iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.82701

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Enhancing Skin Cancer Diagnosis on Resource-Constrained Devices Using a MobileViTv2-BatchFormer Framework

Sameeksha[1], Vaibhav Murarka[2], Tanish Bajaj[3], Sumedha Seniaray[4]
[1, 2, 3]*Department of Applied Mathematics, Delhi Technological University, New Delhi, India*
[4]*Assistant Professor, Department of Applied Mathematics, Delhi Technological University, New Delhi, India*

Abstract*—Skin cancer is a global health issue where early diagnosis significantly improves survival rates. Deep learning shows great promise, but most high-performance models require 28 to 88 million parameters, making them too heavy for deployment in remote clinics with limited resources. To address this, we present a novel training-time architectural augmentation framework for lightweight hybrid vision transformers that significantly improves diagnostic performance without increasing the final inference parameter count. We evaluated our approach on the highly imbalanced HAM10000 benchmark, achieving a competitive 88.32% overall accuracy while keeping the final model at only 2.9 million parameters. While larger state-of-the-art architectures achieve higher accuracy, our lightweight configuration surpasses them in balanced metrics like F1-score and Recall, despite having a significantly smaller parameter footprint. It also prevents majority classes from dominating the learning process, boosting accuracy of rare lesions like DF and BCC to 90.91% and 86.54%, respectively. Supported by Grad-CAM visualizations for transparency, this framework bridges the gap between fair performance and practical real-world deployment on edge devices.

Keywords*—skin cancer classification, biomedical application, edge computing, MobileViTv2, BatchFormer, Explainable AI, HAM10000, automated healthcare, lightweight deep learning.

I. INTRODUCTION

A. Background

Skin cancer has emerged as one of the most prevalent and swiftly growing cancer types worldwide. Melanoma, while constituting a small portion of all diagnosed skin cancers, unfortunately accounts for the majority of skin cancer-related deaths [1]. Dermoscopy is a technique that involves the non-invasive imaging of skin lesions using polarized light. Its implementation along with deep learning models has significantly improved early detection rates [2]. However, using high-performing classification models is still hampered by a fundamental tension of size. For example, leading architectures like SwinV2, ViT, and ConvNeXt are known for their strong benchmark performance but have a parameter count in the range of 28 to 88 million [3]. This makes them unsuitable for integration into mobile dermoscopy devices, point-of-care diagnostic tools, and wearable monitoring systems. Therefore, there is an urgent need for light, accurate, and fair models which can be deployed on edge devices. But this requirement is mostly unaddressed in the skin lesion classification literature.

Another complicating factor is the high-class imbalance that comes with publicly available dermoscopy datasets. The HAM10000 dataset [4], which is the de facto benchmark for seven-class skin lesion classification, comprises 10,015 dermoscopy images distributed over seven diagnostic categories with an extremely skewed distribution. Melanocytic Nevi (NV) has the largest count with 6,705 images, followed by Melanoma (MEL) with 1,113 and Benign Keratosis-Like Lesions (BKL) with 1,099 images. The four remaining classes are severely undersampled: BCC contributes just 514 images, AKIEC only 327, VASC only 142, and DF only 115. This clearly shows that DF is almost 58 times rarer than NV within the same dataset. Models trained without addressing this severe imbalance tend to exploit this statistical skew by overpredicting the majority class (NV), leading to degradation in recall and F1-scores.

B. Problem Statement

Most skin cancer classification methods use large-scale architectures with parameter counts too high for mobile or edge devices. In comparison, MobileViTv2 offers a compact 2.9 million parameter footprint ideal for lightweight deployment.

However, it has a lower baseline accuracy on general vision tasks, creating a performance gap in high-stakes medical diagnosis. Additionally, in the HAM10000 dataset, NV makes up 66.9% of all samples, meaning models trained without explicit balancing tend to favor majority classes while ignoring rarer classes like DF and VASC.

A critical research gap exists in creating lightweight architectures that perform comparably to massive models at a fraction of their parameter count, while also preserving minority-class recognition. Current literature shows hybrid vision transformers have potential, but most existing methods improve accuracy solely by scaling up parameter counts—a strategy that neither reduces deployment costs nor improves balanced metrics like F1-score, Recall, and Precision.

C. Objectives

We make the following primary contributions in this paper:

- A novel training framework integrating MobileViTv2-0.75 and BatchFormer that improves classification accuracy while keeping the final deployed model at just 2.9 million parameters.
- A comprehensive data augmentation and BatchFormer pipeline that boosts rare class accuracy while also improving other classes.
- Evidence of a scaling trend showing that doubling the batch size under BatchFormer consistently improves all performance metrics without increasing the model's parameter count.

II. RELATED WORK

Automated skin cancer diagnosis using deep learning has attracted significant research attention. Seven-class dermoscopy classification on HAM10000 has been conducted in many studies since it was established as a community benchmark by the ISIC 2018 Challenge [5]. Recent comparative work by Sajol et al. [3] evaluated 14 deep learning architectures on HAM10000, reporting top-1 accuracy numbers above 90% using architectures including SwinV2 (92.74%, □88M parameters) and ViT2 (92.66%, □86M parameters). While impressive, the large parameter counts make these models infeasible for integration on edge dermoscopy devices with limited compute budgets.

MobileViT [6] and MobileViTv2 [7] are general purpose lightweight vision transformers proposed by Mehta and Rastegari. MobileViTv2 reduces computational overhead compared to its predecessor but at the cost of lower top-1 accuracy on ImageNet. The 0.75 width multiplier variant has approximately 2.9 million parameters [7], making it naturally suitable for low-power edge devices. However, due to its reduced performance there is a need for a carefully designed training framework to maximize its performance on challenging medical datasets. This is one of the gaps our work addresses.

BatchFormer, proposed by Hou et al. [8], improves representation learning without adding any parameters at inference time, making it a strong candidate for improving overall accuracy and other performance metrics without affecting architecture parameters. Class imbalance in medical imaging has been approached through several strategies like SMOTE [9], cost-sensitive learning with class-weighted loss functions [10], and data augmentation [11]. A popular visual strategy called Gradient-weighted Class Activation Mapping (Grad-CAM) [12] makes deep learning explanations accessible by highlighting the image regions to which the model pays attention—crucial in clinical dermoscopy to gain physician trust.

III. METHODOLOGY

A. Dataset and Preprocessing

The HAM10000 dataset contains 10,015 dermoscopy images distributed across seven diagnostic categories: MEL, BCC, NV, AKIEC, BKL, DF, and VASC. These classes showed a highly imbalanced distribution as evident from Fig. 1. To form reproducible results, we created a train-validation-test split of 80:10:10 while maintaining class proportion, yielding 8,012, 1,001, and 1,002 images for training, validation, and testing respectively. All images were resized to 224 × 224 pixels using BICUBIC interpolation.

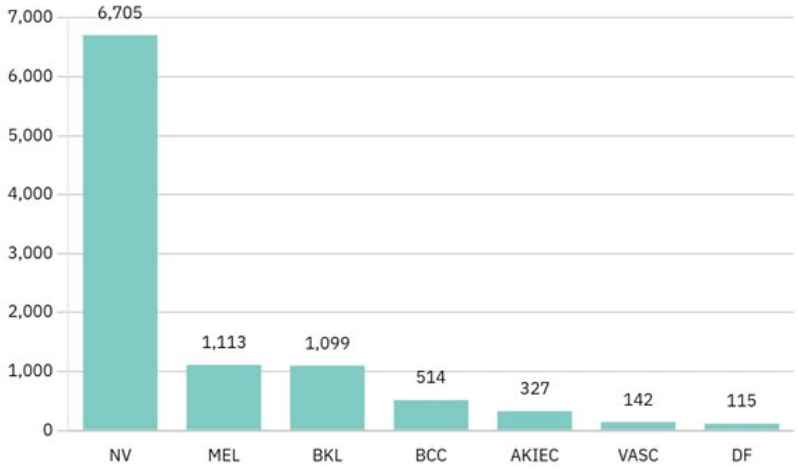


Fig. 1. The HAM10000 dataset's original class distribution.

B. Data Augmentation

The images are augmented as shown in Table I. All classes were brought to approximately 6,500 images each after augmentation. Validation and testing splits were left untouched to prevent data leakage or inflated accuracy. The exact testing split is shown in Table II. The validate and test splits were only resized and normalized.

TABLE I
DATA AUGMENTATION PIPELINE CONFIGURATION

Augmentation Technique	Parameters
Random Resized Crop	Size: 224×224, Scale: [0.67, 1.0], Interpolation: Bicubic
Random Horizontal Flip	Probability: p = 0.5
Random Vertical Flip	Probability: p = 0.2
Random Affine	Rotation: ±30°, Translation: ±10%, Shear: ±10, Interp.: Bicubic
Color Jitter	Brightness: 0.2, Contrast: 0.2, Saturation: 0.15, Hue: 0.02

TABLE II
IMAGE DISTRIBUTION IN THE UNAUGMENTED TEST SPLIT

Class (Lesion Type)	Test Image Count
MEL	112
NV	671
BCC	52
AKIEC	32
BKL	110
DF	11
VASC	14
Total	1,002

C. System Architecture & Pipeline

Our proposed architecture is shown in Fig. 2. It contains two parts: a MobileViTv2-0.75 backbone and a BatchFormer module applied only during training. MobileViTv2-0.75 is a lightweight hybrid vision transformer that uses local features from convolutional layers and global modelling from a separable self-attention mechanism. We use a 0.75 width multiplier to scale the backbone's channel dimensions, resulting in approximately 2.9 million parameters [7]. The backbone is initialized with pretrained ImageNet-1k weights from the timm library [13].

After applying global average pooling, the model extracts a 384-dimensional embedding for each input image. The BatchFormer module is implemented as a single transformer encoder layer using $d_{mo}^{del} = 384$, 4 attention heads, a feedforward dimension of 384, and a dropout rate of 0.5. During training, the module takes a batch of N feature vectors $x \in R^{N\times384}$ from the backbone and applies self-attention across the samples:

$$x_p = \text{TransformerEncoderLayer}(x) \quad (1)$$

The attended features are concatenated with the original features to produce a batch of 2N samples, with labels duplicated accordingly. A linear layer projects these 384-dimensional features to 7 target classes to obtain logits. The total training loss is computed as:

$$L = L^{ce}(\hat{y}_p^{re}, y) + L^{ce}(\hat{y}_{post}, [y; y]) \quad (2)$$

where L^{ce} represents the cross-entropy loss and y stands for the ground truth labels. During inference, BatchFormer is completely bypassed, so the deployed model retains an efficient 2.9M parameter footprint.

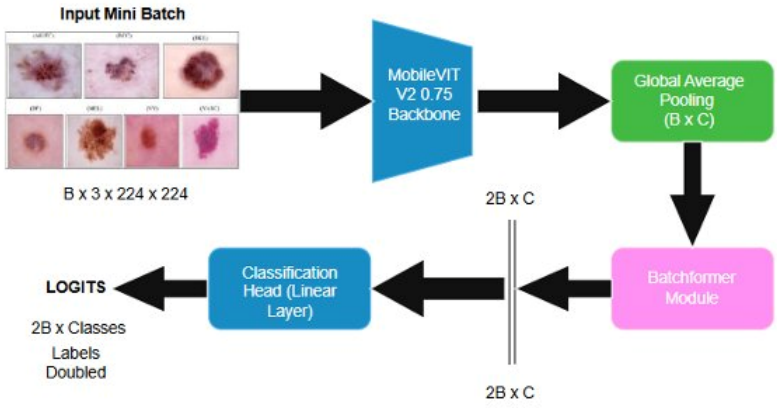


Fig. 2. Proposed pipeline: MobileViTv2-0.75 backbone feeds into the BatchFormer module during training (removed at inference), followed by a linear classifier.

D. Class Balancing Strategy

Our class balancing strategy comprises two complementary components. First, **Label Smoothing**: We used a smoothing value of 0.1 during training to restrict the architecture from making overconfident predictions on minority class samples. Second, **Macro Recall-based Checkpointing**: We select model checkpoints using validation macro recall instead of top-1 accuracy, ensuring the saved model maximizes performance across all seven classes rather than just optimizing for majority class prediction. Macro recall is defined as:

$$\text{Macro Recall} = (1/K)\ \Sigma\ TP_k / (TP_k + FN_k) \quad (3)$$

where K is the count of classes, TP_k is true positives, and FN_k is false negatives for class k.

E. Training Configuration

We train each configuration using the AdamW optimizer [14] with both learning rate and weight decay set to 10^{-4}, cosine annealing scheduling with $T_{ma}^{x} = 100$, and a minimum learning rate of 10^{-6}. Maximum epochs are set to 100, with early stopping activated if validation recall fails to improve for 20 continuous epochs. We also use PyTorch AMP with GradScaler for mixed precision training. Table III shows the three experimental configurations evaluated.

TABLE III
EXPERIMENTAL SETTINGS FOR THE ABLATIONS

Configs	BatchFormer (BF)	Batch Size	Description
Config 1	Disabled	256	Baseline Model
Config 2	Enabled	128	BF (Reduced Batch)
Config 3	Enabled	256	Proposed Best

IV. RESULTS AND DISCUSSION

A. Ablation Studies

We studied all three configurations to understand the contribution of different components. Our baseline (Config 1, MobileViTv2-0.75 only) yielded 84.93% accuracy. Integration of BatchFormer in the training pipeline improved accuracy to 87.43% (Config 2). Our third ablation showed that within BatchFormer, increasing the batch size from 128 to 256 improved metrics across the board: accuracy reaches 88.32%, F1-score 83.19%, precision 83.75%, and recall 82.98%. These gains were realized without any change in the inference-time parameter count of 2.9 million. Furthermore, this improvement also reflects in rare class accuracies: DF moves from 72.73% in the baseline to 90.91% in Config 3, and BCC improves by 5.77 percentage points. Table IV and Table V show the main quantitative results and class-wise accuracy across all three configurations. Fig. 3 and Fig. 4 show confusion matrices and training curves for Config 1 and Config 3 respectively.

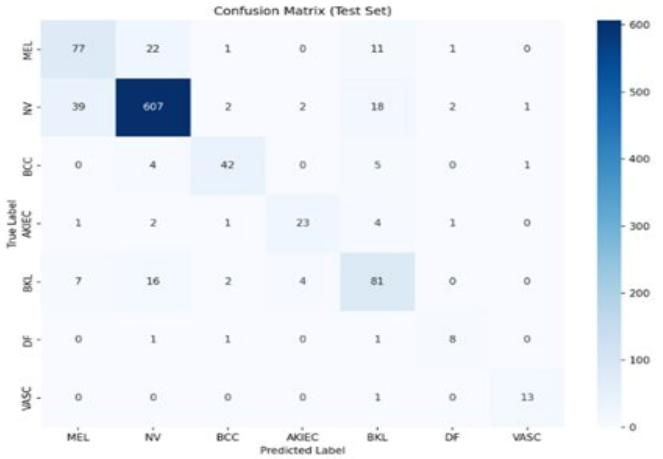


Fig. 3. Confusion matrix for Config. 1 (No BF, B=256), allowing transparent derivations of F1 score, Recall, Precision, and Class-Wise Accuracies.

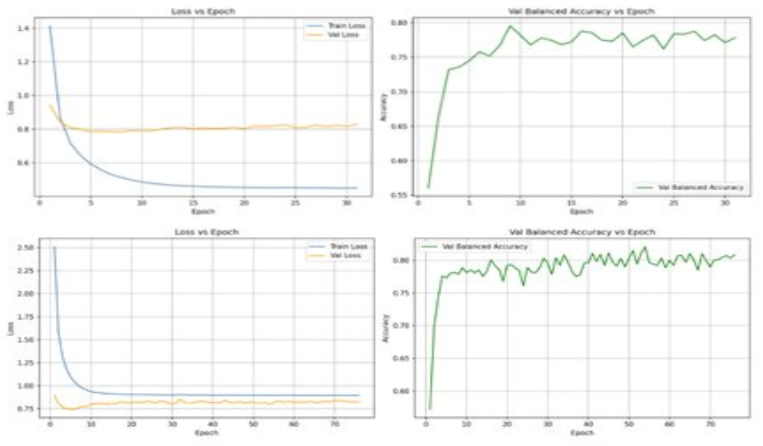


Fig. 4. Loss vs. Epoch and Val Balanced Accuracy vs. Epoch graphs for Config. 1 (No BF, B=256) and Config. 3 (BF, B=256).

TABLE IV
CLASSIFICATION PERFORMANCE ACROSS THREE ABLATIONS

Metric	Config 1	Config 2	Config 3
Accuracy	84.93%	87.43%	88.32%
Precision	77.21%	82.99%	83.75%
Recall	78.73%	81.68%	82.98%
F1 Score	77.85%	82.20%	83.19%

TABLE V
CLASS-WISE ACCURACY ON THE TEST SET ACROSS THREE ABLATION CONFIGURATIONS

Class	Config 1	Config 2	Config 3
MEL	68.75%	63.39%	66.07%
NV	90.46%	94.19%	94.93%
BCC	80.77%	84.62%	86.54%
AKIEC	71.87%	81.25%	75.00%
BKL	73.64%	73.64%	74.55%
DF	72.73%	81.82%	90.91%
VASC	92.86%	92.86%	92.86%

B. Effects of BatchFormer

It is well known in deep learning literature that increasing batch size can lead to worse generalization performance [15]. However, integrating BatchFormer completely changes how the model learns. Instead of looking at one image in isolation, BatchFormer uses an attention mechanism to look at the entire mini-batch at once, allowing the model to compare different skin lesions. Moving to a higher batch size boosted overall F1-score from 77.85% to 83.19%, with Precision and Recall also improving to 83.75% and 82.98% respectively. The steady improvement from batch size 128 to 256 under balanced conditions suggests that further increasing to 512 or 1024 could yield additional gains without architectural changes.

C. Explainability via Grad-CAM

We applied Grad-CAM to Config 3 to enhance model interpretability. Grad-CAM produces heatmaps using gradients that indicate which parts of the image the model paid most attention to, aiding clinician verification. Fig. 6 presents Grad-CAM outputs for some test samples. The visualizations confirm that the model focuses on clinically relevant lesion regions rather than background noise, which is essential for gaining physician trust in a clinical deployment context.

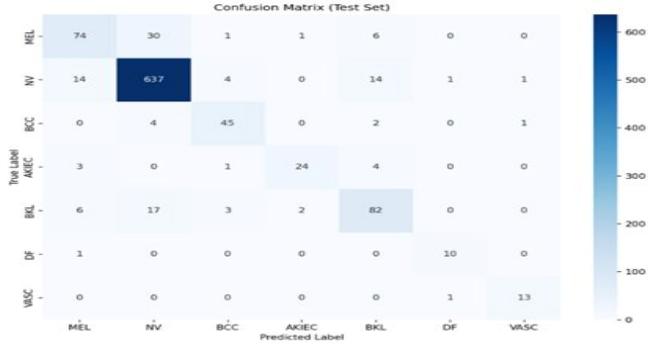


Fig. 5. Confusion matrix for Config. 3 (BF, B=256).

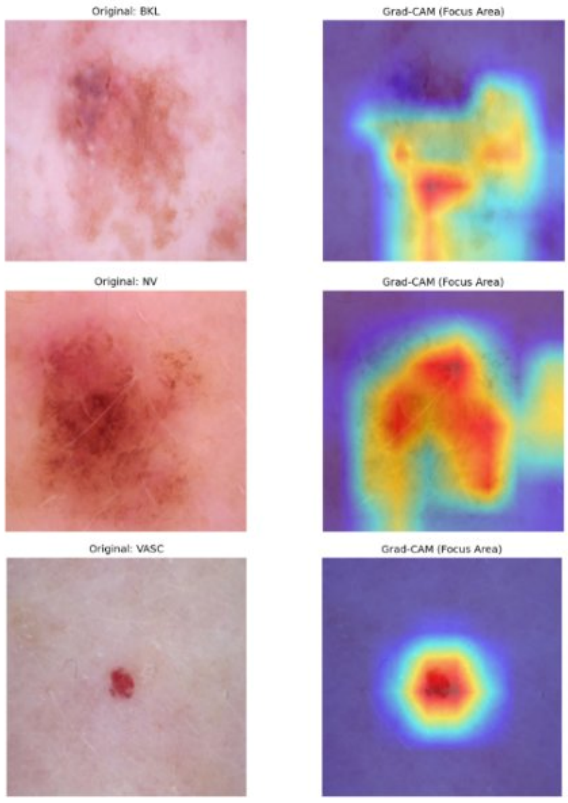


Fig. 6. Grad-CAM visualizations for Config. 3.

D. Comparison with Prior Work

Comparing Config 3 against current leading models underscores the practical advantages of our approach. Heavyweight architectures like SwinV2 and ViT2 achieve accuracy above 92% [3] but have parameter counts of □88M and □86M respectively. By contrast, our proposed framework achieves 88.32% accuracy with only 2.9 million parameters. Even more strikingly, it outperforms the larger models in precision, F1-score, and recall, as depicted in Table VI. The improvements observed with larger batches suggest that even higher performance metrics could be reached through additional training runs, making our proposed framework a highly practical candidate for real-world clinical deployment.

TABLE VI
COMPREHENSIVE PERFORMANCE EVALUATION WITH CURRENT LEADING ARCHITECTURES

Model	Params	Accuracy	Precision	Recall	F1
VGG [3]	~138.0M	91.56%	0.7600	0.7400	0.7500
SwinV2 [3]	~88.0M	92.74%	0.8500	0.7500	0.7900
ViT2 [3]	~86.0M	92.66%	0.8100	0.7400	0.7700
ResNet [3]	~25.6M	91.65%	0.8000	0.7000	0.7300
DenseNet [3]	~20.0M	91.84%	0.7800	0.7400	0.7500
GoogLeNet [3]	~6.8M	90.11%	0.7700	0.6900	0.7200
EfficientNet [3]	~5.3M	91.65%	0.7900	0.7800	0.7800
Proposed	~2.9M	88.32%	0.8375	0.8298	0.8319

V. CONCLUSION

In this paper, we introduced a novel training-time architectural augmentation framework for hybrid vision transformers that handles highly imbalanced dermoscopy data. This approach improves performance without increasing computational overhead. Specifically, we integrated MobileViTv2 with the BatchFormer module to tackle the HAM10000 dataset. The compact design of MobileViTv2 results in a drop from its baseline performance, but our approach counters this by using a strong data augmentation and BatchFormer pipeline, resulting in a capable diagnostic tool that stays at a 2.9 million parameter footprint.

The proposed setup reached a final accuracy of 88.32%, improving balanced metrics like F1-score, recall, and precision to 83.19%, 82.98%, and 83.75% respectively. By prioritizing fairness across all lesions, our model beats state-of-the-art architectures in F1, recall, and precision with only a small fraction of their parameter count. Overall, these results show that massive and heavy models are not required for precise and explainable skin cancer identification. Future work will include exploring further batch size scaling, testing more advanced augmentation strategies, and expanding evaluation to ISIC 2019 and ISIC 2020.

VI. ACKNOWLEDGMENT

The authors would like to thank the HAM10000 dataset contributors and the open-source community behind the timm library and PyTorch framework, which made this research possible.

REFERENCES

[1] R. L. Siegel, K. D. Miller, H. E. Fuchs, and A. Jemal, “Cancer statistics, 2021,” *CA: A Cancer Journal for Clinicians*, vol. 71, no. 1, pp. 7–33, Jan. 2021, doi: 10.3322/caac.21654.

[2] J. Kawahara, A. BenTaieb, and G. Hamarneh, “Deep features to classify skin lesions,” in *Proc. IEEE ISBI*, 2016, pp. 1397–1400, doi: 10.1109/ISBI.2016.7493528.

[3] M. S. I. Sajol, S. T. Alvi, and C. A. A. Era, “Performance assessment of advanced CNN and transformer architectures in skin cancer detection,” in *Proc. 11th Int. Conf. on EECSI*, 2024, pp. 1–8, doi: 10.1109/EECSI63442.2024.10776508.

[4] P. Tschandl, C. Rosendahl, and H. Kittler, “The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions,” *Scientific Data*, vol. 5, Art. no. 180161, Aug. 2018, doi: 10.1038/sdata.2018.161.

[5] N. C. F. Codella et al., “Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the ISIC,” 2019. [Online]. Available: https://arxiv.org/abs/1902.03368

[6] S. Mehta and M. Rastegari, “MobileViT: Light-weight, general-purpose, and mobile-friendly vision transformer,” in *Proc. ICLR*, 2022. [Online]. Available: https://openreview.net/forum?id=vh-0sUt8HlG

[7] S. Mehta and M. Rastegari, “Separable self-attention for mobile vision transformers,” 2022. [Online]. Available: https://arxiv.org/abs/2206.02680

[8] Z. Hou, B. Yu, and D. Tao, “BatchFormer: Learning to explore sample relationships for robust representation learning,” in *Proc. IEEE CVPR*, 2022, pp. 7246–7256, doi: 10.1109/CVPR52688.2022.00711.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic minority over-sampling technique,” *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002, doi: 10.1613/jair.953.

[10] M. Buda, A. Maki, and M. A. Mazurowski, “A systematic study of the class imbalance problem in convolutional neural networks,” *Neural Networks*, vol. 106, pp. 249–259, 2018, doi: 10.1016/j.neunet.2018.07.011.

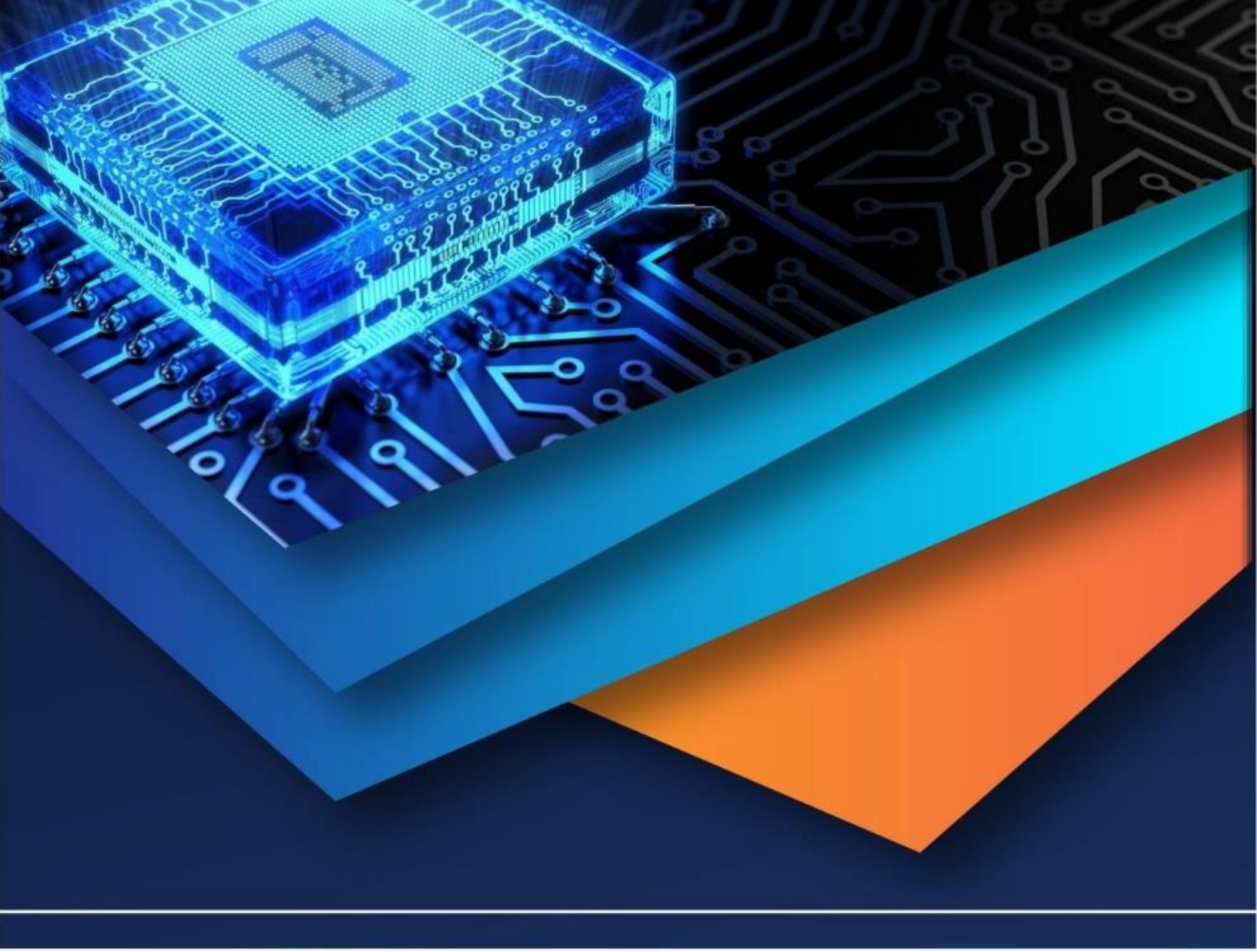
[11] A. Mumuni and F. Mumuni, “Data augmentation: A comprehensive survey of modern approaches,” *Array*, vol. 16, Art. no. 100258, 2022, doi: 10.1016/j.array.2022.100258.

[12] R. R. Selvaraju et al., “Grad-CAM: Visual explanations from deep networks via gradient-based localization,” in *Proc. IEEE ICCV*, 2017, pp. 618–626, doi: 10.1109/ICCV.2017.74.

[13] R. Wightman, “PyTorch image models (timm).” [Online]. Available: https://github.com/rwightman/pytorch-image-models. [Accessed Mar. 2026].

[14] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” in *Proc. ICLR*, 2019, doi: 10.48550/arXiv.1711.05101.

[15] S. Smith, P. Kindermans, C. Ying, and Q. V. Le, “Don’t decay the learning rate, increase the batch size,” in *Proc. ICLR*, 2018. [Online]. Available: https://openreview.net/pdf?id=B1Yy1BxCZ



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY