hand had images of in-flight and grounded aircrafts with contextual labels which could be further adapted to classification, detection, and segmentation tasks, however, the images were not remote sensing-based and were taken from ground level. Other popular land-use remote sensing datasets also had an aircraft classification component. Examples of these were the UC Merced Land Use Dataset [15] and the RESISC45 [16], which were scene classification datasets. However, the ground truths for these two datasets were general land-use/land-cover classes with ''airplane'' as one of many labels, and could not be directly used to supplement the MTARSI dataset.

2) CONVOLUTIONAL NEURAL NETWORKS AND TRANSFORMERS

Convolutional neural networks (CNNs) have revolutionized image processing. LeNet [17], developed by Yann LeCun, was the earliest convolutional neural network. Invented in 1989, it was successfully applied to handwritten zip code identification. However, CNNs and deep learning in general had not achieved wide stream recognition until two decades later. In the early 2010s, deep learning experienced a great explosion in popularity. Convolutional Neural Networks have been the focus of deep learning-based computer vision for the past decade. The most popular among these are AlexNet [18], VGG [4], InceptionNet [19], and ResNet [3] which have dominated the ImageNet competition from 2012 to 2016. ResNet is particularly worth highlighting, as its residual connection alleviated problems with exploding or vanishing gradients, allowing for the construction of very deep convolutional neural networks. It was used as a backbone for many important algorithms such as Mask R-CNN [20], as well as a starting point for more modern CNNs. More modern CNNs included innovations such as the pyramid architecture [21] and attention mechanisms using hybrid attention/convolution models [22], [23]. In the past 5 years, hundreds, potentially thousands of CNN-based models were published, and successfully applied to research fields with major image processing components, such as remote sensing, medicine, manufacturing, autonomous navigation and transportation, and robotics, to name a few.

In 2017, Vaswani et al. developed the Transformer for natural language processing [24]. By using a combination of dense and self-attention layers, the authors produced a highly scalable deep learning model incredibly suited for pre-training on large datasets. This work inspired many super-sized language models, such as the 175 billion parameter GPT3 [25] and the Switch Transformer with 1.6 trillion parameters [26], which were trained using sophisticated unsupervised techniques, and successfully fine-tuned to a variety of downstream tasks. These models dwarf commonly used convolutional neural networks, which were typically less than a hundred million parameters E.g., the ResNet variants, ResNet-50 had 23 million parameters, and ResNet101 had 43 million parameters [3]. Large unsupervised and supervised image datasets also existed. However, in 2017, the transformer architecture was designed for sequential data, and could not easily be applied to computer vision.

In 2020, the Google Brain research team introduced the Vision Transformer (ViT) [27], which adapted the Transformer architecture to computer vision. By treating images as sequences of feature patches, they adapted the scalability and the powerful unsupervised pre-training of giant NLP Transformers to image processing and achieved state-of-the-art results on the common image classification benchmarks ImageNet [1] and CIFAR-10 [28]. Building on ViT, the team at Microsoft Research proposed the Shifted Window Hierarchical Vision Transformer (Swin) [29]. The Swin improved on the ViT with two key innovations; the hierarchical feature mapping, and the shifted window attention. Despite being initially developed for object classification benchmarks, these computer vision transformers were adaptive backbone architectures that have successfully been used for other downstream tasks such as semantic segmentation, object detection, image super-resolution, and instance segmentation

via integration into well-known algorithms such as Mask-RCNN [20] and HTC [30].

Data augmentation and training methodology can greatly improve the results of older convolutional neural networks to near Transformer levels. By using cleaver training techniques, data augmentation, and carefully searching appropriate learning rates and learning schedules, Wightman et al. [31] trained a ResNet-50 a on ImageNet1k, and achieved an extremely impressive top-1 accuracy comparable to Swin-Transformer and Swin-MLP models, greatly exceeding previous ResNet-50 performance.

3) SUPER-RESOLUTION

In many computer vision applications, super-resolution can also be used to improve image quality and final classification/segmentation results. Super-resolution is especially useful in remote sensing due to the limitations on the resolution of aerial and satellite images. Super-resolution methods in the context of remote sensing typically are categorized into two families. The first of which is Joint Image Super-Resolution. It is applied to super-resolve lower resolution hyperspectral images using spatial information from higher resolution multi-spectral images [32]. The other family of methods is Single Image Super-Resolution (SISR). The methods in this family are more applicable in general since most datasets are not built from hyperspectral images. In terms of single image super-resolution, the advancement in deep learning methods in computer vision directly translated to improved image upsampling methods; as such, modern research largely focused on deep learning methods. Examples of CNN-based single-image super-resolution models included VDSR [33], DRCN [34], RED-Net [35], AND DRRN [36]. More recent super-resolution neural networks also used the self-attention mechanism, e.g., RCAN [37], SAN [38], RFANet [39], and MSCA-RFANet [40], which have greatly improved results when integrated into the data pipeline of classification or segmentation tasks when input images were of low resolution.

B. CONTRIBUTIONS

In this paper, we studied aircraft classification on the MTARSI dataset using state-of-the-art deep learning models and training procedures.

- We implemented and trained a super-resolved Swin-Transformer model which greatly exceeded previous MTARSI benchmarks, achieving a validation score that we believed to be at the upper limit of the MTARSI dataset.
- We optimized different models in terms of the training procedure, and showed that the selection of training procedures and schedules greatly impacted model performance to the extent of bringing older models to state-of-the-art performance.
- We identified critical issues with the MTARSI dataset in terms of label errors, separability of training/validation sets, and data heterogeneity, and we suggested future improvements, as well as recommendations for dataset building.
- The generalizability of different models was evaluated on out-of-distribution samples, demonstrating that the aforementioned issues with the MTARSI dataset resulted in models which failed to classify real-world, synthetic, and scale-model aircrafts. Our results suggested that the dataset has limited real-world applications.

II. METHODOLOGY

A. DATASET AND DATA AUGMENTATION

The MTARSI dataset is a supervised image classification dataset containing 20 classes of commercial and military aircraft from Google Earth images (which sourced images from a variety of remote sensing image providers), as well as from other datasets such as FGVC-Aircraft. The aircrafts were landed, and viewed from a top-down point-of-view with only slight deviations in viewing angles. In this paper, we refer (except in tables) to MTARSI class labels with quotation marks (e.g. "F-22") and real-life airplanes without quotation marks (e.g. F-22). The dataset's class distribution is imbalanced, with some classes such as the "F-22" occurring more than twice as often as some other classes (See Fig.1). Moreover, it does not represent real-life distribution with military aircraft being over-represented. We note that the MTARSI dataset only covered 36 airports and contains many images of rare military aircraft. As such, some of the same aircraft appear in multiple images, albeit captured under different imaging conditions. To generate the 9385 images, the authors also performed augmentation on the dataset by segmenting airplanes, performing rotations and flips, and finally switching backgrounds. Despite having a large number of images, the variety of unique aircrafts was relatively low. The dataset was not canonically split into training and validation sets by the original authors [7]. We randomly split the 9385 MTARSI dataset images into 7045 training and 2340 validation images,

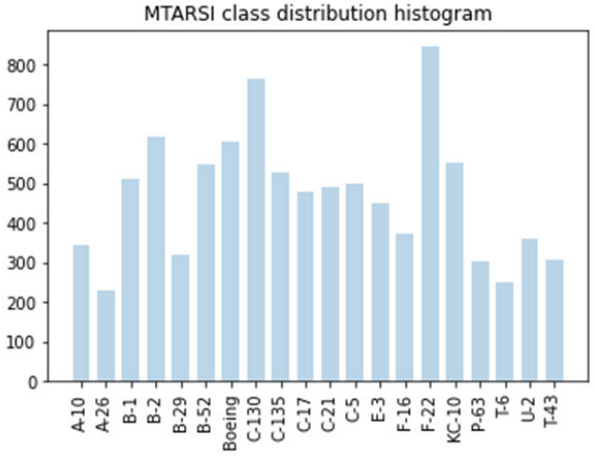


FIGURE 1. MTARSI class distribution histogram. We note a relatively high class imbalance.

FIGURE 2. Sample of the MTARSI images.

resulting in a 75:25 training-to-validation ratio. We note the original MTARSI dataset authors used an 80:20 ratio; our split should in theory contribute to more accurate validation results.

Fig.3 showed that a non-negligible fraction of images had heights and widths which were less than 150 pixels. We saw from the aspect ratio histogram that the images were moderately skewed vertically.

For our experiments, as part of the data pipeline, the images were resized via bi-cubic scaling and center cropping (while maintaining an aspect ratio to not deform images). This was due to the limitations of the pre-trained models, where weights were only available for a fixed input size. 224 × 224 was used for Swin-Transformer and ResNet-50 models. 256 × 256 was used for Swin-MLP. In a separate data pipeline, we also performed experiments using a Single Image Super Resolution deep neural network to super-resolve (upsample) the images by a factor of 2× prior to bi-cubic re-scaling and cropping to 224 × 224 or 256 × 256.

Preliminary experiments showed that the model struggled to learn after reaching training accuracies above 99.9% while validation accuracies remained in the mid-to-high 90%. To prevent overfitting while training complex models for a large number of epochs and encourage further training, advanced data augmentations were then employed. These include the ones suggested by the authors [27], [29], [31], implemented in the TIMM package [41]. They include random color jittering using a factor of 0.5, random erasure (via masking or replacement with noise) of image sections with probability of 0.25, random mixing of images via alpha blending ($\alpha = 0.8$), randomly chosen composed augmentations (TIMM [41] package Class) of Rotation, Equalization, Shear, Pixel translation, Brightness shift, Contrast adjustment, and Sharpness adjustment. Fig.4 shows example images from a training batch with full data augmentation. These data augmentations also had real-world motivations. The geometry of any type/model of aircraft (as viewed from above) is fixed. These augmentations can account for the different paint patterns, lighting conditions, and camera conditions, and help the model generalize to out-of-distribution aircraft. Moreover, the random occlusion of image sections can help the model learn to recognize the specific body,

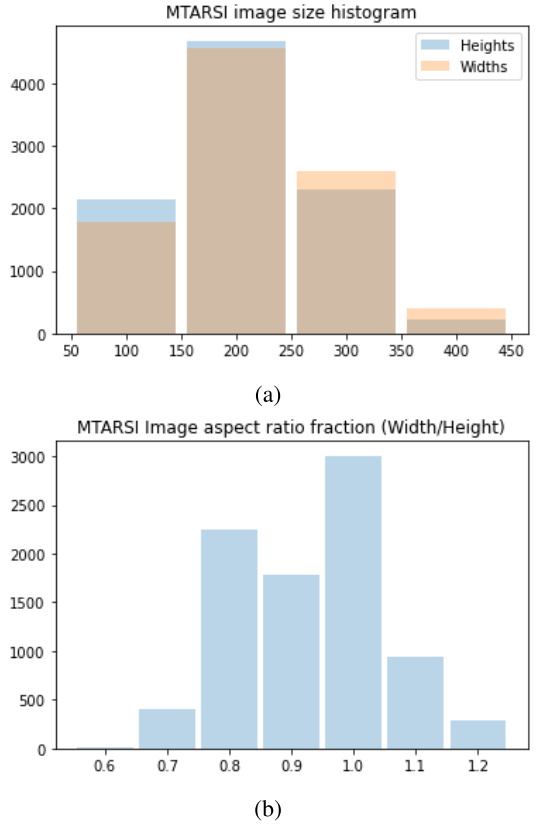


FIGURE 3. Distribution of heights and widths (in pixels) and the aspect ratios of images (as a fraction) in the MTARSI dataset.

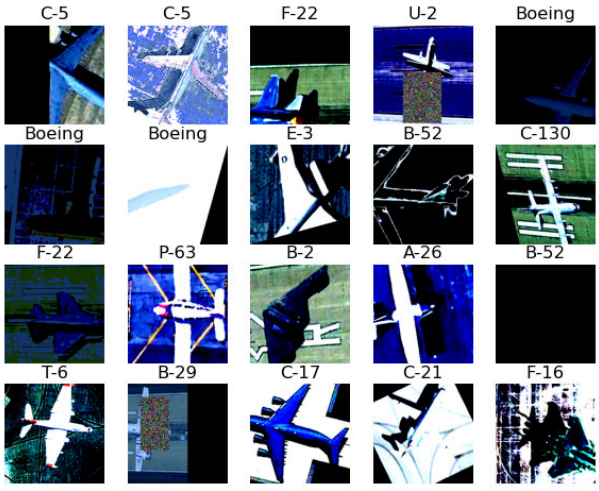


FIGURE 4. Sample of the training images with data augmentation.

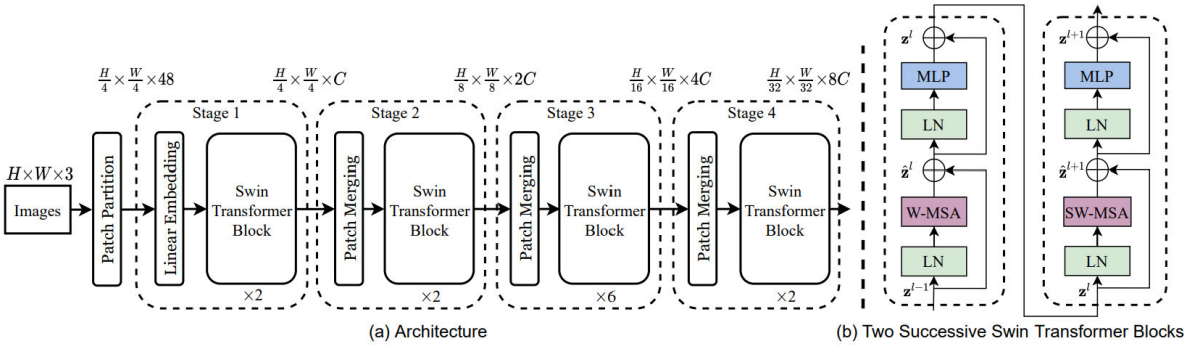


FIGURE 5. Swin Transformer architecture. Sourced from [29].

wings, or tail shapes. In addition to these data augmentations, we also shifted and scaled the brightness values to match ImageNet1k, and resized the images to 224 × 224 for Swin-Transformer and ResNet and 256 × 256 for Swin-MLP. This transformation also helped with pre-trained model convergence. The validation set images were only resized, and the mean/standard deviation shifted to ImageNet with further no data augmentation.

B. ResNet, SWIN TRANSFORMER, AND SWIN MLP

ResNet [3] is a family of deep convolutional neural networks with residual connections. These residual connections mitigated vanishing and exploding gradients, allowing for the construction of deeper neural networks. ResNet also makes use of Batch Normalization after each convolution layer. The ResNet family of convolutional neural networks is one of the most well-known architectures in computer vision, having been used as a neural backbone for a variety of classification, detection, and segmentation tasks. As such, we refer the readers to the original paper [3] for detailed descriptions of this architecture.

The basic building block of the Swin Transformer is the Swin Transformer Block composed of Multi-Layer Perceptron (MLP) and Multi-head Attention modules in both shifted window and standard configuration.

Fig.5 illustrates the Swin-Transformer architecture, which is composed of sequentially arranged Swin-Transformer Blocks interlaced with patch processing layers. The image patches are gradually reduced in height and width, but gain channels as they pass through the transformer blocks. H and W denote the original height and width of the image, respectively. C denotes feature dimension and is user-defined.

Three sequences q, k, v, are mapped through learned embedding layers, where $Q = W^q q$, $K = W^k k$, $V = W^v v$, for learned weight matrices W^q, W^k, W^v. The embedded sequences Q, K, V are then passed onto the attention layer. Q and K are multiplied together and passed to a softmax function. This step generates attention weights, which are used to scale the elements in V.

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

These attention layers usually are composed of multiple attention heads, each of which learns its own embedding matrices and attention weights. The outputs of each head are concatenated and passed through a final linear layer. The attention layer is also position agnostic. A positional encoding layer embeds the position of each token in the input sequence. Readers are referred to the original Transformer paper [24] for further details.

The attention mechanism was initially built for processing sequences. Thus, Transformers were not fit for image processing tasks. Dovovitskiy et al. [27] however adapted Transformers to images by treating images as a sequence of featurized image patches. Liu et al. then improved on this visual attention by introducing self-attention in shifted window configurations and hierarchical feature maps. The Swin-Transformer used relative positional encoding (2) via positional bias, which according to their ablation study, outperformed the traditionally used absolute positional encoding, as well as the self-attention with no positional information. The self-attention with relative positional bias is given by

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}} + B)V \quad (2)$$

where $B \in R^{P \times P}$, is the relative positional bias matrix for an attention window with P patches. Liu et al. [29] also created the Swin-MLP, by improving the MLP-Mixer [42] architecture with hierarchical feature mapping and the shifted window scheme. Mixer layers are comprised of a token-mixing MLP and a channel-mixing MLP, with layer normalization layers and residual connections intermixed. The Swin-MLP uses neither convolutions nor self-attention, relying solely on MLP-mixer layers, achieving only slightly worse results than a Swin-Tranformer of equivalent size for small models (around 20 million parameters). It could refer to the original papers for a detailed description of the architecture.

C. SINGLE IMAGE SUPER RESOLUTION

Super-resolution via Single Image Super-resolution (SISR) Networks was shown to increase the performance of deep learning models in remote sensing research to varying

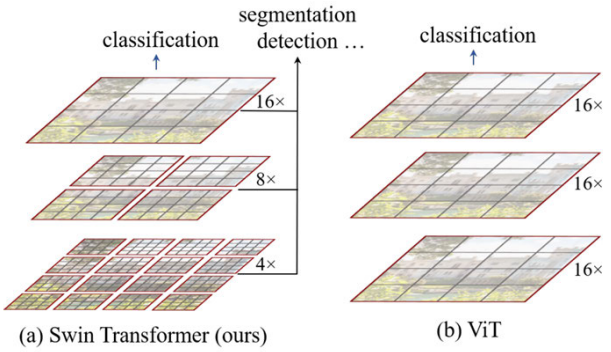


FIGURE 6. Hierarchical feature map, sourced from [29]. The Swin transformer built feature maps hierarchically, which increased the receptive field of each image patch in the latter layers. Furthermore, this limited the maximum sequence length input into attention layers, resulting in a linear complexity Transformer.

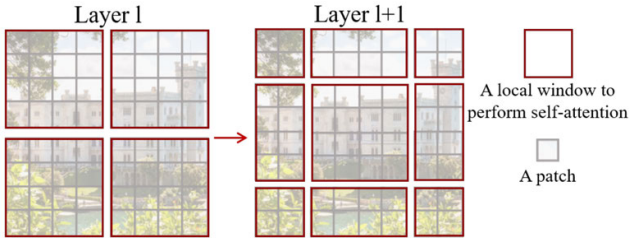


FIGURE 7. Shifted window attention, sourced from [29]. The shifted window attention scheme connected the disjoint attention windows in any fixed layer and according to the ablation study by the original authors drastically improved model performance.

degrees depending on the task. For the MTARSI dataset, when not using super-resolution, we bi-cubically scaled and cropped the images to 224 × 224. As shown in Fig.3, a non negligible portion of images had heights and widths less than 150 pixels. For these images, we believed super-resolution could benefit in enhancing the discernability of airplane features. Visual inspection showed that the bi-cubic upsampling of some of these ''small'' images resulted in visual artifacts in the form of pixelation.

For single-image super-resolution, we used the MSCA-RFANet [40], to super-resolve remote sensing images for semantic segmentation of buildings from remote sensing images, significantly outperforming bi-cubic interpolation. The MSCA-RFANet was based on the RFANet [39], a widely used single-image super-resolution network that used convolutions and spatial attention to generate accurate super-resolution results. The MSCA-RFANet [40] additionally included channel attention blocks in the trunk of the baseline RFANet, and achieved great results on remote sensing images. For the detailed architectures of these super-resolution networks, we refer the readers to the original papers.

D. TRAINING PARAMETERS AND ENVIRONMENT

Pytorch 19.0 compiled with CUDA 11.1 was used to write the training script. Benchmark models were trained under Stochastic Gradient Descent (SGD) with Nesterov Momentum (momentum factor = 0.1), under different learning rate schedules. Exact details are found in Section II-E. A dropout rate of 0.1 and drop-path rate of 0.2 were used for the Swin-Transformer and Swin-MLP models. The loss function used was soft-target categorical cross entropy due to target mix-up augmentation via alpha blending. Gradients with norms greater than 5.0 were clipped. Hardware specifications are i9-10900KF CPU and Nvidia GTX 3080 GPU.

E. BENCHMARK MODELS

For the benchmarks on the MTARSI dataset, we tested a variety of training procedures on ResNet-50, Swin-Transformer (Tiny), and Swin-MLP (Tiny). The Swin-Transformer (Tiny) used in our experiments is characterized by {3,6,12,24} heads in the four Transformer blocks, in order. The feature depths are {2,2,18,2}, in order. The Swin-MLP (Tiny) has {3,6,12,128} heads in the four Swin-MLP blocks, and has the same feature depths as above. We chose the models which were both the closest in numbers of parameters to ResNet-50 and also had available ImageNet pre-trained weights from the original authors [29]. The Swin-Transformer(Tiny), Swin-MLP (Tiny) and ResNet-50 models have 28, 23, and 23 million parameters, respectively. For our experiments, Swin-Transformer, and Swin-MLP refer specifically to these ''Tiny'' variants. A single linear layer was used as the classification head for all three models.

We also included the benchmark results from previous authors, which included baseline models from [7], BD-ELMNet [8], FGATR-Net [10], and the LinearSVM with PCA on features extracted from the author's CNN [9]. We note that the MTARSI dataset was not canonically divided into training and validation splits by the authors of the original paper. Such authors performed their own training and validation splitting.[1]

III. RESULTS

A. THE EFFECTS OF THE TRAINING PROCEDURE

1) TRANSFER LEARNING

Preliminary results showed that from-scratch Transformer models failed to converge using the AdamW [43] optimizer for some low learning rates and some weights initialization. To examine the effect of transfer learning using pre-trained ImageNet weights, we trained ResNet-50, Swin-Transformer, and Swin-MLP for 10 epochs using Stochastic Gradient Descent (SGD) with a cosine learning rate schedule, with 1 warmup epoch, with a maximum learning rate of 2e-3 and a minimum learning rate of 5e-6. We also mean-shifted and scaled our images to the mean and standard deviation of the ImageNet dataset. Table 1 shows the results of the transfer learning experiment. We noticed using pre-trained ImageNet weights significantly improved the convergence speed of models from all three architectures. This was especially true

[1]The training and validation split used in our experiments is available at https://github.com/kyle-gao/Swin-MTARSI

TABLE 1. Accuracy (%) of pre-trained vs. from-scratch models after 10 epochs.

From-scratch	Swin-Transformer	Swin-MLP	ResNet-50
Training Accuracy	22.2	18.4	15.1
Validation Accuracy	19.2	16.1	14.0
Pre-trained	Swin-Transformer	Swin-MLP	ResNet-50
Training Accuracy	88.9	84.6	88.1
Validation Accuracy	85.9	84.1	87.0

for Swin-Transformer and Swin-MLP, which without pre-trained weights, struggled to learn some weight initialization, optimizers, and learning rate schedules. In these failed training scenarios, we observed no loss function decrease.

2) LEARNING RATE AND LEARNING SCHEDULE

We used SGD with Nesterov momentum as the optimizer. We performed a convergence test to find a good baseline learning rate to train our models. Fig.8 shows the results of 20 training epochs at different learning rates for models with pre-trained ImageNet weights. For this experiment, we used a constant learning rate schedule. At learning rates of 5e-3 and 1e-2, both ResNet-50 and Swin-Transformer quickly achieved validation accuracies above 95% converging at around 98%.

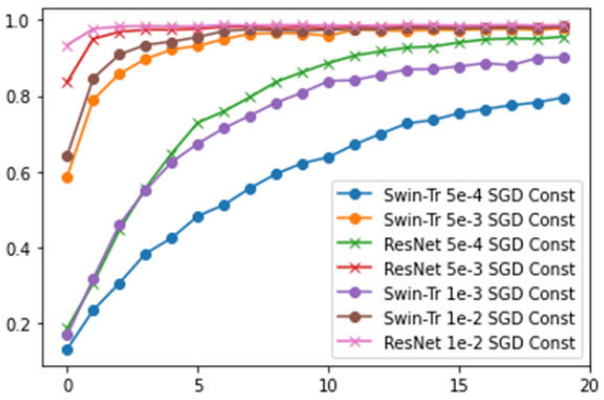


FIGURE 8. Validation accuracy for 20 epochs transfer learning experiments with Stochastic Gradient Descent (SGD) optimizer. All models used pre-trained ImageNet weights.

We noticed that models tended to converge to above 98.5 % validation accuracy, using both constant and cosine learning rate schedules within 20 epochs using a base learning rate of 5e-3, with faster convergence for constant schedules. However, for longer training experiments, we decided to use the cosine schedule based on experiments from previous authors [29], [31]. For the cosine schedule, the learning rate starts low to warm up the optimizer, then the high base learning rate would drive the model toward convergence faster, with the learning rate decreasing to control the training fluctuations at convergence. We achieved excellent results with the cosine learning rate schedule when training for 100+ epochs.

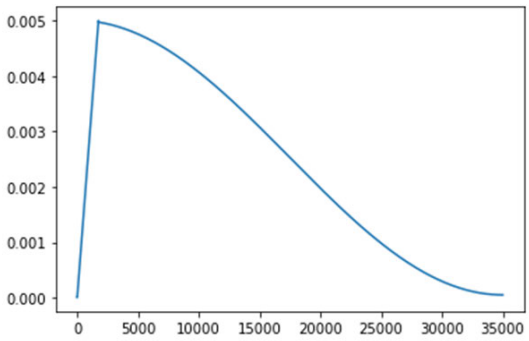


FIGURE 9. One cycle of cosine learning rate schedule as a function of gradient descent iterations for 100 epochs. Base learning rate = 5e-3, 10 epochs warm-up. These settings were used for data augmentation experiments.

TABLE 2. Accuracy (%) of models after 100 epochs with data augmentation vs. no data augmentation.

No augmentation	Swin-Transformer	Swin-MLP	ResNet-50
Training Accuracy	99.9	99.8	99.8
Validation Accuracy	98.7	98.7	98.8
Full augmentation	Swin-Transformer	Swin-MLP	ResNet-50
Training Accuracy	88.7	87.8	91.3
Validation Accuracy	98.9	98.2	98.8

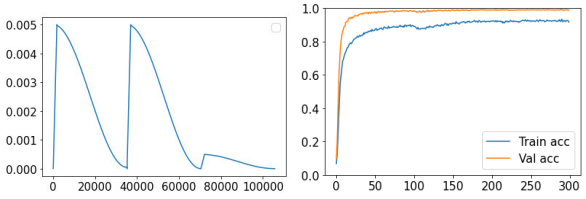
3) DATA AUGMENTATION

We noticed that by using the appropriate training schedule, the Swin models and ResNet-50 were able to reach very high validation accuracy (98.5%+) within 50 epochs, while the training accuracies reached 99.9 %. However, at these extremely high training accuracies (around 7 misclassifications for the 7045 image training set), we were not confident the model could learn any more from the training set data. With data augmentation, the training accuracy and loss converged more slowly for all three models. Moreover, the training accuracy was consistently lower than the validation accuracy, and slowly increased throughout the 100 epochs and beyond. Nonetheless, the validation set performance of all models were mutually similar after 100 epochs, whether or not using data augmentation. However, there was a large difference in training accuracy when using data augmentation as opposed to when not.

For the final data augmentation experiment, we trained the Swin-Transformer for 300 epochs using a cosine schedule with 3 cycles with a baseline learning rate of 5e-3 for the first 200 epochs, and 5e-4 for the last 100 epochs. We believed the Swin-Transformer to be the most promising of the model based on previous benchmarks on ImageNet1K [29]. After 150 epochs, up to 300 epochs, the validation accuracy hovered between 99.0% and 99.4%, despite this, the training accuracy kept slowly increasing from low to mid 90's to maximum of 93.4 %, which we believed would make the model more robust to out-of-distribution images.

B. MTARSI BENCHMARKS

As can be seen in Table 3, even without super-resolution, our best model significantly outperformed the original



(a) 3-cycle cosine schedule, 5e-3 baseline lr for epochs 0-199, 5e-4 baseline lr for epochs 200-299

(b) Accuracy curves for 300 epoch: training accuracy (blue) and validation accuracy (orange)

FIGURE 10. Swin-Transformer transfer learning for 300 epochs with heavy data augmentation: learning rate schedule and accuracy curves. When data augmentation was used, the training accuracy was always lower than the validation accuracy, and slowly increased.

TABLE 3. Model performance on the MTARSI benchmarks.

Models	Validation Accuracy
AlexNet [7]	85.6
VGG [7]	87.7
GoogLeNet [7]	86.6
ResNet [7]	89.6
DenseNet [7]	89.1
EfficientNet [7]	89.8
Zhao's method [8]	78.1
FGATR-Net [10]	93.8
SRARNet-Net [11]	93.4
LinearSVM(CNN-PCA) [9]	96.8
Swin-Transformer [300 Epochs No SR]	99.4
Swin-Transformer [300 Epochs With SR]	99.5

TABLE 4. Class-based performance of Swin-Transformer after 300 epochs on the validation set.

Class	IoU	Precision	Recall	Num Images
A-10	1.000	1.000	1.000	92
A-26	1.000	1.000	1.000	65
B-1	1.000	1.000	1.000	118
B-2	1.000	1.000	1.000	158
B-29	1.000	1.000	1.000	74
B-52	1.000	1.000	1.000	139
Boeing	0.943	0.955	0.987	150
C-130	0.989	0.994	0.994	176
C-135	0.993	1.000	0.993	139
C-17	0.991	1.000	0.991	117
C-21	0.984	1.000	0.984	124
C-5	0.991	0.991	1.000	113
E-3	0.991	0.991	1.000	109
F-16	0.990	0.990	1.000	95
F-22	0.991	0.995	0.995	215
KC-10	0.968	0.992	0.976	123
P-63	1.000	1.000	1.000	79
T-43	1.000	1.000	1.000	86
T-6	1.000	1.000	1.000	65
U-2	0.962	0.990	0.971	103

TABLE 5. Accuracy (%) of models after 100 and 300 epochs with data augmentation and 2× super-resolved images.

100 Epochs	Swin-Transformer	Swin-MLP	ResNet-50
Training Accuracy	90.2	90.0	92.8
Validation Accuracy	99.0	98.4	99.1
300 Epochs	Swin-Transformer	Swin-MLP	ResNet-50
Training Accuracy	94.8	92.1	93.9
Validation Accuracy	99.5	99.1	99.4

benchmarks, as well as the previously published models for the MTARSI dataset. We also note that the ResNet-50 shown in Table 1 also significantly outperforms the ResNet-50 trained by Wu et al. [7]. Wu used a 1e-3 constant learning rate with decay, dividing by 10 every epoch (unspecified optimizer). We believed Wu et al. [7] underfit their model; instead of the model converging, their learning rate vanished, remaining at reasonable learning rates for only two to three epochs. We confidently believed that our models learned the MTARSI classification task. However, due to the low variation in the dataset and potential issues with training/validation set separability (see Subsection IV-B), we could not confirm the generalizability of models trained on MTARSI data to new out-of-distribution data based on MTARSI experiments alone. As such, we performed additional out-of-distribution testing (see Subsection III-D).

Table 4 shows the class-based metrics for our Swin-Transformer. As can be seen, some classes are more easily classified than others. Certain aircraft such as the B-2 has a very distinctive shape, which is easy to classify. The model noticeably struggled the most with "Boeing", which was a class into which the MTARSI authors assigned multiple types of commercial airlines.

C. SUPER-RESOLUTION EXPERIMENTS

For the MTARSI dataset in absence of super-resolution, we scaled the images via bi-cubic interpolation such that the smallest dimension was greater than 224 while maintaining the aspect ratio. We then center-cropped the images to obtain 224 × 224 images. Most images were scaled by a factor of 0.8-1.5, which we considered to be reasonable based on visual inspection. However, a small fraction of images was upsampled by a factor greater than 1.5. In these scenarios, we could visually discern pixelation.

For the super-resolution experiments, we super-resolved the images by a factor of 2× using MSCA-RFANet prior to bi-cubic scaling and cropping of images to the pre-determined sizes 224 × 224 or 256 × 256. The super-resolution visibly improved image quality in some cases, as shown in Fig.11. We compared the results of ResNet50, Swin-MLP, and Swin-Transformer using the cosine learning rate schedules, as shown in Fig.9 and Fig.10.

Comparing Table 2 and Table 5, after 100 epochs, the super-resolution slightly improved the convergence of all three models, with Swin-Transformer gaining 0.1%, Swin-MLP gaining 0.1%, and ResNet50 gaining 0.2% overall accuracy, respectively. We also noted a similar overall accuracy increase after 300 training epochs for the Swin-Transformer.

We also observed changes in class-based metrics for Swin-Transformer after 300 epochs when using super-resolution (Table 4 vs. Table 6). By using super-resolution, we observed a slight reduction in the class-based IoU of "B-52", "Boeing", "C-135", "KC-10", and "U-2". We also noted a slight increase in the class-based IoU of "C-130", "C-21", "C-5",

(a) MTARSI "Boeing" class image "0-48.jpg" original image size: 93 by 93

(b) MTARSI "Boeing" class image "0-48.jpg" super resolved to 186 by 186

FIGURE 11. Image super-resolution results. The super-resolved image (b) is smoother (less pixelated) than the native 93 × 93 image (a) when both are then scaled to the same size. This is especially noticeable at the boundary between the wings/fuselage and the background.

"E-3", "F-16", and "F-22". We would like to highlight the changes in "Boeing", "F-16", and "F-22" for discussion.

D. OUT-OF-DISTRIBUTION TESTING

The extremely high training and validation scores could be because of the high degree of correlations between training and validation images. The MTARSI dataset was taken from 33 airports. Some classes are extremely rare in real life. As such, we suspect many images were of the same planes, taken at different times under different imaging conditions and with different backgrounds. Furthermore, the majority of the 9385 images were generated via data augmentation (isometries and background shifts), thus any single plane appears multiple times. In this scenario, despite using random splitting of the training and validation set, it is possible to overfit the validation set without every training on a single validation image.

To examine this potential overfitting, we performed additional testing on 36 additional out-of-distribution test images. 20 of these 36 images were from a vertical top-down perspective, which we considered to be "easy images" directly comparable to MTARSI images. The other 16 images were from a top-down view, at an angle, and were considered "hard" images. No test images were taken from a "looking up" point of view, showing the underneath of the aircraft. All test images showed the entirety of the aircraft.

TABLE 6. Class-based performance of Swin-Transformer with super resolution after 300 epochs on the validation set.

Class	IoU	Precision	Recall	Num Images
A-10	1.000	1.000	1.000	92
A-26	1.000	1.000	1.000	65
B-1	1.000	1.000	1.000	118
B-2	1.000	1.000	1.000	158
B-29	1.000	1.000	1.000	74
B-52	↓ 0.993	1.000	0.993	139
Boeing	↓ 0.937	0.943	0.993	150
C-130	↑ 1.000	1.000	1.000	176
C-135	↓ 0.986	0.986	1.000	139
C-17	0.991	1.000	0.991	117
C-21	↑ 0.992	1.000	0.992	124
C-5	↑ 1.000	1.000	1.000	113
E-3	↑ 1.000	1.000	1.000	109
F-16	↑ 1.000	1.000	1.000	95
F-22	↑ 1.000	1.000	1.000	215
KC-10	↓ 0.960	0.992	0.967	123
P-63	1.000	1.000	1.000	79
T-43	1.000	1.000	1.000	86
T-6	1.000	1.000	1.000	65
U-2	↓ 0.961	1.000	0.961	103

TABLE 7. Out-of-distribution test score out of 36 (SR denotes models trained on super-resolved images).

Models	Data Augmentation	Score
Swin Transformer 300 Epochs	Yes	17
Swin Transformer(SR) 300 Epochs	Yes	17
ResNet-50(SR) 300 Epochs	Yes	17
Swin-MLP(SR) 300 Epochs	Yes	16
Swin Transformer 100 Epochs	Yes	15
Swin-MLP 100 Epochs	Yes	12
ResNet-50 100 Epochs	Yes	14
Swin Transformer 100 Epochs	No	14
Swin-MLP 100 Epochs	No	11
ResNet-50 100 Epochs	No	15

As shown in Table 7, despite the excellent validation scores, models trained on the MTARSI dataset did not generalize well to new data. The Swin-Transformer used in the benchmark table, trained with heavy data augmentation showed the best performance. With super-resolution, after 300 epochs, ResNet-50 and Swin-MLP also achieved similar results. However, $\frac{17}{36}$ correct classification despite having approximately 99.5% validation accuracy indicates that the validation score did not reflect out-of-distribution performance, and that the MTARSI dataset is unsuited for training generalizable aircraft recognition models.

IV. DISCUSSIONS

A. MTARSI DATASET LABELING PROBLEMS

The Swin-Transformer achieved an extremely high validation accuracy of around 99.4% after 300 epochs, even without super-resolution. Even without considering dataset errors,

these results also indicate that these deep learning models were likely at the upper limit of what this dataset can benchmark (in terms of model complexity), and that more complex aircraft recognition datasets are required to benchmark bigger and more complex aircraft recognition models. We believe these results indicated that the model has ''solved'' the MTARSI aircraft recognition task (but not aircraft classification in general).

A visual inspection of the dataset showed that it had some obvious labeling issues. We focus our discussions on potential issues with the classes ''Boeing'', ''C-17'', ''T-43'', ''F-16'' and ''F-22'', but there are potentially similar issues with classes we were less familiar with. The Boeing class contained many different models of commercial airliners, some of which were quad-engine planes, while others were twin-engine planes. We believed this class should be refined for the future, separating the different models (eg. Boeing 747) within the manufacturer ''Boeing''. In fact, the C-17 and the T-43 models were their own classes in the MTARSI dataset, but are manufactured by Boeing, with the Boeing T-43 model being a modified Boeing 737 variant. We believed the construction of the MTARSI ''Boeing'' class resulted in problems with class separability.

Many images in the MTARSI class ''C-17'' were mislabeled (Fig.13), where the left image does not have the engine, tail, and fuselage shape of a C-17 plane. The MTARSI classes ''F-22'' (Fig.12) and ''F-16'' were also often mislabeled, with many examples of ''F-15'', ''F-16'', and ''F-18'' planes intermixed.

In light of our knowledge of the mislabeling of certain classes, a few interesting results indicated the potential overfitting of the validation set. As aforementioned, we believed the ''Boeing'' class to be problematic since it contained images of both twin-engine (eg. Boeing 737), and quad-engine (eg. Boeing 747) commercial airliners. On the other hand, the ''T-43'' class referred to the Boeing T-43, a modified Boeing 737 used by the United States Air Force for training purposes with an indistinguishable airframe from the commercial Boeing 737. The fact that the ''T-43'' class received perfect classification scores despite being indistinguishable top-down from certain planes in the ''Boeing'' class indicated potential issues with the validation set. We also noted the high accuracy of the classes ''F-16'' and ''F-22'', which increased under super-resolution, as shown in Tables 4 and 6. From visual inspection, we were confident that out of the 95 validation set images in the ''F-16'' class, 20 were not images of the F-16 fighter jets. Moreover, out of the 215 ''F-22'' class images in our validation set, we believed 151 images were not of F-22 fighter jets. Despite this, we achieved extremely high class-based scores on ''F-22'' and ''F-16'', which became perfect after we upsampled the images via super-resolution. Super-resolution enhances the discriminative features of images. Therefore, this behavior strongly suggested that the model instead memorized which planes belonged to the MTARSI assigned classes, rather than learning the correct shape and pattern matching based

(a) MTARSI "F-22" class image "10-115.jpg" (b) MTARSI "F-22" class image "10-128.jpg"

FIGURE 12. Possible mislabeling of "F-22" in MTARSI. the two aircraft have different wing shapes (which also do not match those of an F-22). (a) is likely an F-15 based on the wing (F-16-like) and tail shape (two vertical stabilizers), and (b) is likely an F-18.

(a) MTARSI "C17" class image "7-30.jpg" (b) MTARSI "C17" class image "7-266.jpg"

FIGURE 13. Possible mislabeling of "C-17" in MTARSI. Despite both being quad-engine airplanes, these two are of different fuselages, engines, and tail shapes. (a) is potentially a Boeing 747 and (b) is a C-17.

classification. This in return strongly suggested overfitting of the validation set, despite the models never having been trained on them. We further confirmed this with our out-of-distribution testing.

B. RESULTS AND PROBLEMS WITH MTARSI DATASET GENERALIZABILITY

Our results showed that the effect of a good training schedule was much greater than changing models (as long as the model size was similar). Our ResNet-50 performed very similarly to our Swin-models, and greatly outperformed Wu's [7] ResNet-50 from the original MTARSI paper, as well as all models by previous authors [8], [9], [10], [11] on the MTARSI dataset. We attribute the performance of our models to (1) having chosen a good training regiment, and (2) using pre-trained ImageNet weights and normalizing our dataset to ImageNet's mean and standard deviation (many previous authors have used pre-trained weights but have not performed the additional normalization step). We would like to point out the recent study [31] which corroborated this finding. In the aforementioned paper, the authors trained a ResNet-50 that

is ImageNet1k top-1 accuracy greatly exceeded the previous ResNet-50 score (80.4 vs. 75.3 %), and was comparable to Swin-Transformer's (Tiny) 81.2 %. The training parameters of the authors were very similar to ours, with the same baseline learning rate, cosine scheduler, and similar data pipeline.

Most importantly, we found out using out-of-distribution testing data that the MTARSI dataset is unsatisfactory for the training and evaluation of aircraft classification algorithms for real-life applications. The validation set results simply did not translate to out-of-distribution data to a satisfactory level. This was likely due to (1) the low unique aircraft diversity in the dataset and (2) the construction of the dataset, where Wu et al. [7] had not canonically split the dataset before applying their data augmentation to generate new images (causing subsequent authors to use cross-contaminated training and validation sets). This resulted in potential overfitting of the validation set while only training on the training set, since both sets likely contained the same plane artificially placed on different backgrounds. This cross-contamination also made the judgment of overfitting impossible when using MTARSI data alone.

C. SUGGESTIONS

We suggest future authors consider alternative datasets for aircraft classification for research, and more importantly, for real-life applications. Our results showed training procedure was more important than architecture when considering models of similar sizes. Moreover, we suggest future authors carefully investigate training procedure optimization before building new models. Swin-Transformers has in the recent past shown to be very promising. However, the fact that the Swin-Transformer we trained only marginally exceeded ResNet-50 should not be understood to reflect its true potential, as we believed both models have achieved the upper limit of MTARSI dataset scores. We also note that we only used the Tiny variant of Swin-Transformer and Swin-MLP due to computational limitations. In future experiments, We are considering testing the limits of the Swin-Transformer and improving its architecture while training on more complex aircraft classification tasks, such as the Aircraft Context Dataset [14].

For improving the MTARSI dataset, we suggest reconstructing it without including augmented images, as we believe data augmentation should be part of the training data pipeline, and not the dataset construction. This recommendation broadly applies to dataset construction in general, since the choice of data augmentation depends on the needs of the user. Moreover, it is easy to augment data, but it can be difficult to recover original images from augmented data. The MTARSI labels must be carefully examined for errors. Given the severity of the labeling errors, it could be more reasonable to relabel the images from scratch. We also suggest the ''Boeing'' class be refined either into specific models, or into general ''Quad-engine airliner'' and ''Twin-engine airliner'' into which the T-43 planes should be merged. The labeling errors of MTARSI classes ''F-16'' and ''F-22'' were numerous. We suggest the creation of ''F-15'' and ''F-18'' classes for MTARSI, which would contain images previously misclassified into ''F-16'' and ''F-22''. For testing purposes, we suggest the MTARSI dataset, and future datasets are constructed with canonically split training/validation/testing sets, so that future users can train and test on the same splits. Since modern state-of-the-art methods often only differ from one-another by a fraction of a percent in terms of accuracy scores, controlling for the testing dataset split should result in a less biased benchmark. It can be also useful to include an out-of-distribution test set created entirely separately from the training/validation set, either from photo-realistic simulation images or from independently taken remote sensing images.

V. CONCLUSION

By carefully selecting our training procedure, we have achieved state-of-the-art results on the MTARSI dataset with a 99.4 % validation accuracy on a ∼2000 image validation set, greatly exceeding the results of the previous authors. By making use of pre-trained ImageNet weights, ResNet-50, Swin-Transformer (Tiny), and Swin-MLP (Tiny) were all able to exceed the previously published results using our training procedures. We further improved our results to 99.5% validation accuracy when using a state-of-the-art super-resolution method to upsample our images. We also found that for this dataset, the training procedure was more important than model selection. However, we noticed that the MTARSI dataset has many issues. For example, the number of unique aircraft is low, some images are mislabeled, some classes are problematic in scope, and most importantly, the authors performed data augmentation to generate the 9385 images without canonically splitting the training set and validation set. The validation and training sets generated via any random split would likely be cross-contaminated and contain the same aircraft, except under different augmentation. These data augmentation also artificially raised the number of data samples which could mislead users about the dataset's variety. As such, we performed additional out-of-distribution testing with challenging images taken from various sources, and confirmed that the dataset's validation score did not generalize to true performance. We would like to caution against using the MTARSI dataset for practical applications. Future aircraft classification studies should investigate whether other aircraft datasets generalize to out-of-distribution data, as well as investigate robust generalizable models by training on multiple datasets.

REFERENCES

[1] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, ''ImageNet: A large-scale hierarchical image database,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[2] A. Krizhevsky, V. Nair, and G. Hinton. *CIFAR-10 (Canadian Institute for Advanced Research)*. [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html

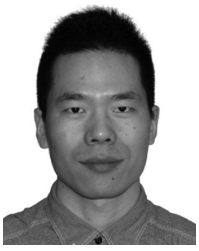
[3] K. He, X. Zhang, S. Ren, and J. Sun, ''Deep residual learning for image recognition,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[4] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[5] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Can semantic labeling methods generalize to any city? The inria aerial image labeling benchmark," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 3226–3229.
[6] A. E. Maxwell, T. A. Warner, B. C. Vanderbilt, and C. A. Ramezan, "Land cover classification and feature extraction from national agriculture imagery program (NAIP) orthoimagery: A review," *Photogramm. Eng. Remote Sens.*, vol. 83, no. 11, pp. 737–747, Nov. 2017.
[7] Z.-Z. Wu, S.-H. Wan, X.-F. Wang, M. Tan, L. Zou, X.-L. Li, and Y. Chen, "A benchmark data set for aircraft type recognition from remote sensing images," *Appl. Soft Comput.*, vol. 89, Apr. 2020, Art. no. 106132.
[8] B. Zhao, W. Tang, Y. Pan, Y. Han, and W. Wang, "Aircraft type recognition in remote sensing images: Bilinear discriminative extreme learning machine framework," *Electronics*, vol. 10, no. 17, p. 2046, Aug. 2021.
[9] F. Azam, A. Rizvi, W. Z. Khan, M. Y. Aalsalem, H. Yu, and Y. B. Zikria, "Aircraft classification based on PCA and feature fusion techniques in convolutional neural network," *IEEE Access*, vol. 9, pp. 161683–161694, 2021.
[10] W. Liang, J. Li, W. Diao, X. Sun, K. Fu, and Y. Wu, "FGATR-Net: Automatic network architecture design for fine-grained aircraft type recognition in remote sensing images," *Remote Sens.*, vol. 12, no. 24, p. 4187, Dec. 2020.
[11] W. Tang, C. Deng, Y. Han, Y. Huang, and B. Zhao, "SRARNet: A unified framework for joint superresolution and aircraft recognition," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 327–336, 2020.
[12] S. Maji, E. Rahtu, J. Kannala, M. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," 2013, *arXiv:1306.5151*.
[13] J. Shermeyer, T. Hossler, A. V. Etten, D. Hogan, R. Lewis, and D. Kim, "RarePlanes: Synthetic data takes flight," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 207–217.
[14] D. Steininger, V. Widhalm, J. Simon, A. Kriegler, and C. Sulzbacher, "The aircraft context dataset: Understanding and optimizing data variability in aerial domains," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 3823–3832.
[15] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst. (GIS)*, 2010, pp. 270–279.
[16] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.
[17] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.
[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 84–90.
[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[20] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2017, pp. 2961–2969.
[21] R. Johnson and T. Zhang, "Deep pyramid convolutional neural networks for text categorization," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2017, pp. 562–570.
[22] L. Chen, H. Zhang, J. Xiao, L. Nie, J. Shao, W. Liu, and T.-S. Chua, "SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5659–5667.
[23] C. Tian, Y. Xu, Z. Li, W. Zuo, and H. Liu, "Attention-guided CNN for image denoising," *Neural Netw.*, vol. 124, pp. 117–129, Apr. 2020.
[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[25] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, and S. Agarwal, "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 1877–1901.
[26] W. Fedus, B. Zoph, and N. Shazeer, "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity," 2021, *arXiv:2101.03961*.
[27] 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 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[28] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009.
[29] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted Windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10012–10022.
[30] K. Chen, W. Ouyang, C. C. Loy, D. Lin, J. Pang, J. Wang, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, and J. Shi, "Hybrid task cascade for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4974–4983.
[31] R. Wightman, H. Touvron, and H. Jégou, "ResNet strikes back: An improved training procedure in timm," 2021, *arXiv:2110.00476*.
[32] I. Marivani, E. Tsiligianni, B. Cornelis, and N. Deligiannis, "Joint image super-resolution via recurrent convolutional neural networks with coupled sparse priors," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2020, pp. 868–872.
[33] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1646–1654.
[34] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1637–1645.
[35] X. Mao, C. Shen, and Y.-B. Yang, "Image restoration using very deep convolutional encoder–decoder networks with symmetric skip connections," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1–9.
[36] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3147–3155.
[37] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 286–301.
[38] X. Dai, M. Ding, W. Zhang, Z. Xuan, J. Liang, D. Yang, Q. Zhang, B. Su, H. Zhu, and X. Jia, "Anti-inflammatory effects of different elution fractions of Er-Miao-San on acute inflammation induced by carrageenan in rat paw tissue," *Med. Sci. Monitor, Int. Med. J. Exp. Clin. Res.*, vol. 25, p. 7958, Apr. 2019.
[39] J. Liu, W. Zhang, Y. Tang, J. Tang, and G. Wu, "Residual feature aggregation network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2359–2368.
[40] H. He, K. Gao, W. Tan, L. Wang, N. Chen, L. Ma, and J. Li, "Super-resolving and composing building dataset using a momentum spatial-channel attention residual feature aggregation network," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 111, Jul. 2022, Art. no. 102826.
[41] R. Wightman. (2019). *Pytorch Image Models*. [Online]. Available: https://github.com/rwightman/pytorch-image-models
[42] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit, and M. Lucic, "MLP-Mixer: An all-MLP architecture for vision," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 1–12.
[43] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.

KYLE (YILIN) GAO (Graduate Student Member, IEEE) received the bachelor's degree in mathematics from the University of Waterloo, Canada, in 2016, and the master's degree in physics from the University of Victoria, Canada, in 2020. He is currently pursuing the Ph.D. degree in systems design engineering with the Geospatial Sensing and Data Intelligence Group, University of Waterloo. He has published papers in the *International Journal of Applied Earth Observation and Geoinformation and Geomatica*. His research interests include computer vision and deep learning.

HONGJIE HE received the B.Sc. degree in geomatics from the China University of Petroleum, Qingdao, in 2016, and the M.Sc. degree in cartography and geographic information systems from Lanzhou University, China, in 2019. He is currently pursuing the Ph.D. degree in geography specializing in applied earth observations with the Geospatial Sensing and Data Intelligence Group, University of Waterloo, Canada. He has published papers in the *International Journal of Applied Earth Observation and Geoinformation*, *Canadian Journal of Remote Sensing*, and *Geomatica*, and flagship conferences, including IGARSS and ISPRS. His research interests include AI-based algorithms and software tools for information extraction from earth observation images.

DENING LU received the B.Sc. and M.Sc. degrees in electrical engineering from the Nanjing University of Aeronautics and Astronautics (NUAA), China, in 2018 and 2021, respectively. He is currently pursuing the Ph.D. degree in systems design engineering with the Geospatial Sensing and Data Intelligence Group, University of Waterloo, Canada. He has published papers in the IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT and ICCV. His research interests include 3D point cloud processing and deep learning.



LINLIN XU (Member, IEEE) received the B.Eng. and M.Sc. degrees in geomatics engineering from the China University of Geosciences, Beijing, China, in 2007 and 2010, respectively, and the Ph.D. degree in geography from the Department of Geography and Environmental Management, University of Waterloo, Waterloo, ON, Canada, in 2014. He is currently a Research Assistant Professor with the Department of Systems Design Engineering, University of Waterloo. He has published various papers on high-impact remote sensing journals and conferences. His research interests include hyperspectral and synthetic aperture radar data processing and their applications in various environmental applications.

LINGFEI MA (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in geomatics engineering from the University of Waterloo, Waterloo, ON, Canada, in 2015, 2017, and 2020, respectively. He is currently an Assistant Professor with the School of Statistics and Mathematics, Central University of Finance and Economics, Beijing, China. He has published more than 30 papers in refereed journals and conferences, including IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, *ISPRS Journal of Photogrammetry and Remote Sensing*, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and IEEE CVPRW. His research interests include autonomous driving, mobile laser scanning, intelligent processing of point clouds, 3D scene modeling, and machine learning. He was a recipient of the 2020 National Best Ph.D. Thesis Award granted by the Canadian Remote Sensing Society. He serves as the Guest Editor for *International Journal of Applied Earth Observation and Geoinformation*.

JONATHAN LI (Senior Member, IEEE) received the Ph.D. degree in geomatics engineering from the University of Cape Town, South Africa, in 2000. He is currently a Professor of geomatics and systems design engineering with the University of Waterloo, Canada. He has supervised more than 120 master's and Ph.D. students as well as postdoctoral fellows to completion and coauthored more than 490 publications, more than 320 of which were published in refereed journals, including the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING, *ISPRS-JPRS*, *RSE*, and *JAG*. He has also published papers in flagship conferences in computer vision and AI, including CVPR, AAAI, and IJCAI. His main research interests include AI-based information extraction from earth observation images and LiDAR point clouds as well as 3D vision and GeoAI. He is a fellow of the Canadian Academy of Engineering and the Engineering Institute of Canada. He is the Editor-in-Chief of the *International Journal of Applied Earth Observation and Geoinformation* (JAG) and the Associate Editor of the IEEE Transactions on Geoscience and Remote Sensing, IEEE Transactions on Intelligent Transportation Systems, and *Canadian Journal of Remote Sensing*.