AI-Driven Deep Learning Architectures for Robust Emotion Recognition

Hamad Ali Abosaq
Computer Science Department-College of Computer Science and Information Systems,
Najran University, Najran, 61441, Saudi Arabia

***Abstract*—Due to an insufficient labeled dataset, class-level variation emotion recognition becomes a challenging task in computer vision. Deep learning (DL) makes it possible to automatically learn meaningful patterns from facial expressions. It captures simple details such as edges, textures at low layers, and gradually builds up to more complex information, including Facial components and the overall meaning of the expression. Despite progress made via end-to-end learning, partial occlusions, inconsistent lighting, and biases within datasets are a few challenges that still remain. In this work, a DL based model is presented to classify two emotional states of human expression. The pipeline depends on several components, including the preparation of data, preprocessing and analysis, and the use of pretrained networks, dimensionality-reduction techniques, and region-based explanation via Grad-CAM. More than 2,000 images of happy and sad faces were derived from Kaggle. These images were used to test a custom-designed CNN and two widely adopted architectures, such as VGG16 and MobileNetV. The custom model attained an accuracy rate of 66% and 67% F1, while the VGG16 performed notably better with 78% accuracy and 77% F1, and the MobileNetV architecture, which achieved 77% accuracy and 73% F1. The statistical comparisons using paired t-tests and Wilcoxon signed-rank tests further confirmed these findings, showing that pre-trained models outperformed a custom CNN with a meaningful effect size. Although deeper networks are more susceptible to overfitting and the hand-crafted CNN suffered exhibited underfitting, the results indicate that pretained architecture provides a clear advantage for facial emotion recognition. This study makes a major contribution to existing computer vision research in removing the trade-off between accuracy and generalization, and opens doors to the application of lightweight yet interpretable models in practical affective computing systems.**

***Keywords*—Deep learning models; computer vision; emotion recognition; image processing; Grad-CAM**

I. Introduction

Facial Emotion Recognition (FER) is one of the prominent computer vision subareas that automatically translates human face emotions into emotional states. Affective computing, human-computer interaction (HCI), driver monitoring, mental illness diagnosis, and social robotics are its applications. Conventional FER systems were primarily based on hand-crafted descriptors such as Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), and Gabor filters with traditional classifiers. Despite these, these methods suffer from poor generalization under illumination change, occlusion, pose variation, and intra-class variation, thereby limiting deployment in the real-world [1], [2], [3], [4], [5].

Beyond FER, emotion and sentiment analysis have also been explored through concept-level and fine-grained sentiment detection methods, demonstrating the potential of machine learning to capture nuanced human affect [6] [7] [8].

Deep learning, specifically convolutional neural networks (CNNs), has progressed FER immensely by allowing hierarchical feature learning from images without requiring manual design. CNN-based architectures have shown better performance on typical datasets, such as FER2013, JAFFE, CK+, and KDEF. However, practical deployment remains faced with some challenges [9] [10]. First, the majority of available datasets are still on a small scale, with class imbalance and sparse demographic diversity, limiting the stability of trained models. Second, shallow CNNs can easily miss subtle expression cues, while excessively deep structures are prone to overfitting when there is limited data. Third, deep models' interpretability tends to be overlooked, decreasing trust and transparency in high-stakes applications like healthcare. Lastly, most FER systems only use frontal face images in isolation, without regard to profile views or naturalistic situations, and that diminishes generalizability to unconstrained settings [11] [12].

The above-discussed issues underscore the need for flexible, explainable, and data-efficient FER systems. Enhancing reliability under real-world scenarios, the existing literature stresses the importance of domain-specific, hybrid learning frameworks and explainability techniques (e.g., Grad-CAM) [13], [14].

The motivation of this study is to design a workflow for deep learning based two-class emotion detection (happy/sad) by utilizing transfer learning, data augmentation, and presentational methods.

It aims not only to improve the model's capacity to recognize facial expressions but also to simplify the feature space through methods like PCA and t-SNE so that the patterns become easier to interpret. Besides, Grad-CAM is used to show which regions of the image influence the decisions of the model, and makes the overall process clear and more reliable.

FER still faces many challenges, such as small datasets, class imbalance, and models that often act like "black boxes". Considering these challenges, the following is the key question pursued in this study:

RQ: Can a transfer-learning model, paired with data augmentation and basic interpretability tools, reliably

differentiate between happy and sad faces when the training data are limited and imbalanced?

While transfer learning, Grad-CAM interpretability, and dimensionality reduction have been applied in previous FER studies, few works have explored their combined application on small binary emotion datasets with a focus on rigorous statistical validation. Indeed, most prior work either addresses multi-class datasets, lacks formal significance testing, or does not explicitly evaluate model interpretability alongside performance. In this study, we apply a deep-learning pipeline to a small happy/sad dataset, incorporate PCA and t-SNE into Grad-CAM for clearer visual interpretation, and statistically confirm performance differences using both parametric and non-parametric tests. This article, therefore, fills the gap in the literature by showing how these methods can be meaningfully combined to advance the frontiers of accuracy, interpretability, and robustness in FER under constrained settings.

Beyond model performance, this work offers a number of high-level insights to the FER and larger computer-vision community. First, it frames an argument for why pretrained transfer-learning models, such as VGG16 and MobileNetV2, outperform custom CNNs on small, binary emotion datasets by emphasizing the importance of leveraging large-scale feature representations for improved generalization. By integrating Grad-CAM with dimensionality-reduction techniques such as PCA and t-SNE, the study further shows how the increased interpretability can divulge the decision-making patterns of models even in the face of scarce data, structuring trust in the automated FER systems being designed. Finally, the combination of transfer learning, data augmentation, and rigorous statistical evaluation offers practical design principles that may guide future affective-computing applications for which dataset size, class imbalance, and interpretability are of paramount importance. These contributions extend beyond procedural execution to concept and method, helping to advance the field.

This makes the following key contributions:

1) Development of a DL workflow for detecting two fundamental emotions (happy/Sad) from facial images.

2) Application of transfer learning to improve model performance on small-scale datasets.

3) Implementation of data augmentation techniques to enhance the diversity and robustness of training data.

4) Integration of flexibility methods to visualize and interpret which facial regions the model focuses on, improving transparency for real-world applications.

5) Through evolution on the benchmark dataset, demonstrating significant improvements in emotion recognition accuracy compared to existing methods.

The remainder of the study is organized as follows: Section II gives the literature review on facial emotion recognition. Section III explains the proposed method with dataset creation, feature selection, balancing, and model evaluation. Section IV displays experimental results and robustness. Section V gives the findings, limitations, and future directions for work. Section VI concludes the study.

II. Literature Review

Human emotion recognition from images is a difficult but strong task in social communication. (DL) techniques surpass conventional image processing in this area. This [6] research suggests a CNN-based structure for emotion recognition using facial expressions, through three essential steps: face detection, feature extraction, and emotion classification. The model is tested on the FERC-2013 and JAFFE datasets with an accuracy of 70.14% and 98.65%, respectively. Automatic (FER) is a challenging computer vision problem with uses in HCI, psychology, and expression generation. Handcrafted features pose challenges in dealing with variability, which leads to deep learning solutions. This work [7] presents FER-net, a softmax classifier CNN, against 21 state-of-the-art approaches on five benchmark datasets (FER2013, JAFFE, CK+, KDEF, RAF). On seven sentiments (neutral, anger, disgust, fear, happiness, sadness, surprise), FER-net had accuracies of 78.9%, 96.7%, 97.8%, 82.5%, and 81.68%, which surpass the existing techniques. FER converts facial expressions into emotional states and has widespread applications. Shallow CNNs in conventional models cannot extract features efficiently and tend to ignore profile views. This research [8] presents a transfer learning–based deep CNN pipeline, with pre-trained dense layers and pre-trained blocks being successively fine-tuned in a bid to attain higher accuracy. The approach is validated with eight DCNN models (VGG, ResNet, Inception-v3, DenseNet) on KDEF and JAFFE datasets. With DenseNet-161, the system achieved 96.51% and 99.52% accuracy on KDEF and JAFFE, respectively, compared to state-of-the-art methods, and demonstrated resistance for both frontal and profile faces. Human Action Recognition (HAR) of RGB videos was enhanced through an ensemble of 3D-AlexNet-RF and InceptionV3. The system achieved 99.54% accuracy on the HMDB51 dataset with good precision, recall, and F1-scores. The multi-tier ensemble is better than individual models and has great potential in healthcare, surveillance, HCI, and robotics through improved reliability in human activity recognition [9]. In [15], the authors presented a feedforward deep CNN for emotion recognition based on full-body motion patterns, with 95.4% accuracy on the York (15 emotions) and GEMEP (5 emotions) datasets, an improvement over previous research works. Although performance results show strong recognition performance, both datasets are small in size and variety, suggesting that real-world, larger data is required to guarantee generalization with a wider scope. Study [16] presented an overview of multimodal emotion recognition systems based on deep learning compared to unimodal approaches. It noted that multi-channel input integration (e.g., face images, physiological signals, neuroimaging) generally increases classification accuracy over single-modality systems. The article also discussed how the number of emotion classes, feature extractors, and dataset quality influence performance and presented difficulties in understanding emotional signals. This study [17] gave a systematic review of human–computer interaction approaches to training and learning, with emphasis on Extended Reality (VR, AR, MR). With a PRISMA-driven search of 1,156 articles, 69 were examined. The review categorized HCI approaches, identified core challenges, and stressed the potential of XR-HCI to enhance training effectiveness, safety, and cost-effectiveness while giving

insight into future research directions. In [18], the authors designed a hybrid text-based emotion recognition model by coupling machine learning (SVM) and deep learning (CNN + BiGRU) to outshine keyword- and lexicon-based methods in textual sentiment analysis. Based on mixed datasets of sentences, tweets, and conversations, the model reported an accuracy rate of 80.11%, indicating better emotion classification from text than using conventional NLP methods. This work [19] suggested a two-way feature extraction method for speech emotion recognition (SER) with deep learning. One followed PCA with DNN, and another used audio conversion to mel-spectrograms input to a pre-trained VGG-16 network. Experiments conducted on RAVDESS and TESS datasets revealed VGG-16 performing better compared to other models, with an accuracy of 81.94% (RAVDESS) and 97.15% (TESS), better than DNN, ResNet-18, and legacy classifiers. In [20], the authors suggested a real-time HCI system based on webcam-based eye gaze recognition and Mask R-CNN instance segmentation. The system identified four types of gazes with ~99% accuracy and segmented tools/parts with ~99% accuracy.

The understanding of human emotions from facial expressions has become a major focus of research as well as industry interest. This work [21] explores those approaches which depend solely on facial images for emotion recognition because facial expressions are one of the most natural means by which humans express their feelings and intentions. With the emergence of sophisticated machine learning methods, these techniques have notably enhanced their accuracy and efficiency. Understanding a person's emotions by looking at their face is beneficial for a wide variety of areas, from improving interactions with computers to enhancing customer service. However, the majority of publicly available facial expression datasets are too small to effectively train advanced learning models in need of large quantities of data. In this study [22], various methods of creating additional face images exhibiting varied expressions through the use of convolutional autoencoders are explored to expand this dataset. The described approach was tested on the Japanese Female Facial Expression (JAFFE) dataset, and the results indicated that this technique increased emotion recognition accuracy by approximately 21%.

III. Methodology

The research makes use of a pipeline based on deep learning for emotion recognition with the help of convolutional neural networks (CNNs) and sophisticated visualization. There are six overall stages involved in the methodological framework: dataset preprocessing, exploratory data analysis, feature extraction, dimensionality reduction, and interpretability of models by Grad-CAM, as shown in Fig. 1.

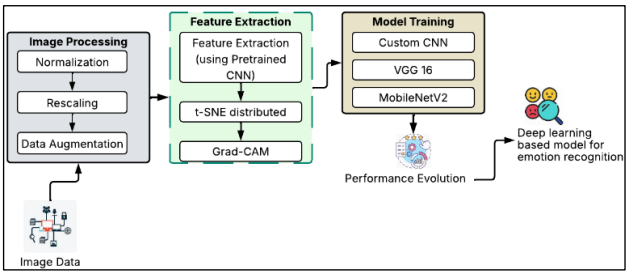

Fig. 1. Research methodology.

A. Dataset Collection

The dataset for this research was collected from Kaggle [23], which contains a group of over 2000 human portrait faces. The images were initially compiled from public sources such as Google Images and Pinterest. The dataset consists of two categories of emotion (happy and sad) and has images in various forms such as JPG, JPEG, PNG, and SVG. This dataset was utilized for training and testing of the deep learning models constructed for binary emotion classification.

For this study, facial images were collected from two benchmark datasets that have been widely used, FER2013 and CK+. FER2013 contains a large set of labeled facial expressions captured in unconstrained conditions, which has significant value for training models under realistic variability. CK+ provides high-quality, posed images with well-documented emotion labels, thus enabling precise evaluation of model performance. Together, these datasets provide a balanced combination of real-world variability and controlled conditions, making them suitable for assessing the performance, generalization, and interpretability of the proposed emotion recognition pipeline.

B. Exploratory Data Analysis (EDA)

For computational efficiency and consistency, all the images were resized to 128 × 128 pixels and normalized between [0, 1] using the ImageDataGenerator class in TensorFlow/Keras. A batch size of eight was used to optimize GPU memory utilization and speed up the training and inference process. Class distribution is shown in Fig. 2, where significant class imbalance exists. Representative samples per class were shown to authenticate dataset quality and detect intra-class diversity. All this analysis is important for ensuring the reliability of follow-up feature extraction and classification procedures [24].

To enhance the model's generalization and robustness, data augmentation was utilized with ImageDataGenerator in TensorFlow/Keras. Table I shows the data augmentation parameters. Pixel values were scaled to the range [0,1] by rescaling (rescale=1./255) for numerical stability during training. Random rotation up to 20° (rotation_range=20) was utilized to handle variations in orientations. Horizontal and vertical translations (width_shift_range=0.1, height_shift_range=0.1) modeled object movement in the frame, and shear transformations (shear_range=0.1) created geometric alterations that allowed for real-world variability simulation. Random zooming (zoom_range=0.1) enhanced model generalization across scales as well. All together, these augmentations lowered the chances of overfitting by artificially increasing the training set and subjecting the model to varied input conditions.

TABLE I. Data Augmentation Parameters

Parameters	Values	Description
Rescale	255./1	Normalize pixel to [0,1]
Rotation_range	20	Random image rotation (Degrees)
Width_shift_range	0.1	Horizontal translation shift
height_shift_range	0.1	Vertical translation shift
Shear_range	0.1	Shear transformation
Zoom_range	0.1	Random Zooming

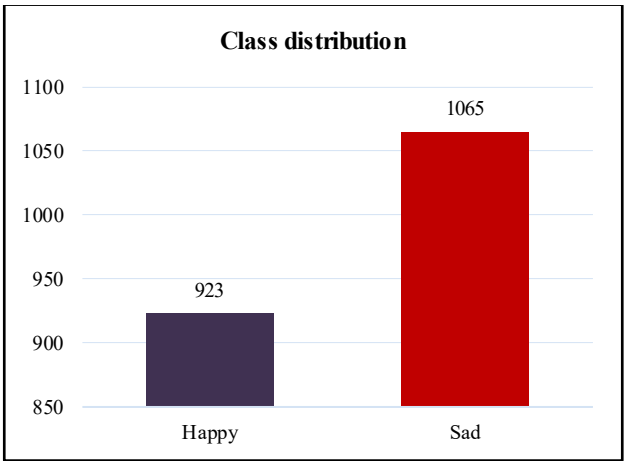

Fig. 2. Class distribution.

The dataset distribution is presented in Fig. 2, where the "Sad" class has 1,065 instances, and the "Happy" class has 923 instances, reflecting a slight imbalance. This imbalance can lead to the model becoming biased toward the majority class. To counter this, the training was done with class weights such that both classes contributed equally towards optimization. This ensures fair learning and enhances the reliability of the evaluation result obtained thereafter.

C. Feature Extraction Using Pre-Trained VGG16

Transfer learning was used to obtain high-level features from face images. The VGG16 model, pre-trained on the ImageNet database, was used with the top classification layers stripped off. A Global Average Pooling (GAP) layer was added to the last convolutional block to produce compact feature vectors for every image [25], [26]. This method leverages the representational powers of VGG16 in order to encourage effective facial structure encoding with reduced computational complexity. Mathematically, for an input image $I \in R^{(128\times128\times3)}$, the feature extractor gives a dense vector $f \in R^d$ whose dimension d corresponds to the pooled feature dimensionality. Such vectors were then used for downstream analysis. These embeddings were used as an input to two dimensionality reduction methods:

- t-distributed Stochastic Neighbor Embedding (t-SNE): A non-linear technique that embeds high-dimensional space into the 2D space without any loss of local neighborhood structures. This enabled emotion clusters to be visualized.
- Principal Component Analysis (PCA): A linear projection method was employed to estimate the proportion of variance captured by the dominant principal components. It assisted in finding the most significant directions of variation in the feature space and provided insights into the redundancy of extracted features [27], [28]

The integration of t-SNE and PCA provided both qualitative and quantitative evaluation of the extracted features.

TABLE II. FEATURE EXTRACTION PARAMETERS

Layer Type	Output Shape	Kernel/units	Activation
Input Layer	(128,128,3)	-	-
Conv2D+Relu (Block 1)	(128,128,64)	3x3,64	ReLu
Conv2D+Relu (Block 1)	(128,128,64)	3x3,64	ReLu
MaxPooling2D	(64,64,64)	2x2	-
COnv2D (Block5, last layer)	(4,4,512)	3x3,512	ReLu
Global Average Pooling	(512,)	3x3,512	-
Feature Vector	(512,)	-	-

Table II demonstrates the parameters for feature extraction using the VGG16 model. The VGG16 model was used as a feature extractor, and the input image of size (128×128×3) undergoes several convolutional and pooling layers. Shallow layers like Conv2D + ReLU (Block 1) learn low-level visual patterns such as edges and textures, whereas deeper layers like Conv2D (Block 5) represent high-level semantic features that pertain to facial expressions. A Global Average Pooling (GAP) layer was performed on the last convolutional maps to produce a compact 512-dimensional feature vector, which was subsequently utilized for dimensionality reduction and visualization. VGG16 hierarchical convolutional filters pre-trained from ImageNet yield transferable and semantically informative embeddings useful for emotion recognition tasks, and applying the GAP operation diminishes spatial dimensions while maintaining the most significant activations, making computation efficient and stable for the extracted features.

Fig. 3 shows the t-SNE plot shows the distribution of deep features learned from the VGG16 model for the "happy" and "sad" emotion classes. While some overlap is apparent, clear clusters exist, indicating that the network learns discriminative patterns that partially distinguish the two states of emotions.

The scatter indicates how high-dimensional feature vectors get mapped into a two-dimensional space, so inter-class similarities and differences become visually interpretable. t-SNE utilizes a probabilistic method that maintains the local neighborhood patterns of high-dimensional data, so even as clusters of similar emotions are preserved despite the reduction in dimensions. Second, the implementation of VGG16-extracted features guarantees that the embeddings are semantically rich and learned from convolutional filters trained on huge datasets, thereby yielding informative representations which make the visualized clusters more reliable. Collectively, these reasons prove t-SNE to be an effective tool to prove the separability of emotion-related features in latent space.

To improve the interpretability of CNN choices, Gradient-weighted Class Activation Mapping (Grad-CAM) was used. Grad-CAM derives class-specific saliency maps by backpropagating the gradients from the predicted class output to the final convolutional layer. For an input image I, the heatmap H is generated as:

$$H = Relu(\sum_k \alpha_k A^k), \alpha_k = \frac{1}{z}\sum_{i,j} \frac{\partial y^c}{\partial A_{ij}{}^k} \quad (1)$$

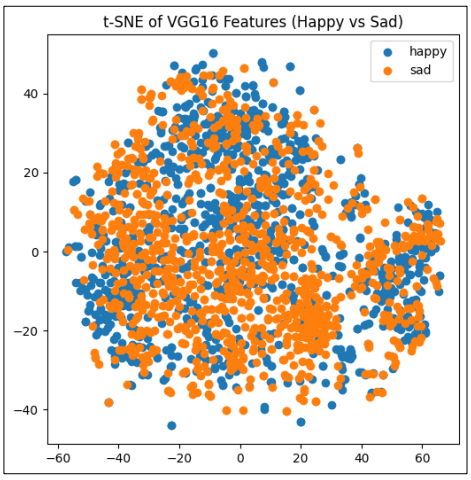

Fig. 3. t-SNE analysis.

where, A^k represents the feature maps of the last layer, y^c is the score of class c_i and α_k denotes the channel-wise importance weights. The consequent heatmaps were overlaid on the original images for maximum contrast to emphasize discriminative facial areas (e.g., eyes, mouth, forehead) responsible for the recognition of particular emotions. This ensures the transparency and provides biological explainability of the decision-making process of the model [29], [30].

Fig. 4 represents the Grad-CAM visualization emphasizes the most contributing facial regions to the model's choice. The network's main attention is towards the forehead, eyes, and mouth regions, which are key markers of emotional expression. Warm-colored areas (red/yellow) are areas of high-importance features, and cooler areas (blue) are less relevant to classification. The Grad-CAM uses the gradients of the target class score with respect to the final convolutional feature maps so that the highlighted areas directly correspond to the discriminative power of the CNN filters. Second, the utilization of convolutional feature maps does not destroy spatial information, enabling the method to localize class-discriminative regions without retraining or changing the base model. Collectively, these explanations corroborate that the predictions from the model are not random but based on physiologically relevant facial signals.

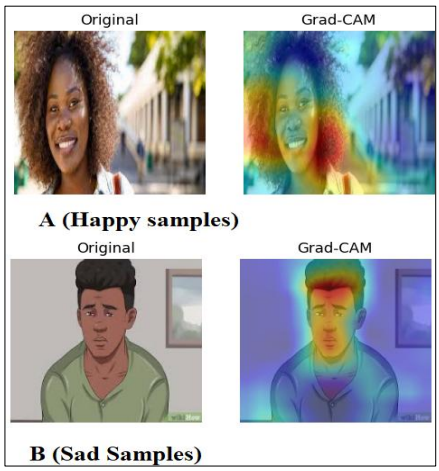

Fig. 4. Grad-CAM.

Algorithm 1 shows the procedure of feature extraction using the VGG16 architecture. To extract meaningful features from facial images, a pre-trained VGG16 model with ImageNet weights was used. Our dataset consists of two classes-Happy and Sad-each resized to 128 × 128 × 3 pixels. Exploratory data analysis was conducted at the start of the feature extraction process, including class distribution balance reviews. Then, all images were normalized by dividing the pixel values by 255 to standardize the input to the network. The pre-trained VGG16 model was used to generate the feature matrix, capturing high-level representations of every image.

Algorithm 1: Feature Extraction using VGG16 Model

Input:
- Image Dataset $\mathcal{D} = \{(x_i, y_i)\}^N_{i=1}$, with two classes: Happy and Sad
- Image Size $S = 128 \times 128 \times 3$
- Pre-trained VGG16 model M_{VGG} with ImageNet weights

Output:
- Extracted Feature Matrix $\mathcal{F} \in \mathcal{R}^{N \times d}$
- Low-dimensional embedding (t-SNE, PCA)
- Grad-CAM outcome

Start
1. Exploratory Data Analysis
2. Compute Class distribution
3. $n_c = |\{x_i \in \mathcal{D} | y_i = c\}| \quad \forall_c \in \{Happy, Sad\}$
4.
5. Image Processing
6. Normalize each image
7. $x_i' = \frac{x_i}{255}$

These features were subsequently transformed into low-dimensional embeddings using PCA and t-SNE algorithms for visualization and interpretability. Finally, the proposed system employed Grad-CAM to highlight regions of the face that highly affected the model's predictions, which provided more interpretability to the emotion recognition task.

D. Model Development

To compare the performance of classification, three deep learning models were tested and compared: a hand-designed Convolutional Neural Network (CNN), VGG16, and MobileNetV2.

1) Custom CNN: A convolutional neural network architecture, a practitioner designs or reconfigures for an application or constraints (Dataset, latency, device, and number of parameters). Custom CNNs change layer numbers, filter sizes, skip connections and regularization to trade accuracy versus cost [31].

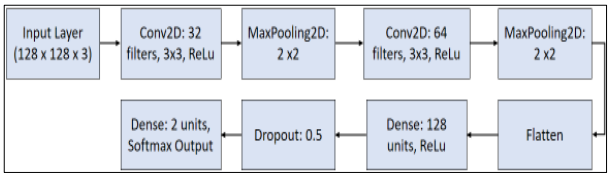

Fig. 5. Custom CNN architecture.

Fig. 5 illustrates the proposed custom CNN architecture. The three convolutional layers with enlarging filter sizes (32, 64, 128) are used to gradually extract low- to high-level spatial features, followed by max-pooling operations to compress dimensions and prevent overfitting. The flattened feature maps are fed into a fully connected dense layer of 128 neurons with ReLU activation to permit non-linear interactions among the features. A dropout layer (rate = 0.5) was used to reduce overfitting by randomly dropping out neurons in training. Lastly, a softmax output layer with two units was used for binary classification ("happy" vs. "sad"). This architecture strikes a balance between computational efficiency and expressive capacity: convolutional layers learn strong hierarchical features, pooling eliminates redundancy, and dropout improves generalization. The compactness ensures GPU trainability while retaining enough depth to acquire intra-class variability.

2) VGG16: A deep, uniform CNN that follows small 3×3 convolutions into numerous layers (16 weight layers in the VGG16 variant), demonstrating that deeper with small convolutions provides robust ImageNet performance and transferable feature representations [32].

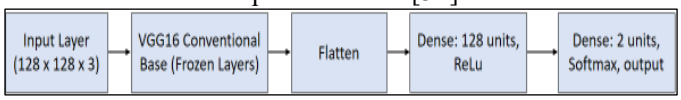

Fig. 6. VGG16 architecture.

The architecture of the VGG16-based model is shown in Fig. 6. The network accepts images of 128 × 128 × 3 as inputs, which are fed into the pre-trained VGG16 convolutional base with all the convolutional layers frozen in order to preserve low-level and mid-level visual features learned from ImageNet. The extracted feature maps are flattened and fed into a dense layer of 128 neurons with ReLU activation in order to capture high-level abstractions pertaining to the target task. A softmax output layer with two units is finally employed for binary classification. Freezing the convolutional base retains strong generalizable features while keeping computational expense and overfitting danger in check with small training data. Fine-tuning just the classifier head enables the model to draw pre-trained expertise while still effectively adjusting to the happy–sad emotion recognition task.

3) MobileNetV2: A mobile-efficient CNN employing inverted residual blocks with thin bottlenecks and depth wise separable convolutions to attain high accuracy with low latency and parameter usage on edge devices [33]

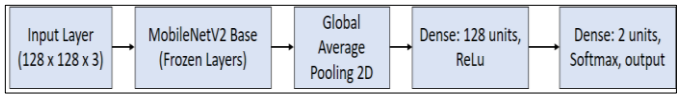

Fig. 7. MobileNetV2 architecture.

Fig. 7 depicts the MobileNetV2-based model. Input images (128 × 128 × 3) are processed through the pre-trained MobileNetV2 base with frozen convolutional blocks in order to capture generic low-level features. A Global Average Pooling layer down-samples feature maps into small representations, which are fed into a dense layer of 128 units with ReLU activation for task-specific learning. Last but not least, a softmax output layer with two units does binary classification. The architecture employs MobileNetV2's depthwise separable lightweight convolutions to be computationally efficient with high representational power. Freezing the convolutional base reduces overfitting using small data, and global pooling cuts parameters relative to flattening, enhancing generalization. This efficiency-accuracy trade-off positions MobileNetV2 appropriately for resource-limited training regimes.

E. Model Evaluation

Model performance was evaluated using an accuracy, precision, recall, F1 score, loss graphs and confusion metrics. Statistical significance is the probability that an observed effect is unlikely to have occurred by random chance under the null hypothesis. It is typically measured by a p-value, compared against a pre-specified threshold ($\alpha = 0.05$ in most studies). Results are considered statistically significant if $p \leq \alpha$, indicating that null hypothesis can be rejected. However, statistically significance does not imply practical importance or the magnitude of the effect. Therefore, it should be reported alongside effect size measures (e.g., Cohen's d) and confidence intervals to convey the reliability and practical impact of the findings [34], [35]

In this research, model evaluation combined descriptive statistics with statistical inference. Training and validation loss and accuracy were monitored over 70 epochs to track learning behavior and identify possible overfitting. To statistically verify performance difference, paired *t-tests* were applied under parametric assumptions with Wilcoxon signed-rank tests used as a non-parametric alternative for robustness. Cohen's d was additionally computed to quantify effect size and assess the magnitude of differences between training and validation performance.

IV. Results and Discussion

Experiments were performed on Kaggle notebook using Keras Deep learning library in Python. The model was trained with a batch size of 32 over 70 epochs. The Adam optimization as employed with a learning rate of 1×10^{-4} to ensure efficient convergence. Categorical cross-entropy was chosen a loss function, as it is well-suited for multi-class classification task.

TABLE III. Performance Evaluation of DL Models

DL models	Accuracy	Precision	Recall	F1 score
Custom CNN	66%	67%	69%	67%
VGG16	78%	79%	76%	77%
MobileNetV	77%	74%	73%	73%

Table III shows the performance on DL models. It demonstrates that Custom CNN has the lowest performance (Accuracy = 66%, F1 = 67%), representing a low generalization capability. Both transfer learning models significantly outperformed it. VGG16 produced the best overall results (Accuracy = 78%, Precision = 79%, Recall = 76%, F1 = 77%), which demonstrates a high precision-recall balance. MobileNetV also achieved well (Accuracy = 77%, F1 = 73%) but with relatively lower precision and recall, implying the slightest trade-off in terms of predictive consistency.

Technically, the superiority of VGG16 and MobileNetV to the nonstandard CNN is due to their pretrained feature extractors, which tap into large-scale ImageNet representations, allowing for improved generalization with fewer task-specific samples. The marginally higher and better-balanced F1 score in VGG16 also reflects its advantage in dealing with class-level variability, while MobileNetV's lightweight structure provides comparable accuracy at lower complexity.

Fig. 8 shows confusion matrices for Custom CNN and VGG16 models on a binary classification problem (happy vs. sad). Fig. 8(A) represents the Custom CNN, the model correctly identifies 226 happy and 169 sad samples but classifies 150 sad as happy, showing bias towards classifying 'happy'. This creates lower sensitivity for the 'sad' class, which implies poor feature extraction ability. Fig. 8(B) illustrates the VGG16 model, the performance is better balanced: 208 happy and 257 sad samples are classified correctly, and fewer misclassifications are made.

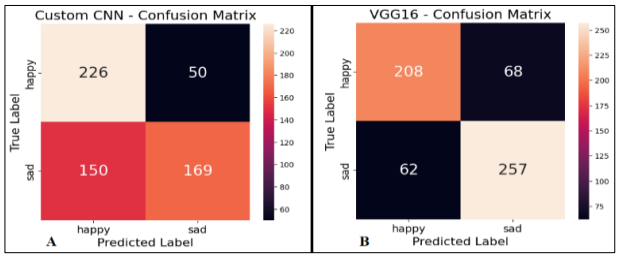

Fig. 8. Confusion matrices for DL models.

This signifies improved generalization and discriminative ability, thanks to VGG16's deeper structure and more detailed feature representations. Though Custom CNN performs quite well, it is adversely affected by class imbalance and feature discrimination, resulting in biased predictions. VGG16, though prone to overfitting as previously observed, shows better classification accuracy and resilience when separating classes. VGG16 is hence better suited for real-world implementation, if regularization is used to counteract overfitting.

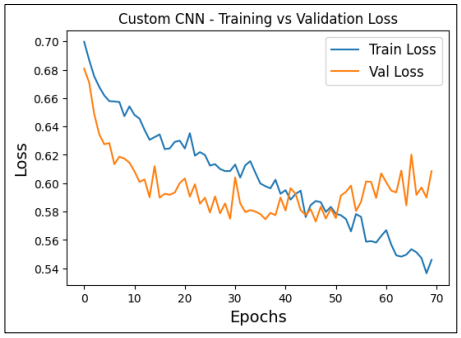

Fig. 9. Loss for CNN model.

Fig. 9 illustrates training versus validation loss for the Custom CNN across 70 epochs. The blue curve shows the training loss, showing how well the model fits the training data during learning and orange line depicts the validation loss, indicating how well the model generalizes to unseen data. Training loss drops gradually from ~0.70 down to ~0.54, whereas validation loss decreases first below the training loss, remains stable around 0.58–0.60, and fluctuates in the later epochs. The fairly small difference between training and validation loss reflects a decrease in overfitting relative to deeper models (e.g., VGG16, MobileNetV2). Nonetheless, the noisy validation curve reveals low model capacity, where the network is finding it hard to generalize reliably and could be underfitting. The Custom CNN shows greater consistency between training and validation loss, which indicates balanced learning. However, the higher overall loss values and failure to converge further suggest the representational capability of the model is inadequate to extract complex features. Optimization methods like tuning the depth of architecture, convolutional layer addition, or utilization of transfer learning may generalize better without overfitting.

Fig. 10 shows comparison of training and validation loss curves of MobileNetV2 and VGG16 models over 70 epochs. The blue curve shows the training loss, showing how well the model fits the training data during learning and orange line depicts the validation loss, indicating how well the model generalizes to unseen data.

Fig. 10(A) depicts the MobileNetV2 performance where the training loss decreases linearly to ~0.15, whereas the validation loss shows a plateau at 0.5–0.55 after approximately ~10 epochs. The difference between training and validation loss suggests overfitting since the model learns training data adequately but cannot generalize. Whereas, Fig. 10(B) shows the VGG16, training loss also reduces steadily, down to ~0.1, but validation loss oscillates and even rises over time, with higher overfitting than MobileNetV2. The increasing gap indicates that although VGG16 is high-capacity to reduce training loss, it has problems generalizing, presumably because it contains more parameters and has a greater risk of memorization. MobileNetV2 has more consistent validation performance than VGG16 and is hence better for cases with sparse data or where generalization is essential. VGG16, though strong, needs higher regularization (e.g., dropout, data augmentation, weight decay) to prevent overfitting.

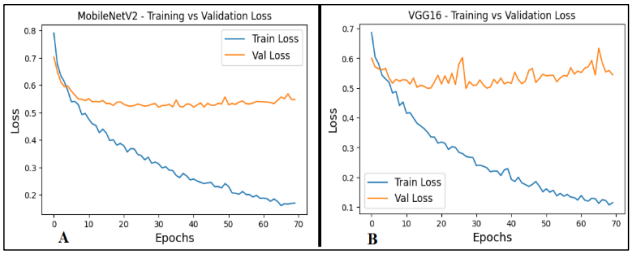

Fig. 10. Loss graphs for DL models.

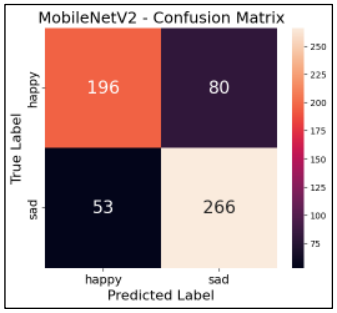

Fig. 11. MobileNet.

Fig. 11 illustrates the confusion matrix which reveals that the model is fairly good at identifying "happy" and "sad" emotions, as it was able to correctly predict 196 "happy" samples and 266 "sad" samples. There are 80 "happy" instances and 53 "sad" instances misclassified as "sad" and "happy" respectively. This gives a slightly better precision and recall for the "sad" class than "happy". The disparity in misclassification implies that the model could be more attuned to sensing "sad" emotions, possibly because of feature representations biased towards that class or class distribution during training. Overall, the matrix shows good generalization with potential to improve further in lowering false negatives for the "happy" class.

Table IV demonstrates the statistical tests which shows the extensive variations between the tested models. Custom CNN was much poorer than the reference point ($t = -4.798$, $p < 0.0001$; $W = 511.0$, $p = 1.8\times10^{-5}$; Cohen's $d = -0.573$), reflecting a moderate negative effect. Conversely, both VGG16 ($t = 17.441$, $p < 0.00001$; $W = 20.0$, $p = 8.42\times10^{-13}$; $d = 2.085$) and MobileNetV ($t = 15.725$, $p < 0.00001$; $W = 37.0$, $p = 1.72\times10^{-12}$; $d = 1.880$) realized highly significant gains with extremely large effect sizes. The accord between non-parametric and parametric tests reasserts the solidity of these findings, with assurance that transfer learning models (MobileNetV and VGG16) significantly outperform the tailored CNN both statistically and practically.

TABLE IV. PAIRED T-TEST

DL models	Paired t-test		Wilcoxon Test		Cohen's d
	T	P-values	W	P-values	Cohen's d
Custom CNN	-4.798	0.00001	511.0	1.8 x e^{-5}	-0.573
VGG16	17.441	0.00000	20.0	8.42 x e^{-13}	2.085
MobileNetV	15.725	0.00000	37.0	1.72 x e^{-12}	1.880

V. DISCUSSION

DL models have revolutionized image-based classification by learning hierarchical features from unstructured data in an automated manner. Their ability to capture intricate patterns makes them especially useful for identifying emotions, medical imaging, and object detection. Key to this success are deep architectures and transfer learning that improve generalization, reduce the necessity of large labeled datasets, and enhance predictive performance compared to traditional machine learning methods. Therefore, this study compared the performances of three DL models, focusing on the benefits of transfer learning over a custom-designed CNN. Based on Table III, the proposed custom CNN had the poorest scores in all categories: 66% in terms of accuracy and 67% in terms of F1 score. This finding is indicative of its poor generalization capability. Similarly, VGG16 generated the highest overall performance with an Accuracy = 78% and F1 = 77%, while MobileNetV yielded the same accuracy of 77% with a slightly lower F1 score of 73%, indicating a small trade-off between precision and recall. Indeed, both transfer learning models showed stronger performances due to their pre-trained feature extractors, which leveraged large-scale ImageNet representations to learn more discriminative and generalizable features. Further differences can be seen in the class-wise predictions, as depicted in Fig. 8. The custom CNN showed bias towards the 'happy' class by mislabeling 150 samples of the 'sad' class, which reflects its weak feature extraction and poor handling of class imbalance. On the other hand, VGG16 produced a more balanced classification: 208 happy and 257 sad samples were correctly identified, demonstrating higher discriminative power given that it is deeper and represents richer features. This is also reflected in the superior F1 and recall values shown in Table III. Analysis of the loss curves shown in Fig. 9 provides further insight. The custom CNN has high validation loss throughout, indicating underfitting given its limited representational capacity. MobileNetV2 has early overfitting, with its validation loss plateauing, whereas VGG16 has more pronounced overfitting given its higher capacity and thus requires regularization techniques, such as dropout or data augmentation. This is reinforced by the statistical analysis shown in Table IV, where the custom CNN presents a moderate negative effect (Cohen's $d = -0.573$), while for both VGG16 and MobileNetV, very large positive effects are obtained (Cohen's $d = 2.085$ and 1.880, respectively). In terms of both the parametric test (paired t-test) and the non-parametric test (Wilcoxon), it is shown that the transfer learning models significantly outperform the custom CNN, hence they have practical utility. Overall, the proposed analysis support that the use of a transfer learning model combined with data augmentation and basic interpretability techniques may be enough for the reliable recognition of happy and sad facial expressions even when the training dataset is limited and imbalanced. This will answer effectively the research question of this study.

The proposed approach offers several benefits compared to existing methods. Firstly, by leveraging transfer-learning models such as VGG16 and MobileNetV2, higher accuracy and F1-score are attained on small binary emotion datasets compared to custom CNNs, resulting in improved generalization. Secondly, incorporating Grad-CAM with dimensionality-reduction techniques such as PCA and t-SNE has the effect of providing interpretable visualizations, allowing a deeper understanding of which features contribute to the predictions and enhancing trust in data-scarce FER systems. Thirdly, statistical analyses using paired t-tests and Wilcoxon tests confirm the significance of performance differences and hence provide rigor often absent in similar studies. Overall, the combination of performance, interpretability, and statistically validated methodology represents a practical framework that generalizes to other affective computing tasks.

- *Practical Implications*: These results indicate that the pretraining models VGG16 and MobileNetV are more suitable for tasks with small datasets, giving better accuracy with balanced precision-recall performances. Their consistent feature extraction methods mean they can be reliably used in emotion recognition and other types of classification that require class balance and generalization.
- *Limitations*: Despite the high accuracy of transfer learning models like VGG16, overfitting is still possible due to a large number of parameters, especially when using small datasets. The poor performance of the custom CNN underlines that in many cases, manually

designing models without pretrained knowledge often presents challenges in obtaining good outcomes.

VI. Conclusion

This research implemented and tested a complete pipeline for the emotions detection using CNNs and pre-trained transfer learning models. The proposed workflow encompasses the entire procedure, from preprocessing the dataset and exploratory analysis to feature extraction, dimensionality reduction, and visual interpretation via Grad-CAM. The results indicated that besides being accurate, the proposed framework was also interpretable, providing insights into how the models reached a particular decision. The performance of pretrained networks, specifically VGG16 and MobileNetV, outperformed the custom CNN in terms of accuracy, F1 score, and overall reliability. While VGG16 yielded the most balanced performances for metrics, MobileNetV attained competitive results with a smaller and more efficient architecture, which makes it ideal in resource-constrained scenarios. Grad-CAM visualizations further ascertained that models focus on meaningful facial regions, hence reinforcing the trustworthiness of their predictions. These findings underline the value of marrying transfer learning with interpretability tools. This study has managed to build models that are strong in performance and easily interpretable. Lightweight models like MobileNetV perform well and show promise for real-time applications, even when constrained by small computing resources. Dimensionality reduction helped to simplify complex feature spaces and made the patterns clearer and more actionable. Putting the interpretation within the pipeline strengthens confidence in results and opens up the system for practical use. In the future, this work could be extended in several ways. The inclusion of other multi-class emotion datasets would introduce more affective states, such as fear, anger, and surprise, thus making the models more generalizable. Deeper models, such as VGG16, could be regularized using dropout, data augmentation, and adversarial training to prevent overfitting. Hybrid methods based on CNN with attention mechanisms or transformer-based architectures could achieve better feature representation and contextual understanding. Lightweight models, such as MobileNetV, should be easily deployable on edge devices or mobile platforms to realize real-time emotion recognition for applications in healthcare, education, and human-computer interaction.

References

[1] W. Mellouk and W. Handouzi, "Facial emotion recognition using deep learning: review and insights," Procedia Computer Science, vol. 175, pp. 689–694, Jan. 2020, doi: 10.1016/j.procs.2020.07.101.

[2] Z.-Y. Huang et al., "A study on computer vision for facial emotion recognition," Sci Rep, vol. 13, no. 1, p. 8425, May 2023, doi: 10.1038/s41598-023-35446-4.

[3] A. Khattak, M. Z. Asghar, M. Ali, and U. Batool, "An efficient deep learning technique for facial emotion recognition," Multimed Tools Appl, vol. 81, no. 2, pp. 1649–1683, Jan. 2022, doi: 10.1007/s11042-021-11298-w.

[4] G. L. Sălăgean, M. Leba, and A. C. Ionica, "Leveraging Symmetry and Addressing Asymmetry Challenges for Improved Convolutional Neural Network-Based Facial Emotion Recognition," Symmetry, vol. 17, no. 3, p. 397, Mar. 2025, doi: 10.3390/sym17030397.

[5] W. Ahmad, A. R. Shahzad, M. A. Amin, W. H. Bangyal, T. J. Alahmadi, and S. H. Khan, "Machine learning driven dashboard for chronic myeloid leukemia prediction using protein sequences," PLOS ONE, vol. 20, no. 6, p. e0321761, Jun. 2025, doi: 10.1371/journal.pone.0321761.

[6] A. Khattak, M. Z. Asghar, Z. Ishaq, W. H. Bangyal, and I. A. Hameed, "Enhanced concept-level sentiment analysis system with expanded ontological relations for efficient classification of user reviews," Egyptian Informatics Journal, vol. 22, no. 4, pp. 455–471, Dec. 2021, doi: 10.1016/j.eij.2021.03.001.

[7] "(PDF) Prediction of Insurance Fraud Detection using Machine Learning Algorithms," ResearchGate, Aug. 2025, doi: 10.22581/muet1982.2201.04.

[8] Z. Ahmad, W. Bangyal, K. Nisar, M. Haque, and M. Khan, "Comparative Analysis Using Machine Learning Techniques for Fine Grain Sentiments," JAI, vol. 4, no. 1, pp. 49–60, 2022, doi: 10.32604/jai.2022.017992.

[9] S. Jaiswal and G. C. Nandi, "Robust real-time emotion detection system using CNN architecture," Neural Comput & Applic, vol. 32, no. 15, pp. 11253–11262, Aug. 2020, doi: 10.1007/s00521-019-04564-4.

[10] M. Khalid, A. Ashraf, W. H. Bangyal, and M. Iqbal, "An Android Application for Unwanted Vehicle Detection and Counting," in 2023 International Conference on Human-Centered Cognitive Systems (HCCS), Dec. 2023, pp. 1–6. doi: 10.1109/HCCS59561.2023.10452502.

[11] D. Sahana, K. S. Varsha, S. Sen, and R. Priyanka, "A CNN-Based Approach for Facial Emotion Detection," in Soft Computing: Theories and Applications, R. Kumar, A. K. Verma, T. K. Sharma, O. P. Verma, and S. Sharma, Eds., Singapore: Springer Nature, 2023, pp. 1–10. doi: 10.1007/978-981-19-9858-4_1.

[12] C. Dalvi, M. Rathod, S. Patil, S. Gite, and K. Kotecha, "A Survey of AI-Based Facial Emotion Recognition: Features, ML & DL Techniques, Age-Wise Datasets and Future Directions," IEEE Access, vol. 9, pp. 165806–165840, 2021, doi: 10.1109/ACCESS.2021.3131733.

[13] F. Z. Canal et al., "A survey on facial emotion recognition techniques: A state-of-the-art literature review," Information Sciences, vol. 582, pp. 593–617, Jan. 2022, doi: 10.1016/j.ins.2021.10.005.

[14] A. R. Khan, "Facial Emotion Recognition Using Conventional Machine Learning and Deep Learning Methods: Current Achievements, Analysis and Remaining Challenges," Information, vol. 13, no. 6, p. 268, Jun. 2022, doi: 10.3390/info13060268.

[15] R. Santhoshkumar and M. K. Geetha, "Deep Learning Approach for Emotion Recognition from Human Body Movements with Feedforward Deep Convolution Neural Networks," Procedia Computer Science, vol. 152, pp. 158–165, Jan. 2019, doi: 10.1016/j.procs.2019.05.038.

[16] S. M. S. A. Abdullah, S. Y. A. Ameen, M. A. M. Sadeeq, and S. Zeebaree, "Multimodal Emotion Recognition using Deep Learning," Journal of Applied Science and Technology Trends, vol. 2, no. 01, pp. 73–79, May 2021, doi: 10.38094/jastt20291.

[17] A. Sadeghi Milani, A. Cecil-Xavier, A. Gupta, J. Cecil, and S. Kennison, "A Systematic Review of Human–Computer Interaction (HCI) Research in Medical and Other Engineering Fields," International Journal of Human–Computer Interaction, vol. 40, no. 3, pp. 515–536, Feb. 2024, doi: 10.1080/10447318.2022.2116530.

[18] S. K. Bharti et al., "Text-Based Emotion Recognition Using Deep Learning Approach," Computational Intelligence and Neuroscience, vol. 2022, no. 1, p. 2645381, 2022, doi: 10.1155/2022/2645381.

[19] A. Aggarwal et al., "Two-Way Feature Extraction for Speech Emotion Recognition Using Deep Learning," Sensors, vol. 22, no. 6, p. 2378, Jan. 2022, doi: 10.3390/s22062378.

[20] H. Chen, N. Zendehdel, M. C. Leu, and Z. Yin, "Real-time human-computer interaction using eye gazes," Manufacturing Letters, vol. 35, pp. 883–894, Aug. 2023, doi: 10.1016/j.mfglet.2023.07.024.

[21] A. Rehman, M. Mujahid, A. Elyassih, B. AlGhofaily, and S. A. O. Bahaj, "Comprehensive Review and Analysis on Facial Emotion Recognition: Performance Insights into Deep and Traditional Learning with Current Updates and Challenges.," Computers, Materials & Continua, vol. 82, no. 1, p. 41, Jan. 2025, doi: 10.32604/cmc.2024.058036.

[22] S. N. Yousafzai et al., "A multi-scale simplicial transformer with graph attention for facial emotion recognition," Ain Shams Engineering Journal, vol. 16, no. 10, p. 103584, Oct. 2025, doi: 10.1016/j.asej.2025.103584.

[23] "Over 2000 Sad and Happy faces Auto Detection." Accessed: Oct. 05, 2025. [Online]. Available: https://www.kaggle.com/datasets/alirezaatashnejad/sad-and-happy-face-detection

[24] S. Ghaffarian, J. Valente, M. van der Voort, and B. Tekinerdogan, "Effect of Attention Mechanism in Deep Learning-Based Remote Sensing Image Processing: A Systematic Literature Review," Remote Sensing, vol. 13, no. 15, p. 2965, Jan. 2021, doi: 10.3390/rs13152965.

[25] S. Sharma, K. Guleria, S. Tiwari, and S. Kumar, "A deep learning based convolutional neural network model with VGG16 feature extractor for the detection of Alzheimer Disease using MRI scans," Measurement: Sensors, vol. 24, p. 100506, Dec. 2022, doi: 10.1016/j.measen.2022.100506.

[26] Z.-P. Jiang, Y.-Y. Liu, Z.-E. Shao, and K.-W. Huang, "An Improved VGG16 Model for Pneumonia Image Classification," Applied Sciences, vol. 11, no. 23, p. 11185, Jan. 2021, doi: 10.3390/app112311185.

[27] H. Perez and J. H. M. Tah, "Improving the Accuracy of Convolutional Neural Networks by Identifying and Removing Outlier Images in Datasets Using t-SNE," Mathematics, vol. 8, no. 5, p. 662, May 2020, doi: 10.3390/math8050662.

[28] J. Xue, Y. Chen, O. Li, and F. Li, "Classification and identification of unknown network protocols based on CNN and T-SNE," J. Phys.: Conf. Ser., vol. 1617, no. 1, p. 012071, Aug. 2020, doi: 10.1088/1742-6596/1617/1/012071.

[29] D. A. Muhtasim, M. I. Pavel, and S. Y. Tan, "A Patch-Based CNN Built on the VGG-16 Architecture for Real-Time Facial Liveness Detection," Sustainability, vol. 14, no. 16, p. 10024, Jan. 2022, doi: 10.3390/su141610024.

[30] I. García-Rubio, R. Gallardo-Cava, D. Ortega-delCampo, J. Guillen-Garcia, D. Palacios-Alonso, and C. Conde, "Grad-CAM Applied to the Detection of Instruments Used in Facial Presentation Attacks," in Artificial Intelligence for Neuroscience and Emotional Systems, J. M. Ferrández Vicente, M. Val Calvo, and H. Adeli, Eds., Cham: Springer Nature Switzerland, 2024, pp. 270–281. doi: 10.1007/978-3-031-61140-7_27.

[31] I. Dixit, C. Prakash, G. Ramya, M. Dinesh, J. Giri, and A. Amer, "Human Emotion Detection using Deep CNN," in 2024 8th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Nov. 2024, pp. 1391–1395. doi: 10.1109/ICECA63461.2024.10800985.

[32] S. Mascarenhas and M. Agarwal, "A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification," in 2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON), Nov. 2021, pp. 96–99. doi: 10.1109/CENTCON52345.2021.9687944.

[33] Y. Zhang, A. Zhou, F. Zhao, and H. Wu, "A Lightweight Vehicle-Pedestrian Detection Algorithm Based on Attention Mechanism in Traffic Scenarios," Sensors, vol. 22, no. 21, p. 8480, Jan. 2022, doi: 10.3390/s22218480.

[34] S. J. Dankel and J. P. Loenneke, "Effect Sizes for Paired Data Should Use the Change Score Variability Rather Than the Pre-test Variability," The Journal of Strength & Conditioning Research, vol. 35, no. 6, p. 1773, Jun. 2021, doi: 10.1519/JSC.0000000000002946.

[35] J. Carrasco, S. García, M. M. Rueda, S. Das, and F. Herrera, "Recent trends in the use of statistical tests for comparing swarm and evolutionary computing algorithms: Practical guidelines and a critical review," Swarm and Evolutionary Computation, vol. 54, p. 100665, May 2020, doi: 10.1016/j.swevo.2020.100665.