Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

Deep Convolutional Generative Adversarial Network-Enhanced Data Augmentation for Imbalance Facial Acne Severity Classification Using a Fine-Tuned EfficientNet-B1

Khoirun Nisya[*1], Sugiyarto Surono[2], Aris Thobirin[3]

[1,2,3]Department of Mathematic, Universitas Ahmad Dahlan, Indonesia

Email: [1]2200015033@webmail.uad.ac.id

Received : Dec 8, 2025; Revised : Dec 30, 2025; Accepted : Jan 5, 2026; Published : Apr 16, 2026

Abstract

Imbalanced datasets often hinder the generalization capability of Convolutional Neural Networks (CNNs) in medical image classification, leading to overfitting and reduced performance on minority classes. This study aims to develop an acne severity classification model using EfficientNet-B1 combined with geometric and photometric augmentation, as well as and Deep Convolutional Generative Adversarial Network (DCGAN)-based augmentation to address class imbalance. The dataset consists of 1,380 facial images categorized into four acne severity levels: Normal, Level 0, Level 1, and Level 2. Preprocessing includes RGB conversion, bilinear resizing, and center cropping. The data are split into training (80%), validation (10%), and testing (10%) sets. Geometric and photometric augmentation applies horizontal flipping, 45° rotation, color jittering, and random resized cropping, while DCGAN generates synthetic samples to balance minority classes. The EfficientNet-B1 model is fine-tuned using compound scaling, MBConv blocks, Swish activation, Batch Normalization, Cross-Entropy loss, and AdamW optimizer, with 5-fold cross-validation for robustness. Experimental results demonstrate that DCGAN-based augmentation achieves superior performance, with a test accuracy of 94% and an average F1-score of 0.93, outperforming geometric and photometric data augmentation (90% accuracy and 0.88 F1-score). DCGAN augmentation also significantly reduces misclassification between visually similar acne severity levels, particularly Level 0 and Level 1. These findings indicate that integrating DCGAN with EfficientNet-B1 effectively enhances generalization on imbalanced medical image datasets, providing a robust and replicable framework for acne severity classification and related medical imaging applications.

Keywords : *Acne Severity Classification, Convolutional Neural Network, Data Augmentation, DCGAN, Deep Learning, EfficientNet-B1.*

1. INTRODUCTION

Significant progress in deep learning has provided a robust mathematical basis for numerous applications in pattern recognition, especially within digital image processing. Convolutional Neural Networks (CNNs) excel at pulling out spatial features by using convolution operations. These operations are, mathematically, linear multiplications of filter kernels with image matrices, subsequently enhanced by non-linear activation functions [1][2]. The hierarchical nature of convolutional neural networks allows for a progressive feature extraction. Initial layers capture fundamental components such as edges and corners, while subsequent deeper layers synthesize these into more complex structures, textures, and distinctive patterns [3]. This step-by-step learning approach empowers CNNs to detect key patterns in images effectively, proving them a powerful tool for tasks involving image-based pattern recognition and classification [4]. As a result, CNNs perform exceptionally well in diverse image classification areas, including object recognition, medical diagnostics from images, and similar visual identification challenges [5][6].

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

Increasing the amount and variety of data is an important factor in the success of CNN model training, so data augmentation becomes a crucial part in image processing [7][8]. Challenges such as overfitting and data imbalance, especially in medical image classification, often cause model bias towards the majority class and reduce performance on the minority class. To overcome these problems, data augmentation is widely applied through geometric and photometric transformations such as rotation, translation, flipping, and color changes, as well as through Generative Adversarial Network (GAN)-based approaches [9]. Recent research shows that GAN-based augmentation is able to produce synthetic images that represent the minority data distribution more realistically than geometric and photometric augmentation, thereby increasing accuracy and F1-score values on imbalanced datasets [10][11]. This approach has also been proven effective in various medical image classification applications, such as ultrasound and mammography, especially when the amount of training data is limited [12][13]. Recent research shows that GAN models are effective in dealing with limited data volume and class imbalance in medical image datasets, with an increase in classification accuracy of about 6% compared to conventional methods on imbalanced medical data [14]. One of the widely used GAN variants is the Deep Convolutional Generative Adversarial Network (DCGAN), which is designed to generate high-quality synthetic images and enrich the variety of training data to improve the generalization capabilities of CNN models [15].

The choice of architecture is also a critical factor in determining CNN performance. Recent studies highlight that the EfficientNet architecture introduces the concept of compound scaling, a strategy that simultaneously adjusts depth, width, and input resolution to achieve improved efficiency [16]. Studies have shown that this architecture can reach high precision while using far fewer parameters than other models like ResNet and Inception [16]. For example, one investigation into lung cancer histopathology images found that EfficientNet-B1 attained a classification accuracy of 99.8% [17]. These benefits position EfficientNet-B1 as a well-suited choice for medical image classification, such as in the analysis of dermatological images [18][19].

Previous research applying CNNs to classify acne severity has reported a range of outcomes. MobileNetV2 was used in one study to differentiate four severity levels, reaching an accuracy of 87.29%[20]. Another study that evaluated architectures including VGG16, ResNet50, and InceptionV3 found the best accuracy to be 71% for a three-class acne severity task [21]. Such work confirms that CNN models can detect fine textural and color differences in images of acne-affected skin, achieving strong classification results [22].

Most studies on acne severity classification still rely on CNN with conventional augmentation, making it less effective in handling data imbalance in limited medical datasets [23][24]. Although GAN-based augmentation has been shown to improve performance on various medical images, the application of DCGAN combined with EfficientNet-B1 for acne severity classification is still rare in previous studies [25][26].

Based on existing literature, there remains potential to improve model efficiency and feature learning in medical image classification tasks. Therefore, this study implements the EfficientNet-B1 architecture combined with data augmentation using DCGAN. DCGAN is utilized to generate synthetic images that closely resemble the original dataset, thereby increasing training data diversity while preserving essential feature characteristics of acne images. This strategy aims to address data scarcity and reduce the risk of overfitting. EfficientNet-B1 is selected due to its *compound scaling* mechanism, which optimally balances network depth, width, and input resolution in a unified manner. Accordingly, this study develops an EfficientNet-B1 model integrated with DCGAN to achieve accurate and computationally efficient acne severity classification with strong generalization capability, even when trained on limited datasets.

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

2. METHOD

This study aims to develop a model for classifying acne severity levels using the EfficientNet-B1 architecture in combination with two augmentation strategies: geometric and photometric augmentation and DCGAN-based augmentation. The research workflow is organized sequentially, beginning with data preparation, image preprocessing, dataset partitioning, augmentation implementation, model training, and concluding with performance evaluation.

An overview of the entire methodology is depicted in Figure 1, outlining the key stages from dataset collection and preprocessing to splitting, augmentation, training, and evaluation.

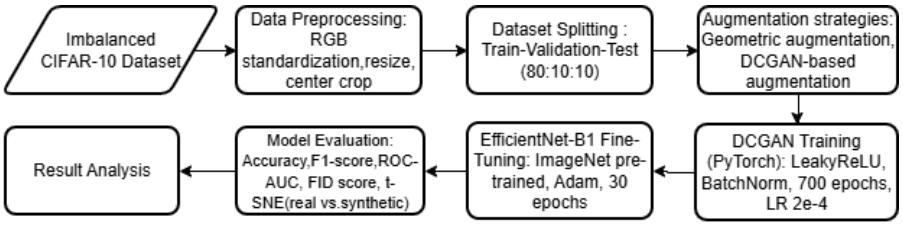


Figure 1. Research Flow

2.1. Research Data

The image data for this project was sourced from the Roboflow platform, specifically from a repository named "Acne Classification" (accessible at https://universe.roboflow.com/pytorchenv/acne-classification-kz7m8). It comprises facial images labeled into four acne severity categories: Normal, Level 0, Level 1, and Level 2. The images within each class show differences in skin appearance, redness, the presence of blackheads or whiteheads, lighting conditions, and camera angles. The original dataset exhibited an uneven distribution of samples across these classes, necessitating the use of augmentation in later steps. This dataset formed the foundation for training, validating, and ultimately testing the EfficientNet-B1 model developed in this work.

2.2. Data Preprocessing

The preprocessing phase involves standardizing image dimensions and formats before they are used for DCGAN augmentation and fed into the EfficientNet-B1 model. This step corrects inconsistencies in image sizes and color channels. Following this standardization, the complete dataset is divided into separate sets for training, validation, and testing.

2.2.1. Color Format Conversion

The color conversion step ensures every image has a uniform channel structure. Images in grayscale (with a single channel) are transformed into RGB format by replicating their single intensity value across three identical channels. For images with an RGBA format (which includes a transparency or alpha channel), the alpha channel is discarded, retaining only the red, green, and blue channels. Images already in standard RGB format are left unchanged.

This process guarantees all inputs adhere to the three-channel format expected by the neural network. Converting one-channel images to three channels also allows CNN models to learn from more detailed feature information than would be possible with grayscale inputs alone [27].

2.2.2. Resize

In this study, all images were resized to match the pixel dimensions required for input into both the DCGAN augmentation and the EfficientNet-B1 model. This resizing process can be expressed by the following equation:

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

$$T_{\text{resize}}: \mathbb{R}^{H_i \times W_i \times 3} \rightarrow \mathbb{R}^{240 \times 240 \times 3}. \quad (1)$$

The scaling technique applied is bilinear interpolation. This method determines the value of each new pixel by taking a weighted average of the four closest pixel values from the original image.

$$I_{\text{resized}}(x', y') = \sum_{i=0}^{1} \sum_{j=0}^{1} I(x_i, y_j)\ w(x', i)\ w(y', j), \quad (2)$$

The equation provided models this interpolation. In it $x'$and y' are the coordinates of a pixel in the new, resized image. The terms $x_i$and $y_j$correspond to the coordinates of the four original pixels that are nearest to this new location, used as reference points in the calculation.

$$x_i = \lfloor x' \rfloor + i, \quad (3)$$

$$y_j = \lfloor y' \rfloor + j, \quad (4)$$

The weights, $w(x', i)$and $w(y', j)$ are linear coefficients that define how much each of these four neighboring pixels influences the final calculated value for the new pixel.

This bilinear approach is a standard practice in image processing [28]. It results in resized images that have smoother gradients in color and brightness, reducing unwanted visual artifacts like blockiness or jagged edges when an image's dimensions are changed.

2.2.3. Center Crop

Preprocessing included a center crop operation to ensure that the main region of the images remained consistently positioned. This process is commonly used in image processing workflows to maintain composition and reduce edge noise. Center cropping has also been applied in recent studies using crop-based preprocessing to enhance the quality of visual features in deep learning models. [29].

2.2.4. Data Splitting

The dataset was divided using three parameters: α (alpha), β (beta), and γ (gamma). These represent the fractions of data designated for the training set, validation set, and test set, respectively. Each parameter is a value between 0 and 1, and they must collectively sum to 1.

$$\alpha + \beta + \gamma = 1 \quad (5)$$

$$| D_{\text{train}} | = \alpha N, | D_{\text{val}} | = \beta N, | D_{\text{test}} | = \gamma N \quad (6)$$

The number of samples in each subset is determined by multiplying its corresponding parameter (α, β, and γ) by N which is the total count of images in the full dataset. How the data is partitioned between training and evaluation phases is known to influence a model's final performance and its ability to generalize to new data, a consideration highlighted in meta-learning research [30].

2.3. Data Augmentation

2.3.1. Geometric and Photometric Augmentation

In this study, four distinct geometric and photometric augmentation methods were utilized: horizontal flipping, 45-degree rotation, color jittering, and random resized cropping. Flipping images horizontally varies the orientation of features within them. Rotating images by 45 degrees adds diversity to the apparent viewpoint, aiding the model's adaptability. Color jitter mimics different lighting environments, and random resized cropping forces the model to identify features regardless of their scale or exact position in the frame.

a. Horizontal Flip

Mathematically, a horizontal flip reflects an image of width W and height H across its vertical centerline. A pixel located at coordinates (x, y) is moved to its mirrored position at$(W - x - 1, y)$, producing a left-right reversed version of the original image.

$$I'(x, y) = I(W - x - 1, y) \quad (7)$$

Transformations like flipping and rotation are proven methods for making the training data more varied, which assists the model in developing stronger, more adaptable internal representations. Horizontal flipping, specifically, is a widely adopted technique to artificially broaden the range of object positions and angles seen during training, without changing what the objects fundamentally are [31].

b. Rotation

The rotation transform creates different perspectives by rotating every pixel's coordinates by a set angle. This enriches the dataset, helping the model become less sensitive to how an object is turned or oriented. In mathematical terms, this 2D rotation is expressed as a linear transformation.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (8)$$

This process maps each original pixel (x, y) to a new position (x', y'), allowing the same visual pattern to appear in multiple orientations. This approach is particularly effective for improving a model's generalizability when working with limited data, as rotation can reduce a network's reliance on specific poses and broaden the variety of spatial patterns it encounters [32].

c. Color Jitter

Color jitter augmentation alters an image's brightness and contrast to replicate the variations found in natural light. Brightness is changed by uniformly shifting pixel values, while contrast is adjusted by scaling how much pixels deviate from the image's average intensity. This process improves how well the model copes with different lighting scenarios and strengthens its performance across varied visual conditions [33], [34].

d. Random Resized Crop 6

Random Resized Crop is used as a spatial augmentation by randomly selecting a portion of the image and then resizing it back to the model's input dimensions. This technique increases the variation of visible object regions and prevents the model from relying on a fixed position, as objects can appear in different locations within the image [35].

Mathematically, the area of the selected region is expressed as follows:

$$A_{\text{crop}} = s \times A_{\text{orig}}, s \in [s_{\min}, s_{\max}] \quad (9)$$

where A_{orig} denotes the area of the original image, and $s$is the ratio of the randomly selected region within the range $[s_{\min}, s_{\max}]$. Selecting a random ratio allows the model to observe objects at varying scales and compositions, consistent with findings that random cropping can enhance model performance [36].

2.3.2. DCGAN-Based Augmentation

DCGAN is a generative architecture based on GANs that learns the distribution of the original data $p_{\text{data}}(x)$ using two parametric functions: the generator $G(z; \theta_G)$and the discriminator $D(x; \theta_D)$. The Generator learns to create new, synthetic samples that mimic the real data, while the Discriminator is trained to tell real images apart from the fake ones produced by the Generator, through an adversarial training process [37].

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

The Generator takes in a random vector of noise as its starting point.

$$z \sim p_z(z), p_z(z) = \mathcal{N}(0, I) \quad (10)$$

This noise vector acts as a source of randomness and a compressed representation of features [37]. The continuous nature of this noise space allows the Generator to produce a diverse array of image outputs. The Generator's function is to transform this noise vector into a synthetic image.

$$\hat{x} = G(z) \quad (11)$$

In this study, the Generator is built using a sequence of transposed convolutional layers, along with normalization and activation functions, to upscale a 100-dimensional noise vector into a 64x64 pixel image [38]. The full design is shown in Table 1, which details five successive ConvTranspose2D layers [37]. Batch normalization and a ReLU activation function follow each of these layers, with one exception: the final output layer. The last layer instead uses a Tanh activation function to adjust the pixel values so they fall within the range of [-1, 1].

Table 1. DCGAN Generator Architecture

Tipe Layer	Input (C×H×W)	Kernel	Output (C×H×W)	Aktivasi
ConvTranspose Block + BN	100 × 1 × 1	4×4	512 × 4 × 4	ReLU
ConvTranspose Block + BN	512 × 4 × 4	4×4	256 × 8 × 8	ReLU
ConvTranspose Block + BN	256 × 8 × 8	4×4	128 × 16 × 16	ReLU
ConvTranspose Block + BN	128 × 16 × 16	4×4	64 × 32 × 32	ReLU
ConvTranspose Layer (Output)	64 × 32 × 32	4×4	3 × 64 × 64	Tanh

The Discriminator outputs a probability score estimating how likely an input image is to be real.

$$D(x) \in [0,1] \quad (12)$$

The Discriminator *D* undergoes training to improve its skill in identifying fake images [38]. Its design incorporates four Conv2D blocks with a stride of 2, which progressively reduces the image's spatial dimensions. Each block uses a LeakyReLU activation and batch normalization, and the entire network finishes with a final Conv2D layer that outputs a single score indicating "real" or "fake". The complete layout is presented in Table 2 [37].

Table 2. DCGAN Discriminator Architecture

Tipe Layer	Input (C×H×W)	Kernel	Output (C×H×W)	Aktivasi
Conv Block	3 × 64 × 64	4×4	64 × 32 × 32	LeakyReLU
Conv Block	64 × 32 × 32	4×4	128 × 16 × 16	LeakyReLU
Conv Block + BN	128 × 16 × 16	4×4	256 × 8 × 8	LeakyReLU
Conv Block + BN	256 × 8 × 8	4×4	512 × 4 × 4	LeakyReLU
Conv Layer (Output)	512 × 4 × 4	4×4	1 × 1 × 1	Sigmoid

To enhance methodological transparency and reproducibility, the Deep Convolutional Generative Adversarial Network (DCGAN) architecture can be presented in a concise, framework-oriented manner. Prior studies describe DCGAN as a sequential composition of generator and discriminator networks, where the generator progressively transforms a latent noise vector through transposed convolutional layers and nonlinear activations, while the discriminator employs convolutional layers to distinguish real and synthetic samples. Such architectural descriptions emphasize layer-wise construction without revealing full training scripts, making them suitable for representation using PyTorch-style pseudocode [39][40].

Algorithm 1 : DCGAN Architecture (PyTorch-style pseudocode)

Input: Noise vector $z \in R^{100}$, real image x ∈ $R^{3\times64\times64}$
Output: Trained Generator G and Discriminator D
1: Initialize Generator G and Discriminator D
2: G ← Sequential(
ConvTranspose2d(100,512,4),
BatchNorm2d, ReLU,
ConvTranspose2d(512,256,4,2,1),
BatchNorm2d, ReLU,
ConvTranspose2d(256,128,4,2,1),
BatchNorm2d, ReLU,
ConvTranspose2d(128,64,4,2,1),
BatchNorm2d, ReLU,
ConvTranspose2d(64,3,4,2,1),
Tanh)
3: D ← Sequential(
Conv2d(3,64,4,2,1),
LeakyReLU(0.2),
Conv2d(64,128,4,2,1),
BatchNorm2d, LeakyReLU(0.2),
Conv2d(128,256,4,2,1),
BatchNorm2d, LeakyReLU(0.2),
Conv2d(256,512,4,2,1),
BatchNorm2d, LeakyReLU(0.2),
Conv2d(512,1,4),
Sigmoid)

The objective of DCGAN training is to solve a min–max optimization problem.

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D(G(z)))] \quad (13)$$

Discriminator Loss

$$L_D = -(\mathbb{E}_{x\sim p_{\text{data}}} \log D(x) + \mathbb{E}_{z\sim p_z} \log(1 - D(G(z)))) \quad (14)$$

Generator Loss (commonly used non-saturating version):

$$L_G = -\mathbb{E}_{z\sim p_z} \log D(G(z)) \quad (15)$$

This loss formulation is consistent with the original GAN objective [37].

For training the DCGAN in this project, the Adam optimizer was selected. This choice follows earlier successful implementations, as this setup promotes steady parameter updates throughout learning. This stability allows the model to learn progressively and yield synthetic images that are both consistent and of higher quality [41].

Once training stabilizes, the Generator produces images that closely resemble the real data, having learned to approximate the true data distribution at this point [37].

$$\hat{x} \approx p_{\text{data}} \quad (16)$$

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

The synthetic images created for each acne severity class are gathered as additional samples and merged with the original dataset.

$$\mathcal{D}_{\text{aug}} = \mathcal{D}_{\text{real}} \cup \mathcal{D}_{\text{syn}}. \quad (17)$$

This method is applied to rectify imbalances between classes and ensure that underrepresented classes have sufficient examples during training [42]. The final objective is to create a training set where every class is represented by an equal number of images.

Synthetic images generated by DCGAN do not represent real individuals and cannot be traced back to original facial images, as the generator only learns the statistical distribution of the data without reproducing identifiable patient information. Thus, the use of synthetic data maintains privacy and meets ethical principles in medical image analysis. To ensure reproducibility, all experiments were conducted with a fixed random seed (42) used in weight initialization, data randomization, and noise extraction during DCGAN training.

2.4. Model Training

2.4.1. Efficientnet-B1 Model Architecture

EfficientNet-B1 belongs to the EfficientNet series, which is designed around a concept called compound scaling. This method scales up three fundamental aspects of the network together: how deep it is, how wide it is, and the resolution of its input images. This coordinated scaling approach was developed to find an ideal trade-off between model accuracy and the computational resources required [43]. EfficientNet performs coordinated model scaling using three coefficients α, β, and γ, which are calibrated through grid search (neural architecture search). The complete formulation is as follows:

$$d = \alpha^{\phi}, w = \beta^{\phi}, r = \gamma^{\phi} \quad (18)$$

In the compound scaling formula, $\boldsymbol{d}$ represents the growth of network depth, $\boldsymbol{w}$ denotes the expansion of channel width, and $\boldsymbol{r}$ controls the increase in input resolution. The parameter $\boldsymbol{\phi}$ serves as a global scaling factor that determines the overall level of model scaling. Meanwhile, $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$ are scaling coefficients optimized through neural architecture search, ensuring that the three dimensions grow proportionally. EfficientNet-B1 uses $\boldsymbol{\phi} = \mathbf{1}$, which means all dimensions are increased in a coordinated manner relative to EfficientNet-B0 [43].

2.4.2. Mobile Inverted Bottleneck Convolution (MBConv)

The core building blocks of EfficientNet-B1 are MBConv modules. This efficient design first expands the number of channels to capture richer features, then applies depthwise convolution to process each channel separately (reducing computation), incorporates a squeeze-and-excitation (SE) module to emphasize important channels, and finally projects the data back to a lower channel count. Initially, the block increases its channel count to allow for a more detailed feature representation. The following depthwise convolution handles each input channel independently, making it far more efficient than a standard convolution [44]. The SE component then acts as an attention mechanism, recalibrating channel importance by boosting useful features and dampening less relevant ones, a technique known to improve model performance [45]. Lastly, the projection layer shrinks the channel dimension. When the input and output dimensions align, a residual connection is added to help gradients flow smoothly during training.

The complete layer-by-layer design of the EfficientNet-B1 model used in this work is detailed in Table 3. The table lists the progression of stages, the type of operation at each stage, the number of layers, the output channels, and the resulting feature map sizes [46]. This table provides the blueprint for the initial model setup before training and fine-tuning begin.

Table 3. Arsitektur Efficientnet-B1

Stage	Operator	Layer	Channels	Resolution (H x W)
1	Conv 3 x 3	1	32	240x240
2	MBConv1, k3	2	16	120x120
3	MBConv6, k3	3	24	120x120
4	MBConv6, k5	3	40	60x60
5	MBConv6, k3	4	80	30x30
6	MBConv6, k5	4	112	15x15
7	MBConv6, k5	5	192	15x15
8	MBConv6, k3	2	320	8x8
9	Conv 1x1+Pool +FC	1	1280	8x8

To reduce performance variance caused by a single data split, a 5-fold cross-validation strategy was applied on the balanced training dataset. In each fold, four subsets were used for training and one subset for validation. Model performance was reported as the mean and standard deviation across folds. The test set was kept separate and used only for final evaluation.

2.4.3. Swish Activation function

EfficientNet employs the Swish activation function, defined as:

$$\text{Swish}(x) = x \cdot \sigma(x) \tag{19}$$

Here, σ(x) refers to the standard sigmoid function.

$$\sigma(x) = \frac{1}{1+e^{-x}} \tag{20}$$

In this formula, the input $\boldsymbol{x}$ is multiplied by its sigmoid, allowing the activation to retain small negative values instead of discarding them as ReLU does. This property makes Swish more informative and capable of capturing more complex non-linear patterns. Its smooth and non-monotonic behavior has consistently demonstrated improved performance compared to ReLU across various modern classification architectures [45][47].

2.4.4. Batch Normalization

In this study, each convolutional layer in the architecture is paired with Batch Normalization to normalize the layer's activations before the non-linear activation function is applied. Batch Normalization operates by computing the mini-batch mean $\boldsymbol{\mu_B}$and the mini-batch variance $\boldsymbol{\sigma_B^2}$,

$$\hat{\boldsymbol{x}}_{\boldsymbol{i}} = \frac{\boldsymbol{x_i - \mu_B}}{\sqrt{\boldsymbol{\sigma_B^2 + \epsilon}}} \tag{21}$$

Specifically, each activation value is first centered by removing the mini-batch average, and then its scale is adjusted by dividing by the mini-batch standard deviation $\sqrt{\boldsymbol{\sigma_B^2 + \epsilon}}$. After the normalization step, the activations are re-adjusted through a learnable scaling and shifting process using the parameters $\boldsymbol{\gamma}$ (scale) and $\boldsymbol{\beta}$ (shift), which are optimized during training.

$$\boldsymbol{y_i = \gamma \hat{x}_i + \beta} \tag{22}$$

This step ensures that the activation values remain centered around zero with a variance close to one, while still allowing the network to adapt the scale and shift as needed during training. Batch Normalization not only accelerates convergence but also strengthens the model's generalization

capability. Its learnable scale and shift parameters play a crucial role in maintaining training stability and influencing the final accuracy [48][49]. In addition, the fluctuations in batch statistics act as an implicit regularizer that helps reduce overfitting.

2.4.5. Loss Function dan Optimizer

For this project, the Categorical Cross-Entropy Loss function served as the primary training criterion for the classification model. As a standard in the field, it measures the divergence between the model's predicted probability scores and the actual distribution of the target classes. This measurement is crucial for guiding the model during optimization. Furthermore, its proven reliability and stability make it a cornerstone for contemporary image classification tasks [50]. For each training sample $\boldsymbol{i}$ with $\boldsymbol{K}$ classes, the model's predicted probability is denoted as $\hat{\boldsymbol{p}}_{\boldsymbol{i,k}}$, while the target label is represented in one-hot form as $\boldsymbol{y}_{\boldsymbol{i,k}}$, the loss is calculated using the following equation:

$$\mathcal{L}_{CCE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} y_{i,k} \log \hat{p}_{i,k} \quad (23)$$

This method is commonplace in large-scale image classification research because it aligns with the principle of maximum likelihood estimation and avoids creating unstable gradient updates [50].

To optimize the model's parameters, this study uses AdamW, an enhanced version of the Adam optimizer that applies weight decay separately from the gradient updates. This decoupling has been demonstrated to lead to better model generalization and more stable training dynamics in large neural networks compared to the original Adam optimizer [51][52].

In each training step t, AdamW computes the first and second moments of the gradient as follows:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\, g_t, v_t = \beta_2 v_{t-1} + (1-\beta_2)\, g_t^2 \quad (24)$$

Which are then bias-corrected as:

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \hat{v}_t = \frac{v_t}{1-\beta_2^t} \quad (25)$$

With learning rate η and regularization coefficient λ, the parameter update is performed as:

$$\theta_{t+1} = (1-\eta\lambda)\,\theta_t \; - \; \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t}+\varepsilon} \quad (26)$$

This formulation emphasizes that the contribution of weight decay is no longer mixed into the gradient but is applied separately, allowing the regularization effect to operate more consistently [51]. In this study, a learning rate of $\boldsymbol{\eta = 1 \times 10^{-5}}$was used to ensure a more stable fine-tuning process and to prevent overly large parameter updates. Furthermore, the training procedure adopts a staged fine-tuning strategy, which has been shown to improve optimization stability and enhance model adaptation to the target dataset [53].

2.5. Model Evaluation

2.5.1. Classification Performance Evaluation

To assess how well the model performed, we used a confusion matrix. This table helps visualize the relationship between what the model predicted and the actual ground-truth labels.

In a binary classification setting, the confusion matrix is built from four basic counts. The True Positives (TP) count shows how many positive examples the model correctly labeled as positive. True

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

Negatives (TN) are the negative examples correctly identified as negative. On the other hand, False Positives (FP) occur when the model mistakenly labels a negative example as positive. False Negatives (FN) happen when it incorrectly labels a positive example as negative. These four values are arranged in the following matrix structure:

$$Confusion\ Matrix\ = \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} \qquad (27)$$

To provide a threshold-independent evaluation of the model's discriminative ability, the Receiver Operating Characteristic (ROC) curve is employed. The ROC curve represents the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR) across different classification thresholds. Mathematically, TPR and FPR are defined as:

$$\text{TPR} = \frac{TP}{TP+FN},\ \text{FPR} = \frac{FP}{FP+TN} \qquad (28)$$

where TP, FN, FP, and TN denote the number of correct and incorrect predictions for the positive and negative classes. Model performance is quantified using the Area Under the Curve (AUC), which corresponds to the area under the ROC curve and is defined as the integral of TPR with respect to FPR:

$$\text{AUC} = \int_0^1 \text{TPR}\ d(\text{FPR}) \qquad (29)$$

In practice, the AUC is estimated numerically using the trapezoidal rule based on discrete ROC points [54]. The AUC value represents the probability that the model assigns a higher prediction score to a randomly selected positive sample than to a randomly selected negative sample. For multi-class classification, the one-versus-rest strategy is applied by computing the AUC for each class, and the overall performance is summarized using the macro-average AUC, defined as the average AUC across all classes [54].

Based on the confusion matrix values, several threshold-dependent performance metrics are then computed. Accuracy measures the proportion of correct predictions over all samples and is calculated as:

$$Accuracy = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}} \qquad (30)$$

This shows that accuracy gives a general view of the model's overall correctness, but it can be misleading if the dataset has a severe class imbalance, as it might not reflect performance on the minority class [55].

To provide a balanced evaluation between precision and recall, the F1-score is used, which represents the harmonic mean of the two metrics. Mathematically, the F1-Score is defined as:

$$F1\text{-}score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (31)$$

The F1-score offers a more reliable single metric when dealing with imbalanced classes because it combines both the model's precision and its success rate in finding positive examples into one balanced figure [57].

2.5.2. GAN Quality Evaluation

The quality of synthetic images generated by GANs is evaluated using the Fréchet Inception Distance (FID), which measures the statistical distance between the feature distributions of real and synthetic images. Features are extracted using a pre-trained Inception-v3 network, and each feature

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

distribution is assumed to follow a multivariate Gaussian distribution with parameters $(\boldsymbol{\mu_r}, \boldsymbol{\Sigma_r})$ for real data and $(\boldsymbol{\mu_g}, \boldsymbol{\Sigma_g})$ for synthetic. Mathematically, FID is defined as:

$$\text{FID} = \| \mu_r - \mu_g \|_2^2 + \text{Tr}(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}) \quad (32)$$

where a smaller FID value indicates a higher similarity between the synthetic and real data distributions, thus reflecting better image quality and realism from GANs [58].

The evaluation of distributional similarity between real and synthetic data is performed using t-Distributed Stochastic Neighbor Embedding (t-SNE), which projects high-dimensional features into a two-dimensional space while preserving the local proximity structure between samples. Mathematically, t-SNE minimizes the Kullback–Leibler divergence between the probability distributions of data pairs in the original space and the low-dimensional projection space. In GAN evaluation, real and synthetic data are projected simultaneously; a high distributional overlap indicates similarity in feature characteristics between the two [59].

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a non-linear dimensionality reduction technique that projects high-dimensional data into a two-dimensional space while preserving the local proximity structure between samples [60],[61]. The underlying theory of t-SNE is based on minimizing the Kullback–Leibler divergence between the probability distributions in the original and projection spaces, resulting in a mapping that facilitates the visualization of data clusters [60]. In synthetic image quality evaluation, t-SNE can simultaneously visualize the distributions of real and synthetic features; a high overlap between the two indicates good feature similarity [61].

The evaluation of the significance of differences between the distributions of real and synthetic data is performed through statistical hypothesis testing at the individual feature level. The null hypothesis states that the distribution of synthetic data comes from the same population as the real data, and the difference is considered significant if the p-value < 0.05, indicating that the synthetic does not completely replicate the real distribution [62].

3. RESULT

3.1. Dataset

The images for this study are grouped into four acne severity categories: Normal, Level 0, Level 1, and Level 2. Each category represents a unique stage of facial skin condition. Consequently, the core task involves recognizing and distinguishing the specific visual patterns that characterize every class. The original, unprocessed dataset comprises 1,380 images in total. Figure 2 displays how these images are distributed among the different classes.

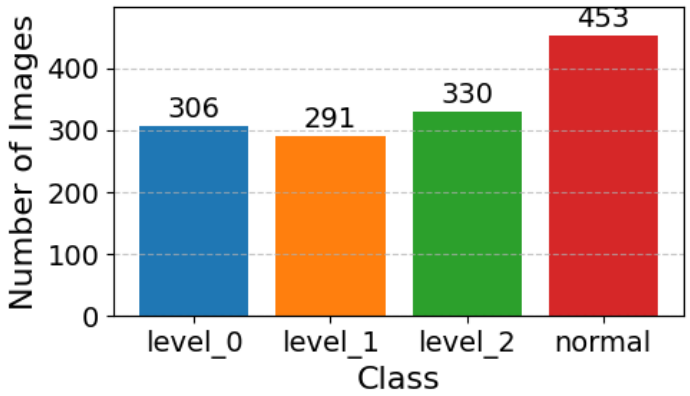


Figure 2. Data Distribution per Class Before Augmentation

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

From Figure 2, we can see the dataset is not balanced. The 'normal' class has the most images, while the 'level 0' and 'level 1' classes have a much smaller number of samples. This imbalance could negatively affect the model's learning, highlighting the need for augmentation to improve representation for all classes.

To provide a visual overview of the dataset used in this study, representative sample images from each class are presented below.

Figure 3. Example images from each class: (a) Normal, (b) Level 0, (c) Level 1, and (d) Level 2.

The images in Figure 3 show the visual traits of each class. 'Normal' skin is healthy, clear, and free of acne, blackheads, or redness. 'Level 0' represents very mild acne, typically just blackheads or whiteheads without inflammation. 'Level 1' is mild to moderate acne, with more visible papules, pustules, and some redness. 'Level 2' shows moderate to severe acne, characterized by a higher density of lesions, more pronounced redness, and potentially small, inflamed nodules.

3.2. Data Preprocessing

All images are first converted to a three-channel (RGB) format in accordance with the EfficientNet-B1 input standard. A resize operation followed by a center crop was then applied to ensure that the acne-affected region remained centered and consistent across samples. Each image was subsequently converted into the RGB color space to comply with the feature-extraction requirements of the model. Through these preprocessing steps, all images were standardized in terms of spatial dimensions and color format, making them ready for data splitting, augmentation, and model training.

The final preprocessing step involved partitioning the data into three distinct subsets for training, validation, and testing. Using a stratified strategy, the images were distributed to preserve the class balance: 80% were allocated for training, with 10% each reserved for validation and final evaluation.

Table 4. Distribution of images per class before augmentation

Class	Total	Train (80%)	Validation (10%)	Test (10%)
Normal	453	362	45	46
Level 0	306	244	31	31
Level 1	291	232	29	30
Level 2	330	264	33	33

Based on Table 4, all classes have been proportionally allocated into the training, validation, and test sets. These results form the basis for applying the augmentation stage to address the existing class imbalance.

3.3. Data Augmentation

To address class imbalance and increase dataset diversity for better model learning, two augmentation strategies were implemented: geometric and photometric augmentation and DCGAN-based augmentation.

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

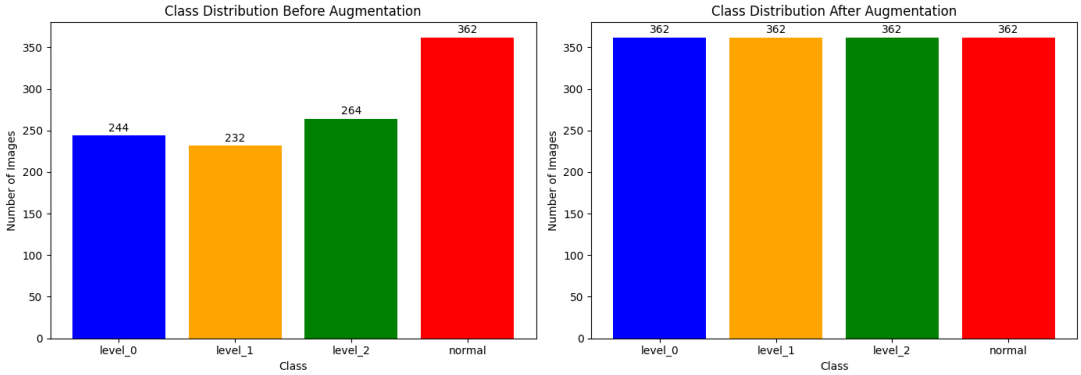


Figure 4. Distribution Before and After data augmentation

Figure 4 illustrates the class distribution before and after data augmentation. Before augmentation, the dataset exhibited clear class imbalance, with the Normal class containing the highest number of samples (362 images), while the Level 0, Level 1, and Level 2 classes had significantly fewer samples. After applying the augmentation strategies, the dataset became fully balanced, with each class containing 362 images. This balancing process aims to reduce bias toward the majority class and improve model learning robustness.

3.3.1. Geometric and Photometric Augmentation

Geometric and photometric augmentation was performed using simple transformations such as rotation, flipping, brightness adjustment, and cropping.

Table 5. Example of Geometric and Photometric Augmentation

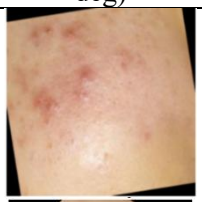
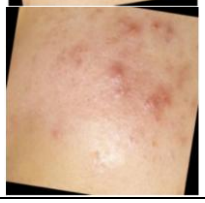
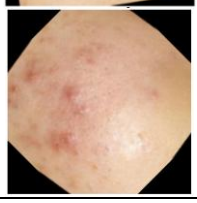
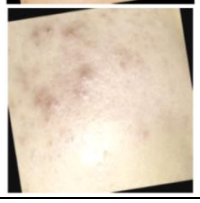
Augmentation Methods	Horizontal Flip	Rotation (45 deg)	ColorJitter	RandomResizedCrop
Original Image				
Geometric and Photometric Augmentation Synthetic Image				

Table 5 shows that the geometric and photometric augmentations preserve the acne characteristics within each class. Following augmentation, the dataset achieved class balance, with every category containing an equal number of samples equivalent to the count of the largest original class.

3.3.2. DCGAN-Based Augmentation

DCGAN-based augmentation was used to generate synthetic images, particularly for classes with fewer samples. DCGAN augmentation was applied only to Level 0, Level 1, and Level 2 classes. The goal was to bring the sample count for each of these classes up to 362 images, matching the 'Normal' class. The DCGAN was trained for 700 epochs until its generator could produce convincing synthetic images.

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

Table 6. Example of DCGAN Augmentation

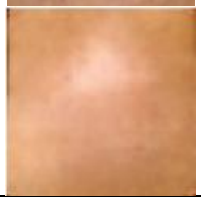
Class	Level 0	Level 1	Level 2
Original Image			
DCGAN-Based Synthetic Image			

Table 6 shows that the generated images exhibit acne patterns closely resembling the original images, making them suitable for use as additional data.

3.4. Model Training Results

We train our model using the EfficientNet-B1 architecture and evaluate it with two types of augmentations, namely geometric and photometric augmentation and DCGAN-based augmentation. Both augmentation methods were applied to assess their impact on the model's performance in classifying acne severity levels. The training was done in two, an initial training run for 20 epochs, followed by a fine-tuning stage for 10 more epochs. During fine-tuning, most of the model parameters were unfrozen except for the Batch Normalization layers. Throughout training, model performance was monitored using the training and validation datasets, while the progression of accuracy and loss was visualized through graphs.

3.4.1. Training with Geometric and Photometric Augmentation

Training the model on conventionally augmented data produced the accuracy and loss patterns shown in the following figure.

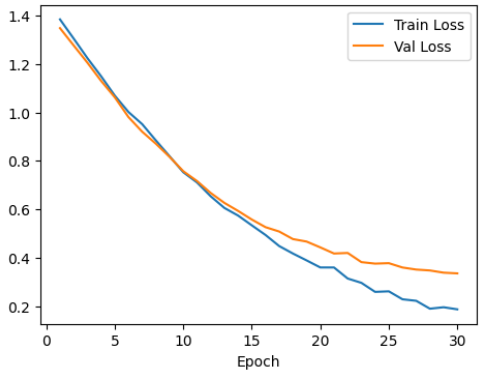


Figure 5. Training and Validation Loss with Geometric and Photometric Augmentation

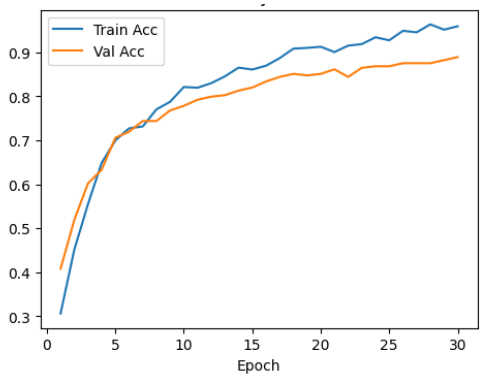


Figure 6. Training and Validation Accuracy with Geometric and Photometric Augmentation

Figure 5 illustrates the changes in training and validation loss across epochs. Both training and validation losses show a consistent downward trend, indicating effective learning. The training loss decreases steadily to around 0.18 by the final epoch, while the validation loss also declines smoothly and stabilizes at approximately 0.33. The close gap and parallel behavior between the two curves suggest good model convergence with minimal overfitting and stable generalization performance.

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

Figure 5 presents the training and validation accuracy over the training epochs. The training accuracy increases steadily and reaches approximately 0.95 toward the final epochs. Similarly, the validation accuracy shows a consistent upward trend, stabilizing around 0.88–0.90 with only minor fluctuations. The close alignment between the training and validation curves indicates that the model generalizes well without significant overfitting.

3.4.2. Training with DCGAN Augmentation

In this section, training was conducted using a dataset composed of synthetic images generated by DCGAN. DCGAN augmentation was applied to the Level 0, Level 1, and Level 2 classes to increase their sample sizes to match the Normal class, with 362 images per class.

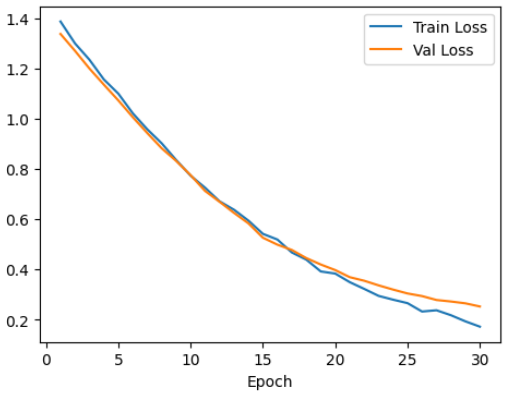


Figure 6. Training and Validation with DCGAN Augmentation

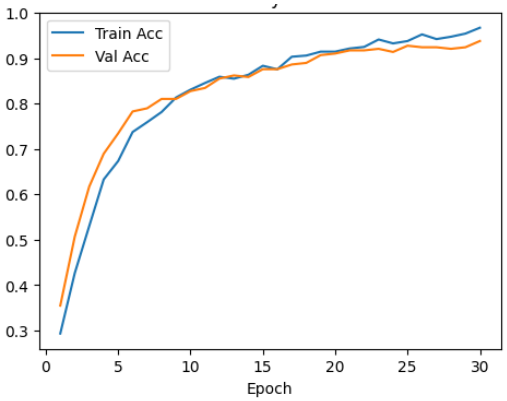


Figure 7. Training and Validation Accuracy with DCGAN Augmentation

Figure 7 illustrates the training and validation loss trends after applying DCGAN-based augmentation. Both losses decrease smoothly and consistently throughout the epochs. The training loss gradually drops to approximately 0.18 by the final epoch, while the validation loss follows a similar trajectory and stabilizes around 0.25. The minimal gap and parallel decline between the two curves indicate stable convergence and a more controlled learning process, suggesting that DCGAN augmentation effectively enhances model generalization compared to geometric and photometric augmentation.

Figure 8 illustrates the training and validation accuracy of the model across epochs. Training accuracy increases consistently and reaches approximately 0.96 at the final epoch, while validation accuracy follows a similar upward trend and stabilizes around 0.93–0.94. The small gap and strong alignment between the two curves indicate stable learning behavior with minimal overfitting. This suggests that the applied data augmentation strategy effectively enhanced data diversity, enabling the model to generalize well to unseen data.

3.4.3. K-Fold Cross-Validation Result and Descriptive Statistic

Table 7. K-Fold Cross-Validation Performance Comparison Between Augmentation Methods

Augmentation Method	Mean Accuracy	Std Accuracy	Mean F1-score	Std F1-score
No Augmentation	0.8802	0.0158	0.8788	0.0163
Geometric and Photometric	0.8791	0.0127	0.8778	0.0129
DCGAN	0.9157	0.0181	0.9150	0.0183

Table 7 presents the 5-fold cross-validation results for three training configurations: no augmentation, geometric and photometric augmentation, and the proposed DCGAN-based

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

augmentation. Performance is reported using mean accuracy and F1-score with their corresponding standard deviations to assess both effectiveness and stability.

Without augmentation, the model achieved a mean validation accuracy of 0.8802 and a mean F1-score of 0.8788, indicating a reasonable baseline but limited robustness when trained on imbalanced data. Applying geometric and photometric augmentation resulted in comparable performance (mean accuracy 0.8791, mean F1-score 0.8778), suggesting that conventional transformations provided minimal improvement over the baseline.

In contrast, DCGAN-based augmentation significantly improved performance, achieving a mean validation accuracy of 0.9157 and a mean F1-score of 0.9150. This corresponds to an absolute improvement of approximately 3.6% in accuracy and 3.7% in F1-score compared to geometric and photometric augmentation. These results demonstrate that DCGAN-based augmentation more effectively enhances feature diversity and model generalization across folds.

3.5. Model Evaluation

We conducted a final evaluation to measure the real-world performance of our EfficientNet-B1 model on the held-out test data, comparing results from both augmentation approaches. The results are summarized in a classification report with F1-score for each class, and visualized using confusion matrices.

Table 8. Comparison of Model Evaluation Results on Test Data

Class	F1-score Pre-Augmentation	F1-score (Geometrik and Photometric)	F1-score (DCGAN)
Level 0	0.91	0.88	0.93
Level 1	0.82	0.85	0.90
Level 2	0.92	0.95	0.92
Normal	0.93	0.93	0.98

Table 9. Comparison of Accuracy Results Between (Geometric and Photometric Augmentation) and (DCGAN-based Augmentation)

Method	Test
EfficientnetB1	0.90
Augmentasi Geometrik + EfficientNetB1	0.91
Augmentasi DCGAN + EfficientNetB1	0.94

We conducted a final evaluation to assess the real-world performance of the EfficientNet-B1 model on a held-out test dataset by comparing three training strategies: without augmentation, with geometric and photometric augmentation, and with the proposed DCGAN-based augmentation. Model performance was analyzed using class-wise F1-scores derived from the classification report and overall test accuracy, as summarized in Tables 8 and 9.

As shown in Table 8, DCGAN-based augmentation yields higher and more balanced F1-scores across all acne severity classes compared to geometric and photometric augmentation and the non-augmented baseline. Notable improvements are observed for minority and visually ambiguous classes, particularly Level 0 (0.93) and Level 1 (0.90), which are more challenging to classify. The Normal class also achieves the highest F1-score of 0.98, indicating improved feature representation and reduced misclassification. In contrast, geometric and photometric augmentation provides moderate improvements but remains less effective in enhancing performance for minority classes.

Furthermore, Table 9 demonstrates that DCGAN-based augmentation achieves the highest overall test accuracy of 0.94, outperforming geometric and photometric augmentation (0.91) and the non-augmented model (0.90). This consistent improvement across both class-wise and global metrics

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

indicates that DCGAN-generated synthetic data effectively mitigate class imbalance, reduce overfitting, and enhance the generalization capability of the EfficientNet-B1 model on unseen data.

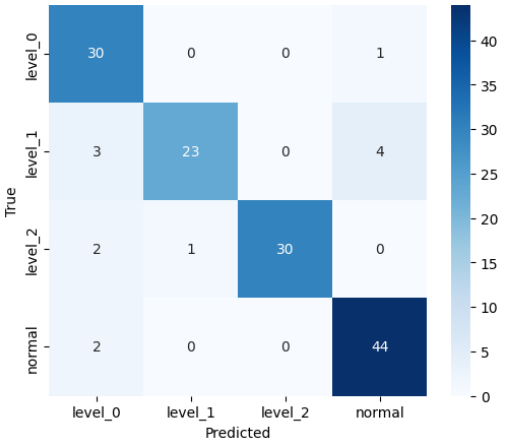


Figure 8. Confusion Matrix of the Model with Geometric and Photometric Augmentation

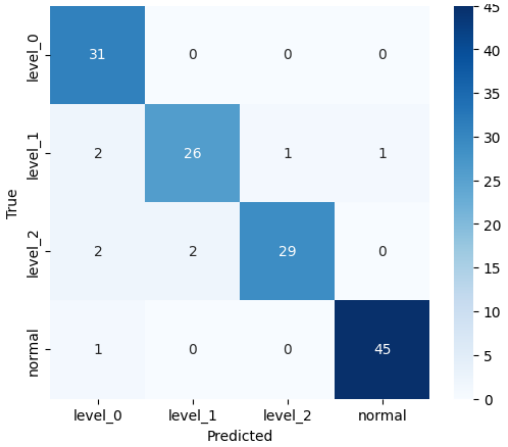


Figure 9. Confusion matrix of the Model with DCGAN Augmentation

As shown in Figure 9, under geometric and photometric augmentation, notable misclassification occurs in Level 1, where 23.3% of samples are incorrectly classified, primarily as Normal (13.3%) and Level 0 (10.0%), indicating difficulty in separating mild-to-moderate acne from adjacent classes. A smaller error is also observed for Level 0, with 3.2% of samples misclassified as Normal.

In contrast, Figure 10 demonstrates that DCGAN-based augmentation reduces misclassification across all classes. Notably, misclassification between Level 0 and Normal is completely eliminated (0%), while misclassification in Level 1 decreases to 13.3% and becomes more evenly distributed. These results indicate that DCGAN-generated synthetic data enhance feature diversity and improve class separability for visually similar acne severity levels.

Overall, compared to geometric and photometric augmentation, DCGAN-based augmentation consistently reduces both the frequency and concentration of misclassification errors, particularly between adjacent acne severity levels. This improvement highlights the effectiveness of DCGAN in enhancing feature diversity and improving class separability.

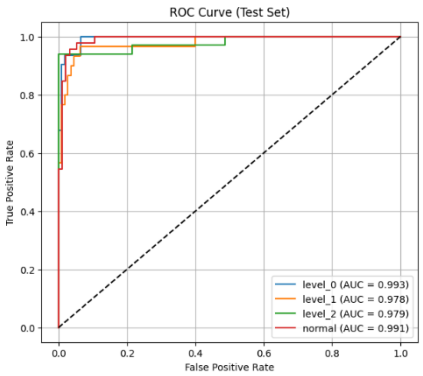


Figure 10. ROC–AUC Analysis Geometric and Photometric

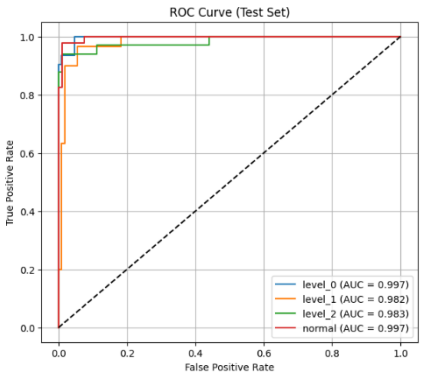


Figure 11. ROC-AUC Analysis DCGAN

As shown in Figures 11 and 12, both geometric and photometric augmentation, as well as DCGAN-based augmentation, achieve high ROC–AUC values across all classes, indicating strong overall discriminative capability. Under geometric and photometric augmentation (Figure 11), the AUC

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

values reach 0.993 (Level 0), 0.978 (Level 1), 0.979 (Level 2), and 0.991 (Normal). With DCGAN-based augmentation (Figure 12), the AUC values further improve to 0.997, 0.982, 0.983, and 0.997, respectively. The consistent AUC increase for Level 1 and Level 2 aligns with the reduced misclassification observed in the confusion matrix analysis, indicating improved discrimination among visually similar acne severity levels. Overall, while both augmentation strategies yield strong performance, DCGAN-based augmentation provides more robust and consistent class separation, supporting its effectiveness in enhancing model generalization.

3.6. GAN Quality Evaluation

To ensure that the synthetic images generated by DCGAN preserve the underlying visual characteristics of real acne images, an additional quality validation was conducted. This validation aims to assess the similarity between real and synthetic image distributions before the synthetic data are used for classification. Two complementary approaches were employed: quantitative evaluation using the FID and qualitative analysis using t-SNE visualization of feature embeddings.

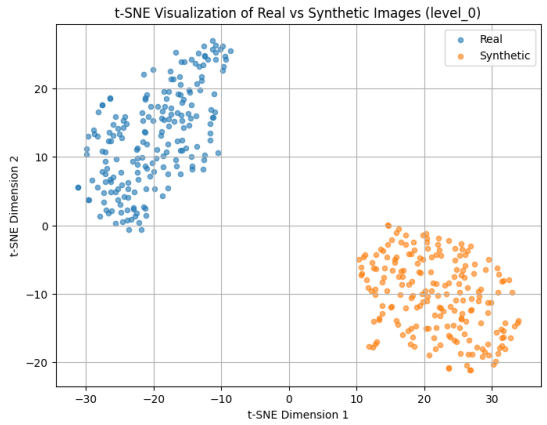
Table 10. FID Score per Acne Severity Level

Class	FID Score
Level 0	138.11
Level 1	137.17
Level 2	135.38

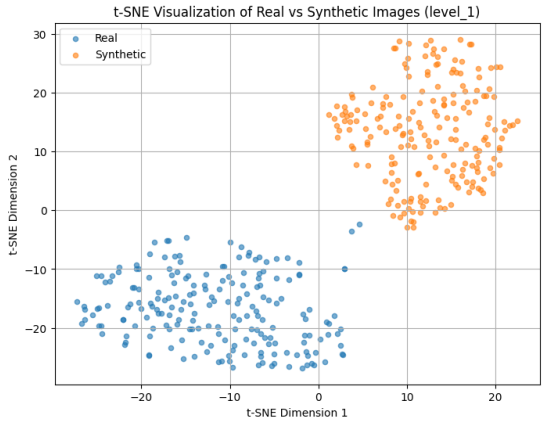
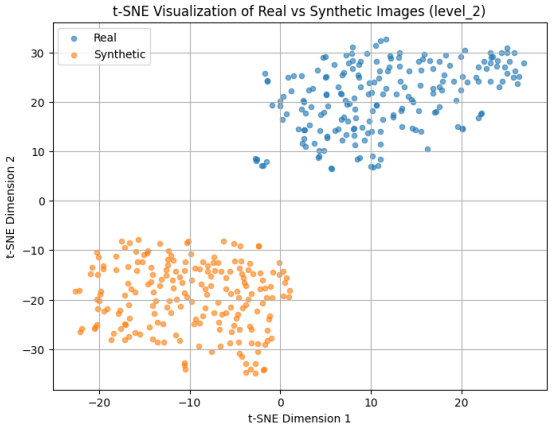
The FID measures the distance between feature distributions of real and synthetic images extracted using a pretrained Inception network, where lower values indicate higher similarity. As shown in Table 10, the FID scores for the generated images across Level 0, Level 1, and Level 2 classes range from 135.38 to 138.11. Although the absolute FID values are relatively high, the scores are consistent across classes, indicating that the DCGAN generates synthetic images with comparable distributional characteristics for each acne severity level.

It is important to note that the primary objective of DCGAN in this study is not to generate photorealistic images, but to augment minority classes and enrich feature diversity for improved classification performance. Therefore, FID is used as a complementary validation metric rather than the sole indicator of generative quality.

Table 11. t-SNE Visualization of Real and Synthetic Feature Distribution.

Level	t-SNE Visualization
Level 0	

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

Level 1	
Level 2	

Based on the t-SNE visualization in Table Y, it can be seen that at all three levels, the feature clusters of real and synthetic images remain clearly separated, with limited overlap. This pattern is consistent with the relatively high FID values, indicating differences in statistical distribution between the two data types.

At Level 0, cluster separation is most pronounced, consistent with the highest FID values. At Levels 1 and 2, despite a decrease in FID values, the t-SNE visualization still shows separation between clusters, although the synthetic distribution structure has become more stable and varied. This indicates that while generation quality has improved gradually, it is not yet sufficient to produce a feature distribution that truly resembles the real data.

Thus, t-SNE serves as a visual aid to strengthen the interpretation of FID results, rather than as the primary evaluation metric.

Referring to the average F1-score results in Table 7, further statistical analysis was performed using a paired t-test to evaluate the significance of the performance differences between the augmentation methods. The average F1-score differences (ΔF1) along with the test statistic values are summarized in Table.

Table 12. Paired t-test Results (Validation F1-score)

Comparison	Mean Difference (ΔF1)	t-statistic	p-value	Significance
No Augmentation vs Geometric and Photometric	−0.0010	−0.21	> 0.05	Not significant
No Augmentation vs DCGAN	+0.0362	5.74	< 0.001	Significant
Geometric and Photometric vs DCGAN	+0.0372	6.01	< 0.001	Significant

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

The paired t-test results indicate that geometric and photometric augmentation does not provide a statistically significant improvement over the no-augmentation baseline. This suggests that simple geometric and photometric transformations alone are insufficient to substantially enhance feature discrimination for acne severity classification.

In contrast, DCGAN-based augmentation yields a statistically significant improvement in validation F1-score compared to both no augmentation and geometric and photometric augmentation ($p < 0.001$). The consistent improvement across all folds demonstrates that the performance gain introduced by DCGAN is robust and not caused by random variation.

These findings confirm that DCGAN-generated synthetic images effectively enrich feature diversity in minority classes, leading to improved generalization performance. When interpreted alongside the FID and t-SNE analyses, the results indicate that even though the synthetic images do not fully replicate the real data distribution, they provide complementary feature representations that significantly benefit the classification task.

4. DISCUSSIONS

The results of this study demonstrate that DCGAN-based augmentation provides more consistent improvements in classification performance compared to geometric and photometric augmentation or no augmentation. The EfficientNet-B1 model trained with DCGAN-augmented data achieved a test accuracy of 94% and a macro F1-score of 0.915, while also reducing the gap between training and validation losses. This indicates that DCGAN is able to mitigate overfitting by enriching the feature diversity of the training data, particularly in the Level 0 and Level 1 minority classes, which previously experienced higher misclassification rates.

These findings align with recent studies reporting that GANs are effective in addressing size limitations and class imbalance in medical image datasets by generating synthetic data variations that enrich the training feature distribution [38], [63]. Several studies have also confirmed that GAN-based synthetic images can improve the performance of CNNs in dermatology and other medical image classification tasks [64], [42]. A recent systematic review even emphasized that GANs are highly relevant for AI-based medical applications, which typically have small and highly imbalanced datasets [65].

Quantitatively, the ablation study in this study showed that DCGAN removal caused a decrease in the macro F1-score of approximately 3.6% (from 0.915 to 0.878), confirming that the primary contribution to the performance improvement came from the presence of synthetic GAN-generated data. This strengthens the argument that DCGAN not only improves accuracy but also improves model generalization toward minority classes, as reflected in the increased recall at Level 0 and Level 1.

However, the quality of the synthetic data generated by DCGAN still has limitations. FID evaluation and t-SNE visualizations indicate that the feature distribution of the synthetic data does not fully resemble the real data. This condition aligns with previous findings that GANs still face issues of training instability and potential mode collapse, which can limit the realism of the synthetic data [4]. Furthermore, the DCGAN in this study is still limited to 64x64 image resolution, so high-resolution clinical details cannot be optimally utilized.

In terms of contributions to informatics and computer science, this study demonstrates that the integration of GAN–CNN forms an efficient synthetic augmentation framework to improve the generalization of classification models on small and imbalanced datasets. This framework is relevant for the development of more robust AI-based medical diagnostic systems and has potential applications in the healthcare and smart agriculture domains, which typically face limitations in labeled data.

In terms of scalability and future directions, several further steps are suggested. Future research can explore more stable GAN variants such as WGAN-GP to improve the quality and consistency of

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

synthetic data. Furthermore, the use of high-resolution CNN or Vision Transformer (ViT) architectures should be evaluated to capture finer clinical details. The integration of cross-domain transfer learning and the use of multi-ethnic datasets are also important to improve model robustness and generalization. Furthermore, CycleGAN can be considered for mitigating domain shift in clinical scenarios across devices and healthcare centers, thereby improving the model's readiness for real-world applications.

5. CONCLUSION

This This study demonstrates that integrating DCGAN-based data augmentation with EfficientNet-B1 effectively addresses data imbalance in acne severity classification. This approach achieves a test accuracy of up to 94% and an average F1-score of 0.92, while reducing overfitting compared to both unaugmented and geometric and photometric augmentation. Specifically, minority classes (Level 0 and Level 1) experience a post-GAN recall increase of up to ±15%, indicating improved model generalization.

From an informatics perspective, this study contributes to deep learning-based pattern recognition by demonstrating that synthetic GAN data can improve the generalization of CNN models on small medical datasets, which is relevant for AI-based clinical applications. Future research can be expanded through variations in CNN/ViT architectures, DCGAN hyperparameter tuning, CycleGAN integration for domain shift, and the use of multi-ethnic and high-resolution datasets to enhance the system's scalability and robustness.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest regarding the authorship or the subject matter of this paper.

ACKNOWLEDGEMENT

Our sincere gratitude goes to Universitas Ahmad Dahlan for the essential facilities and unwavering support that made this research possible. We are also grateful to the Roboflow platform for making the dataset available, which was essential for our data processing and experiments. Our thanks also go to all friends who offered help, encouragement, and support during this project. Finally, we sincerely thank the supervising lecturers in the Mathematics Study Program for their invaluable guidance, feedback, and mentorship throughout this study, as well as our families for their continuous prayers, patience, and moral support during the completion of this research.

REFERENCES

[1] M. Krichen, “Convolutional Neural Networks: A Survey,” *Computers*, vol. 12, no. 8, pp. 1–41, 2023, doi: 10.3390/computers12080151.

[2] Z. Wang *et al.*, “A Comprehensive Survey on Data Augmentation,” vol. 14, no. 8, pp. 1–20, 2025, [Online]. Available: http://arxiv.org/abs/2405.09591

[3] H. K. Jeong, C. Park, R. Henao, and M. Kheterpal, “Deep Learning in Dermatology: A Systematic Review of Current Approaches, Outcomes, and Limitations,” *JID Innov.*, vol. 3, no. 1, p. 100150, 2023, doi: 10.1016/j.xjidi.2022.100150.

[4] P. M. Shah *et al.*, “DC-GAN-based synthetic X-ray images augmentation for increasing the performance of EfficientNet for COVID-19 detection,” *Expert Syst.*, vol. 39, no. 3, pp. 1–13, 2022, doi: 10.1111/exsy.12823.

[5] I. D. Mienye, T. G. Swart, G. Obaido, M. Jordan, and P. Ilono, “Deep Convolutional Neural Networks in Medical Image Analysis: A Review,” *Inf.*, vol. 16, no. 3, pp. 1–28, 2025, doi: 10.3390/info16030195.

[6] J. Lee, T. Y. Kim, S. Beak, Y. Moon, and J. Jeong, “Real-Time Pose Estimation Based on ResNet-50 for Rapid Safety Prevention and Accident Detection for Field Workers,” *Electron.*,

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

vol. 12, no. 16, pp. 1–22, 2023, doi: 10.3390/electronics12163513.

[7] S. R. Yang, H. C. Yang, F. R. Shen, and J. Zhao, "Image Data Augmentation for Deep Learning: A Survey," *Ruan Jian Xue Bao/Journal Softw.*, vol. 36, no. 3, pp. 1390–1412, 2025, doi: 10.13328/j.cnki.jos.007263.

[8] A. Mumuni, F. Mumuni, and N. K. Gerrar, "A Survey of Synthetic Data Augmentation Methods in Machine Vision," *Mach. Intell. Res.*, vol. 21, no. 5, pp. 831–869, 2024, doi: 10.1007/s11633-022-1411-7.

[9] T. Kumar, R. Brennan, A. Mileo, and M. Bendechache, "Image Data Augmentation Approaches: A Comprehensive Survey and Future Directions," *IEEE Access*, vol. 12, pp. 187536–187571, 2024, doi: 10.1109/ACCESS.2024.3470122.

[10] S. Guan *et al.*, "Strip Steel Defect Classification Using the Improved GAN and EfficientNet," *Appl. Artif. Intell.*, vol. 35, no. 15, pp. 1887–1904, 2021, doi: 10.1080/08839514.2021.1995231.

[11] S. Rahmadani, A. Subekti, and M. Haris, "Improving Classification Performance on Imbalance Medical Data using Generative Adversarial Network," vol. 1, pp. 9–17, 2024.

[12] Z. Gao, "Enhancing Image Classification Performance via GAN-based Data Augmentation," vol. 0, pp. 10–20, 2025, doi: 10.54254/2755-2721/2025.22710.

[13] Y. Jim and D. Carri, "Biomedical Signal Processing and Control Gan-based data augmentation to improve breast ultrasound and mammography mass classification," vol. 94, no. February, 2024, doi: 10.1016/j.bspc.2024.106255.

[14] I. Khazrak, S. Takhirova, M. M. Rezaee, and M. Yadollahi, "Addressing Small and Imbalanced Medical Image Datasets Using Generative Models : A Comparative Study of DDPM and PGGANs with Random and Greedy K Sampling".

[15] T. Agustin, I. A. Saputro, and M. L. Rahmadi, "Optimizing Rice Plant Disease Classification Using Data Augmentation with GANs on Convolutional Neural Networks," *INTENSIF J. Ilm. Penelit. dan Penerapan Teknol. Sist. Inf.*, vol. 9, no. 1, pp. 97–114, 2025, doi: 10.29407/intensif.v9i1.23834.

[16] M. W. Purbandanu, R. Yanuarta, and A. Kurniawan, "Optimization of Skin Cancer Detection to Improve Accuracy with the Application of Efficient Convolutional Neural Network and EfficientNetB2 Models," *J. Intell. Comput. Heal. Informatics*, vol. 5, no. 2, pp. 43–50, 2024, doi: 10.26714/jichi.v5i2.14338.

[17] R. Javed, T. Saba, T. J. Alahmadi, S. Al-Otaibi, B. AlGhofaily, and A. Rehman, "EfficientNetB1 Deep LearningModel for Microscopic Lung Cancer Lesion Detection and Classification Using Histopathological Images," *Comput. Mater. Contin.*, vol. 81, no. 1, pp. 809–825, 2024, doi: 10.32604/cmc.2024.052755.

[18] N. S. Suriani, S. S. A. Tarmizi, M. N. H. Mohd, and S. M. Shah, "Acne Severity Classification on Mobile Devices using Lighweight Deep Learning Approach," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 6, pp. 680–687, 2024, doi: 10.14569/IJACSA.2024.0150668.

[19] H. Wen *et al.*, "Acne detection and severity evaluation with interpretable convolutional neural network models," *Technol. Heal. Care*, vol. 30, no. S1, pp. S143–S153, 2022, doi: 10.3233/THC-228014.

[20] F. Ramadhani, S. Rahardiantoro, and M. Masjkur, "Acne Severity Classification Study Using Convolutional Neural Network Algorithm with MobileNetV2 Architecture," *Indones. J. Stat. Its Appl.*, vol. 8, no. 2, pp. 112–128, 2024, doi: 10.29244/ijsa.v8i2p112-128.

[21] A. A. Odho, A. Bilal, N. Ur, and R. Malik, "ISSN (e) 3007-3138 (p) 3007-312X SKIN ACNE SKIN DISEASE CLASSIFICATION BY USING FINE TUNED CONVOLUTIONAL NEURAL NETWORK," vol. 3138, pp. 639–648, 2025, [Online]. Available: https://doi.org/10.5281/zenodo.15260008

[22] M. S. Junayed, M. B. Islam, A. A. Jeny, A. Sadeghzadeh, T. Biswas, and A. F. M. S. Shah, "ScarNet: Development and Validation of a Novel Deep CNN Model for Acne Scar Classification with a New Dataset," *IEEE Access*, vol. 10, pp. 1245–1258, 2022, doi: 10.1109/ACCESS.2021.3138021.

[23] N. Gao *et al.*, "Evaluation of an acne lesion detection and severity grading model for Chinese population in online and offline healthcare scenarios," pp. 1–11, 2025.

[24] K. Watanabe, K. Iinuma, C. Nakashima, H. Yamamoto, N. Oiso, and A. Otsuka, "Deep

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

Learning-Based Acne Severity Classification Using Standardized Facial Images of Japanese Patients," vol. 17, no. 9, 2025, doi: 10.7759/cureus.91944.

[25] J. Hussain, M. Båth, and J. Ivarsson, "Generative adversarial networks in medical image reconstruction : A systematic literature review," *Comput. Biol. Med.*, vol. 191, no. July 2024, p. 110094, 2025, doi: 10.1016/j.compbiomed.2025.110094.

[26] O. Rainio, "applied sciences Exploring Generative Adversarial Network-Based Augmentation of Magnetic Resonance Brain Tumor Images," 2024.

[27] J. Wang and S. Lee, "Data augmentation methods applying grayscale images for convolutional neural networks in machine vision," *Appl. Sci.*, vol. 11, no. 15, 2021, doi: 10.3390/app11156721.

[28] A. Kotte and S. S. Ahmad, "Implementation of innovative approach for detecting brain tumors in magnetic resonance imaging using NeuroFusionNet model," *Int. J. Electr. Comput. Eng.*, vol. 14, no. 6, pp. 6628–6641, 2024, doi: 10.11591/ijece.v14i6.pp6628-6641.

[29] D. A. Kurnia, O. Mohd, M. F. Abdollah, D. Sudrajat, D. M. Efendi, and S. Rahmatullah, "Digital Image Processing (DIP) and Generative Adversarial Networks (GANs) Techniques for Improvement Low-Resolution Face Recognition," *Ing. des Syst. d'Information*, vol. 29, no. 6, pp. 2251–2263, 2024, doi: 10.18280/isi.290615.

[30] Y. Bai *et al.*, "How Important is the Train-Validation Split in Meta-Learning?," *Proc. Mach. Learn. Res.*, vol. 139, pp. 543–553, 2021.

[31] A. Mumuni and F. Mumuni, "Data augmentation: A comprehensive survey of modern approaches," *Array*, vol. 16, no. November, p. 100258, 2022, doi: 10.1016/j.array.2022.100258.

[32] I. Hrga and M. Ivasic-Kos, "Effect of Data Augmentation Methods on Face Image Classification Results," *Int. Conf. Pattern Recognit. Appl. Methods*, vol. 1, no. Icpram, pp. 660–667, 2022, doi: 10.5220/0010883800003122.

[33] T. Islam, M. S. Hafiz, J. R. Jim, M. M. Kabir, and M. F. Mridha, "A systematic review of deep learning data augmentation in medical imaging: Recent advances and future research directions," *Healthc. Anal.*, vol. 5, no. April, p. 100340, 2024, doi: 10.1016/j.health.2024.100340.

[34] J. Ma, C. Hu, P. Zhou, F. Jin, X. Wang, and H. Huang, "Review of Image Augmentation Used in Deep Learning-Based Material Microscopic Image Segmentation," *Appl. Sci.*, vol. 13, no. 11, pp. 1–18, 2023, doi: 10.3390/app13116478.

[35] R. Yang, R. Wang, Y. Deng, X. Jia, and H. Zhang, "Rethinking the random cropping data augmentation method used in the training of cnn-based sar image ship detector," *Remote Sens.*, vol. 13, no. 1, pp. 1–23, 2021, doi: 10.3390/rs13010034.

[36] S. Mishra *et al.*, "Object-aware Cropping for Self-Supervised Learning," *Trans. Mach. Learn. Res.*, vol. 2022-Decem, no. 2012, pp. 1–15, 2022.

[37] S. Surono, C. W. Onn, A. Mujhid, N. Irsalinda, and G. K. Wen, "Deep convolutional generative adversarial networks for data imbalance in convolutional neural networks for facial expression classification," vol. 1, no. 1, pp. 1–8, 2023.

[38] A. Makhlouf, M. Maayah, N. Abughanam, and C. Catal, "The use of generative adversarial networks in medical image augmentation," *Neural Comput. Appl.*, vol. 35, no. 34, pp. 24055–24068, 2023, doi: 10.1007/s00521-023-09100-z.

[39] A. H. Basori and S. J. Malebary, "applied sciences Hybrid Deep Convolutional Generative Adversarial Network (DCGAN) and Xtreme Gradient Boost for X-ray Image Augmentation and Detection," 2023.

[40] J. Blarr, S. Klinder, W. V Liebig, K. Inal, L. Kärger, and K. A. Weidenmann, "Deep convolutional generative adversarial network for generation of computed tomography images of discontinuously carbon fiber reinforced polymer microstructures," *Sci. Rep.*, pp. 1–13, 2024, doi: 10.1038/s41598-024-59252-8.

[41] X. Du, X. Ding, M. Xi, Y. Lv, S. Qiu, and Q. Liu, "A Data Augmentation Method for Motor Imagery EEG Signals Based on DCGAN-GP Network," *Brain Sci.*, vol. 14, no. 4, 2024, doi: 10.3390/brainsci14040375.

[42] M. S. Meor Yahaya and J. Teo, "Data augmentation using generative adversarial networks for images and biomarkers in medicine and neuroscience," *Front. Appl. Math. Stat.*, vol. 9, 2023, doi: 10.3389/fams.2023.1162760.

[43] M. Tan and Q. V. Le, "EfficientNetV2: Smaller Models and Faster Training," *Proc. Mach.*

Learn. Res., vol. 139, pp. 10096–10106, 2021.
[44] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, *A review of convolutional neural networks in computer vision*, vol. 57, no. 4. Springer Netherlands, 2024. doi: 10.1007/s10462-024-10721-6.
[45] D. B. Mulindwa and S. Du, "An n-Sigmoid Activation Function to Improve the Squeeze-and-Excitation for 2D and 3D Deep Networks," *Electron.*, vol. 12, no. 4, 2023, doi: 10.3390/electronics12040911.
[46] R. Raza *et al.*, "Lung-EffNet: Lung cancer classification using EfficientNet from CT-scan images," *Eng. Appl. Artif. Intell.*, vol. 126, no. PB, p. 106902, 2023, doi: 10.1016/j.engappai.2023.106902.
[47] W. Zhao, Y. Wang, X. Xiong, and Y. Li, "Cfm-rfm: A cascading failure model for inter-domain routing systems with the recovery feedback mechanism," *Inf.*, vol. 12, no. 6, 2021, doi: 10.3390/info12060247.
[48] Y. Peerthum and M. Stamp, "An empirical analysis of the shift and scale parameters in BatchNorm," *Inf. Sci. (Ny).*, vol. 637, no. February, p. 118951, 2023, doi: 10.1016/j.ins.2023.118951.
[49] S. Lange, K. Helfrich, and Q. Ye, "Batch Normalization Preconditioning for Neural Network Training," *J. Mach. Learn. Res.*, vol. 23, pp. 1–41, 2022.
[50] Y. Yao *et al.*, "Jo-SRC: A Contrastive Approach for Combating Noisy Labels," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 5188–5197, 2021, doi: 10.1109/CVPR46437.2021.00515.
[51] Z. Zhuang, M. Liu, A. Cutkosky, and F. Orabona, "Understanding AdamW through Proximal Methods and Scale-Freeness," *Trans. Mach. Learn. Res.*, vol. 2022-Augus, no. 2019, 2022.
[52] P. Zhou, X. Xie, Z. Lin, and S. Yan, "Towards Understanding Convergence and Generalization of AdamW," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 9, pp. 6486–6493, 2024, doi: 10.1109/TPAMI.2024.3382294.
[53] Z. Shen, Z. Liu, J. Qin, M. Savvides, and K. T. Cheng, "Partial Is Better Than All: Revisiting Fine-tuning Strategy for Few-shot Learning," *35th AAAI Conf. Artif. Intell. AAAI 2021*, vol. 11A, pp. 9594–9602, 2021, doi: 10.1609/aaai.v35i11.17155.
[54] J. Sun, C. Tang, W. Xie, and X. Zhou, "Nonpa ra metric re c eiv e r ope rating cha racte ris tic curve analysis with an imperfect gold standard," vol. 80, no. 3, 2024.
[55] L. Ferrer, "Analysis and Comparison of Classification Metrics," pp. 1–36, 2023, [Online]. Available: http://arxiv.org/abs/2209.05355
[56] Jude Chukwura Obi, "A comparative study of several classification metrics and their performances on data," *World J. Adv. Eng. Technol. Sci.*, vol. 8, no. 1, pp. 308–314, 2023, doi: 10.30574/wjaets.2023.8.1.0054.
[57] J. Opitz, "A Closer Look at Classification Evaluation Metrics and a Critical Reflection of Common Evaluation Practice," *Trans. Assoc. Comput. Linguist.*, vol. 12, no. 2018, pp. 820–836, 2024, doi: 10.1162/tacl_a_00675.
[58] Y. Benny, T. Galanti, S. Benaim, and L. Wolf, "Evaluation Metrics for Conditional Image Generation," *Int. J. Comput. Vis.*, 2021, doi: 10.1007/s11263-020-01424-w.
[59] W. Zhao, W. Chen, L. Fan, Y. Shang, Y. Wang, and W. Situ, "MAN-GAN : a mask-adaptive normalization based generative adversarial networks for liver multi- phase CT image generation," pp. 1–11, 2025.
[60] T. T. Cai, "Theoretical Foundations of t-SNE for Visualizing High-Dimensional Clustered Data," vol. 23, pp. 1–54, 2022.
[61] C. G. Cess and L. Haghverdi, "Gene expression Compound-SNE : comparative alignment of t-SNEs for multiple single-cell omics data visualization," vol. 40, no. February, 2024.
[62] L. Gonzalez-abril, C. Angulo, and J. A. Ortega, "Statistical Validation of Synthetic Data for Lung Cancer Patients Generated by Using Generative Adversarial Networks," pp. 1–15, 2022.
[63] D. N. S. Radhika, M. P. Shyamasunder, and N. B. Manohara, "A review of deep learning and Generative Adversarial Networks applications in medical image analysis," *Multimed. Syst.*, vol. 30, no. 3, pp. 1–25, 2024, doi: 10.1007/s00530-024-01349-1.
[64] M. M. Abdulqader and A. M. Abdulazeez, "A Comparative Study of Generative Adversarial

Vol. 7, No. 2, April 2026, Page. 1549-1574
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2026.7.2.5548

Networks in Medical Image Processing," pp. 1–23, 2025.

[65] J. J. Jeong, A. Tariq, T. Adejumo, H. Trivedi, J. W. Gichoya, and I. Banerjee, "Systematic Review of Generative Adversarial Networks (GANs) for Medical Image Classification and Segmentation," *J. Digit. Imaging*, pp. 137–152, 2022, doi: 10.1007/s10278-021-00556-w.