Lightweight Plant Leaf Disease Classification Using CLAHE-Enhanced EfficientNet-B0 with CBAM Attention

Asif Muhammad Sheharyar[1], Soo Kyun Kim[2*]
[1]Student, Department of Computer Engineering, Jeju National University
[2]Professor, Department of Computer Engineering, Jeju National University

CLAHE 영상 향상과 CBAM 어텐션을 적용한 EfficientNet-B0 기반 경량 식물 잎 질병 분류

아시프 무하마드 셰하르야르[1], 김수균[2*]
[1]제주대학교 컴퓨터공학과 학생, [2]제주대학교 컴퓨터공학과 교수

Abstract Plant leaf diseases threaten crop yield and quality, while manual diagnosis requires considerable time and expert knowledge. This paper proposes a lightweight plant leaf disease classification framework combining Contrast Limited Adaptive Histogram Equalization (CLAHE), EfficientNet-B0, and the Convolutional Block Attention Module (CBAM). CLAHE enhances lesion contrast, EfficientNet-B0 extracts hierarchical features, and CBAM emphasizes disease-relevant features. The proposed framework was evaluated using 20,638 images from 15 PlantVillage classes through repeated experiments with three random seeds (42, 52, and 62) and compared with MobileNetV2, ShuffleNetV2, and SqueezeNet1.1. Experimental results showed that the proposed model achieved the highest classification performance, with a mean accuracy of 99.860 ± 0.019% and a macro-F1 score of 99.868 ± 0.036%. These findings suggest that the proposed framework has potential as a lightweight model for plant leaf disease classification in resource-constrained computing environments.

Key Words : Plant disease classification, CLAHE, EfficientNet-B0, CBAM, Lightweight CNN, Attention mechanism, edge AI

요 약 본 식물 잎 질병은 수확량과 품질을 위협하지만, 수작업 진단은 시간과 전문 인력이 필요하다. 본 논문은 제한 대비 적응형 히스토그램 평활화(CLAHE), EfficientNet-B0 및 합성곱 블록 어텐션 모듈(CBAM)을 결합한 경량 식물 잎 질병 분류 프레임워크를 제안한다. CLAHE는 병변의 대비를 향상시키고, EfficientNet-B0는 계층적 특징을 추출하며, CBAM은 질병과 관련된 주요 특징을 강조한다. PlantVillage 데이터셋의 15개 클래스, 20,638장 이미지를 이용하여 세 가지 랜덤 시드(42, 52, 62)로 반복 실험을 수행하고, MobileNetV2, ShuffleNetV2 및 SqueezeNet1.1과 성능을 비교하였다. 실험 결과, 제안 모델은 평균 정확도 99.860 ± 0.019%와 Macro-F1 99.868 ± 0.036%를 달성하여 가장 우수한 분류 성능을 보였고, 이를 통해 제안 프레임워크가 제한된 컴퓨팅 환경에서 식물 잎 질병을 분류하기 위한 경량 모델로 활용될 가능성을 검증하였다.

주제어 : 식물 잎 질병 분류, 제한 대비 적응형 히스토그램 평활화(CLAHE), EfficientNet-B0, 합성곱 블록 어텐션 모듈(CBAM), 경량 합성곱 신경망, 어텐션 메커니즘, 엣지 인공지능

이 논문은 2026학년도 제주대학교 교원성과지원사업에 의하여 연구되었음
*교신저자 : 김수균(kimsk@jejunu.ac.kr)
접수일 2026년 08월 06일 수정일 2026년 10월 08일 심사완료일 2026년 10월 12일

www.kci.go.kr

1. Introduction

Plant diseases threaten agricultural productivity, while expert inspection is slow and difficult to scale. Image-based diagnosis offers a practical alternative, and the PlantVillage collection provided more than 50,000 curated images of healthy and diseased leaves and established a widely used benchmark for plant disease recognition[1,2].

CNNs and transfer learning achieve strong performance on controlled leaf images, but accuracy may decrease under changes in illumination, background, camera, or field conditions. In the original PlantVillage study, performance dropped substantially when models were evaluated on images collected outside the controlled dataset, highlighting the need for robustness[2,3]. Practical deployment also requires lightweight models that balance accuracy, model size, latency, and power consumption. Large CNN and transformer models improve representation but their memory, latency, and power requirements hinder deployment on smartphones, embedded computers, or edge devices. Lightweight architectures such as EfficientNet-B0, MobileNetV2, ShuffleNetV2, and SqueezeNet reduce computation through compound scaling, depthwise separable convolution, channel shuffling, or compact fire modules[5,8,9].

Input enhancement and attention can further improve recognition. CLAHE enhances local lesion contrast, while CBAM emphasizes informative channels and spatial regions[4,6]. Biomedical AI studies also support domain-aware preprocessing [15], local-global and lesion-focused attention [16], [17], and explainable data augmentation [18]. These principles motivate the proposed lightweight combination of CLAHE, EfficientNet-B0, and CBAM.

2. Related work

2.1 Deep Learning for Plant Disease Recognition

Early automated plant-disease systems relied on manually designed color, texture, and shape descriptors followed by conventional classifiers. Deep CNNs replaced this multi-stage process with end-to-end feature learning. Mohanty et al. trained AlexNet and GoogLeNet on PlantVillage and showed that deep representations could separate many crop-disease combinations under controlled acquisition conditions. Ferentinos subsequently evaluated multiple deep architectures across a broad disease dataset and further demonstrated the effectiveness of CNN-based diagnosis. These studies established deep learning as the dominant approach, but they also exposed a generalization gap between laboratory-like images and real agricultural scenes. [2,3]

2.2 Lightweight CNNs and Edge-Oriented Models

Lightweight CNN research attempts to preserve discriminative power while reducing parameters and multiply–accumulate operations. EfficientNet uses compound scaling to balance depth, width, and input resolution, providing a strong accuracy–efficiency trade-off. MobileNetV2 uses inverted residual blocks and linear bottlenecks; ShuffleNetV2 emphasizes memory-access efficiency through channel splitting and shuffling; and SqueezeNet reduces parameters through fire modules composed of squeeze and expand operations[5,12,13,14].

In agricultural applications, Guan et al. proposed Dise-Efficient for mobile and embedded recognition, while Khan et al. used MobileNetV3-small and post-training quantization for edge deployment. Chang and Lai applied RegNetY-400MF to potato disease recognition, illustrating the growing interest in compact backbones. These studies show that lightweight models can achieve strong controlled-dataset accuracy, but different papers often use different crops, data splits, image resolutions, and optimization settings, making direct comparison difficult[8,9,10].

www.kci.go.kr

2.3 Attention Mechanisms and Image Enhancement

CBAM is a lightweight attention module that sequentially emphasizes informative feature channels and discriminative spatial regions. In plant disease classification, EfficientNet-B0 with CBAM improved pear leaf symptom recognition with limited computational overhead, while Grad-CAM showed greater focus on diseased regions[6,7,11].

CLAHE complements attention by enhancing local contrast under uneven illumination. Because excessive enhancement may create artificial textures, this study applies CLAHE only to the luminance channel and uses the same settings for all compared backbones[4].

2.4 Transferable Strategies from Biomedical Deep Learning

Although biomedical diagnosis and plant disease recognition involve different domains, both require identifying subtle and localized abnormalities. Biomedical studies provide methodological insights rather than direct performance baselines. Previous research demonstrated the benefits of domain-aware preprocessing in ECG analysis [15] and lesion-focused attention in ocular disease recognition [16, 17]. Similarly, CLAHE enhances low-contrast lesion features in plant leaves, while CBAM emphasizes disease-relevant channels and spatial regions. Explainability and data augmentation approaches from biomedical studies [18] also offer directions for future research. Based on these principles, this study integrates CLAHE, EfficientNet-B0, and CBAM into a lightweight plant disease classification framework and compares its accuracy and computational efficiency with MobileNetV2, ShuffleNetV2 x1.0, and SqueezeNet1.1 under identical experimental conditions.

3. Dataset

3.1 Dataset Source and Composition

The Kaggle PlantVillage archive contains 20,638 readable RGB images from 15 healthy and diseased classes of pepper, potato, and tomato leaves. Class sizes range from 152 to 3,208 images, indicating substantial imbalance. No unreadable images were found[1,2]. Table 1 summarizes the composition of the dataset used in this study.

〈Table 1〉 Summary of the dataset used in this study

Images	Classes	Crops	Min class	Max class	Unreadable
20,638	15	3	152	3,208	0

3.2 Disease Classes

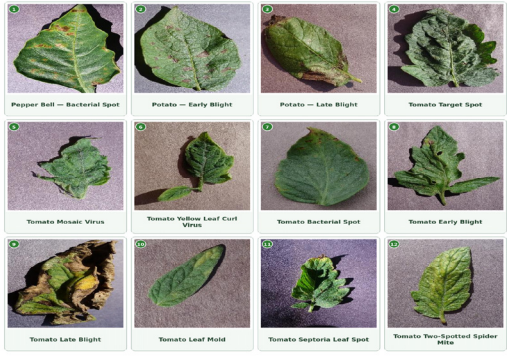
The dataset includes 15 classes representing healthy leaves and bacterial, fungal, viral, and pest-related diseases of pepper, potato, and tomato, as listed in Table 2.

〈Table 2〉 Fifteen PlantVillage leaf disease and healthy classes used in this study

Pepper bell-Bacterial spot	Pepper bell-Healthy	Potato-Early blight
Potato-Late blight	Potato-Healthy	Tomato-Bacterial spot
Tomato-Early blight	Tomato-Late blight	Tomato-Leaf mold
Tomato-Septoria leaf spot	Tomato-Spider mites	Tomato-Target spot
Tomato-Yellow leaf curl virus	Tomato-Mosaic virus	Tomato-Healthy

3.3 Representative Leaf Images

Figure 1 presents separate representative leaf appearances used to explain the visual differences learned by the classifier.



[Fig. 1] Separate representative examples of healthy tissue and common disease-symptom appearances

www.kci.go.kr

Healthy leaves contain relatively uniform green texture, while diseased leaves may show yellow spots, dark lesions, necrotic regions, or broad discoloration. These examples are included for visual explanation; the training procedure uses all images in the corresponding class directories.

3.4 Data Splitting and Preparation

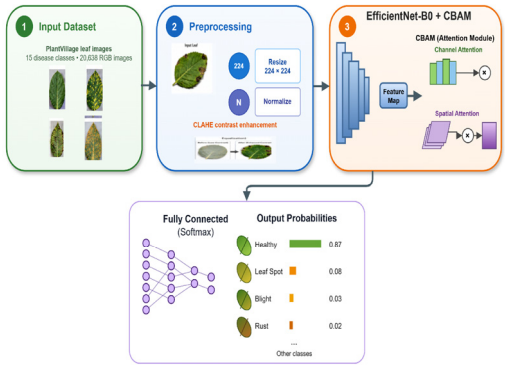
Images were resized to 224 × 224 pixels and stratified into training, validation, and test sets at a 70/15/15 ratio. Since the PlantVillage dataset lacks verified plant or leaf identifiers, image-level stratification was adopted to preserve class distributions and ensure consistent model comparisons. To improve evaluation reliability, experiments were repeated using three random seeds (42, 52, and 62) under identical preprocessing and training conditions. Results are reported as mean ± standard deviation across the three runs. However, image-level splitting may introduce bias due to visually similar images from the same plant appearing in different subsets, potentially overestimating performance. Future work will address this limitation through plant/leaf-level partitioning and independent field-image evaluation.

4. Proposed Methodology

4.1 Framework Overview

The proposed framework consists of five stages: input preparation, CLAHE enhancement, EfficientNet-B0 feature extraction, CBAM refinement, and Softmax classification.

The proposed EfficientNet-B0 model is compared with MobileNetV2, ShuffleNetV2 x1.0, and SqueezeNet1.1. To isolate the effect of the backbone, each comparison model receives the same CLAHE-enhanced images and the same CBAM module. Figure 2 summarizes the complete experimental workflow.



[Fig. 2] Proposed CLAHE-Efficient-B0-CBAM methodology and controlled lightweight-backbone comparison

4.2 CLAHE-Based Contrast Enhancement

Each RGB image is first converted to the CIELAB color space. CLAHE is applied only to the L (luminance) channel using a clip limit of 2.0 and an 8 × 8 tile grid. The enhanced luminance channel is then recombined with the original a and b chromatic channels and converted back to RGB. Applying CLAHE to luminance instead of independently processing red, green, and blue channels reduces color distortion while improving the visibility of weak lesions[4].

$$I_{CLAHE}(x,y) = \frac{L-1}{N_t} \sum_{i=0}^{I(x,y)} \tilde{h}(i) \qquad (1)$$

In (1), L is the number of intensity levels, N_t is the number of pixels in a local tile, and $\tilde{h}(i)$ is the clipped and redistributed tiles is used to reduce boundary artifacts. After enhancement, the image is resized and normalized before entering the CNN.

4.3 Efficient-B0 Feature Extractor

EfficientNet-B0 is selected as the proposed backbone because it provides a strong balance between representation quality and computational cost. Its mobile inverted bottleneck convolution (MBConv) blocks use expansion, depthwise convolution, squeeze-and-excitation, and projection operations. The network's original classification

www.kci.go.kr

layer is removed, and the final convolutional feature map is passed to CBAM. ImageNet-pretrained weights provide a stable initialization, while all layers are fine-tuned on the leaf dataset[5].

Let $F \in R^{C \times H \times W)}$ denote final backbone feature map, where C, H and W are the channel, height, and width dimensions. EfficientNet-B0 encodes local lesion boundaries in shallow layers and increasingly abstract disease patterns in deeper layers. Because these features still contain background and non-discriminative responses, attention is applied before global pooling.

4.4 CBAM Attention Refinement

CBAM sequentially applies channel attention and spatial attention. Channel attention aggregates each feature map using global average pooling and global max pooling, then passes both descriptors through a shared multi-layer perceptron. Spatial attention aggregates the channel-refined feature map along the channel dimension and applies a 7×7 convolution to estimate a location-sensitive attention map[6].

$$F_c = M_c(F) \odot F,\ F_s = M_s(F_c) \odot F_c \qquad (2)$$

In (2), M_c and M_s denote the channel and spatial attention maps, respectively, and odot operation represents element-wise multiplication with broadcasting. The first operation emphasizes useful feature types, such as color variation or lesion texture; the second emphasizes their most informative image regions. The resulting feature map F_s is forwarded to global average pooling.

4.5 Classification Layer and Objective

The refined feature map is compressed by global average pooling, followed by dropout and a fully connected layer with C outputs, where $C = 15$ for the present dataset, The logits are converted to class probabilities using Softmax:

$$p_k = \frac{\exp(z_k)}{\sum_{j=1}^{C} \exp(z_j)} \qquad (3)$$

In (3), z_k is the logit for class k and p_k is its predicted probability. Training minimizes weighted cross-entropy with label smoothing. Class weights are calculated from the training distribution so that small classes contribute more strongly to the loss. The checkpoint with the hightes validation macro-F1 is retained for final testing.

4.6 Lightweight Comparison Models

Three lightweight CNNs are used as controlled comparison backbones. MobileNetV2 relies on inverted residuals and linear bottlenecks; ShuffleNetV2 x1.0 uses channel splitting and shuffling to reduce practical computation; and SqueezeNet1.1 uses compact fire modules. The same preprocessing, CBAM block, classifier, augmentation, optimizer, learning rate, batch size, early stopping rule, split, and random seed are used for all models. Consequently, differences in performance primarily reflect backbone behavior rather than inconsistent experimental settings[12,13,14].

〈Table 3〉 Backbones included in the controlled comparison

Backbone	Core design	Pipeline	Study role
EfficientNet-B0	Compound scaling and MBConv	CLAHE + CBAM	Proposed model
MobileNetV2	Inverted residual and linear bottleneck	CLAHE + CBAM	Comparison 1
ShuffleNetV2 x1.0	Channel split and channel shuffle	CLAHE + CBAM	Comparison 2
SqueezeNet1.1	Squeeze-expand fire modules	CLAHE + CBAM	Comparison 3

4.7 Training and Evaluation Protocol

Model are trained with AdamW using an initial learning rate of 3×10^{-4}, weight decay of 1×10^{-4}, a batch size of 32, and maximum of 30 epochs. Early stopping is applied with a patience of eight epochs, and the checkpoint with the

www.kci.go.kr

highest validation macro-F1 is retained. Mixed-precision computation is enabled when CUDA is available. All results reported in this paper use the shared seed-42 split. The primary predictive metrics are accuracy, balanced accuracy, macro-precision, macro-recall, macro-F1, weighted-F1, and test loss. Macro-F1 is used for checkpoint selection because the dataset is imbalanced. Efficiency is assessed using trainable parameters, multiply-accumulate operations (MACs), approximate floating-point operations (FLOPs), checkpoint size, and batch-one inference latency. Latency values are hardware dependent and are therefore interpreted together with model complexity rather than as an architecture-only property.

5. Experimental Results

5.1 Predictive Performance

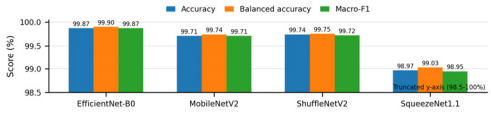
Table 4 summarizes the mean predictive performance across three independent runs using Seeds 42, 52, and 62.

〈Table 4〉 Single-seed predictive performance on the 15-class test set (seed 42)

Backbone	Accuracy (%)	Macro-F1 (%)
EfficientNet-B0	99.860 ± 0.019	99.868 ± 0.036
MobileNetV2	99.742 ± 0.056	99.748 ± 0.045
ShuffleNetV2 x1.0	99.742 ± 0.065	99.707 ± 0.109
SqueezeNet1.1	99.031 ± 0.065	99.002 ± 0.076

Among the evaluated models, EfficientNet-B0 achieved the best overall performance, with an average accuracy of 99.860 ± 0.019% and a macro-F1 score of 99.868 ± 0.036% indicating stable results across different random seeds. MobileNetV2 and ShuffleNetV2 delivered similarly high accuracy, whereas SqueezeNet1.1 showed comparatively lower predictive performance.

Figure 3 visualizes the small but consistent differences in accuracy, balanced accuracy, and macro-F1 across the four backbones.



[Fig. 3] Comparison of accuracy, balanced accuracy, and macro-F1 for the four lightweight backbones. The y-axis begins at 98.5% to make the small score differences visible

5.2 Computational Efficiency

The accuracy advantage of EfficientNet-B0 is accompanied by the largest computational footprint: 4.23 million parameters, 828.60 million approximate FLOPs, 6.62 ms batch-one latency, and a 16.43 MB checkpoint.

〈Table 5〉 Model complexity and batch-one inference efficiency

Backbone	Parameters (M)	MACs (M)	FLOPS (M)	Latency (ms)	Size (MB)
EfficientNet-B0	4.23	414.30	828.60	6.62	16..43
MobileNetV2	2.45	326.70	653.41	4.46	9.58
ShuffleNetV2 x1.0	1.40	152.02	304.04	6.30	5.51
SqueezeNet1.1	0.76	262.80	525.60	1.96	2.93

MobileNetV2 provides the strongest speed-accuracy compromise, reaching 99.71% accuracy with 4.46 ms latency. ShuffleNetV2 has the lowest operation count at 304.04 million FLOPs and only 1.40 million parameters, although its measured latency remains 6.30 ms, illustrating that FLOPs do not fully predict practical execution time. SqueezeNet1.1 is the smallest and fastest model at 0.76 million parameters, 2.93 MB, and 1.96 ms, but it yields the lowest predictive performance.

6. Discussion

EfficientNet-B0 achieved the best overall performance, although the margin over MobileNetV2 and ShuffleNetV2 was small. Thus, model

www.kci.go.kr

selection should consider deployment needs: EfficientNet-B0 for accuracy, MobileNetV2 for balanced speed and accuracy, ShuffleNetV2 for lower computation, and SqueezeNet1.1 for minimal storage and latency. The three-seed evaluation improved result reliability, while limitations remain due to the controlled PlantVillage setting, image-level partitioning, and unavailable leaf identifiers. Future work should include leaf-disjoint and field-based validation. The biomedical connection is methodological: CLAHE enhances subtle lesion patterns, while CBAM emphasizes informative spatial and channel features [15-17]. Explainability and augmentation also remain important future directions [18].

7. Conclusion

Across three independent cases, the proposed method, CLAHE-EfficientNet-B0-CBAM framework achieved 99.860 ± 0.019% accuracy and 99.868 ± 0.036% macro-F1, consistently outperforming the lightweight comparison models. The results indicate that the proposed framework offers an effective balance between predictive performance and computational efficiency for resource-constrained plant disease diagnosis, while future work will further evaluate its generalization on leaf-disjoint and field-acquired datasets.

REFERENCES

[1] D.P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," arXiv:1511.08060, 2015.

[2] S.P. Mohanty, D.P. Hughes, and M. Salathe, "Using deep learning for image-based plant disease detection," Frontiers in Plant Science, vol. 7, art. 1419, 2016, doi: 10.3389/fpls.2016.01419.

[3] K.P.Ferentinos, "Deep learning models for plant disease detection and diagnosis," Computers and Electronics in Agriculture, vol. 145, pp. 311-318, 2018, doi: 10.1016/j.compag.2018.01.009.

[4] K.Zuiderveld, "Contrast limited adaptive histogram equalization," in Graphics Gems IV, P.S.Heckbert, Ed. San Diego, CA, USA: Academic Press, 1994, pp. 474-485.

[5] M.Tan and Q.V.Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in Proc. 36th International Conference on Machine Learning, 2019, pp. 6105-6114.

[6] S.Woo, J.Park, J.-Y.Lee, and I.S.Kweon, "CBAM: Convolutional block attention module," in Proc. European Conference on Computer Vision, 2018, pp. 3- 19, doi: 10.1007/978-3-030-01234-2_1.

[7] P.Alirezazadeh, M.Schirrmann, and F.Stolzenburg, "Improving deep learning-based plant disease classification with attention mechanism," Gesunde Pflanzen, vol. 75, pp. 49-59, 2023, doi:10.1007/s10343-022-00796-y.

[8] A.T. Khan, S.M. Jensen, A.R. Khan, and S.Li, "Plant disease detection model for edge computing devices," Frontiers in Plant Science, vol. 14, art. 1308528, 2023, doi: 10.3389/fpls.2023.1308528.

[9] H.Guan, C.Fu, G.Zhang, K.Li, P.Wang, and Z.Zhu, "A lightweight model for efficient identification of plant diseases and pests based on deep learning," Frontiers in Plant Science, vol. 14, art. 1227011, 2023, doi: 10.3389/fpls.2023.1227011.

[10] C.-Y.Chang and C.-C.Lai, "Potato leaf disease detection based on a lightweight deep learning model," Machine Learning and Knowledge Extraction, vol. 6, no. 4, art. 114, 2024, doi: 10.3390/make6040114.

[11] R.R.Selvaraju et al., "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in Proc. IEEE International Conference on Computer Vision, 2017, pp. 618-626, doi: 10.1109/ICCV.2017.74.

[12] M.Sandler, A.Howard, M.Zhu, A.Zhmoginov, and L.-C.Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in Proc. IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510-4520, doi: 10.1109/CVPR.2018.00474.

[13] N.Ma, X.Zhang, H.-T.Zheng, and J.Sun,"ShuffleNet V2: Practical guidelines for efficient CNN architecture design," in Proc. European Conference on Computer Vision, 2018, pp. 116-131.

[14] F.N. Iandola et al., "SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size," arXiv:1602.07360, 2016.

[15] M.S.Asif, M.S.Faisal, M.N.Dar, M.Hamdi, H.Elmannai, A. Rizwan, and M. Abbas, "Hybrid deep learning and discrete wavelet transform-based ECG biometric recognition for arrhythmic patients and healthy controls," Sensors, vol. 23, no. 10, art. 4635, 2023, doi: 10.3390/s23104635.

www.kci.go.kr

[16] M.A.Iqbal, I.Han, and S.K.Kim, "Bilateral lesion-guided transformers with patient-level graph reasoning for multi-label retinal disease diagnosis," in Ubi-Media Computing, Pervasive Systems, Algorithms and Networks, Communications in Computer and Information Science, vol. 2940. Singapore: Springer, 2026, pp. 331-342, doi: 10.1007/978-981-95-9843-4_27.

[17] M.A.Iqbal, J.Kim,I.Han, and S.K.Kim,"Attention-driven feature fusion integrating Swin Transformer and CNN models for improved ocular disease classification," in Proc. 2024 International Conference on Engineering and Emerging Technologies(ICEET), 2024, pp.1-6 doi:10.1109/ICEET65156.2024

[18] M.D.Ali, M.A.Iqbal, S.Lee, X.Duan, and S.K.Kim, "Explainable AI based multi class skin cancer detection enhanced by meta learning with generative DDPM data augmentation," Applied Sciences, vol. 15, no. 21, art. 11689, 2025, doi: 10.3390/app152111689.

아시프 무하마드 셰하르하르
(Asif Muhammad Sheharyar) **[정회원]**



- 2020년 9월 : 콤사츠 대학교 (COMSATS University Islamabad) 컴퓨터과학과 졸업 (이학학사)
- 2023년 3월 : 콤사츠 대학교 (COMSATS University Islamabad) 컴퓨터과학과 졸업 (이학석사)
- 2026년 3월 ~ 현재 : 제주대학교 컴퓨터공학과 박사과정

〈관심분야〉
컴퓨터 비전, 딥러닝, 인공지능 (영상 처리, 객체 탐지, 이미지 분류)

김 수 균(Soo Kyun Kim) **[종신회원]**



- 2006년 2월 : 고려대학교 컴퓨터학과 (이학박사)
- 2006년 3월 ~ 2008년 2월 : 삼성전자 책임연구원
- 2008년 3월 ~ 2020년 2월 : 배재대학교 게임공학과 교수
- 2020년 3월 ~ 현재 : 제주대학교 컴퓨터공학과 교수

〈관심분야〉
컴퓨터비전 및 지능형 영상처리, 인공지능 기반 컴퓨터그래픽스 및 3차원 복원, 혼합현실

www.kci.go.kr