Transfer Learning Architecture Selection for Remote Sensing Scene Classification

Akhiyar Waladi Universitas Jambi
Jambi, Indonesia
akhiyar.waladi@unja.ac.id

***Abstract*—Most published comparisons of deep learning architectures for remote sensing scene classification differ in their training protocols, making it difficult to determine whether reported accuracy gaps reflect genuine architectural advantages or merely differences in hyperparameters and augmentation. In this paper we evaluate eight models from three architecture families (classical CNNs, vision transformers, and a modernized CNN) on EuroSAT (Sentinel-2, 10 classes) and UC Merced (aerial imagery, 21 classes), training all of them with exactly the same ImageNet-pretrained weights, optimizer, augmentation pipeline, and early-stopping criterion. The highest accuracy was achieved by ConvNeXt-Tiny on EuroSAT (99.11%) and by Swin Transformer on UC Merced (99.76%), but no single architecture family dominated both benchmarks. Across all eight models the accuracy spread was less than half a percentage point on EuroSAT and under two points on UC Merced, and McNemar's test showed that most pairwise differences were not statistically significant. The smallest model we tested performed on par with architectures sixteen times its size. These findings indicate that, for standard scene classification with transfer learning, the shared pretrained initialization and fine-tuning protocol exert a stronger influence on accuracy than the choice of backbone architecture.**

***Index Terms*—Scene classification, remote sensing, deep learning, convolutional neural networks, vision transformers, transfer learning, EuroSAT, UC Merced**

I. Introduction

ASSIGNING a land-use label to a satellite or aerial image patch is one of the older problems in remote sensing. Deep neural networks now handle it far better than anything that came before [1], [2], and the labels they produce feed straight into national land cover inventories and change-monitoring programs.

The pre-deep-learning workflow was tedious. An analyst would engineer features by hand (color histograms, texture statistics, bag-of-visual-words vectors), feed them to an SVM, and hope for the best [3]. Performance was acceptable when the number of classes was small but fell apart once datasets like NWPU-RESISC45 pushed past 40 categories [4].

Convolutional neural networks eliminated most of that manual effort. VGGNet [5], ResNet [6], DenseNet [7], and their descendants learn hierarchical features directly from raw pixel values, and the common practice of initializing them with ImageNet [8] weights before fine-tuning on remote sensing data has proven extremely effective [9], [10]. The reason is straightforward: labeled satellite imagery is expensive to produce, so letting the network start from features it already learned on millions of natural photographs saves a great deal of training effort [11], [12].

Vision transformers appeared next, importing the self-attention mechanism from natural language processing [13] into image classification by treating each image as a grid of patch tokens. ViT [14] and Swin Transformer [15] soon reached or surpassed CNNs on large-scale benchmarks, and remote sensing researchers adopted them almost immediately [16], [17]. Around the same time, Liu et al. [18] demonstrated with ConvNeXt that a purely convolutional design, when equipped with training recipes borrowed from transformers, could recover the same accuracy without any attention mechanism at all.

Practitioners now face three competing architecture families, each with published evidence of superiority: classical CNNs, vision transformers, and modernized CNNs. The trouble is that most existing comparisons evaluate only two or three models at a time and rarely standardize the training protocol across them. One paper may conclude that ViT outperforms ResNet; another, using a different learning rate schedule or a different augmentation policy, reaches the opposite conclusion. It is often impossible to tell whether the reported gap comes from the architecture or from the experimental setup.

We set out to resolve this ambiguity by running a controlled experiment. Eight architectures spanning all three families were initialized with the same ImageNet-pretrained weights, trained with the same optimizer and augmentation pipeline, and evaluated on two datasets that differ substantially in spatial resolution and number of classes. We then applied McNemar's test [19] to every pair of models to determine which accuracy gaps, if any, were statistically meaningful. Parameter counts and wall-clock training times were also recorded so that efficiency could be compared alongside accuracy.

The outcome was not what we anticipated. All eight models ended up within a narrow accuracy band on both benchmarks, and the majority of pairwise differences did not survive McNemar's test. EfficientNet-B0, the smallest architecture we tested (5.3 M parameters), performed on par with models carrying up to sixteen times as many weights. ViT-B/16, by far the largest at 86.6 M parameters, finished near the bottom of the ranking on both datasets. The data point toward a simple conclusion: when transfer learning from ImageNet is the starting point, the shared pretrained features and the fine-tuning protocol overshadow the architectural differences.

II. Related Work

A. CNN-Based Scene Classification

The residual learning framework of He et al. [6] introduced identity shortcut connections that allow the optimizer to learn only the deviation from the input at each block. Training remained stable even at 100+ layers, and the 50- and 101-layer configurations rapidly became the dominant backbones in remote sensing classification. Huang et al. [7] pursued a different strategy: each layer in their DenseNet receives as input the concatenated feature maps of every preceding layer, a form of aggressive feature reuse that keeps individual channel counts low while still rivaling much wider residual networks in accuracy. The EfficientNet family of Tan and Le [20] contributed another idea, compound scaling, which enlarges network depth, width, and input resolution simultaneously through a single coefficient rather than tuning each axis independently; the baseline model, discovered via neural architecture search, established a new Pareto frontier for accuracy versus computational cost on ImageNet. From an application standpoint, the systematic comparison by Nogueira et al. [9] across six architectures and three aerial scene datasets demonstrated that fine-tuning ImageNet-pretrained weights reliably outperforms training from random initialization, a result that has been confirmed repeatedly since [2], [21].

B. Transformer-Based Approaches

The Vision Transformer (ViT) of Dosovitskiy et al. [14] abandoned convolutions altogether. An image is partitioned into non-overlapping 16×16-pixel tiles; each tile is projected into a fixed-length embedding and the full sequence is processed by a standard multi-head self-attention encoder. Global attention means that any token can interact with any other, which is attractive for remote sensing scenes where relevant objects (say, an airplane and a runway) may lie far apart in the image [17]. The drawback is quadratic memory growth with the number of tokens. Liu et al. [15] sidestepped this limitation by confining attention to small local windows and cyclically shifting the window boundaries between layers so that adjacent windows can exchange information without incurring the quadratic cost. Despite this restriction, the resulting Swin Transformer produces hierarchical, multi-scale feature maps that function much like those of a traditional convolutional backbone. Hong et al. [16] later showed that the same attention-based architecture generalizes naturally to hyperspectral inputs with dozens of bands.

C. Modernized CNNs

Liu et al. [18] performed what amounts to an ablation study in reverse: beginning from a standard ResNet, they incrementally adopted transformer-inspired modifications (a patchify stem with 4×4 non-overlapping convolutions, 7×7 depthwise kernels sized to match Swin's local window, an inverted bottleneck with 4× channel expansion, and layer normalization with GELU replacing batch normalization and ReLU). Crucially, self-attention was never introduced at any stage. The resulting ConvNeXt model, still entirely convolutional, equaled or surpassed Swin Transformer on ImageNet classification, COCO detection, and ADE20K segmentation. The implication for our study is direct: if borrowing a handful of design choices is sufficient to close the CNN–transformer gap on natural images, there is little reason to expect that gap to persist on remote sensing scenes either.

D. Benchmark Datasets

Yang and Newsam [3] assembled the first publicly available high-resolution scene dataset by manually cropping 256×256 aerial patches from the USGS National Map Urban Area Imagery collection. The dataset contains 100 images per class across 21 land-use categories at 0.3 m spatial resolution. Helber et al. [22] later introduced a medium-resolution counterpart built entirely from freely available Sentinel-2 acquisitions spanning 34 European countries; its 27,000 geo-referenced patches (64×64 pixels, 10 m ground sampling distance, 10 classes) made it the first large-scale scene benchmark derived from operational satellite data. Larger alternatives have since appeared. NWPU-RESISC45 [4] offers 31,500 images over 45 classes; AID [23] contains 10,000 images over 30 classes. Yet EuroSAT and UC Merced persist as standard benchmarks because their manageable size permits exhaustive multi-model experiments within a single GPU budget. We picked these two because they differ on almost every axis: 0.3 m aerial photographs versus 10 m satellite imagery, fine-grained urban classes versus broad land-cover categories. An architecture that wins on both gives a stronger signal than any single-dataset comparison.

III. Methodology

Fig. 1 outlines the experimental pipeline. Both datasets go through the same preprocessing before being fed to all eight architectures under identical training settings. Models are compared on accuracy, statistical significance, error patterns, and training cost.

A. Datasets

1) EuroSAT: EuroSAT [22] contains 27,000 geo-referenced Sentinel-2 patches in 10 land-use categories. Each patch is 64×64 pixels at 10 m ground sampling distance, covering natural covers (Forest, SeaLake, River), agricultural types (AnnualCrop, PermanentCrop, Pasture), and built-up areas (Highway, Industrial, Residential). We use the RGB bands only, since all eight architectures expect three-channel input. Fig. 2 (left) shows one sample per class.

2) UC Merced Land Use: The UC Merced dataset [3] has 2,100 aerial images at 0.3 m resolution, split evenly across 21 classes (100 images each, 256×256 pixels). The fine resolution means individual buildings, tennis courts, and storage tanks are clearly visible, but it also means that classes like denseresidential, mediumresidential, and sparseresidential differ only in the spacing between structures, which makes them easy to confuse. Samples from both datasets appear in Fig. 2.

3) Data Partitioning: We applied an 80/20 stratified random split with a fixed seed (42) for both datasets. This produces 21,600 training and 5,400 test images for EuroSAT, and 1,680 training and 420 test images for UC Merced.

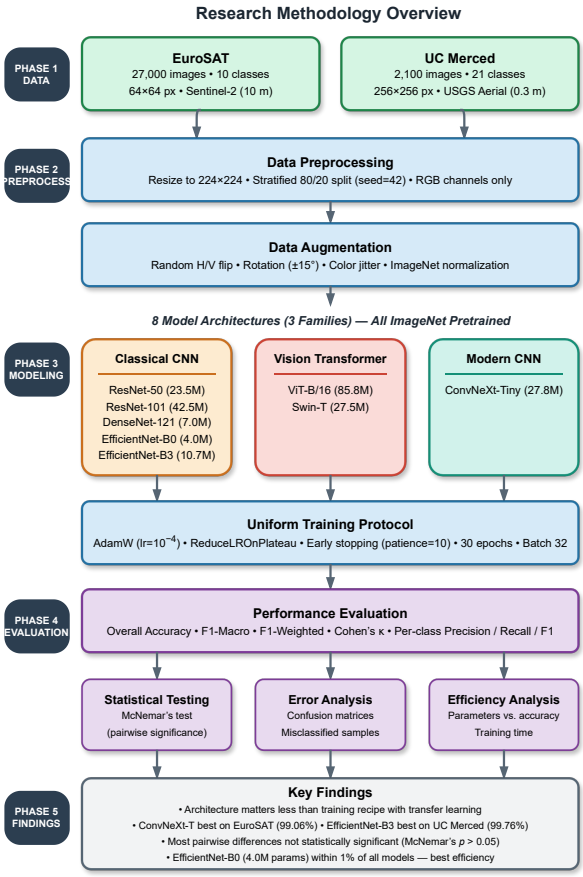

Fig. 1. Overview of the research methodology. Both datasets pass through the same preprocessing and train all eight architectures under identical conditions. Evaluation covers accuracy, McNemar significance testing, confusion analysis, and training cost.

B. Model Architectures

We selected eight architectures to cover three families. Table I lists them along with their parameter counts and publication years.

TABLE I
MODEL ARCHITECTURES EVALUATED IN THIS STUDY. PARAMETER COUNTS REFER TO THE IMAGENET-PRETRAINED BACKBONE BEFORE REPLACING THE CLASSIFIER HEAD.

Model	Family	Params (M)	Year
ResNet-50 [6]	CNN	25.6	2016
ResNet-101 [6]	CNN	44.5	2016
DenseNet-121 [7]	CNN	8.0	2017
EfficientNet-B0 [20]	CNN	5.3	2019
EfficientNet-B3 [20]	CNN	12.2	2019
ViT-B/16 [14]	Transformer	86.6	2021
Swin-T [15]	Transformer	28.3	2021
ConvNeXt-T [18]	Modern CNN	28.6	2022

Classical CNNs. ResNet-50 and ResNet-101 are residual networks with 50 and 101 layers. DenseNet-121 connects each layer to every other in a feed-forward fashion, reusing features while keeping the parameter count at 8.0M. EfficientNet-B0 and B3 use depthwise separable convolutions with compound scaling and are the lightest and second-lightest models in our lineup.

Vision Transformers. ViT-B/16 divides each 224×224 image into 16×16 patches and processes them through 12 self-attention layers. At 86.6M parameters it is the largest model we test. Swin-T computes attention inside local windows that shift across layers, trading global receptive field for much lower memory use (28.3M parameters).

Modernized CNN. ConvNeXt-Tiny is a pure convolution network that borrows design choices from transformers: 7×7 kernels, LayerNorm, GELU activations, and an inverted bottleneck layout. It sits between the CNN and transformer families in both design philosophy and parameter count (28.6M).

All models start from ImageNet-1K pretrained weights loaded via the timm library [24]. We replace the final classification head with a new linear layer matching the target class count.

C. Training Protocol

Every model was trained under identical conditions so that any accuracy difference can only come from the architecture. Images are resized to 224×224 pixels and augmented with random horizontal and vertical flips, rotation up to ±15°, and color jitter (brightness and contrast ±0.2, saturation ±0.1). Each channel is then normalized to ImageNet mean and standard deviation. The optimizer is AdamW [25] with a learning rate of 10^{-4} and weight decay of 10^{-4}. Learning rate is halved whenever validation loss stalls for five consecutive epochs, and training stops entirely after ten epochs without improvement. Batch size is 32, the epoch budget is 30, and all runs use a single NVIDIA GPU with PyTorch 2.0 [26].

Fig. 3 shows the augmentation pipeline applied to sample images. The perturbations expand the effective training set and reduce overfitting. The effect is strongest on UC Merced, where each class has only 80 training images.

D. Evaluation Metrics

1) Classification Metrics: Three metrics are reported for each model. Overall accuracy (OA) is the fraction of test images classified correctly. Macro-averaged F1-score averages per-class F1 values without weighting by class size, so rare classes count as much as common ones. Cohen's kappa (κ) [27] corrects for chance agreement and is standard in remote sensing accuracy assessment because class distributions are often uneven [28], [29].

2) Statistical Significance: To test whether two models really differ or just got lucky on different test images, we use McNemar's test [19], [30] with continuity correction. Dietterich showed this is the most reliable option when each classifier is run only once on a fixed test set. The test statistic is:

$$\chi^2 = \frac{(|n_{01} - n_{10}| - 1)^2}{n_{01} + n_{10}} \tag{1}$$

where n_{01} counts samples that model A got right but B got wrong, and n_{10} the reverse. Under the null hypothesis of equal performance this follows a χ^2 distribution with one degree of freedom. We use significance thresholds of α = 0.05 and 0.01.

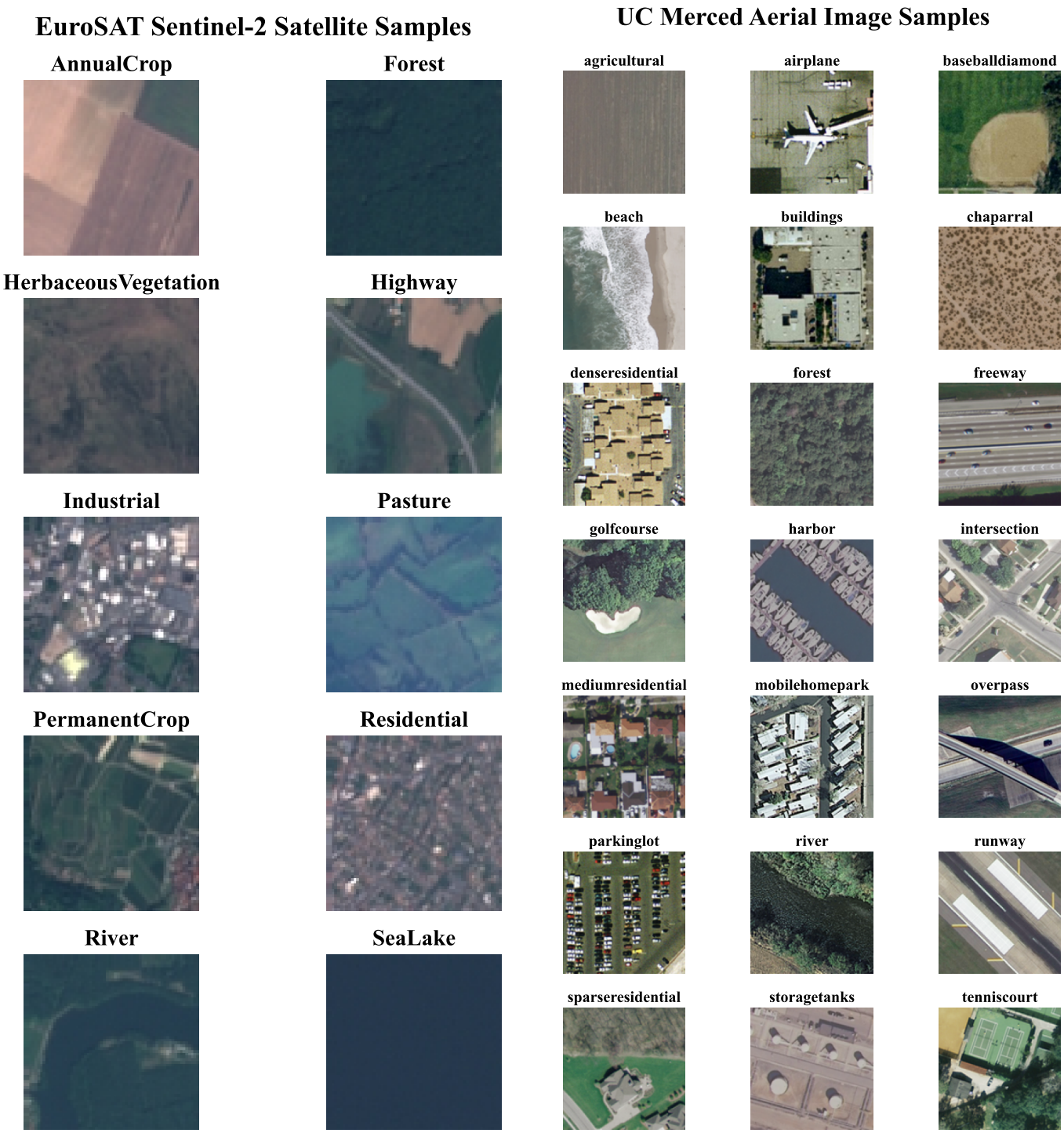

Fig. 2. Dataset samples. Left: one randomly selected Sentinel-2 patch per class from EuroSAT (64×64 pixels, 10 m resolution, 10 classes). Right: one randomly selected aerial image per class from UC Merced (256×256 pixels, 0.3 m resolution, 21 classes).

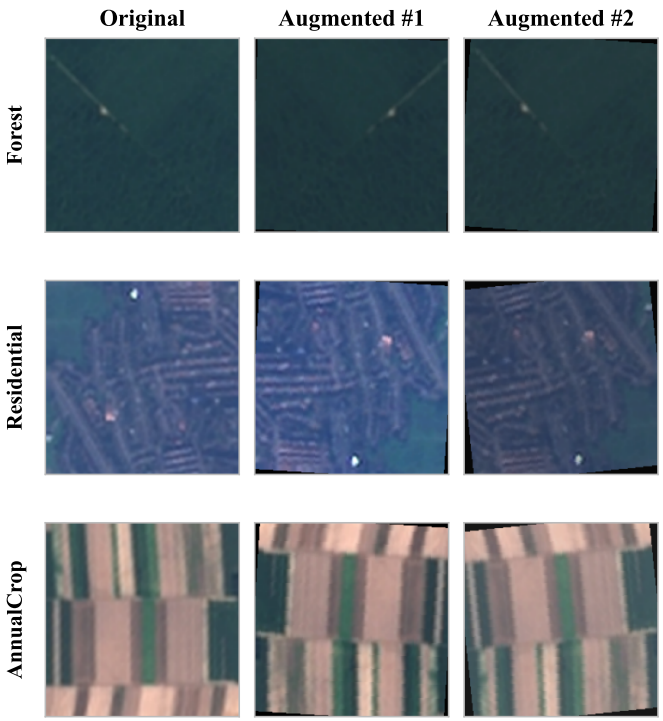

Fig. 3. Data augmentation pipeline. The leftmost column shows original EuroSAT images resized to 224×224 pixels. The two columns to the right show augmented versions produced by random flips, rotation, and color jitter.

3) Computational Efficiency: We record total trainable parameters and wall-clock training time for each model on each dataset.

IV. Results

A. Overall Performance

Tables II and III list the classification results on both datasets. On EuroSAT, ConvNeXt-Tiny is first at 99.11%, followed by Swin-T (99.02%) and EfficientNet-B3 (98.98%). On UC Merced, Swin-T leads at 99.76%, with four models tied at 99.52%.

TABLE II
Classification performance on EuroSAT (5,400 test images). Best results in bold.

Model	OA (%)	F1-Mac	κ	Params
ConvNeXt-T	**99.11**	**0.9908**	**0.9901**	28.6M
Swin-T	99.02	0.9899	0.9891	28.3M
EffNet-B3	98.98	0.9895	0.9887	12.2M
DenseNet-121	98.96	0.9893	0.9885	8.0M
ResNet-101	98.91	0.9887	0.9878	44.5M
EffNet-B0	98.76	0.9870	0.9862	5.3M
ViT-B/16	98.72	0.9870	0.9858	86.6M
ResNet-50	98.70	0.9865	0.9856	25.6M

What stands out in both tables is how little the numbers vary. On EuroSAT the full range spans just 0.41 percentage points (from 98.70% for ResNet-50 to 99.11% for ConvNeXt-Tiny), which is narrow enough that a different random partition of the data could easily reshuffle the ranking. UC Merced shows a wider spread of 1.67 points, but this is almost entirely attributable to ViT-B/16, which drops to 98.10% despite being the largest model at 86.6 M parameters. All other seven models surpass 99% on UC Merced. The simplest reading of these results is that ImageNet pretraining establishes a high performance floor and most architectures, regardless of their internal wiring, manage to reach it. Section V discusses why ViT-B/16 is the exception.

TABLE III
Classification performance on UC Merced (420 test images). Best results in bold.

Model	OA (%)	F1-Mac	κ	Params
Swin-T	**99.76**	**0.9976**	**0.9975**	28.3M
ConvNeXt-T	99.52	0.9952	0.9950	28.6M
EffNet-B0	99.52	0.9952	0.9950	5.3M
EffNet-B3	99.52	0.9952	0.9950	12.2M
ResNet-50	99.52	0.9952	0.9950	25.6M
ResNet-101	99.29	0.9929	0.9925	44.5M
DenseNet-121	99.05	0.9905	0.9900	8.0M
ViT-B/16	98.10	0.9809	0.9800	86.6M

Fig. 4 plots both datasets side by side. The bars cluster tightly near the top of the axis; without the numerical labels it would be hard to tell most models apart.

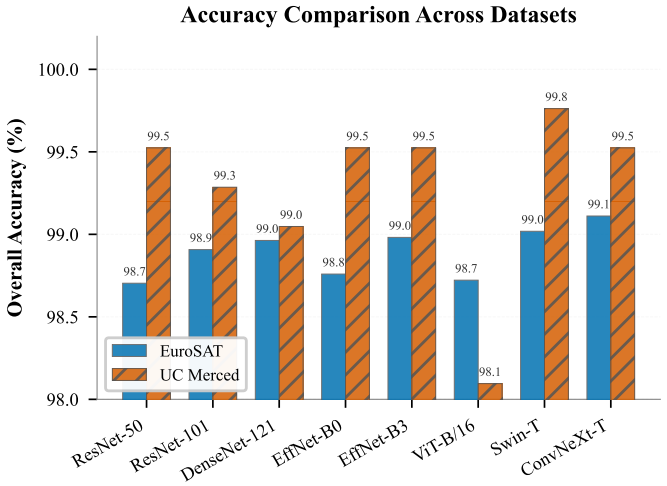

Fig. 4. Accuracy comparison across both datasets. All models exceed 98%, and the entire performance range spans less than one percentage point per dataset.

B. Training Dynamics

Fig. 5 plots the training and validation loss/accuracy curves for EuroSAT. Most models converge within 15 to 20 epochs; early stopping pulls several of them out before the 30-epoch limit. EfficientNet-B0 and DenseNet-121 reach near-optimal accuracy within the first few epochs. ViT-B/16 and ResNet-50 need the full budget, with best checkpoints at epochs 30 and 19. The slower convergence of ViT-B/16 is consistent with its 86.6M parameters: more weights to adapt means more gradient steps before the model settles.

On UC Merced (Fig. 6), convergence is faster across the board. The training set is much smaller (1,680 images versus 21,600), so fewer gradient updates are needed to adapt the pretrained features.

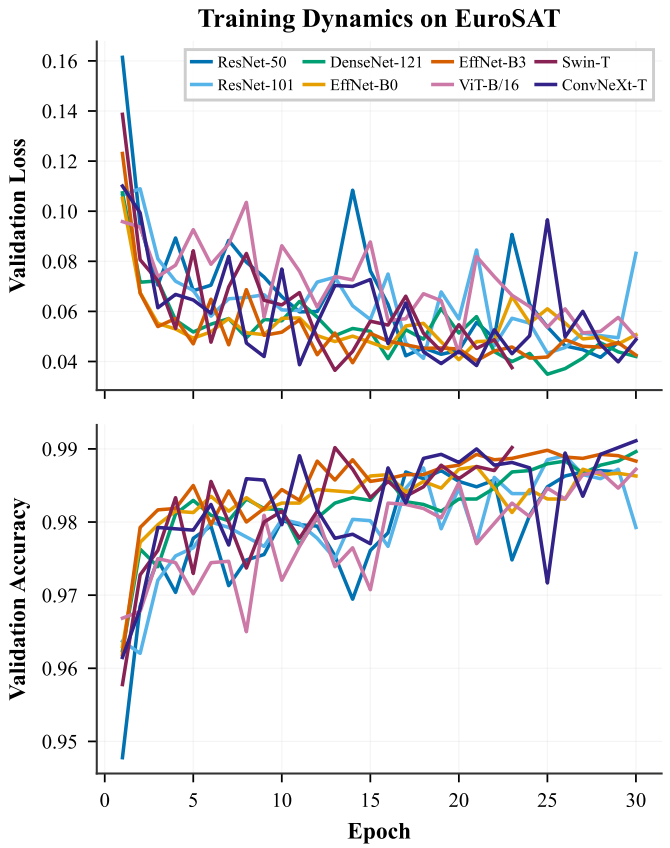

Fig. 5. Training dynamics on EuroSAT. Top: validation loss. Bottom: validation accuracy. Each line represents one architecture.

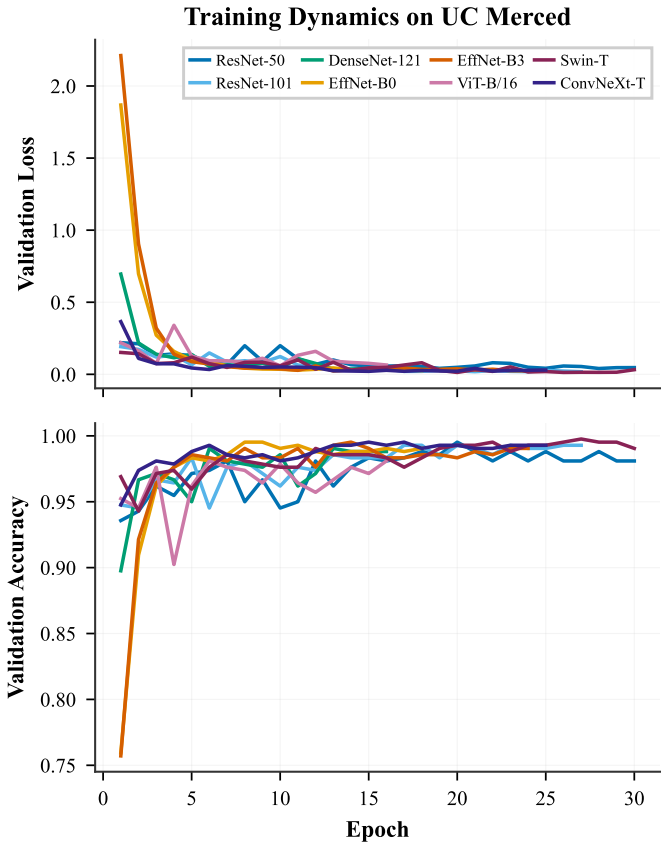

Fig. 6. Training dynamics on UC Merced. Top: validation loss. Bottom: validation accuracy. Convergence is faster due to the smaller dataset.

C. Per-Class Analysis

1) EuroSAT: The per-class F1 heatmap for EuroSAT (Fig. 7) shows that most classes are easy for every model. SeaLake and Forest both exceed 0.99 F1 across all eight architectures; their spectral signatures are distinct enough that none of the models struggle. PermanentCrop is the hardest class, with F1 between 0.96 (ResNet-101) and 0.98 (ConvNeXt-Tiny). River also varies more than average, likely because narrow river channels in 64×64 patches can resemble roads.

2) UC Merced: On UC Merced (Fig. 8), the picture is even more uniform. Most cells sit at F1 = 1.0, perfect classification. The exceptions are the residential sub-types (denseresidential, mediumresidential, sparseresidential) and, to some extent, buildings. These classes share rooftops, streets, and tree cover, and the boundary between "dense" and "medium" residential is blurry enough that a human interpreter would sometimes disagree.

D. Statistical Significance

Fig. 9 shows the McNemar p-value matrix for EuroSAT. Out of 28 pairwise comparisons, only a handful produce $p < 0.05$, mostly involving the bottom-ranked models against the top. The three leaders (ConvNeXt-Tiny, Swin-T, EfficientNet-B3) are not significantly different from each other, and most adjacent pairs in the ranking also show $p > 0.05$. A different random split of the test set could easily shuffle their order.

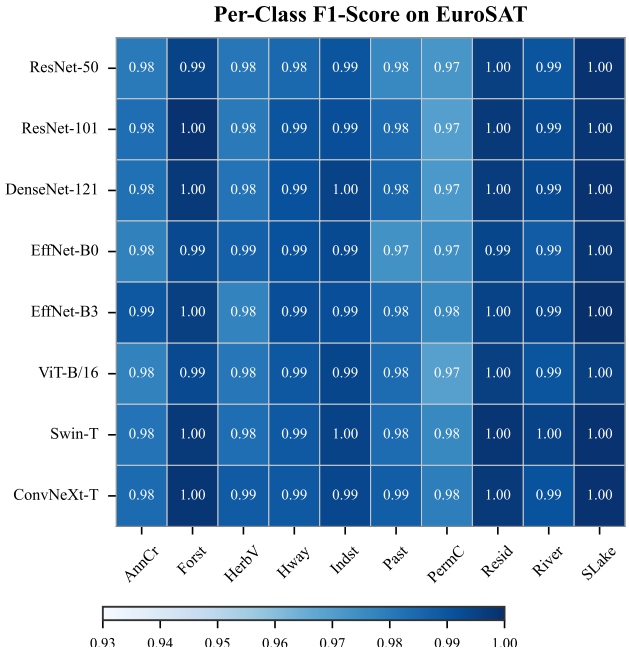

Fig. 7. Per-class F1-score heatmap on EuroSAT. Rows are models, columns are classes. Darker blue indicates higher F1. PermanentCrop and Pasture show the most variation across models.

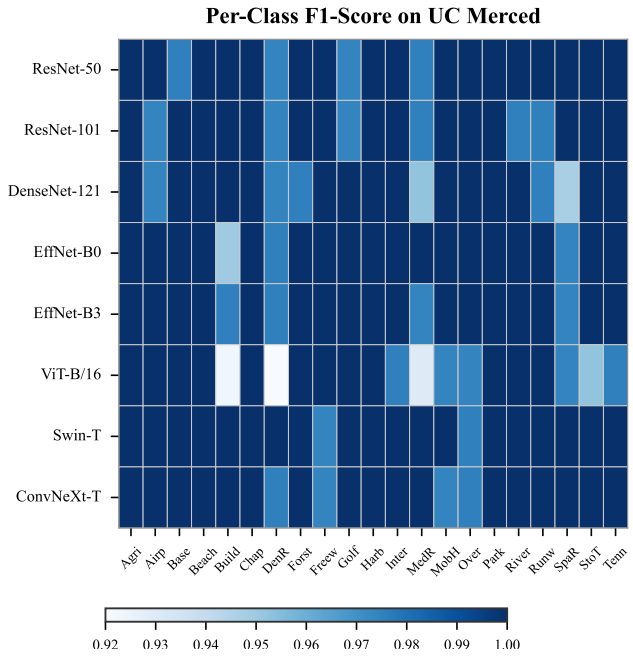

Fig. 8. Per-class F1-score heatmap on UC Merced (21 classes). Most cells are at F1 = 1.0 (dark blue). Residential sub-types show the most variation.

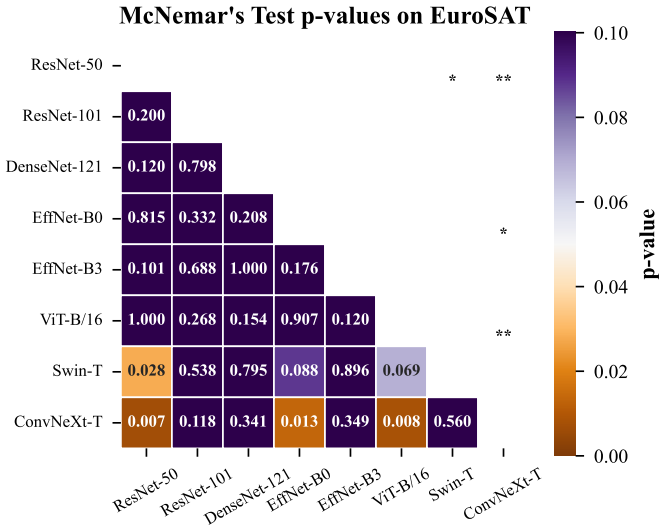

Fig. 9. McNemar's test p-value matrix for EuroSAT. Cells shaded green indicate model pairs whose accuracy difference is not statistically significant ($p > 0.05$); red cells denote pairs where the null hypothesis of equal error rate is rejected ($p < 0.05$). Significance levels: * $p < 0.05$, ** $p < 0.01$.

Fig. 10 shows the same analysis for UC Merced. Fewer pairs reach significance, as expected given the smaller test set (420 images). Most comparisons among the top-performing models produce $p > 0.05$. The exception is ViT-B/16, whose drop to 98.10% puts it far enough from the pack that several of its pairwise comparisons cross the significance threshold. For the remaining seven models, the accuracy ordering is largely a product of which specific images ended up in the test split.

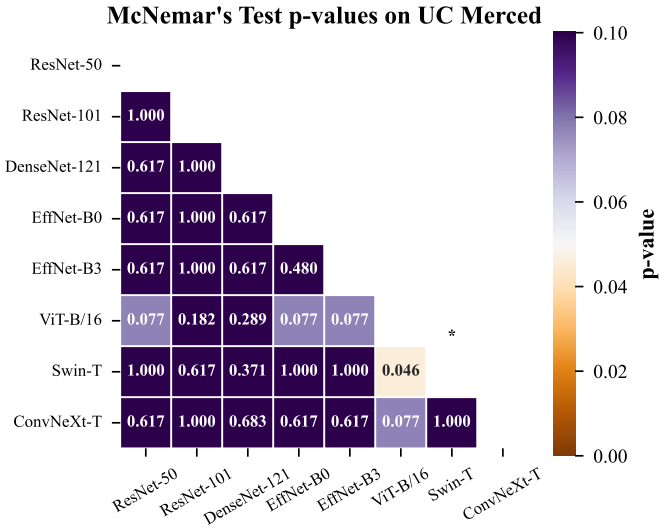

Fig. 10. McNemar's test p-value matrix for UC Merced. Same color scheme as Fig. 9. Pairs involving ViT-B/16 show significant differences; all other models are statistically indistinguishable at $p > 0.05$.

E. Error Analysis

Fig. 11 breaks down the predictions of ConvNeXt-Tiny on EuroSAT by class. Most classes have zero or near-zero misclassifications. The errors concentrate in two places: PermanentCrop gets confused with HerbaceousVegetation, and Pasture gets confused with AnnualCrop. In both cases the vegetation classes share similar green tones at Sentinel-2 resolution.

To understand these confusions better, we traced each misclassified test image back to its source file and placed it alongside a correctly classified example from the class the model predicted. Fig. 12 (left) aggregates these error pairs across all eight models on EuroSAT. In each row the red-bordered image is the misclassified sample and the blue-bordered image is a correct instance of the predicted class. The visual resemblance between the two is remarkable: a PermanentCrop patch containing mixed crop rows and green field margins is, at 64×64-pixel resolution, virtually indistinguishable from a HerbaceousVegetation patch. The same applies to Pasture versus AnnualCrop, where uniform grass cover makes the two classes overlap spectrally. In these cases the models are not making errors in any meaningful sense; the images themselves are ambiguous.

A similar phenomenon appears on UC Merced, despite its much finer spatial resolution of 0.3 m. Denseresidential and mediumresidential patches contain the same rooftop shapes

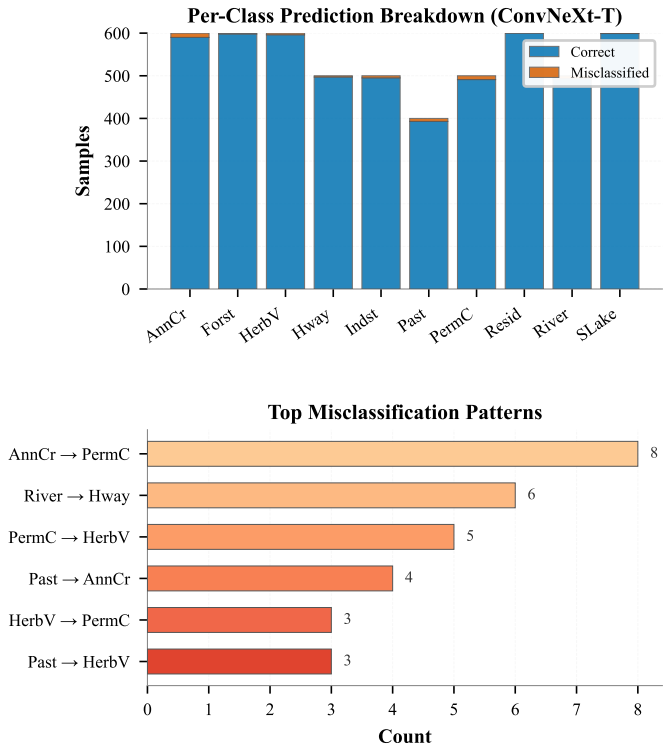

Fig. 11. Error analysis for ConvNeXt-Tiny on EuroSAT. Top: per-class correct (green) and misclassified (red) counts. Bottom: the most common misclassification patterns.

and street patterns; the sole distinguishing feature is the spacing between buildings, and some borderline cases would be difficult for a human annotator as well. Mobilehomepark images frequently contain scattered low-rise structures surrounded by green space that closely resembles sparseresidential layouts. The fact that all eight architectures confuse the same class pairs suggests that these errors originate in the dataset definitions rather than in any particular model's weaknesses. Fig. 12 shows representative pairs from both datasets.

F. Computational Efficiency

Fig. 13 plots accuracy against parameter count for EuroSAT. EfficientNet-B0 reaches 98.76% with 5.3M parameters. ViT-B/16 uses 86.6M parameters for 98.72%, a 16× larger model for slightly lower accuracy. The two EfficientNet variants trace an efficiency frontier that no other family matches.

Table IV lists training times. EfficientNet-B0 is fastest on EuroSAT (1,088 s); DenseNet-121 is fastest on UC Merced (211 s). ViT-B/16 is slowest on EuroSAT at 3,403 s, consistent with the cost of global self-attention over 86.6M parameters. Training time does not always track parameter count: Swin-T (28.3M) trains faster than ResNet-50 (25.6M) on EuroSAT, partly because windowed attention avoids the quadratic cost and partly because early stopping pulls different models out at different epochs.

TABLE IV
TRAINING TIME COMPARISON (SECONDS). MODELS SORTED BY EUROSAT TIME.

Model	EuroSAT	UC Merced
EffNet-B0	1,088	216
Swin-T	1,474	421
EffNet-B3	1,559	312
ResNet-50	1,630	402
DenseNet-121	1,686	211
ResNet-101	1,928	374
ConvNeXt-T	2,470	385
ViT-B/16	3,403	282

V. DISCUSSION

A. Architecture Family Comparison

A common expectation is that self-attention should give transformers an advantage in scene classification because it can relate distant regions of an image in ways that local convolution kernels cannot. Our results only partially support this view. ConvNeXt-Tiny, which contains no attention mechanism at all, placed first on EuroSAT (99.11%), while Swin Transformer took the lead on UC Merced (99.76%). Neither architecture family won on both datasets. Liu et al. [18] observed a similar pattern on ImageNet, where the accuracy gap between CNNs and transformers shrank to almost nothing once the CNN adopted modern training practices. Our contribution is the observation that the reverse also holds: on one of our two benchmarks, a transformer beat the modernized CNN.

We suspect the explanation lies in the nature of single-label scene classification at 224×224 resolution. Each image depicts one dominant land-use type, and the texture within the patch tends to be fairly homogeneous. Under these conditions, local color and shape cues are sufficient to distinguish classes, and there is little spatial reasoning for global attention to exploit. Tasks that require recognizing configurations of objects (for instance, identifying a harbor because boats sit adjacent to docking structures and open water) would likely give transformers more room to differentiate themselves, but such compositional reasoning is rarely needed when the entire image is "forest" or "highway."

The contrast between ViT-B/16 and Swin-T is especially instructive. Both models belong to the transformer family, yet they occupy opposite ends of our leaderboard: Swin-T placed first on UC Merced (99.76%) and second on EuroSAT (99.02%), whereas ViT-B/16 placed second-to-last on EuroSAT (98.72%) and last on UC Merced (98.10%). In other words, the gap between these two transformers exceeds the gap between any CNN and any transformer in our study. Swin-T's local windowed attention and hierarchical pooling endow it with an implicit multi-scale structure similar to that of convolutional networks, while ViT-B/16's flat global attention across all 196 tokens appears to demand substantially more fine-tuning data before it adapts. UC Merced, with only 80 training images per class, does not supply enough. Treating "transformers" as a single group conceals this distinction; the specific architectural decisions within the transformer family (local vs. global attention, flat vs. hierarchical features)

Fig. 12. Misclassified examples from EuroSAT (left) and UC Merced (right). In each pair, the red-bordered image was misclassified and the blue-bordered image is a correctly classified example from the predicted class. The visual similarity within each pair explains the confusion. On UC Merced, residential density classes are the main source of errors.

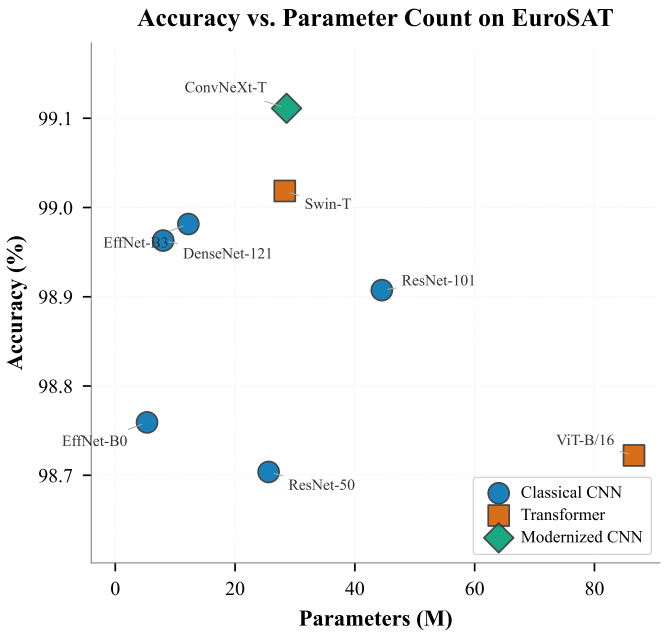

Fig. 13. Accuracy vs. parameter count on EuroSAT. Blue circles = classical CNNs, red squares = transformers, green diamonds = modernized CNN.

proved at least as consequential as the broader CNN-versus-transformer divide.

B. Deeper Is Not Always Better

ResNet-101 edged out ResNet-50 on EuroSAT (98.91% vs. 98.70%), which is the conventional expectation: more layers, more capacity, slightly higher accuracy. On UC Merced, however, the ranking inverted. ResNet-50 reached 99.52%; ResNet-101 stopped at 99.29%. The most likely explanation is data scarcity. UC Merced provides only 80 training images per class, which is probably not enough to push all 101 layers meaningfully away from their ImageNet initialization. The deeper model overfits to whatever signal it extracts early and plateaus at its early-stopping checkpoint. On EuroSAT, where each class has 2,160 training images, the additional layers have enough gradient signal to justify their extra capacity.

C. Parameter Efficiency

Perhaps the most striking result in our experiments is the performance of EfficientNet-B0. At 5.3 M parameters it is sixteen times smaller than ViT-B/16, yet it scored 98.76% on EuroSAT and 99.52% on UC Merced, both figures higher than those of ViT-B/16. In practical terms, multiplying the parameter budget by a factor of sixteen purchased no accuracy improvement whatsoever. EfficientNet-B3, roughly twice the size of B0 at 12.2 M parameters, gives up very little inference speed while matching or exceeding every other architecture we evaluated. For settings where model size is constrained (embedded sensors, on-board satellite processing, field-deployable drones), either EfficientNet variant represents a strong default choice.

D. Dataset Saturation

When the worst-performing model in a comparison already scores 98.70% and the total spread across eight architectures is less than half a percentage point, one has to ask whether the benchmark can still discriminate between methods. We believe EuroSAT has essentially reached that ceiling for single-label classification with transfer learning. UC Merced shows somewhat more variation (1.67 percentage points end to end), but almost all of it comes from a single outlier: ViT-B/16 at 98.10%. Remove that one model and the remaining seven span only 0.71 points. Several earlier works have called for more demanding evaluation protocols, such as datasets with tens of thousands of training images per class, cross-sensor generalization, or few-shot learning with only a handful of labeled examples [1], [4]. Our findings reinforce the urgency of that call.

E. Why Architecture Mattered So Little

The McNemar matrices (Figs. 9 and 10) are dominated by green cells, indicating that the vast majority of pairwise accuracy differences fail to reach statistical significance at $p < 0.05$. A different random partition of the test set would, in all likelihood, rearrange the ranking. The practical takeaway is that a researcher who spends weeks experimenting with different backbone architectures may obtain less benefit than one who invests the same time in curating training data or refining the augmentation policy. On the two benchmarks examined here, the pretrained initialization and the fine-tuning protocol appear to matter considerably more than the choice of backbone. Whether this conclusion extends to tasks with richer output structure, such as per-pixel semantic segmentation or multi-temporal change detection, remains to be tested.

F. Limitations

Several caveats apply. First, we evaluated only two datasets; results on larger or more heterogeneous collections could differ. Second, we used only the RGB bands of EuroSAT rather than the full 13-band multispectral product, which may disadvantage architectures that benefit from additional spectral information. Third, all hyperparameters were held constant across models. While this decision is necessary for a fair comparison, it likely understates the peak accuracy achievable by any individual architecture with dedicated tuning. Finally, we used a single stratified train-test partition rather than k-fold cross-validation, so our rankings depend on one particular split of the data.

VI. Conclusion

We compared eight deep learning architectures drawn from three design families on two remote sensing scene classification benchmarks under strictly identical training conditions. Every model exceeded 98% overall accuracy once initialized with ImageNet-pretrained weights and fine-tuned with the same optimizer, augmentation pipeline, and early-stopping schedule. The top-ranked model was ConvNeXt-Tiny on EuroSAT (99.11%) and Swin Transformer on UC

Merced (99.76%), but no single architecture family dominated both datasets. McNemar's test indicated that the majority of pairwise accuracy differences were not statistically significant. EfficientNet-B0 (5.3 M parameters) came within half a percentage point of the best model on EuroSAT and shared second place on UC Merced, whereas ViT-B/16 (86.6 M parameters) finished near the bottom of the ranking on both benchmarks despite carrying sixteen times more parameters.

From a deployment perspective, EfficientNet-B0 or EfficientNet-B3 are attractive defaults: both are compact enough for resource-constrained platforms and sacrifice almost no accuracy relative to much larger models. When computational resources are not a constraint, Swin Transformer is the strongest option among the two transformers we tested, having placed first or second on both datasets. ViT-B/16, by contrast, should be used with caution unless the practitioner has access to considerably more labeled training data than these benchmarks provide. The overarching lesson of our experiments is that the shared pretrained initialization and fine-tuning protocol exerted a larger influence on accuracy than the architectural differences among the eight models. To determine whether meaningful performance gaps emerge under more demanding conditions, future studies should consider larger-scale datasets with several thousand labeled images per class, cross-sensor transfer experiments, and few-shot learning scenarios in which only a handful of labeled examples are available.

REFERENCES

[1] G. Cheng, X. Xie, J. Han, L. Guo, and G.-S. Xia, "Remote sensing image scene classification meets deep learning: challenges, methods, benchmarks, and opportunities," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 4697–4713, 2020.

[2] L. Ma, Y. Liu, X. Zhang, Y. Ye, G. Yin, and B. A. Johnson, "Deep learning in remote sensing applications: a meta-analysis and review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 152, pp. 166–177, 2019.

[3] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 270–279, 2010.

[4] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015.

[6] 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 (CVPR)*, 2016, pp. 770–778.

[7] 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 (CVPR)*, 2017, pp. 2261–2269.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," pp. 248–255, 2009.

[9] K. Nogueira, O. A. B. Penatti, and J. A. dos Santos, "Towards better exploiting convolutional neural networks for remote sensing scene classification," *Pattern Recognition*, vol. 61, pp. 539–556, 2017.

[10] Y. Li, H. Zhang, X. Xue, Y. Jiang, and Q. Shen, "Deep learning for remote sensing image classification: a survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 6, p. e1264, 2018.

[11] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[12] M. Neumann, A. S. Pinto, X. Zhai, and N. Houlsby, "In-domain representation learning for remote sensing," *arXiv preprint arXiv:1911.06721*, 2019.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 5998–6008.

[14] 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," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.

[15] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 9992–10 002.

[16] D. Hong, Z. Han, J. Yao, L. Gao, B. Zhang, A. Plaza, and J. Chanussot, "SpectralFormer: rethinking hyperspectral image classification with transformers," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[17] D. Wang, Q. Zhang, Y. Xu, J. Zhang, and Y. Zhong, "Advancing plain vision transformer toward remote sensing foundation model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 61, pp. 1–15, 2023.

[18] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A ConvNet for the 2020s," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 11 966–11 976.

[19] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, 1947.

[20] M. Tan and Q. V. Le, "EfficientNet: rethinking model scaling for convolutional neural networks," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2019, pp. 6105–6114.

[21] X. X. Zhu, D. Tuia, L. Mou, G.-S. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: a comprehensive review and list of resources," *IEEE Geoscience and Remote Sensing Magazine*, vol. 5, no. 4, pp. 8–36, 2017.

[22] P. Helber, B. Bischke, A. Dengel, and D. Borth, "EuroSAT: a novel dataset and deep learning benchmark for land use and land cover classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 7, pp. 2217–2226, 2019.

[23] G.-S. Xia, J. Hu, F. Hu, B. Shi, X. Bai, Y. Zhong, L. Zhang, and X. Lu, "AID: a benchmark data set for performance evaluation of aerial scene classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 7, pp. 3965–3981, 2017.

[24] R. Wightman, "PyTorch Image Models," 2019, gitHub repository, https://github.com/huggingface/pytorch-image-models.

[25] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

[26] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "PyTorch: an imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.

[27] J. Cohen, "A coefficient of agreement for nominal scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37–46, 1960.

[28] R. G. Congalton, "A review of assessing the accuracy of classifications of remotely sensed data," *Remote Sensing of Environment*, vol. 37, no. 1, pp. 35–46, 1991.

[29] G. M. Foody, "Status of land cover classification accuracy assessment," *Remote Sensing of Environment*, vol. 80, no. 1, pp. 185–201, 2002.

[30] T. G. Dietterich, "Approximate statistical tests for comparing supervised classification learning algorithms," *Neural Computation*, vol. 10, no. 7, pp. 1895–1923, 1998.