I.J. Intelligent Systems and Applications, **2024, 5, 39-52**
Published Online on October 8, 2024 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2024.05.04

Deep Learning for Robust Facial Expression Recognition: A Resilient Defense Against Adversarial Attacks

Tinuk Agustin*
Informatics, STMIK AMIKOM Surakarta, Sukoharjo, 57163, Indonesia
E-mail: Agustin.amikom@gmail.com
ORCID iD: https://orcid.org/0000-0002-4165-8146
*Corresponding Author

Moch. Hari Purwidiantoro
Informatics, STMIK AMIKOM Surakarta, Sukoharjo, 57163, Indonesia
E-mail: hariamikom@gmail.com
ORCID iD: https://orcid.org/0009-0006-0776-6756

Mochammad Luthfi Rahmadi
Informatics, Universitas Siber Muhammadiyah, Yogyakarta, 55253, Indonesia
E-mail: datasaintis@gmail.com
ORCID iD: https://orcid.org/0009-0007-5044-2770

Received: 10 February 2024; Revised: 27 June 2024; Accepted: 10 September 2024; Published: 08 October 2024

Abstract: Adversarial attacks can be extremely dangerous, particularly in scenarios where the precision of facial expression identification is of utmost importance. Hiring adversarial training methods proves effective in mitigating these threats. Although effective, this technique requires large computing resources. This study aims to strengthen deep learning model resilience against adversarial attacks while optimizing performance and resource efficiency. Our proposed method uses adversarial training techniques to create adversarial examples, which are permanently stored as a separate dataset. This strategy helps the model learn and enhances its resilience to adversarial attacks. This study also evaluates models by subjecting them to adversarial attacks, such as the One Pixel Attack and the Fast Gradient Sign Method, to identify any potential vulnerabilities. Moreover, we use two different model architectures to see how well they are protected against adversarial attacks. It compared their performances to determine the best model for making systems more resistant while still maintaining good performance. The findings show that the combination of the proposed adversarial training technique and an efficient model architecture outcome in increased resistance to adversarial attacks. This also improves the reliability of the model and saves more resources for computation. This is evidenced by the high accuracy results achieved at 98.81% accuracy on the CK+ datasets. The adversarial training technique proposed in this study offers an efficient alternative to overcome the limitations of computational resources. This fortifies the model against adversarial attacks, resulting in significant increases in model resilience without loss of performance.

Index Terms: Adversarial Training, Convolutional Neural Network, Adversarial Example, Model Robustness.

1. Introduction

Convolutional Neural Networks (CNN) are a component of Deep Learning (DL) and have made substantial progress in improving information processing and data analysis capabilities [1]. This development is a significant assortment of disciplines, including Natural Language Processing (NLP) [2], Sound Processing [3], Image Sentiment Analysis [4], Facial Recognition (FR) [5], Security [6], and Medical Image Analysis [7, 8]. In Facial Expression Recognition (FER), CNNs are magnificent for the precise identification of emotions [9], such as in automated customer service or feedback analysis [10]. CNN allows the system to identify and recognize complex patterns in human faces [11].

CNN can also understand the unique characteristics of each expression, such as eye folds, lip movements, and changes in expression on certain parts of the face. With precise feature extraction, the model improves the precision of

emotion recognition [12]. However, the success of DL faces serious challenges, namely adversarial attacks (Adv.Attack). Attackers use Adv.Attack to mislead DL models [13]. Small perturbations intentionally introduced to the input are often invisible to humans. These attacks can manipulate the model to provide inaccurate or undesirable predictions [14]. These perturbations are significant enough to result in an incorrect prediction by the model. Ultimately, this can lead to a loss of generalizability of new data [15]. Previous research has suggested how adversarial attacks can make the system unsafe [16]. Even though they are a threat, Adv.Attack can be a valuable security testing tool. Researchers can use Adv.Attack. Researchers can systematically explore model weaknesses to identify and address potential vulnerabilities while enhancing the robustness and security of DL [17]. By understanding how Adv.Attack works and responding to them effectively, researchers can create DL models that are more resilient to potential threats. The proposed method improves both models' safety and resilience encounters of adversarial manipulations [18]. This encourages technological development in a safer and more trustworthy direction [19].

Researchers categorize a defensive approach to attacks as the development of methods that strengthen DL. One commonly used technique is adversarial training (Adv.Train) [20]. Researchers use this technique to train a model using data indicated by Adv.Attack [21, 22]. This method detects unusual or anomalous activities that indicate an attack [23]. Researchers train the model in adversarial examples (Adv.Example) to make it more sensitive to unusual patterns that may suggest an attack. However, this technique faces significant challenges, particularly the high computing resources required [24].

The focus of this research is to increase the robustness of DL CNN in the domain of FER. Besides that, it also guarantees that models can resist potential assaults while simultaneously comprehending and responding to human emotions with precision. In addition, this research aims to address excessive computing resource consumption. Therefore, this study proposes a resource-efficient Adv.Train method. This method entails creating and storing Adv.Example as separate entities to improve the model’s resistance to Adv.Attack.

The following are the contributions made to this research:

- We will generate a new FER dataset. In this phase, we will implement the Adv.Train technique by employing the One Pixel Attack (OPA) and the Fast Gradient Sign Method (FGSM). This new dataset enhances the security and the ability of models to withstand advanced attacks.
- Uniqueness in the FER domain. This research contributes to the adversarial defense model in the FER domain. This research addresses an unexplored knowledge gap in FER.
- Alternative Solutions with Low Computational Costs. Finding alternative solutions for Adv.Train that minimize the use of computing resources is a significant innovation. This could open the door to the development of robust models that are economically viable and widely applicable.

2. Literature Review

Adv.Attack are a serious challenge that can threaten the reliability and security of algorithms in various sectors [25]. Adv.Attack can undermine the accuracy and trustworthiness of the algorithm. Apart from that, Adv.Attack also poses risks to user privacy and ethics. To overcome this challenge, various studies have been conducted to develop algorithms that are more resistant to Adv.Attack [26,27].

Research on Adv.Attack in FER is limited. One study relevant to this topic is [28], which discusses Adv.Attack in FER. This research proposes the Deep Learning Geometry-Aware Adversarial Vulnerability Estimation (GAAVE) method. The GAAVE method effectively addressed class imbalances and enhanced the algorithm's resilience against Adv.Attack. This was achieved using adversarial techniques to identify noisy labels, as well as the implementation of dataset splitting, subset refactoring, and self-annotator modules. Another study, [29], analyzed various factors that influence the security and robustness of FER algorithms, such as network architecture, data quality and quantity, and evaluation metrics. This research shows that although the FER algorithm is vulnerable to Adv.Attack, an adversarial defense approach can reduce the negative impact. This research also proposes techniques and strategies, such as data augmentation, regularization, and ensemble learning, to protect the algorithm from Adv.Attack. The relevance of our research is to increase the algorithm's resilience to Adv.Attack in FER.

A study [30] indicates that, while Adv.Attack can greatly reduce the accuracy of face recognition, defense methods like Adv.Train, feature squeezing, and spectral defense can enhance model robustness. Researchers [31] used Adv.Train techniques too, demonstrating that Adv.Train is an effective method for developing models resistant to Adv.Attack. This research employs Adv.Example as a method for data augmentation. Both studies agree that Adv.Train can significantly increase model resilience against various types of Adv.Attack, although it introduces a complexity that needs to be considered. In addition, [32] illustrates how optimizing the trajectory for re-weighting can enhance Adv.Train. Findings from this investigation inspired the usage of Adv.Train in creating techniques for handling Adv.Attack.

Lastly, certain studies aim to minimize the training cost while preserving the model's resilience against Adv.Attack, particularly in situations where computational abilities are restricted. For example, research [33] proposed a new algorithm for Adv.Train that reduces the computational cost by reusing the computed gradient information to update the model parameters. However, this method is sensitive to the hyperparameters. Again, research [34] attempted to reduce the training cost while maintaining the model's robustness against Adv.Attack. This method diminishes the quantity of

iterations needed to generate Adv.Example and employs faster optimization techniques. However, fewer iterations may lower the quality of the generated Adv.Example and decrease the model's capacity to generalize to assaults that have not been observed before. This research inspires our work on developing resource-efficient Adv.Train methods.

Furthermore, the last studies indicate that the selection of a model for an attack can significantly impact its resilience and performance. Research [35, 36] highlights that lightweight and efficient models can significantly improve resistance to adversarial attacks. Meanwhile, research [37] emphasizes that ensemble models offer a powerful and complex approach to improving resilience against adversarial attacks. This research serves as the foundation for our efforts to test two distinct model architectures.

3. Methodology

This chapter outlines the research procedure applied in the exploration and development of the model. This type of research is experimental and involves developing several test scenarios. The steps and approaches taken will be explained in depth. This is to ensure that the results are in line with the stated research objectives.

3.1. Dataset

The CK+ dataset [38] labels seven categories of facial expressions. The part of the data used is 54 for contempt, 75 for fear, 84 for sadness, 135 for anger, 177 for disgust, 207 for happiness, and 249 for surprise. This dataset contains 593 frames taken from 123 subjects of diverse ages and genders. The piece frame displays a transformation from a neutral expression to a labeled peak expression. This dataset was very often found in computer vision tasks. The dataset was partitioned into data train and data test, with a proportion of 80:20 of the whole dataset. Using sci-kit-learn to divide the training and testing data subsets was done randomly. This dataset was chosen for its wide acceptance in FER research.

3.2. Research Scenario

This research involved four scenarios designed to test and improve the model’s resilience to Adv.Attack. The experimental process was performed according to the following scenario:

- First Scenario: We selected MobileNetV2 and an Ensemble Deep Learning (EDL) model. Both models were trained on the CK+ dataset. After training, we attacked both models using FGSM and OPA techniques to compare their robustness, accuracy, and computation time.
- Second Scenario: We generated Adv.Example from the original CK+ dataset using FGSM (epsilon = 0.01) and OPA (one-pixel random change). These Adv.Example were combined with the original dataset and stored as separate entities.
- Third Scenario: We performed Adv.Train on the models using the Adv.Example generated in the second scenario to enhance the models' resistance to Adv.Attack.
- Fourth Scenario: We retested the models after Adv.Train using the same FGSM and OPA attacks to evaluate their improved robustness.

3.3. Research Steps

Pre-processing is essential for the development of models by improving data quality and preparing it for effective machine learning. This study implemented several essential pre-processing steps. Initially, we resized the data to a uniform size of 75x75 pixels with RGB channels, ensuring dimensional consistency across the dataset. Following resizing, promoting stability, and facilitating convergence during model training. To scale data values between 0 and 1, we implemented normalization. Additionally, the data type was converted to float32 to ensure accurate representation in subsequent mathematical operations. In addition, data augmentation techniques are applied to enrich dataset diversity and the model’s capacity to identify intricate patterns. Use a categorical cross-entropy loss function with a batch size of 32. We then optimized the model over 50 epochs. To evaluate the model's performance, we implemented evaluation metrics, including accuracy, recall, precision, and F1 score. Visual assessment using epoch loss graphs facilitated the monitoring of model convergence and performance trends throughout training. The study also evaluated the efficacy of Adv.Train to fortify the model’s resilience to Adv.Attack. Fig. 1 illustrates the detailed stages of the research process.

3.4. Model

We chose models for this test because each represents different characteristics. We chose EDL to represent complex architectures, leveraging the combined power of multiple models to increase resilience against Adv.Attack. Meanwhile, we selected MobileNetV2 to represent an efficient architecture due to its ability to handle Adv.Attack efficiently, thanks to its simple structure and controlled parameters.

A. Finetuning MobileNetV2

MobileNetV2 was fine-tuned by adapting its final layers for the FER task, utilizing transfer learning from pre-trained weights on Image Net. The model was configured using a learning rate of 0.01, a batch size of 32, trained over 50 epochs, and optimized using the Adam optimizer. MobileNetV2 was chosen due to its low complexity, efficient

inference capabilities, and demonstrated performance even with constrained computational resources. Its advantages include fewer parameters and a lighter structure, facilitating fast and efficient execution, which is ideal for scenarios with constrained computing capabilities [39]. The details of the fine-tuning MobileNetV2 structure are presented in Fig. 2. This modified MobileNetV2 CNN architecture has the following key components:

- Input: Receive an image of size 75 x 75 x 3 pixels.
- Convolution (Conv): In the first layer, it uses a 3 x 3 filter with step 1 and 32 output channels.
- Depth-wise Separable Convolution (DSC). Uses DSC for efficiency, consisting of DSC and point-wise convolution.
- Bottleneck DSC: Used for dimensionality reduction and restoration.
- Global Average Pooling (GAP) takes a global average of cross-feature channels, resulting in one value per feature channel for the entire image. This helps decrease the number of parameters and prevents overfitting.
- Fully Connected (FC): This layer uses 7 units to generate a classification score. The ReLU activation function can be applied after this layer, helping to introduce non-linearity and increase the rigidity of the model.
- Dropout: To reduce overfitting.
- SoftMax converts the scores into image class probabilities.

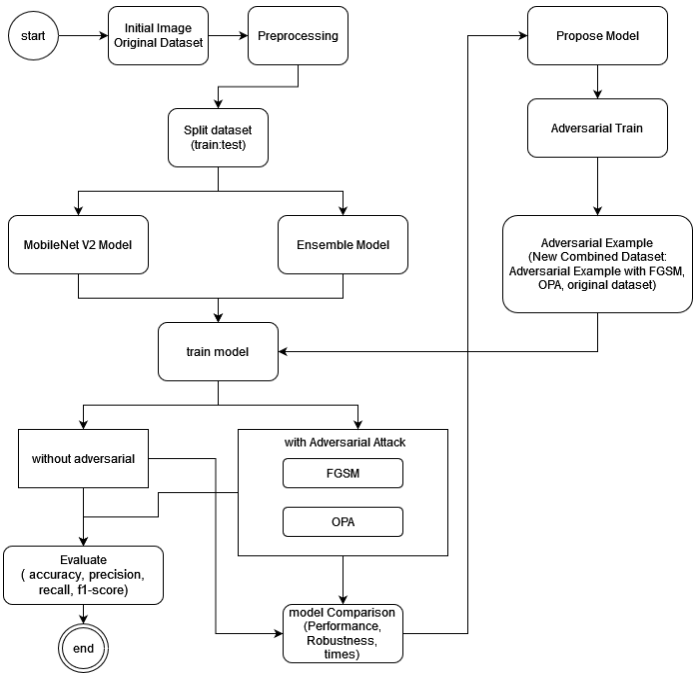

Fig.1. Research flow

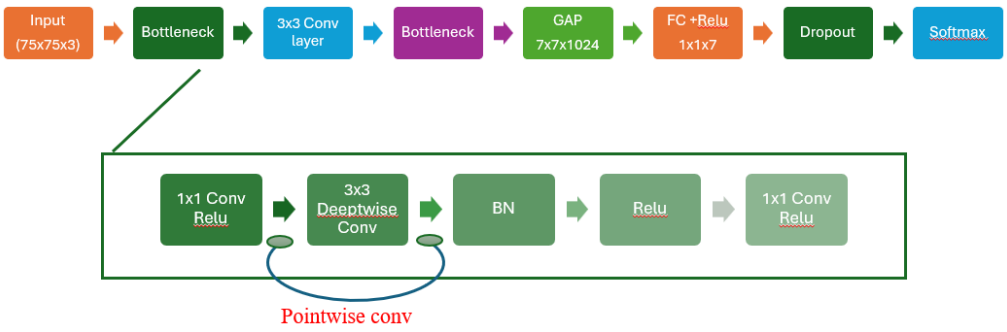

Fig.2. Architecture of The MobileNetV2 Fine-tuning model

B. Ensemble Deep Learning (EDL)

This complex model combines the architectures of Dense Net, MobileNetV2, and VGG19, which collectively have demonstrated excellent accuracy in prior studies [40]. The EDL model integrates these diverse architectures using the concatenate technique to make final predictions. This approach capitalizes on the diversity of representations offered by each model, thereby enhancing robustness to data variations and improving classification accuracy. By leveraging the strengths and weaknesses of each component architecture, EDL effectively identifies intricate features within datasets,

facilitating more generalized decision-making and achieving superior performance in classification tasks. For a detailed structural overview, please refer to Fig. 3.

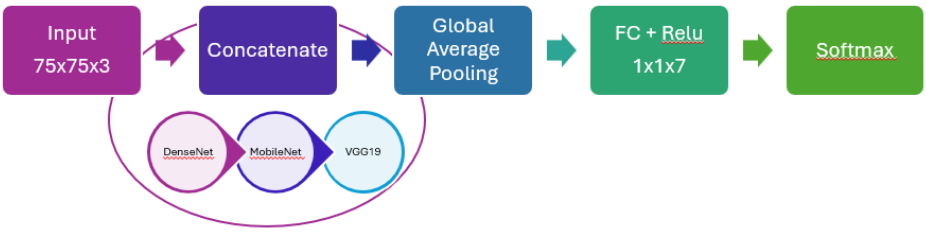

Fig.3. Architecture EDL

3.5. Fast Gradient Sign Method (FGSM)

FGSM can generate an adversarial image by determining the gradient of the model's loss function. Implements the input data and subsequently introduces perturbations that are proportional to the gradient's sign. This perturbation can increase the model loss value and decrease the prediction accuracy. This method was first proposed by [41]. This method effectively makes the model make incorrect predictions by adding noise to the input data. However, models trained with specialized techniques to overcome Adv.Attack, such as FGSM, can become more resilient against such attacks.

FGSM involves several key steps in performing Adv.Attack. First, calculate the gradient ($\nabla_x J$ (x, y)) of the cost function concerning the input data (x) using the model under attack. Next, determine the direction of sign(.) change that will direct the model prediction to the class desired by the attacker based on the gradient. Finally, include disturbance in the input data by the gradient's order, with a magnitude controlled by the epsilon parameter (ϵ). The following is the mathematical formula for the FGSM function use (1) [42].

$$x_{adv} = x + \epsilon \cdot sign\left(\nabla_x J\ (x, y)\right) \tag{1}$$

With these steps, the FGSM efficiently creates input data that appears like the original data but can confuse the model and lead to undesirable predictions.

Epsilon (ϵ) is a way to measure how much noise is applied to the data when involving defense against adversarial attack. The larger the epsilon value, the greater the perturbation to the data. For example, an epsilon of 0.1 will cause more perturbation than an epsilon of 0.01. A large epsilon value may provide a stronger defense, but it may also make the perturbation more visible or change the visual appearance of the image [43]. FGSM was employed both for testing the model's robustness and for Adv.Train purposes. The parameters used to generate Adv.Example in this process included an epsilon value of 0.01.

3.6. One Pixel Attack (OPA)

OPA is a form of adversarial attack that modifies only one pixel in the image but can trick the model and result in erroneous predictions. It uses a random search algorithm to identify pixels that have the greatest effect on the model's predicted class. Introduced by [44], this attack shows that even small pixel changes can have a significant impact on the model's prediction result. The OPA process involves selecting one or more pixels in the image to be modified. This pixel selection can be done by various methods, including using optimization algorithms to determine the optimal location. The optimized pixel value is usually limited by the range of values that the pixel can take (e.g., 0 to 255 for RGB images). Once the optimal location is determined, the pixel values in the image are changed according to the predetermined values to create an attack that can fool the model.

3.7. Adversarial Example

An Adv.Example changes the input data by introducing small disturbances that can affect the model's decisions. Although small and difficult to spot, input data changes can have a big impact on the model. The purpose of creating an Adv.Example was to investigate the model's weaknesses and determine how it could make incorrect or unexpected predictions [38]. To create the Adv.Example dataset, the researcher used two adversary attack methods: the FGSM attack and the OPA attack. The FGSM attack uses an epsilon value of 0.01 to create an adversarial example. This OPA attack uses a pixel value of 1. Next, we permanently store the adversarial data produced by these two techniques as separate entities. After completing these steps, the Adv.Train process will use the resulting dataset to train the model.

3.8. Adversarial Training

The objective of Adv.Train development is to enhance the reliability and resilience of DL models in the face of Adv.Attack. During the Adv.Train process, the model intentionally incorporates Adv.Example. It increases the resistance model to manipulations that may appear on the input data. During the adversarial training phase, the model is modified by utilizing a dataset that includes the original dataset and an Adversarial example dataset, where the results are then

incorporated into the training dataset. This technique allows the model to explore and understand the modified samples. Thus, increasing the model's robustness. Following the training stage, the model was tested again with a test dataset that may include Adv.Attack.

However, Adv.Train involves an iterative process that requires a long computation time. In each iteration, the model must generate Adv.Example and learn to classify them correctly. This process needs to be repeated many times, especially for large datasets and complex models. During Adv.Train, the model must also store Adv.Example for each sample in the dataset, causing a significant memory load. This is compounded by storing the biases and weights of each neuron network, which consume a lot of memory [45]. Adv.Train primary benefit is the enhanced model's resilience against Adv.Attack [46]. In addition, models that have been trained with the adversarial method can provide stable predictions even in the face of data variations that may arise due to Adv.Attack.

3.9. Evaluation Techniques

The application of data analysis techniques can help identify and improve the weaknesses of the DL model in detecting Adv.Attack. Techniques that can be used include the Confusion Matrix (CM). This matrix presents a comprehensive picture of the model's performance in classifying data. It can help identify specific weaknesses in the model. Such as in recognizing certain classes or in responding to Adv.Attack. Using CM, we can evaluate precision, accuracy, F1-score, and recall. This information allows us to understand the extent to which the model can be trusted to classify the data [47]. The model performance evaluation technique used is the epoch graph. This graph shows the change in loss value and model accuracy during the training process using the gradient descent method [48]. This method is an optimization algorithm used to determine the most suitable parameters to reduce the size of a function.

4. Results and Discussion

4.1. The Result of First Scenario: Model Testing Results Against Attacks

In the initial test, we applied the FGSM and OPA methods to MobileNetV2 and EDL to measure their resistance to attacks. The FGSM attack employs an epsilon value of 0.01%, whereas the OPA attack inserts only one intrusive pixel into the image. The test results show that the MobileNetV2 and EDL models have different levels of resistance to FGSM and OPA attacks. In the absence of attacks, the test results on fine-tuning MobileNetV2 showed excellent prediction performance, with a confidence level of 100%. However, when tested with FGSM attacks, there was a significant change in predictions, indicating vulnerability to Adv.Attack and a reduction in confidence in the original class. In contrast, the OPA attack did not change the predictions of the MobileNetV2 model, indicating a higher level of resilience to the OPA attack. Fig. 4 presents the test results.

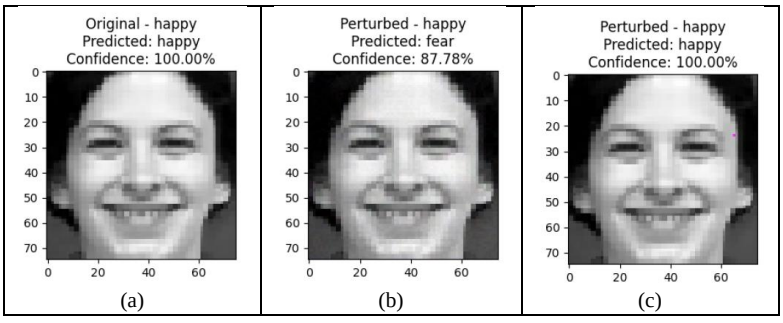

Fig.4. Initial performance without attack (A), model performance against FGSM attack (B), model performance against OPA attack (C)

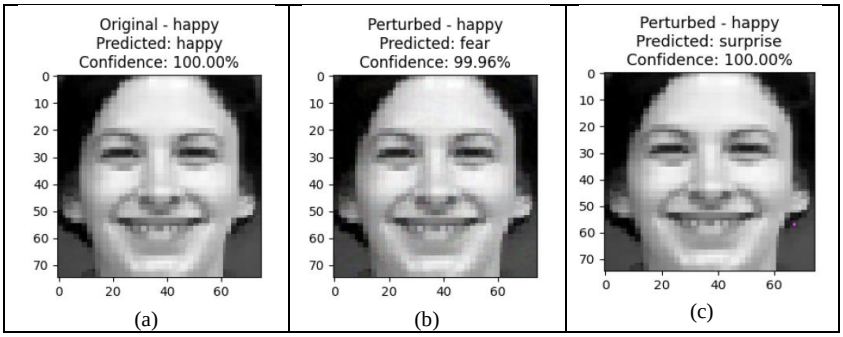

Fig.5. Initial Performance without attack (A), model performance against FGSM attack (B), model performance against OPA attack (C)

We conclude that the EDL model exhibits excellent prediction performance in the absence of attacks, demonstrating a high level of confidence. However, when tested with FGSM and OPA attacks, the model experienced significant changes in prediction. Despite the predictions' inaccuracies, the level of confidence remains high. This indicates the model's inability to predict correctly, which has a more significant impact on the model's performance against attacks. This test's results are illustrated in Fig. 5.

The analysis confirmed the difference in security levels between MobileNetV2 and EDL against Adv.Attack. MobileNetV2 shows better resilience against OPA attacks, but not against FGSM attacks. On the other hand, the EDL model showed no resilience against either type of attack. These findings highlight the importance of considering specific types of assaults when assessing the effectiveness and safety of models, as well as customizing the model choice according to the usage scenario.

Table 2 presents a comparison of the models' performance, parameters, and training time. With this information, we can conclude that the MobileNetV2 fine-tuning model surpasses the EDL model in efficiency and performance. These considerations led to the selection of MobileNetV2 as the base model for subsequent tests, given its superior performance and faster training time compared to the EDL model. In addition, this model also showed better resilience to Adv.Attack in the initial trials.

Table 1. Model comparison

Model	The Number of Model Parameters	Time	Accuracy
MobileNetV2 fine-tuning	2,422,855	264.4638018608093	96.95%
EDL	32,920,391	517.3725578784943	88.32%

This finding has several important implications. First, it shows that certain models are resistant to assault. Therefore, it is important to choose the right model based on the type of attack expected. Secondly, this result shows that Adv.Attack has a significant effect on the performance models, even if it has been well-trained. This highlights the importance of considering model security when designing and training AI models.

4.2. The Result of Second Scenario: Adversarial Example Dataset Result

Adv.Train effectively improves system reliability. The main challenge, however, is the long training time and high computational load required to train the model with adversarial attack data. These constraints limit researchers' ability to conduct experiments and cause computer performance issues. We can apply an approach that stores the original dataset and the adversarial example dataset as separate entities to overcome these constraints. Eliminating constantly generated Adv.Example during training can improve computational efficiency. This approach optimizes the utilization of computational resources. It also provides the flexibility to use the two datasets independently or together according to experimental needs.

Fig.6 shows the original dataset and the new adversarial example dataset. The adversarial example generated by FGSM shows tiny but significant differences in pixel values. The loss function gradient causes perturbations in the input, potentially tricking the model. Equation (1) calculates the sign of the gradient and adds a small perturbation to the input. In the adversarial example of FGSM generation, we use the epsilon parameter of 0.01 to control the perturbation's intensity. This smallest epsilon value results in changes that have an almost invisible appearance of the human, however sufficient to make the model make incorrect predictions.

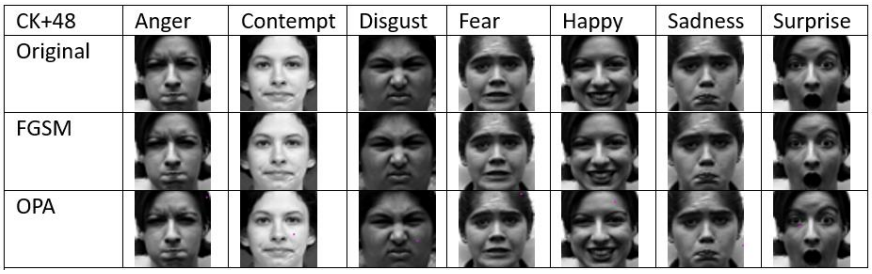

Fig.6 New dataset, original dataset and adversarial dataset

OPA and FGSM both aim to create Adv.Example that can mislead the model. Although the goal is the same, the approaches they adopt are very different. OPA performs direct changes to pixel values in the pixel space. To maximize the loss function, it uses a random search to determine which pixels to change and what values to assign to them. This research bases the OPA algorithm on a random search. This method is simple to implement and does not require a deep understanding of optimization or gradients. OPA produces almost invisible changes in the image, as it only changes one or a few pixels. In this adversarial OPA example, we can still see the differences in the pixels added to the data, especially if the changed pixels have a high contrast with the surrounding pixels.

4.3. The Result of The Third Scenario: Building an Attack-resilient Model

In the third scenario, we performed Adv.Train on the MobileNetV2 model using a combination of the original dataset and Adv.Example generated from the second scenario. This Adv.Train process strengthened the resulting model, which we stored in the module.h5 file. Adv.Train is a crucial strategy in building an attack-resistant model. By incorporating Adv.Example into the training process, the model learns to recognize and respond to attacks more effectively. During training, the model updates itself by incorporating errors discovered during adversarial example processing. Each iteration of training includes both original and newly generated adversarial data, which allows the model to continuously improve its ability to recognize and reject Adv.Attack. This iterative process exposes the model to modifications, enhancing its robustness against attacks by allowing it to explore and understand adversarial samples.

We used MobileNetV2 in this Adv.Train process, resulting in a robust model that can handle Adv.Attack more effectively. We then tested this strengthened model on datasets containing both Adv.Attack and unseen data to assess its resilience. The Adv.Train helped reduce the model's sensitivity to noise and disturbances, allowing it to focus on relevant information and ignore disruptions.

4.4. The Result of Fourth Scenario: Testing the Reinforced Model

The fourth test scenario involved evaluating the strengthened MobileNetV2 model against Adv.Attack. The test results demonstrated how effectively the model can respond to these attacks, showcasing its enhanced robustness and reliability. As seen in Fig. 7, the model provides a very confident prediction for the Happy class on the data without attack, indicating its ability to recognize and classify original data with high confidence. Even when subjected to an FGSM attack with an epsilon of 0.03, the model maintains accurate predictions with only a slight reduction in confidence. This shows that low-intensity attacks can slightly affect the model's stability without completely disrupting the classification. However, increasing the FGSM attack to an epsilon of 0.04 causes a significant shift in the model's predictions, erroneously classifying the data as the Anger class with high confidence. This indicates that higher epsilon values in FGSM attacks can successfully manipulate the model into making incorrect predictions. Interestingly, despite this sensitivity to FGSM attacks, the model remains robust to multi-pixel attacks, producing correct predictions with complete confidence.

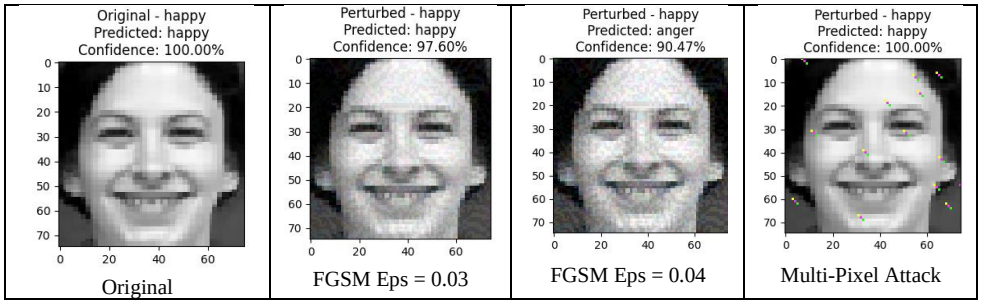

Fig.7 Model performance against attacks

To further validate our experiment, we tested the robust model with images from unseen datasets. Using a completely new dataset for testing is a robust approach to evaluating the model's generalizability and resilience. The results, as shown in Fig. 8, indicate that the model remains highly robust against attacks, even with very high epsilon values, without altering its predictions.

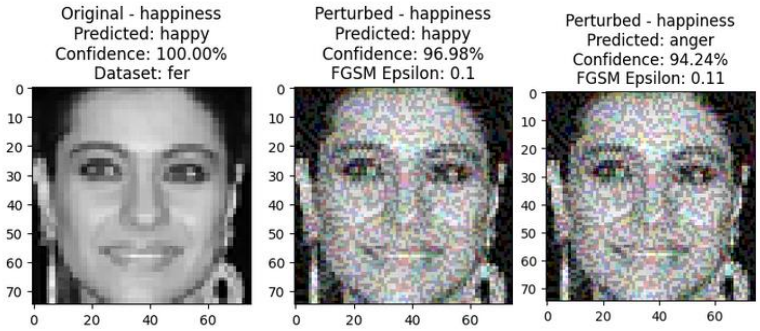

Fig.8. Test on the other dataset

The model's resilience to high epsilon attacks indicates that significant perturbations do not easily affect it. We can attribute this robustness to the combination of extensive training with Adv.Example and the robust architecture of MobileNetV2, which effectively manages disturbances. These findings demonstrate that the Adv.Train process enables the model to learn and adapt to various attacks, thus maintaining high robustness against intentional data manipulation.

4.5. Analysis of Model Sensitivity to FGSM and Multi-pixel Attacks

The observed phenomenon highlights the different natures of different adversary attacks and the importance of model architecture and training in determining resilience. The nature of FGSM attacks exploits information regarding the model's gradient information to create Adv.Example by making small but strategically significant perturbations in the input data. The epsilon value determines the magnitude of these perturbations. As the epsilon value increases, the perturbations become more pronounced, causing the input data to deviate further from its original distribution. This higher deviation can manipulate the model's decision boundaries, leading to incorrect predictions. Research indicates that models such as MobileNetV2 are susceptible to perturbations in their input space aligned with the gradient direction, as these perturbations directly exploit the model's learned features. When epsilon is low, the perturbations are minor, and the model can still rely on its robust features to make correct predictions. However, with higher epsilon values, the perturbations significantly alter the input, causing the model to misclassify the data.

In contrast, multi-pixel attacks involve altering multiple pixels in the input image. These alterations are often spread across the image and do not necessarily align with the gradient direction. While multi-pixel attacks can be disruptive, they may not be as effective as FGSM attacks in targeting the specific features the model relies on for classification. This is because the perturbations are more dispersed and less aligned with the model's decision boundaries. Consequently, the model's robust features can still recognize the underlying patterns in the input data, allowing it to maintain accurate predictions even in the presence of multi-pixel perturbations. This underscores the need for comprehensive Adv.Train and the development of robust models capable of withstanding various types of attacks.

4.6. Evaluation of MobileNetV2 Fine-tuning Performance

The interpretation of the MobileNetV2 fine-tuning model is evaluated by employing the epoch graph depicted in Fig. 9. The graph demonstrates the model's proficiency in recognizing facial expressions. The model's superior performance is expressed in several key aspects. Firstly, the stability of the loss value on the training data throughout the epochs indicates that the model is neither underfitting nor overfitting. Underfitting, a condition where the model fails to learn the training data effectively, resulting in low accuracy, is not observed. Overfitting occurs when the model acquires knowledge at an extreme rate from the data train. It leads to subpar performance when predicting test data. It is also not evident in this model. In addition, the loss value of the model on the test data consistently decreases with the progression of epochs. The observation of this trend confirms the model's reliability in consistently predicting new data that was not previously observed. In conclusion, the model exhibits excellent performance in recognizing and classifying facial expressions.

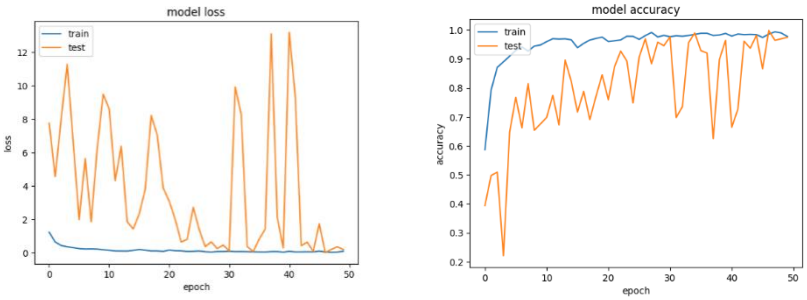

Fig.9. Epoch graph. (left) loss of graph. (right) accuracy graph

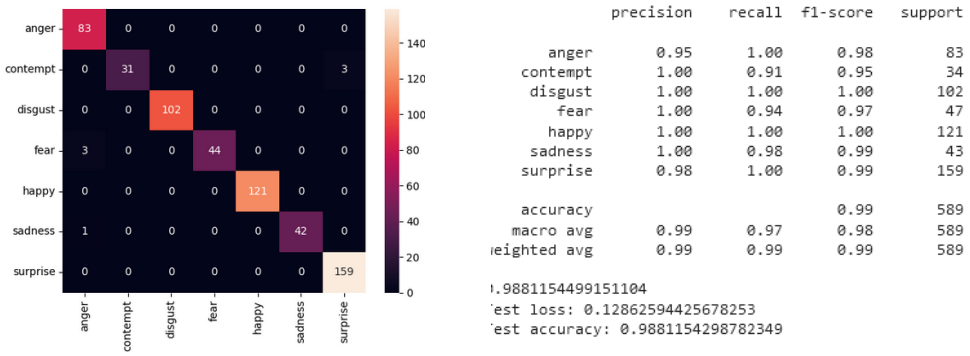

Fig.10. Confusion matrix (left) summary of confusion matrix (right)

The confusion matrix represents actual and predicted values. Fig. 10(a) depicts this matrix. From this matrix, the model is capable of relatively accurately classifying emotions, with most data lying on the diagonal line, representing correct predictions. However, the model exhibits difficulty in distinguishing certain emotions, such as fear, disgust, happy,

and anger. This could be due to the similarities in facial expressions between these emotions, the influence of noise, or labeling errors. Differences in the amount of data between emotion categories, especially in the fear, contempt, and sadness datasets, may also cause the model to struggle to classify emotions correctly. The limitations of this study do not include this issue, which requires further research.

The MobileNetV2 fine-tuning model with Adv.Train has significantly improved its facial expression, classification performance, and resilience to Adv.Attack. Fig. 10(b) presents the confusion matrix results, achieving an overall accuracy of 98.81%. Compared to the initial accuracy before Adv.Train of 96.95%, there is a significant improvement. This shows that Adv.Train is sufficient for increasing the model's resilience to attacks that manipulate inputs to influence classification results. It also shows that the model becomes more resilient without losing performance. The high precision of 99% indicates that the model is very effective at predicting the correct class. The model can reliably recognize specific threats or circumstances with little error. The high recall of 99% shows the model's capability of correctly identifying the most positive samples. So, the model can detect facial expressions thoroughly, including in situations where numerous variations in expression or lighting conditions might affect facial appearance. The high f1-score of 99% suggests the model has a satisfactory balance between precision and recall. It provides a more comprehensive explanation of the model's ability to classify various emotional expressions. We derive these values by calculating the weighted average of each class's proportion in the dataset. This provides a more accurate representation of the model's effectiveness where the classes exhibit an uneven distribution.

4.7. Results of Testing Computing Resources: Adversarial Training Processes

We conduct a comparison between our proposed Adv.Train technique and general Adv.Train methods, applying the FGSM adversarial attack. We emphasize that both processes involve calculating the gradient of the loss over the input to create an adversarial example. The main difference lies in the process of creating Adv.Example. In our proposed method, we create an adversarial dataset using an FGSM attack and store it permanently. Using this dataset and the original data, we train the model. In the meantime, the model training process generates Adv.Example using widely used methods. The Adv.Example generated at each training iteration updates the model weights. Each iteration regenerates this example, not saving it permanently. Our proposed method leverages pre-built adversarial datasets, optimizing the use of computational resources and allowing a better focus on the model training process.

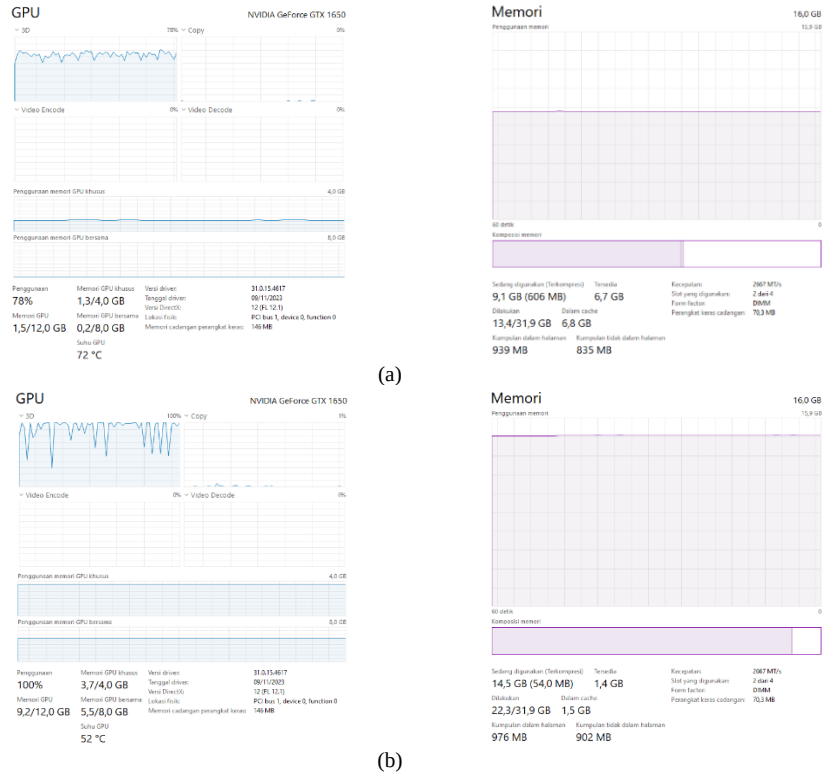

Fig.11. GPU and memory Adv.train resources. (a) Usage of GPU and memory resources of our proposed Adv.train method, (b) Use of GPU and memory resources in general Adv.Train methods

Fig. 11 illustrates the GPU and RAM consumption during Adv.Train. The observations highlight our proposed technique's superior computational efficiency, which is critical for applications with limited computing infrastructure. Fig. 11(b) illustrates how commonly used Adv.Train methods consume significant GPU and RAM resources, reaching 100% GPU usage and nearly full RAM utilization. In contrast, our proposed method proves more efficient resource utilization, as shown in Fig. 11(a). This underscores the effectiveness of our approach, which involves storing and using

adversarial datasets separately to optimize computational usage. By doing so, we reduce the GPU and RAM load required during the Adv.Train process.

4.8. Discussion

Adv.Attack: Our findings are consistent with previous research [27, 46] that shows Adv.Attack can manipulate model predictions with minimal data disruption. However, our research provides an additional contribution by developing a model of FER that shows increased resistance to Adv.Attack.

Adv.Train Technique: Our research verifies the effectiveness of Adv.Train methods in enhancing model resistance to attacks. These results are consistent with prior research that has underscored the significance of this approach in dealing with Adv.Attack. We also found that this method requires significant computational resources for the creation of Adv.Example, as has been noted in previous literature [36].

Resource Optimization: Our research provides empirical evidence that model resistance to Adv.Attack varies significantly depending on the type of attack and architecture used. This variation is consistent with previous findings [45] emphasizing the importance of considering these factors in designing AI systems resistant to attacks.

Our research findings support previous findings [37] showing that lightweight and efficient models can significantly enhance resistance to Adv.Attack. Our adaptation of the MobileNetV2 architecture highlights different resistances to various types of attacks, revealing differences in model characteristics that affect the model's ability to withstand Adv.Attack.

5. Conclusions and Future Works

We have successfully developed a robust and reliable FER model capable of withstanding Adv.Attack. By fine-tuning the effective MobileNetV2 CNN architecture for better performance, the model not only saves computer resources but also indicates improved resistance to Adv.Attack. Evaluation results suggest high accuracy levels, with precision, recall, and f1-score values reaching 97%, 98%, and 97%, respectively, allowing the model to classify facial expressions with exceptional accuracy. The study also makes a big contribution to Adv.Train methods by suggesting a new way to combine and store adversarial datasets (Adv.Example) as separate, permanent entities. This approach proves effective in optimizing the advanced training process, particularly with constrained computational resources. Models trained with the Adv.Example dataset exhibit superior resistance to Adv.Attack compared to those without such examples.

The findings underscore the critical role of security in AI model development. Experimental findings reveal that the resilience to attacks varies across different models, influenced by model characteristics, types of attacks encountered, and training processes applied. The approach of maintaining adversarial datasets as distinct entities presents a practical solution for improving model robustness. In conclusion, this investigation offers deep insights into the vulnerability of AI models to Adv.Attack and underscores the necessity of implementing robust security strategies in artificial intelligence model development.

Future research should focus on:

- Continue to explore the robust model's efficacy in defending against diverse Adv.Attack. By analyzing how different models respond to these various attack scenarios, researchers conceive a deeper comprehension of the advantages and disadvantages of each architecture is inherent. Developing new training techniques or more efficient attack prevention techniques will be critical to optimizing limited resources.
- Developing robust models using a variety of adversarial defense techniques will be critical for increasing model resilience.
- The practical application of this research can apply to various applications, such as security and safety systems, where such attacks are common.

Addressing these areas will advance our understanding of adversarial robustness and facilitate the creation of more resilient AI models across various applications, from security to comprehensive emotional understanding.

Acknowledgment

Thanks to STMIK Amikom Surakarta.

Conflict of Interest

All authors have declared that they have no competing interests.

References

[1] W. Wei et al., "Adversarial Deception in Deep Learning: Analysis and Mitigation," Proc. - 2020 2nd IEEE Int. Conf. Trust. Priv. Secur. Intell. Syst. Appl. TPS-ISA 2020, pp. 236–245, 2020. 10.1109/TPS-ISA50397.2020.00039.

[2] D. W. Otter, J. R. Medina, and J. K. Kalita, "A Survey of the Usages of Deep Learning for Natural Language Processing," IEEE Trans. Neural Networks Learn. Syst., vol. 32, no. 2, pp. 604–624, Feb. 2021. doi: 10.1109/TNNLS.2020.2979670.
[3] Y. Yang and Y. Yue, "English speech sound improvement system based on deep learning from signal processing to semantic recognition," Int. J. Speech Technol., vol. 23, no. 3, pp. 505–515, Sep. 2020. doi: 10.1007/s10772-020-09733-8.
[4] A. Ortis, G. Farinella, and S. Battiato, "An Overview on Image Sentiment Analysis: Methods, Datasets and Current Challenges," in Proceedings of the 16th International Joint Conference on e-Business and Telecommunications, 2019, pp. 290–300. doi: 10.5220/0007909602900300.
[5] W. Wu, Z. Sun, Y. Song, J. Wang, and W. Ouyang, "Transferring Vision-Language Models for Visual Recognition: A Classifier Perspective," Int. J. Comput. Vis., Sep. 2023. doi: 10.1007/s11263-023-01876-w.
[6] Z. Meng, M. Zhang, and H. Wang, "CNN with Pose Segmentation for Suspicious Object Detection in MMW Security Images," Sensors, vol. 20, no. 17, p. 4974, Sep. 2020. doi: 10.3390/s20174974.
[7] M. Puttagunta and S. Ravi, "Medical image analysis based on deep learning approach," Multimed. Tools Appl., vol. 80, no. 16, pp. 24365–24398, Jul. 2021. doi: 10.1007/s11042-021-10707-4.
[8] D. R. Sarvamangala and R. V. Kulkarni, "Convolutional neural networks in medical image understanding: a survey," Evol. Intell., vol. 15, no. 1, pp. 1–22, Mar. 2022. doi: 10.1007/s12065-020-00540-3.
[9] M. Senthil Sivakumar, T. Gurumekala, L. Megalan Leo, and R. Thandaiah Prabu, "Expert System for Smart Virtual Facial Emotion Detection Using Convolutional Neural Network," Wirel. Pers. Commun., vol. 133, no. 4, pp. 2297–2319, Dec. 2023. doi: 10.1007/s11277-024-10867-0.
[10] L. Alzubaidi et al., Review of deep learning: concepts, CNN architectures, challenges, applications, future directions, vol. 8, no. 1. Springer International Publishing, 2021. doi: 10.1186/s40537-021-00444-8.
[11] J. Damilola Akinyemi and O. F. Williams Onifade, "An Individualized Face Pairing Model for AgeInvariant Face Recognition," Int. J. Math. Sci. Comput., vol. 9, no. 1, pp. 1–12, Feb. 2023, doi: 10.5815/ijmsc.2023.01.01.
[12] D. Graupe, "Deep Learning Convolutional Neural Network," Deep Learn. Neural Networks, pp. 41–55, 2016. doi: 10.1142/9789813146464_0005.
[13] C. Szegedy et al., "Intriguing properties of neural networks," Dec. 2013, [Online]. Available: http://arxiv.org/abs/1312.6199. Access, May 5, 2024.
[14] N. Akhtar, A. Mian, N. Kardan, and M. Shah, "Advances in Adversarial Attacks and Defenses in Computer Vision: A Survey," IEEE Access, vol. 9, pp. 155161–155196, 2021.
[15] L. Rice, E. Wong, and J. Z. Kolter, "Overfitting in adversarially robust deep learning," Feb. 2020, [Online]. Available: http://arxiv.org/abs/2002.11569. Access, May 5, 2024.
[16] Y. Wang et al., "Adversarial Attacks and Defenses in Machine Learning-Powered Networks: A Contemporary Survey," Mar. 2023, [Online]. Available: http://arxiv.org/abs/2303.06302. Access, May 5, 2024.
[17] N. Mani, M. Moh, and T.-S. Moh, "Defending Deep Learning Models Against Adversarial Attacks," Int. J. Softw. Sci. Comput. Intell., vol. 13, no. 1, pp. 72–89, Jan. 2021. doi: 10.4018/IJSSCI.2021010105.
[18] I. Rosenberg, A. Shabtai, Y. Elovici, and L. Rokach, "Adversarial Machine Learning Attacks and Defense Methods in the Cyber Security Domain," Jul. 2020, [Online]. Available: http://arxiv.org/abs/2007.02407. Access, May 5, 2024.
[19] C. Wang, J. Wang, and Q. Lin, "Adversarial Attacks and Defenses in Deep Learning: A Survey," in Intelligent Computing Theories and Application, 2021, pp. 450–461. doi: 10.1007/978-3-030-84522-3_37.
[20] E. Nowroozi, M. Mohammadi, P. Golmohammadi, Y. Mekdad, M. Conti, and A. S. Uluagac, "Resisting Deep Learning Models Against Adversarial Attack Transferability Via Feature Randomization," IEEE Trans. Serv. Comput., pp. 1–12, 2023. doi: 10.1109/TSC.2023.3329081.
[21] Y. Ganin et al., "Domain-Adversarial Training of Neural Networks," May 2015, [Online]. Available: http://arxiv.org/abs/1505.07818. Access, May 5, 2024.
[22] W. Zhao, S. Alwidian, and Q. H. Mahmoud, "Adversarial Training Methods for Deep Learning: A Systematic Review," Algorithms, vol. 15, no. 8, p. 283, Aug. 2022. doi: 10.3390/a15080283.
[23] F. V. Massoli, F. Carrara, G. Amato, and F. Falchi, "Detection of Face Recognition Adversarial Attacks," Comput. Vis. Image Underst., vol. 202, no. September 2020, p. 103103, 2021. doi: 10.1016/j.cviu.2020.103103.
[24] S. Liu and Y. Han, "ATRA: Efficient adversarial training with high-robust area," Vis. Comput., vol. 40, no. 5, pp. 3649–3661, May 2024. doi: 10.1007/s00371-023-03057-9.
[25] H. Ren, T. Huang, and H. Yan, "Adversarial examples: attacks and defenses in the physical world," Int. J. Mach. Learn. Cybern., vol. 12, no. 11, pp. 3325–3336, Nov. 2021. doi: 10.1007/s13042-020-01242-z.
[26] I. Kraidia, A. Ghenai, and S. B. Belhaouari, "Defense against adversarial attacks: robust and efficient compressed optimized neural networks," Sci. Rep., vol. 14, no. 1, p. 6420, Mar. 2024. doi: 10.1038/s41598-024-56259-z.
[27] A. Zolfi, S. Avidan, Y. Elovici, and A. Shabtai, "Adversarial Mask: Real-World Universal Adversarial Attack on Face Recognition Model," Nov. 2021, [Online]. Available: http://arxiv.org/abs/2111.10759. Access, May 5, 2024.
[28] J. Zheng, B. Li, S. Zhang, S. Wu, L. Cao, and S. Ding, "Attack Can Benefit: An Adversarial Approach to Recognizing Facial Expressions under Noisy Annotations," Proc. AAAI Conf. Artif. Intell., vol. 37, no. 3, pp. 3660–3668, Jun. 2023. doi: 10.1609/aaai.v37i3.25477.
[29] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial Examples: Attacks and Defenses for Deep Learning," IEEE Trans. Neural Networks Learn. Syst., vol. 30, no. 9, pp. 2805–2824, Sep. 2019. doi: 10.1007/978-3-030-87664-7_7.
[30] Y. Xu, K. Raja, R. Ramachandra, and C. Busch, "Adversarial Attacks on Face Recognition Systems," 2022, pp. 139–161.
[31] H. Zheng, Z. Zhang, J. Gu, H. Lee, and A. Prakash, "Efficient Adversarial Training with Transferable Adversarial Examples," Dec. 2019, [Online]. Available: http://arxiv.org/abs/1912.11969. Access, May 5, 2024.
[32] T. Huang et al., "Enhancing Adversarial Training via Reweighting Optimization Trajectory," Jun. 2023, [Online]. Available: http://arxiv.org/abs/2306.14275. Access, May 5, 2024.
[33] A. Shafahi et al., "Adversarial Training for Free!" Apr. 2019, [Online]. Available: http://arxiv.org/abs/1904.12843. Access, May 5, 2024.
[34] Z. Wang, X. Li, H. Zhu, and C. Xie, "Revisiting Adversarial Training at Scale," Jan. 2024, [Online]. Available: http://arxiv.org/abs/2401.04727. Access, May 5, 2024.

[35] X. Wei et al., "Learning Extremely Lightweight and Robust Model with Differentiable Constraints on Sparsity and Condition Number," 2022, pp. 690–707. doi: 10.1007/978-3-031-19772-7_40.
[36] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial Examples: Attacks and Defenses for Deep Learning," Dec. 2017, doi: 1712.07107. Access, May 5, 2024.
[37] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression," in 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, Jun. 2010, pp. 94–101. doi: 10.1109/CVPRW.2010.5543262.
[38] J. Peck and B. Goossens, "Robust width: A lightweight and certifiable adversarial defense," May 2024, [online]. Available: http://arxiv.org/abs/2405.15971. Access, May 5, 2024.
[39] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," Jan. 2018, [Online]. Available: http://arxiv.org/abs/1801.04381. Access, May 5, 2024.
[40] T. Agustin, M. H. Purwidiantoro, and M. L. Rahmadi, "Enhancing Facial Expression Recognition through Ensemble Deep Learning," in 2023 5th International Conference on Cybernetics and Intelligent System (ICORIS), Oct. 2023, pp. 1–6. doi: 10.1109/ICORIS60118.2023.10352183.
[41] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and Harnessing Adversarial Examples," Dec. 2014, doi: 1412.6572. Access, May 5, 2024.
[42] W. Villegas-Ch, A. Jaramillo-Alcázar, and S. Luján-Mora, "Evaluating the Robustness of Deep Learning Models against Adversarial Attacks: An Analysis with FGSM, PGD and CW," Big Data Cogn. Comput., vol. 8, no. 1, p. 8, Jan. 2024. doi: 10.3390/bdcc8010008.
[43] Y. Liu, S. Mao, X. Mei, T. Yang, and X. Zhao, "Sensitivity of Adversarial Perturbation in Fast Gradient Sign Method," 2019 IEEE Symp. Ser. Comput. Intell. SSCI 2019, no. 2, pp. 433–436, 2019. doi: 10.1109/TEVC.2019.2890858.
[44] J. Su, D. V. Vargas, and S. Kouichi, "One-pixel attack for fooling deep neural networks," Oct. 2017, doi: 10.1109/TEVC.2019.2890858. Access, May 5, 2024.
[45] T. Bai, J. Luo, J. Zhao, B. Wen, and Q. Wang, "Recent Advances in Adversarial Training for Adversarial Robustness," Feb. 2021, [Online]. Available: http://arxiv.org/abs/2102.01356. Access, May 5, 2024.
[46] S. Hussain et al., "ReFace: Real-time Adversarial Attacks on Face Recognition Systems," Jun. 2022, [Online]. Available: http://arxiv.org/abs/2206.04783. Access, May 5, 2024.
[47] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," BMC Genomics, vol. 21, no. 1, p. 6, Dec. 2020. doi: 10.1186/s12864-019-6413-7.
[48] I. M. Ross, "An optimal control theory for nonlinear optimization," J. Comput. Appl. Math., vol. 354, pp. 39–51, Jul. 2019, doi: 10.1016/j.cam.2018.12.044.

Authors' Profiles

Tinuk Agustin is a faculty member in the Department of Informatics, STMIK AMIKOM Surakarta (The College of Management, Informatics and Computer "AMIKOM" Surakarta), Indonesia. Her research interests include computer vision and deep learning. She has several published papers in refereed national and international journals.

Moch. Hari Purwidiantoro is a faculty member and Chairman of STMIK AMIKOM Surakarta (The College of Management, Informatics and Computer "AMIKOM" Surakarta), Indonesia. He has a diverse educational background, ranging from Associate Degree Electrical Engineering at Universitas Gajah Mada (UGM), Indonesia, Bachelor of Electrical Engineering at Institut AKPRIND Yogyakarta, Indonesia, Master of Management at STIE Artha Budhi Iswara Surabaya, Indonesia, to Master of Computer at Universitas AMIKOM Yogyakarta, Indonesia. He has interests and research in machine learning.

Mochammad Luthfi Rahmadi works as a System Administrator and is currently completing studies at the Informatics Department of the Universitas Siber Muhammadiyah, Indonesia. He has good programming skills and an interest in Informatics and Artificial Intelligence (AI) research.

How to cite this paper: Tinuk Agustin, Moch. Hari Purwidiantoro, Mochammad Luthfi Rahmadi, "Deep Learning for Robust Facial Expression Recognition: A Resilient Defense Against Adversarial Attacks", International Journal of Intelligent Systems and Applications(IJISA), Vol.16, No.5, pp.39-52, 2024. DOI:10.5815/ijisa.2024.05.04