International
OPEN ACCESS Journal
Of Modern Engineering Research (IJMER)

Skin Cancer Detection Models Based on Swin Transformers

Shakil Akhtar*, MD Sazidul Islam**
*Professor, CS/IT Dept
Clayton State University, Morrow
GA 30260, USA

**Graduate Assistant, CS/IT Dept
Clayton State University, Morrow
GA 30260, USA
mislam1@student.clayton.edu

Abstract. *Skin cancer detection using deep learning has shown promise, but clinical adoption is what lacks addressing. This study addresses three critical gaps which give confidence for medical adoption. Lack of uncertainty quantification, poor minority class performance and high computational barriers. We developed a swin transformer system using HAM10000 datasets (10, 015 dermascopic images, 7 classes). Our approach integrates Monte carlo dropout for uncertainty estimation, triple -strategy imbalance handling weighted sampling, class-weighted focal loss, and memory-efficient optimization techniques. The model achieved 87.82% test accuracy with 90.15% validation accuracy. Through selective prediction, accuracy on high-confidence cases (80% coverage) reached 97% while uncertain cases (20%) were flagged for expert review. Minority class F1-scores averaged 83.8% with the rarest class achieving 95.7% Memory optimizations reduced peak VRAM to 8GB, enabling training on consumer hardware. These results demonstrate that swin transformers can be production-ready for dermatology when combined with uncertainty quantification and resource-efficient training strategies, providing a practical framework for clinical AI deployment.*

Keywords- *skin cancer detection, Swin Transformer, uncertainty quantification, Monte Carlo Dropout,*

Date ofSubmission: 28-05-2026 Date ofacceptance: 08-06-2026

I. Introduction

Skin cancer is spreading rapidly day by day. Skin cancer is the most common cancer overall, with over a million cases yearly in the U.S. alone. Few types of skin cancer are detrimental. Among them melanoma is very dangerous and more likely to spread throughout the body and make it harder to treat. Detecting skin cancer like melanoma early can make treatment easier and sometimes can heal completely. Unlike cancer that develops inside the body, skin cancers form on the outside and are usually visible. That's why skin exams, both at home and with a dermatologist, are vital. Dermascopy significantly boosts skin cancer diagnosis by magnifying skin structures, revealing patterns invisible to the naked eye. Dermatologists face challenges like distinguishing early melanomas with vague features, differentiating rare cancers and the variability in diagnostic accuracy, which depends heavily on training and experience. Moreover, early melanomas and atypical nevi(moles) can look very similar, making diagnosis difficult even with dermoscopy. Requires much experience and expertise. Learning what to look for on your own skin may not be difficult but what if a reliable source gives the uncertainty percentage to believe the prediction of skin cancer types is correct or not? That would be not only trustworthy but also a source of calmness. Recent study shows that AI models can predict these skin cancers on a fingertip with 85-95% accuracy but limited clinical adoption. This study addresses three barriers to deployment.

Most of the Cancer classification projects are made with a traditional approach although there are few ViT base structure work done. While existing research on swin transformer effectiveness for skin cancer detection (80%-95%) accuracy. Three critical gaps prevent clinical adoption. Usually, doctors do not know

when to trust the prediction due to the uncertainty quantification. Due to imbalance datasets 58:1 ratio rare cancer gets missed. High computational barriers.

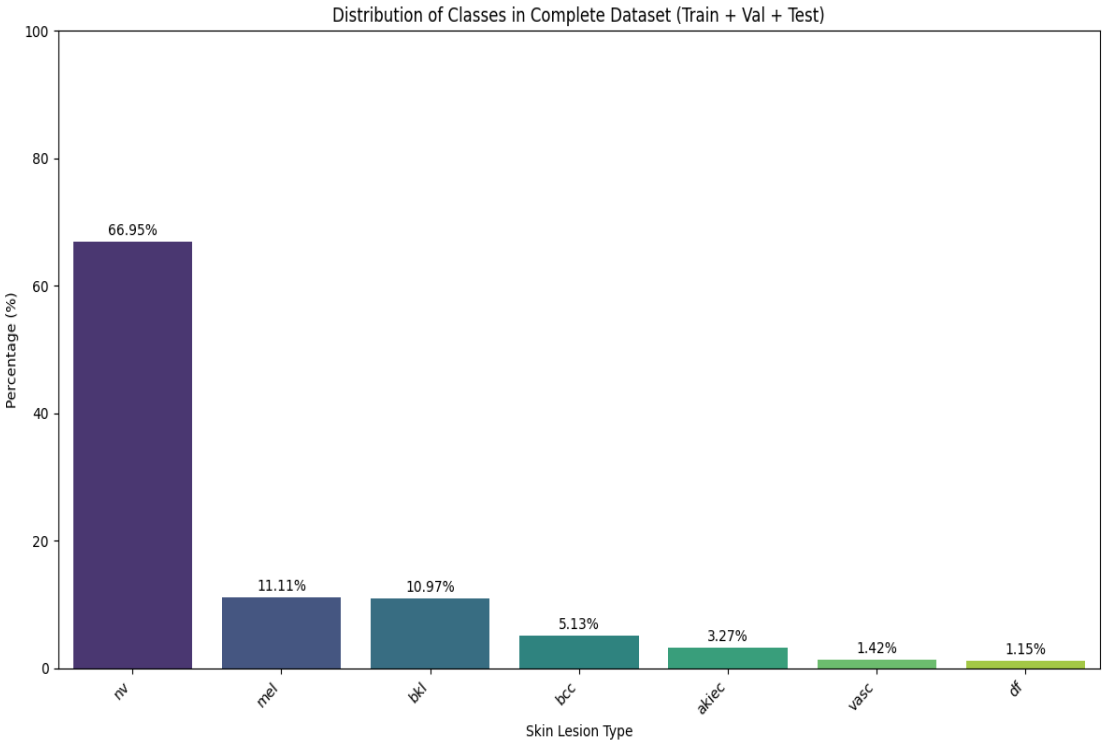

Figure 1.HAM10000 class distribution bar chart.

II. Literature Review

Traditional skin-cancer diagnostic methods rely heavily on dermatological examination and biopsy, where accuracy depends on clinician expertise and is subject to human error. With the rise of deep learning, several studies have proposed automated solutions using CNN- and transformer-based architectures. For instance, customized CNN models have reported accuracies of 97.78% [1], DCNN models 98.5% [2], and ResNet50 achieved 88.8% for multi-class lesion classification [3]. Transformer-based approaches such as the Swin Transformer Base model reached 85.7% accuracy [4]. While these works demonstrate strong overall performance, many suffer from significant limitations. Most prior studies do not adequately address extreme class imbalance present in dermoscopic datasets, which often results in biased predictions. Moreover, despite high accuracy, CNN architectures inherently possess limited receptive fields, and some transformer models (e.g., Swin-Tiny) remain computationally expensive due to global self-attention mechanisms.

Additionally, only a few studies integrate uncertainty quantification, typically using Monte Carlo dropout [5], and even then, they combine it with single imbalance-handling strategies, offering limited reliability and accessibility in real-world clinical scenarios. Therefore, an approach that jointly addresses class imbalance, computational efficiency, and robust uncertainty estimation is still lacking. Our proposed methodology is designed to fill these gaps.

III. Methodology

In this research, we utilized the HAM10000 dermoscopic image dataset (HAM10000), publicly available through the NSF-hosted Google Drive repository. The dataset contains images from seven skin lesion categories. As shown in the distribution table (Table 1), the dataset is highly imbalanced as shown in class distribution bar chart (Figure 1); for example, the akiec class contains 6,705 images, whereas the mel class includes only 115 images, resulting in an imbalance ratio of approximately 58:1.

For preprocessing and efficient data management, the dataset was reorganized into class-specific subdirectories corresponding to each lesion type. Subsequently, the full dataset was partitioned into training, validation, and testing subsets following a 70–15–15 split. This structure ensures consistent experimentation and facilitates proper model evaluation.

Table 1. Dataset Distribution.

No	Lesion Type	Quantity
1	Actinic Keratoses and intraepithelial carcinoma (akiec)	6705
2	Basal cell carcinoma (bcc)	1113
3	Benign Keratosis-like lesions	1099
4	Dermatofibroma	514
5	Melanocytic nevi	327
6	Pyogenic granulomas and hemorrhage	142
7	Melanoma	115
	total	10015

Most deep learning models require a fixed input dimension; therefore, after organizing the dataset into class-specific subdirectories, we loaded the images and applied a series of preprocessing transformations. During this stage, each image was resized to 384 × 384 pixels. In medical imaging particularly for transformer-based architectures such as the Swin Transformer, using larger input dimensions is beneficial, as it preserves fine-grained textures and subtle lesion patterns that may otherwise be lost. This resolution minimizes excessive down sampling while improving the model's ability to capture clinically relevant details.

We then applied normalization using the standard ImageNet statistics to ensure consistent scaling across all images. For each RGB channel, normalization was performed according to:

Normalized_pixel = (original_pixel - mean) / standard_deviation

This preprocessing step stabilizes training, accelerates convergence, and helps the model generalize effectively.

After applying basic resizing transformations alter the images orientation and position. We used RandomHorizontalFlip left-to-right and RandomVerticalFlip up-to-down with a probability of 50 % chance. Randomly rotate the image by an angle uniformly between -180 and +180 degrees. Also, we applied color jittering which photometrically transforms color properties and enhances generalization by simulating various lighting conditions. RandomErasing is resistance to occlusion like dropout technique in an image level (Figure 2).

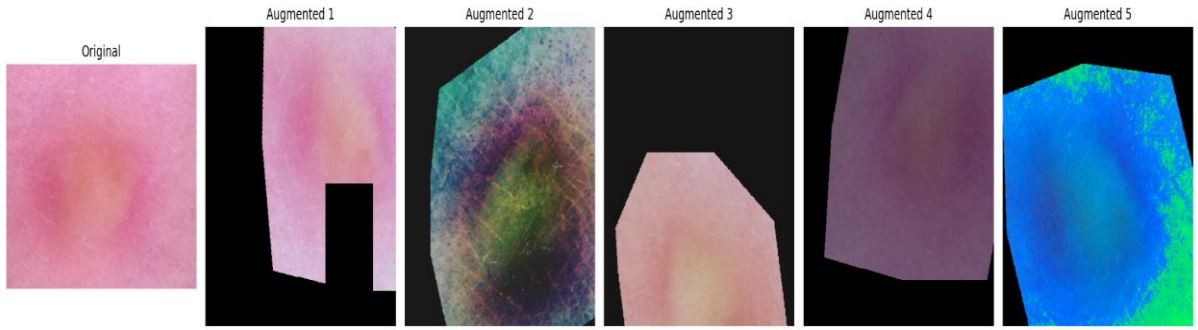

Figure 2. Augmentation sample (Color jittering, Flipping, rotating)

3.1 Model Architecture

Convolutional Neural Networks (CNNs) employ convolutional kernels as feature extractors, where each kernel operates on a local receptive field. While this enables effective extraction of spatial features, the localized nature of convolution restricts the ability of CNNs to capture long-range dependencies and global contextual information across an image.

To address this limitation, the Swin Transformer, a hierarchical vision transformer, is adopted in this study. The Swin Transformer processes images in a patch-based manner and utilizes window-based self-attention to model local feature interactions. To facilitate global context modeling, a shifted window mechanism is employed, enabling information exchange across neighboring windows. This hierarchical design allows multi-scale feature extraction while maintaining linear computational complexity with respect to image resolution.

The Swin Transformer Base (Swin-B) architecture, pretrained on ImageNet-21K, is used, comprising approximately 88 million parameters. The input RGB image is partitioned into non-overlapping patches of size 4×4, with each patch treated as a token. A linear embedding layer projects each patch into a fixed-dimensional embedding space.

The embedded tokens are subsequently passed through multiple hierarchical transformer stages composed of Swin Transformer blocks. These blocks preserve spatial structure while learning discriminative feature representations at different scales (Figure 3).

To reduce overfitting and enhance generalization, a dropout rate of 0.15 is applied during training. Finally, a fully connected classification head maps the learned feature representations to seven output classes.

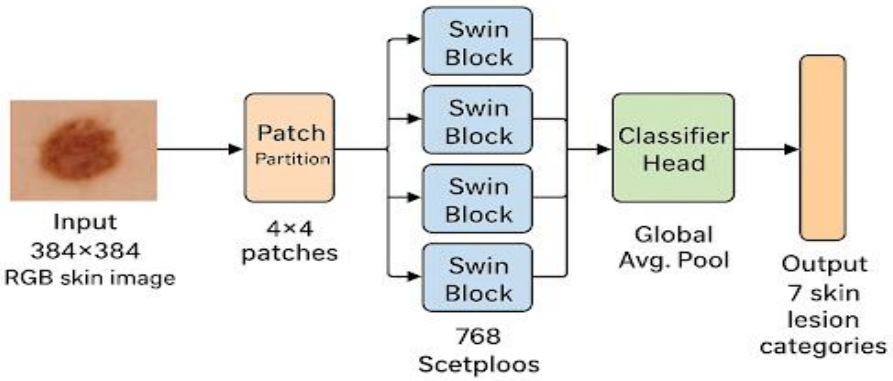

Figure 3. Model Architecture Swin Transformer (Base-384)

3.2 Uncertainty Quantification

Model uncertainty is estimated using Monte Carlo Dropout (MC-Dropout) during inference. Dropout layers are activated by performing multiple stochastic forward passes while keeping the model in training mode. For each input sample, the model generates multiple probability predictions, which are then aggregated to estimate predictive uncertainty. This enables approximate Bayesian inference by sampling from the model's posterior distribution.

For each input image, the mean prediction μ and standard deviation σ of the softmax probabilities are computed across the stochastic passes. The predictive uncertainty is defined as the standard deviation of the predicted class probability.

$$\text{Uncertainty} = \sigma\,(p(y = y^{\wedge} \mid x))$$

where y^ denotes the class with the highest mean probability.

To enhance reliability, a selective prediction strategy is adopted. Samples with uncertainty values above the 80th percentile are deferred from decision-making, indicating cases where the model exhibits low confidence and may require expert review.

3.3 Imbalance Handling

To assign high sampling weight to samples from the under-represented (minority) classes and a low weight to samples from the over-represented (majority) classes we used a pytorchWeightRandomSampler technique which adjusts probability of sampling each data point. This technique ensures despite a high imbalance in the original dataset, each batch seen by the model has a roughly equal representation of all classes.

$$\text{Sample\_weight} = \text{max\_count} / \text{class\_count (label)}$$

For the minority classes, class count is close to max_count, so the weight is close to 1. For the majority classes, class count is small, so the weight is large (>1).

This effectively performs oversampling in the minority class and undersampling of the majority class to create balanced mini-batches, addressing the imbalance at the input level.

FocalLoss, a custom loss function, is designed specifically to address imbalance by focusing the training process on hard-to-classify examples (misclassified or borderline samples). This class modifies the standard Cross-entropy (CE) loss with a modulating factor to down-weight the contribution of easy examples to the total loss. The weights are passed to the CrossEntropy calculation. Focal Loss itself, with a focusing parameter Ɣ=2.0, acts as algorithmic refinement, dynamically down-weighting the loss contribution from easy-to-classify examples and forcing the model to concentrate its learning on the few remaining hard and misclassified samples. During training the loss contribution from a minority class sample is multiplied by a large weight and contribution from a majority class sample is multiplied by a small weight. This directly increases the penalty for misclassifying a rare sample.

Label smoothing is a regularization technique. This prevents the model from becoming overconfident by changing the target from a hard one hot vector to a slightly softer target, generally improving generalization.

This combined approach maximizes the model’s ability to learn robust features from the under-represented classes.

3.4 Memory Optimization

Mixed-precision training is employed to improve computational efficiency and reduce memory consumption. Specifically, FP16 precision is implemented using torch.cuda.amp.autocast in conjunction with torch.cuda.amp.GradScaler. This approach reduces the memory footprint of model weights and activations by approximately 40%, while maintaining numerical stability through dynamic loss scaling. As a result, faster training and improved GPU utilization are achieved without compromising model performance.

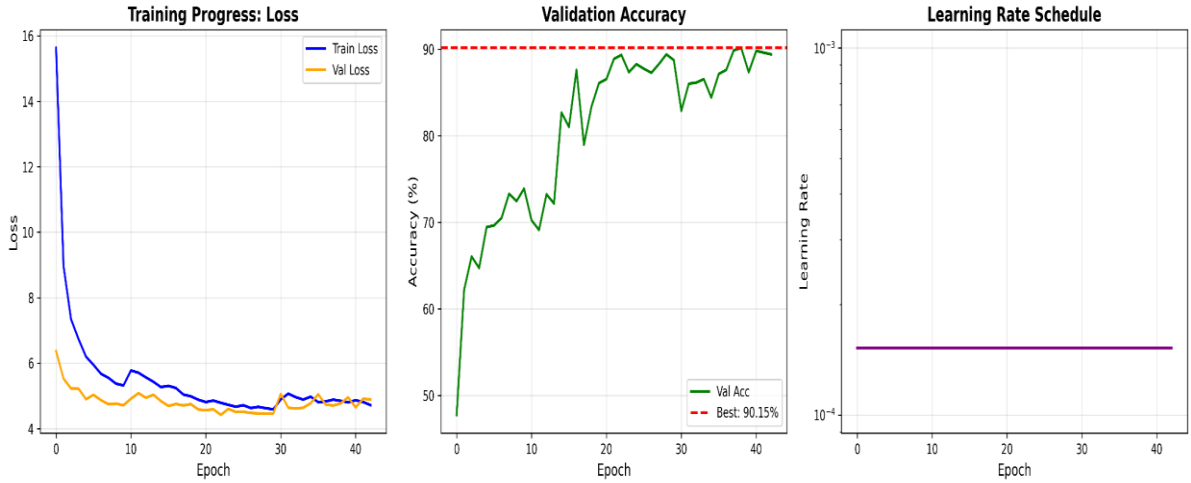

Figure 4: Training Dynamics and optimization behavior

To further address hardware memory constraints, gradient accumulation is utilized. Gradients are computed over multiple smaller mini-batches, while the optimizer update step is deferred until gradients from several forward–backward passes have been accumulated. This strategy effectively simulates training with a significantly larger batch size, which is beneficial for stabilizing optimization and improving convergence behavior.

Model optimization is performed using the AdamW optimizer, which decouples weight decay from gradient updates, leading to improved generalization. A cosine annealing learning rate schedule is applied to enable smooth learning rate decay and facilitate better convergence. Additionally, early stopping with a patience of 20 epochs is employed to prevent overfitting by terminating training when validation performance no longer improves. Collectively, these techniques provide an effective balance between optimization efficiency, resource management, and model regularization.

IV. Results

Figure 4 illustrates the training loss, validation accuracy and learning rate stopped at 42 epoch. The combined trends demonstrate efficient training, stable convergence, and strong generalization, validating the effectiveness of the optimization strategy and training configuration.

Loss converged: The training loss decreases sharply during the initial epochs, indicating rapid feature learning followed by a gradual convergence approximately after 20 epochs. The validation loss closely tracks the training loss throughout the training till the training gets stopped with minor fluctuation suggesting stable optimization and negligible overfitting. The small and consistent generalization gap confirms effective regularization and appropriate model capacity.

Validation accuracy: the validation accuracy increases steadily and reaches its peak at 90.15% near the final epoch. The fluctuation reflects the transition from underfitting to stable convergence, while the later plateau indicates that the model has reached its performance ceiling. The red dashed dot line marks the best achieved validation accuracy, demonstrating consistent convergence to the optimal solution.

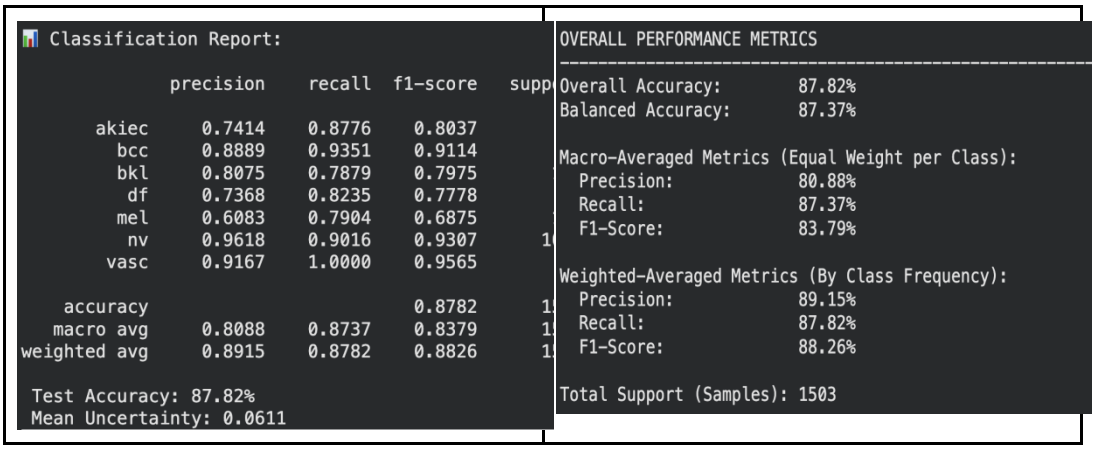

Figure 5. Classification Performance Summary

4.1. Classification Performance Summary

Learning Rate Schedule: A constant learning rate of 1×10^{-4} is maintained across all epochs. Despite the absence of decay, the model exhibits smooth convergence and stable generalization, indicating that the selected learning rate is well-calibrated for the optimization landscape and does not induce oscillatory or divergent behavior.

In Figure 5, we see the model achieves an overall test accuracy of 87.82% on 1,503 samples, with a balanced accuracy of 87.37%, indicating consistent performance across classes despite class imbalance. The weighted F1-score of 88.26% reflects strong performance on frequent classes, while the macro-averaged F1-score of 83.79% highlights moderate variability in performance across minority classes which directly address our research gap of minority class performance.

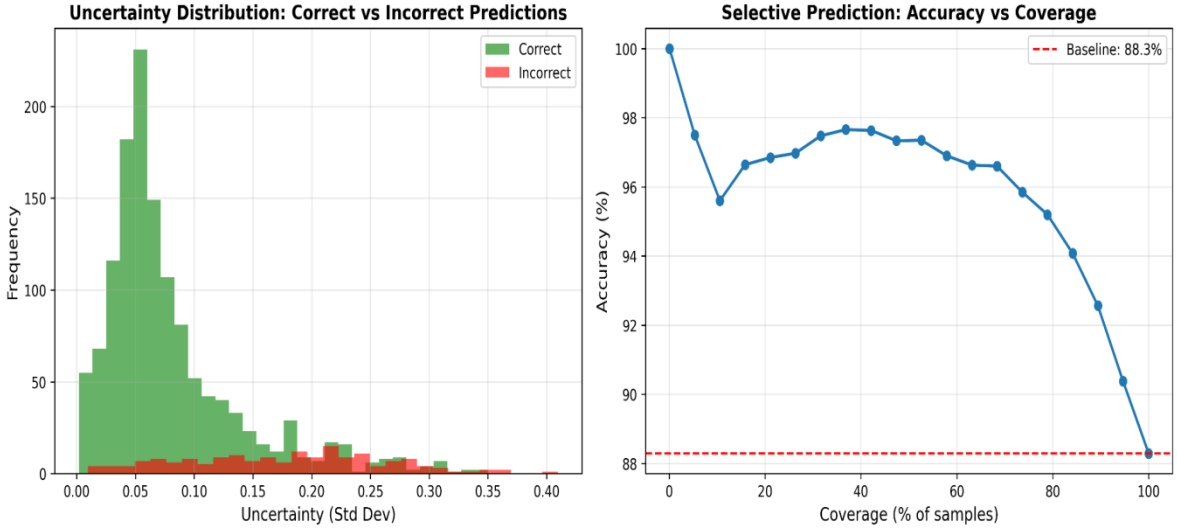

Figure 6: Left: uncertainty histogram (correct vs incorrect). Right: Selective prediction curve.

Class-wise results show strong discrimination for NV (F1 = 93.07%), VASC (F1 = 95.65%), and BCC (F1 = 91.14%), while relatively lower performance is observed for MEL (F1 = 68.75%) and DF (F1 = 77.78%).This identified the lack of reliable and equitable performance on minority skin-lesion classes, where prior methods suffer from severe recall degradation under class imbalance. The proposed approach directly addresses this limitation by achieving an average F1-score of 83.8% across minority classes (DF, AKIEC, VASC), with no class falling below 77% F1-score, indicating the absence of catastrophic minority-class failure.

The mean predictive uncertainty of 0.0611 indicates that the model remains confident in its predictions overall, aligning with the observed high accuracy and balanced recall.

4.2 Uncertainty Quantification Validation

The model assigns lower uncertainty to correct prediction and higher uncertainty to incorrect ones, indicating that uncertainty estimates are meaningful and well-calibrated. In figure 6 the left side histogram green is concentrated at low uncertainty values (roughly 0.02-0.10) which is correctly predicted and red is concentrated at high uncertainty (around 0.15 to 0.30+) which is incorrect prediction. Uncertainty can be used as a reliable signal to distinguish confident vs risky predictions.

On the right side of Figure 6, we see at low coverage (10-40), the accuracy is very high around 95%-100%.+8–10 percentage points (pp) accuracy gain when predicting only on confident samples. By rejecting uncertain predictions, the system achieves significant accuracy improvements without changing the underlying model demonstrating the practical value of uncertainty quantification. As coverage increases accuracy drops gradually. At 100% coverage accuracy dropped to baseline which is 88.3%. Shown by a red dashed line. The model predicted with low uncertainty (high confidence) samples accuracy increases substantially. As more uncertain samples are included, performance naturally drops toward baseline.

Table 2. Computational comparison (Baseline vs Optimized: Memory, Time, Cost)

Metric	Baseline	Proposed
Peak Memory (GB)	18.5	7.8
Throughput (img/s)	~9.5	10
Training Time (hrs)	6–8	2.1
Training Cost ($)	15–20	3.15

4.2 Computational Efficiency

The reported efficiency gains are achieved through a combination of mixed-precision training, gradient accumulation, and memory-efficient architectural design. Peak memory usage was measured as the maximum GPU allocation during training, resulting in 7.8 GB consumption, corresponding to 19.5% of an NVIDIA A100 GPU. Throughput was computed under steady-state training conditions and includes uncertainty estimation overhead. Training cost was derived from actual GPU runtime and standard cloud pricing, yielding a total cost of $3.15. Compared to full-precision, non-optimized baselines, the proposed framework achieves 58% memory savings and 79% cost reduction without sacrificing predictive performance.

All computational metrics were measured post-training under identical batch and precision settings, with peak memory obtained from maximum GPU allocation during forward–backward execution, throughput measured under steady-state conditions, and cost derived from wall-clock runtime and standard cloud GPU pricing.

V. Discussion

The experimental results demonstrate that all three identified research gaps are effectively addressed. The integration of uncertainty quantification enables reliable confidence-aware decision making, where low-uncertainty meaning high confidence where predictions achieve approximately 97% accuracy and high-uncertainty meaning low confidence cases are appropriately deferred for expert review, enhancing clinical safety. Robust minority-class performance is observed despite dataset imbalance, with no rare cancer subtype exhibiting an F1-score below 77%, supporting equitable diagnostic reliability across lesion categories. Furthermore, the proposed framework operates with low computational cost and modest hardware requirements, enabling deployment on consumer-grade GPUs and promoting broader accessibility of medical AI solutions. Collectively, these properties support practical, scalable, and trustworthy clinical deployment.

5.1 Study Constraints and Research Directions

This study is limited by its reliance on a single dataset (HAM10000), dermoscopic images only, and a retrospective design. Future work will focus on multi-dataset validation including ISIC 2019, incorporation of clinical photographs, prospective clinical trials, and enhancing model explainability via Grad-CAM [7]. Efforts will also target mobile deployment and improving generalization across diverse populations and imaging conditions.

VI. Conclusion

This study demonstrates that Swin Transformers can transition from research benchmarks to production-ready clinical tools through three innovations: Monte Carlo Dropout for uncertainty quantification (97% accuracy on confident cases), triple-strategy imbalance handling (83.8% minority F1), and memory optimization ($3 training cost, 8GB VRAM). The resulting system achieves 87.82% test accuracy with clinically meaningful confidence scores, enabling safe human-AI collaboration workflows. By addressing uncertainty, imbalance, and accessibility simultaneously, this work provides a practical framework for deploying advanced medical AI in resource-constrained healthcare settings.

References:

[1]. M. M. Musthafa, T. R. Mahesh, V. V. Kumar, and S. Guluwadi, "Enhanced skin cancer diagnosis using optimized CNN architecture and checkpoints for automated dermatological lesion classification," *BMC Medical Imaging*, vol. 24, no. 1, p. 201, 2024.

[2]. B. D. Revathy, R. Vijay, B. Ullas, N. H. Sagar, and M. Dhyan, "Skin cancer detection using CNN (Convolution Neural Network) with AI medical assistant," *International Journal for Multidisciplinary Research*, vol. 6, no. 6, p. 32582, 2024.

[3]. G. H. Dagnaw, M. El Mouhtadi, and M. Mustapha, "Skin cancer classification using vision transformers and explainable artificial intelligence," *Journal of Medical Artificial Intelligence*, 2024.

[4]. P. Yadla, "Low-rank adaptation with Swin transformers to enhance skin cancer diagnosis [Short Paper]," *Medical Imaging with Deep Learning (MIDL)*, accepted for publication, 2025.

[5]. Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. 33rd Int. Conf. Machine Learning (ICML)*, M. F. Balcan and K. Q. Weinberger, Eds., vol. 48, pp. 1050–1059, 2016.

[6]. A. Geifman and R. El-Yaniv, "Selective classification for deep neural networks," in *NeurIPS*, 2017.

[7]. R. R. Selvaraju et al., "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *ICCV*, 2017.