Article

Not peer-reviewed version

Simple Histogram Equalization Technique Improves Performance of VGG models on Facial Emotion Recognition Datasets

Jaher Hassan Chowdhury , Qian Liu , Sheela Ramanna *

Posted Date: 23 April 2024

doi: 10.20944/preprints202404.1540.v1

Keywords: Facial emotion recognition; Convolutional Neural Network; Histogram Equalization; Transfer Learning; VGG architecture

Article

Simple Histogram Equalization Technique Improves Performance of VGG models on Facial Emotion Recognition Datasets

Jaher Hassan Chowdhury, Qian Liu and Sheela Ramanna *

Department of Applied Computer Science, The University of Winnipeg, Manitoba, Canada
* Correspondence: s.ramanna@uwinnipeg.ca;

Abstract: Facial Emotion Recognition (FER) is crucial across psychology, neuroscience, computer vision, and machine learning due to the diversified and subjective nature of emotions, varying considerably across individuals, cultures, and contexts. This paper investigates the impact of histogram equalization, data augmentation and model optimization strategies on three well-known FER datasets using pre-trained VGG models. Additionally, this paper showcases the effectiveness of different regularization techniques, callbacks, and learning schedulers in enhancing model performance by conducting extensive experiments. The model evaluation is discussed in terms of the following metrics: Accuracy, Area Under the Receiver Operating Characteristic Curve (AUC-ROC), Area Under the Precision-Recall Curve (AUC-PRC), and Weighted F1 score. Notably, the fine-tuned VGG models exhibit state-of-the-art performance compared to more complex conventional transfer learning models resulting in accuracies of 100%, 95.92%, and 69.65% on the CK+, KDEF, and FER2013 datasets, respectively.

Keywords: facial emotion recognition; convolutional neural network; histogram equalization; transfer learning; VGG architecture

1. Introduction

Emotions play a significant role in human interactions, serving as essential mediators in social communication systems[1]. Human expression of emotions incorporates diverse modalities, including facial expressions, speech patterns [2], and body language [3]. According to Darwin and Prodger [4], human facial expressions indicate their emotional states and intentions. Recently, automatic emotion detection through computer vision techniques has shown a growth in interest and application across many domains, including hospital patient care [5], neuroscience research [6], smart home technologies [7], and even in cancer treatment [8,9]. This diversity has established emotion recognition as a distinct and growing field within research, primarily due to its wide range of applications and intense impact on various phases of human life.

Emotion recognition from images mainly consists of two steps: feature extraction and classification. Facial images encompass a multitude of features including geometric, texture, color, intensity, landmarks, shape, and histogram-based features. Handcrafted techniques for feature extraction in facial images involve manual identification of landmarks for geometric features, texture analysis using methods like Local Binary Patterns (LBP) [10], and color distribution analysis through histograms. To enhance the feature extraction process, dimensionality reduction techniques such as PCA (Principal Component Analysis) [11]/t-SNE (t-Distributed Stochastic Neighbor Embedding) [12] have been employed to obtain crucial features for classification. Traditional machine learning algorithms like Support Vector Machine (SVM) [13], and Random Forest (RF) [14] were used to classify the emotions from these features. However, hand-crafted features often struggle to capture the important information required for effective face identification. Moreover, kernel-based methods frequently produce feature vectors that are excessively large, leading to overfitting of the model [15].

Deep learning, particularly Convolutional Neural Networks (CNNs) [16], is renowned for its capability to autonomously learn hierarchical features and complex patterns. However, CNNs frequently face challenges such as overfitting, which arises from limited data availability and computational complexity. Additionally, issues like vanishing or exploding gradients can undermine the stability of training processes [17].

Transfer learning has gained popularity in machine learning as a method for accelerating tasks. Transfer learning provides a framework for leveraging well-known pre-trained models such as VGG (Visual Geometry Group) [18], ResNet [19], DenseNet121 [20] trained on millions of image data and is particularly relevant for FER application in the related domain of facial images.

A schematic representation of the proposed framework is presented in Figure 1, utilizing images sourced from the Karolinska Directed Emotional Faces (KDEF) [21] dataset for illustrative purposes. As presented in Figure 1, the first step (1) showcases facial images retrieved from the KDEF, Filtered Facial Expression Recognition 2013 (FER2013) [22], and Cohn-Kanade (CK+) [23] datasets. In the subsequent step (2), the data undergoes preprocessing, wherein data augmentation techniques such as horizontal flipping, zooming, rotating, and histogram equalization [24] methods are applied. These techniques serve to augment the dataset, enhancing image contrast and thereby facilitating improved feature extraction. Moving forward to step (3), fine-tuning and modification of the pre-trained VGG19 and VGG16 models are undertaken. During this process, the last convolutional block of the models is kept unfrozen, while the remaining layers of the base models (pretrained VGG16, VGG19) are frozen. Additionally, the fully connected layers are then connected to these models. Moreover, diverse learning rate schedulers, including cosine annealing [25] are implemented in the models. The evaluation phase encompasses training the models on datasets such as KDEF, Filtered FER2013, and CK+, followed by assessment using various evaluation metrics. These metrics include Accuracy, AUC-ROC [26], AUC-PRC, and Weighted F1 score [27], respectively.

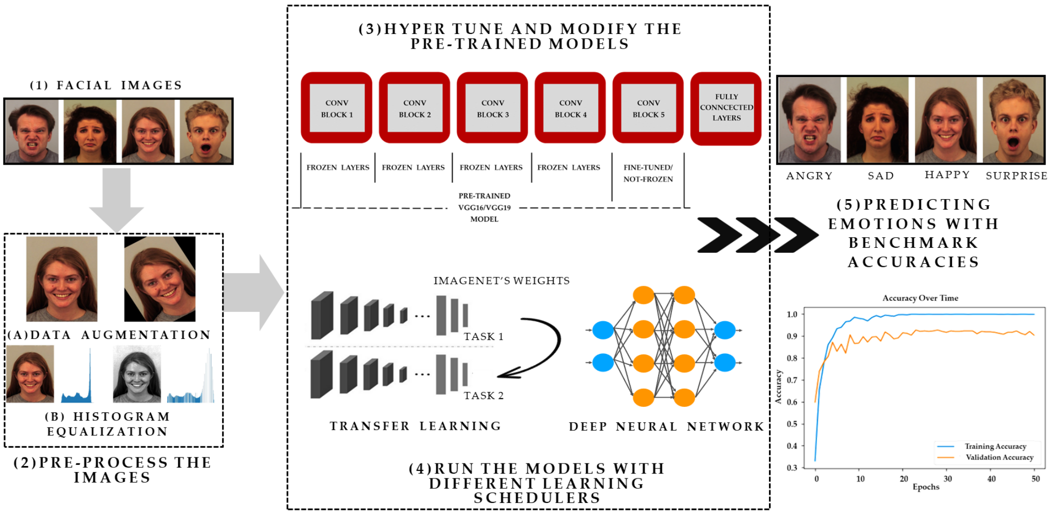

Figure 1. The overall workflow of the proposed framwork.

A significant contribution of this paper is on optimizing the performance of pre-trained simple architecture such as VGG on well-known FER image datasets. Instead of opting for complex deep neural network models, this study demonstrates that careful fine-tuning can lead to better classification accuracy with simpler architectures. This study investigates the efficacy of histogram equalization and data augmentation in improving FER accuracy, alongside optimizing the performance of pre-trained architectures like VGG on three benchmark FER datasets. Additionally, this paper showcases the effectiveness of different regularization techniques, callbacks, and learning schedulers in enhancing model performance for FER by conducting extensive experiments.

The subsequent sections of this paper are structured as follows: Section 2 presents a review of related literature in the field. Section 3 introduces histogram equalization and cosine annealing to aid in understanding the proposed models. Section 4 elaborates on the transfer learning-based model, datasets used, and the experiment pipeline. Section 5 presents the experimental results, while Section 6 provides a thorough discussion of these results. Finally, Section 7 concludes the paper by discussing its significance and outlining potential future work.

2. Related Works

Numerous studies conducted in recent years have focused on FER, employing various techniques. Traditional machine learning approaches have been used alongside CNN models to get important information for classifying emotions extracted from visual objects.

Xiao-Xu et al. [28] employed an ensemble approach using Wavelet Energy Feature (WEF) and Fisher's Linear Discriminants (FLD) for feature extraction and classification of seven facial expressions (anger, disgust, fear, happiness, normal, sadness, surprise) within the Japanese Female Facial Expression (JAFFE) dataset [29]. Abhinav Dhall et al. utilized the Pyramid of Histogram of Gradients (PHOG) [30] and Local Phase Quantization (LPQ) [31] features to encode shape and appearance information. They selected keyframes via K-means clustering [32] of normalized shape vectors from Constrained Local Models (CLM) based face tracking. Emotion classification on the SSPNET [33] and GEMEP-FERA [34] datasets was conducted using SVM and Largest Margin Nearest Neighbor (LMNN) [35]. Pu et al. proposed a framework employing two-fold RF classifiers to recognize Action Units (AUs) from image sequences. Facial motion measurement involved tracking Active Appearance Model (AAM) [36] facial feature points with Lucas-Kanade optical flow [37], using displacement vectors between neutral and peak expressions as motion features. These features were fed into a first-level RF for AU determination, followed by a second-level RF for facial expression classification [38]. Golzadeh et al. focused on spatio-temporal feature extraction based on tracked facial landmarks, aiming to develop an automatic emotion recognition system [39]. They employed the KDEF dataset to identify features that represent different human facial expressions, subsequently evaluating them through various classification methods. Through experimentation, employing K-fold cross-validation, they achieved precise recognition of facial expressions, attaining up to 87% accuracy with the newly devised features and a multiclass SVM classifier. Liew et al. propose five feature characteristics for FER and compare their performance using different classifiers and datasets (KDEF, CK+, JAFFE, and MUG [40]). Among Gaussian-based filtering and response (GABOR), Haar [41], LBP, and Histogram of oriented gradients (HOG) [42] classifiers, HOG performs best for FER with higher image resolutions (above 48x48 pixels), averaging 80% accuracy in these datasets [43].

The researchers found that the most straightforward method for classifying emotions is through CNN models. CNNs are well-suited for image tasks due to their ability to capture various levels of features efficiently and recognize patterns and objects in images regardless of their positions or sizes. Thakare et al. used several classifiers such as ConvNet, RF classifiers, and Extreme Gradient Boosting (XGBoost) classifiers [44], with the CNN model ConvNet consistently yielding the highest accuracy in emotion classification [45]. The researchers proposed a novel FER approach that integrates CNN with image edge detection to bypass traditional feature extraction. This method involves normalizing facial images, extracting edges, and merging this information with features to preserve structural composition. Subsequently, implicit features are reduced using maximum pooling, followed by softmax classification for emotion recognition. Testing on FER2013 [46] and LFW datasets [47] resulted in an average emotion detection rate of 88.56% with faster training, approximately 1.5 times quicker than comparative models [48]. Badrulhisham et al. focused on real-time FER, employing MobileNet [49] to train their model, achieving an 85% recognition accuracy for four emotions (happy, sad, surprise, disgust) on their custom dataset [50]. Puthanidam et al. propose a hybrid approach for facial expression recognition, integrating image pre-processing steps with diverse CNN architectures to enhance accuracy and reduce training time [51]. Experimental validation across multiple databases and facial orientations resulted in significant findings: achieving an accuracy of 89.58% on the KDEF dataset, 100% accuracy on the JAFFE dataset, and 71.975% accuracy on the combined dataset (KDEF + JAFFE + SFEW). These results were obtained using cross-validation techniques to minimize bias.

A significant amount of work has focused on employing transfer learning techniques with CNN models such as AlexNet [52], SqueezeNet [53], and VGG19, evaluating their efficacy on benchmark datasets including FER2013, JAFFE, KDEF, CK+, SFEW [54], and KMU-FED. VGG19 demonstrated notable performance, achieving 99.7% accuracy on the KMU-FED database and competitive results across other benchmark datasets. Specifically, VGG19 attained performance accuracies of 98.98% for the CK+ dataset, 92.99% for the KDEF dataset with all data variations, 91.5% for the selected KDEF FrontalView dataset, 84.38% for JAFFE, 66.58% for FER2013, and 56.02% for SFEW [55]. Additionally, some researchers have explored visual emotion recognition through social media images by employing pre-trained VGG19, ResNet50V2, and DenseNet-121 architectures as their base. Through fine-tuning and regularization techniques, these models demonstrated improved performance on Twitter images from the Crowdflower dataset, with DenseNet-121 exhibiting superior accuracies of 73%, 75%, and 89%, respectively [56]. Furthermore, Subudhiray et al. investigated dual transfer learning for facial emotion classification, experimenting with pre-trained CNN architectures including VGG16, ResNet50, Inception ResNet [57], Wide ResNet [58], and AlexNet. By combining extracted feature vectors into various pairs and inputting them into an SVM classifier, this approach showed promising results in terms of accuracy, kappa, and overall accuracy compared to state-of-the-art methods across benchmark datasets such as JAFFE, CK+, KDEF, and FER2013 [59]. Kaur et al. introduce FERFM, a novel approach using fine-tuned MobileNetV2 [60] for FER on mobile devices. A pipeline strategy was introduced, where the pre-trained MobileNetV2 architecture is fine-tuned by eliminating the last six layers and adding dropout, max pooling, and dense layer. Using transfer learning from ImageNet, the method achieves an accuracy of 85.7% in RGB-KDEF dataset. It surpasses VGG16 with faster processing at 43ms per image and fewer trainable parameters totaling 1,510,599. [61]. In another research, they proposed a system that employs a CNN framework using AlexNet's features, achieving higher accuracy compared to other methods across various datasets like JAFFE, KDEF, CK+, FER2013, and AffectNet [62]. Moreover, they proved it is more efficient and requires fewer device resources than other state-of-the-art deep learning models like VGG16, GoogleNet [63], and ResNet [64]. In another study, Zavarez et al. fine-tuned the VGG-Face Deep CNN model pre-trained for face recognition, the study investigates the impact of a cross-database approach [65]. Results reveal significant accuracy improvements, with average accuracies of 88.58%, 67.03%, 85.97%, and 72.55% on CK+, MMI, RaFD, and KDEF databases, respectively.

3. Methodology

In this section, we briefly review the histogram equalization technique and introduce the cosine annealing strategy and evaluation metrics used in this work.

3.1. Histogram Equalization

Histogram equalization is a technique used in image processing to enhance the contrast and improve the overall appearance of an image by redistributing the intensity values across the image. Dark and light regions in an image might not have optimal contrast, making details hard to perceive. Histogram equalization spreads out the intensity levels, making darker areas darker and brighter areas brighter.

The histogram of a digital image with intensity levels in the range [0, L - 1] is a discrete function $h(r_k) = n_k$, where L is the number of the level, r_k is the k^{th} intensity value and n_k is the number of pixels in the image with intensity r_k. It is common practice to normalize a histogram by dividing each of its components by the total number of pixels in the image, which is denoted by MxN, where M and N are the row and column dimensions of the image. A normalized histogram is given by:

$$p(r_k) = \frac{n_k}{MN} \text{ for } k = 0, 1, 2, ...L - 1 \tag{1}$$

$p(r_k)$ can be seen as an estimate of the probability of occurrence of intensity level r_k, in an image. Where,

$$\sum_{k=1}^{L-1} p(r_k) = 1 \tag{2}$$

Consider the continuous intensity values and let the variable r denote the intensities of an image. We assume that r is in the range [0, L- 1]. We focus on transformations (intensity mappings) of the form: s = T(r) where $0 \leq r \leq (L-1)$ that produces an output intensity level s for every pixel in the input image having intensity [66].

Figure 2 illustrates an image after histogram equalization. In the figure, 2(A) displays the original images from the KDEF dataset. Figure 2(B) depicts the corresponding histogram, illustrating the distribution of pixel intensities. Subsequently, 2(C) exhibits the image post-histogram equalization, a process aimed at enhancing contrast and brightness. Finally, 2(D) shows the resulting histogram, reflecting the altered intensity distribution after equalization. In Figures 2(B) and 2(D), the X-axis denotes the intensity values, while the Y-axis represents the pixel counts.

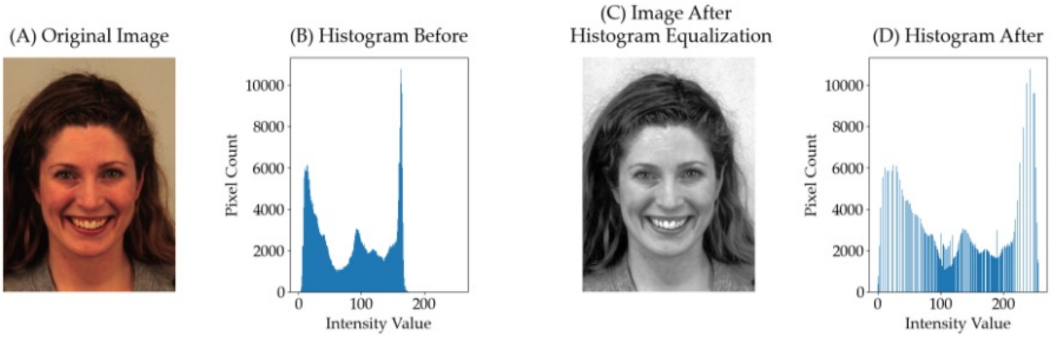

Figure 2. A Histogram Equalized Image from KDEF Dataset.

3.2. Cosine Annealing

The cosine annealing strategy is a learning rate adjustment strategy used to dynamically adjust the learning rate in optimization algorithms. Its main idea is to simulate the annealing process of the cosine function, periodically changing the learning rate during the training process, so that the model can better make fine adjustments in the later stage of training and improve convergence performance. The original formula for the cosine annealing strategy is as follows:

$$\eta_t = \frac{1}{2} \cdot (\eta_{\text{max}} - \eta_{\text{min}}) \cdot \left(1 + \cos\left(\frac{t}{T} \cdot \pi\right)\right) + \eta_{\text{min}} \tag{3}$$

Here, t in the numerator represents the total number of training cycles to the current stage, T in the denominator represents the set cosine annealing cycle, and η represents the learning rate at a given point during the training process. Based on the ratio between t and T, the learning rate exhibits a cosine-like variation.

The characteristic of the cosine annealing strategy is that the learning rate will undergo periodic changes within the range of maximum and minimum learning rates according to the annealing curve of the cosine function. This helps to make the model more stable in the later stages of training, avoiding oscillations or jumping out of local optima caused by the excessive learning rate, as well as situations where the learning rate is too small and the convergence speed is too slow [67].

3.3. Evaluation Matrices

Precision is the number of correctly predicted samples out of all predicted samples, which is a measure of classifier exactness. Recall is the number of correctly predicted samples out of the number of actual samples, which is a measure of classifier completeness. F-measure is a harmonized mean of

doi:10.20944/preprints202404.1540.v1

precision and recall, useful with uneven class distributions [68].

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{5}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{6}$$

$$\text{F-measure} = \frac{2\times\text{Recall}\times\text{Precision}}{\text{Recall}+\text{Precision}} \tag{7}$$

Where, True Positive (TP) is when a positive sample is correctly classified as positive, and True Negative (TN) is when a negative sample is correctly classified as negative. False Negative (FN) when a positive sample is classified as negative, and False Positive (FP) when a negative sample is classified as positive. AUC-ROC and AUC-PRC are performance metrics commonly used to evaluate the performance of binary classification models. The false positive rate (fpr) is calculated to determine this evaluation metrics. Here are the equations for AUC-ROC and AUC-PRC:

$$\text{AUC-ROC} = \int_0^1 \text{TPR}(fpr)\, d(fpr) \tag{8}$$

Here, TPR (fpr) represents the true positive rate (TPR) at a given false positive rate (fpr), and d (fpr) represents the differential element for fpr.

$$\text{AUC-PRC} = \int_0^1 \text{Precision}(recall)\, d(recall) \tag{9}$$

Precision (recall) represents the precision at a given recall value, and d (recall) represents the differential element for recall.

4. Implementation

In this section, we describe the datasets and the deep CNNs used in our work.

4.1. Datasets & Augmentation Techniques

KDEF

The KDEF dataset comprises 4900 colored images depicting various human facial emotions. Additionally, the Averaged KDEF (AKDEF) dataset consists of averaged images derived from the original KDEF photos. Both KDEF and AKDEF were built in 1998 and have since been made freely available to the academic community. Over the years, KDEF has become widely utilized in research, with over 1500 publications using its data. The KDEF dataset encompasses seven distinct emotion classes: *anger, neutral, disgust, fear, happy, sad, and surprise*. Each image in the dataset is carefully labeled to denote the specific emotion portrayed by the individual. The images are in RGB format with a resolution of 224x224 pixels.

CK+

The CK+ dataset [23] serves as a prominent benchmark dataset in the field of facial expression recognition research. It comprises a total of 981 images collected from 123 subjects, with each sequence depicting one of seven facial expressions: *anger, contempt, disgust, fear, happy, sadness, and surprise*. These expressions were elicited using the Facial Action Coding System (FACS), a standardized method for analyzing facial movements.

doi:10.20944/preprints202404.1540.v1

Each sequence within the CK+ dataset begins with a neutral expression, transitions to the target expression, and concludes with a return to the neutral expression. The images are captured under controlled laboratory conditions and are presented in a grayscale format, with a resolution of 640 by 490 pixels [69].

The Filtered FER2013

The original FER2013 dataset comprises 35,887 grayscale images, each depicting cropped faces with dimensions of 48 × 48 pixels. These images are categorized into seven emotions: *angry, disgust, fear, happy, neutral, sad, or surprise*.

One notable aspect of the FER2013 dataset is its class imbalance, where the number of images varies significantly across emotion categories. Despite this, the dataset captures a diverse range of facial expressions encountered in real-life scenarios, including variations in lighting conditions, camera distance, and facial poses. The individuals depicted in the images represent diverse demographics, encompassing differences in age, race, and gender. Additionally, the dataset exhibits variations in the intensity of expressed emotions.

To address issues such as non-class-associated photos or non-face images, a filtered version of the FER2013 dataset was created by Bialek et al. [22] This involved manual cleaning, removing images that did not correspond to any specific emotion category or were not depicting faces. Furthermore, instances of mislabeling were corrected, ensuring images were assigned to the appropriate emotion group.

Data Augmentation

We have used various data augmentation techniques to enhance the diversity of the training dataset of the CK+ and KDEF and improve the generalization performance of the models. During the preprocessing phase, we applied augmentation methods such as rotation, horizontal shifting, and vertical shifting to the input images. Specifically, we configured the *rotation_range* parameter to allow random rotations within a range of -20 to +20 degrees. Additionally, we used *width_shift_range* and *height_shift_range* to introduce random horizontal and vertical shifts to the images, respectively, with a maximum displacement of 20% of the total width and height. Furthermore, we enabled horizontal flipping using the *horizontal_flip* parameter to further increase dataset variability. To ensure seamless augmentation, we have used the 'nearest' fill mode to interpolate pixel values for newly created pixels. Lastly, we applied pixel normalization by rescaling the pixel values of all images to a range between 0 and 1 using the rescale parameter, aiding in the convergence of the model during training.

For the FER2013 dataset, we employed different augmentation techniques compared to the KDEF and CK+ datasets. Specifically, we applied a rotation range of 10 degrees clockwise or counterclockwise, along with horizontal flipping. Additionally, we utilized a zoom range of [1.1, 1.2], allowing for random zooming between 1.1x and 1.2x the original size during training.

Regarding the data split, we adopted an 80% training, 10% validation, and 10% test split for both the CK+ and KDEF datasets. For the KDEF dataset specifically, we use three facial postures instead of the original five, resulting in 420 images per class and it yields 2940 images in total. For the filtered FER2013 dataset, we preserved the identical data split as described by Bialek et al. [22], comprising training (27,310), validation (3,410), and test (3,420) sets. Table 1 presents the counts of training, testing, and validation samples for the datasets used in our experiment.

Table 1. Data split for each dataset.

Dataset	Training Size	Testing Size	Validation Size
KDEF	2350	294	294
CK+	784	99	98
Filtered FER2013	27310	3410	3420

4.2. Experimental Setup

This work was conducted within a Docker environment, using the NVIDIA RTX 2080 GPU on a Windows 10 Education 64-bit system. The system is equipped with 32 GB of RAM and an Intel Core i7-9700k 3.60 GHz CPU. The transfer learning model was developed and executed using the Keras Python library (https://keras.io/api/). Visualizations of the results were generated using the matplotlib library (https://matplotlib.org/) and the seaborn library (https://seaborn.pydata.org/). Additionally, the SciKit-learn library (https://scikit-learn.org/stable/about.html) was used to create evaluation matrices.

4.3. VGG Architectures

The task of FER was based on fine-tuned transfer learning from pre-trained VGG16 and VGG19 architectures. VGG, developed by Simonyan and Zisserman, achieved significant success as the runner-up in the ImageNet Large Scale Visual Recognition Competition (ILSVRC) in 2014. This architecture was chosen for several reasons: i) Demonstrable success on a variety of image classification tasks, ii) Native support provided by Keras 5 which offers pre-trained models with publicly available weights, iii) and, simpler implementation.

However, more complex models such as ResNet50 or DenseNet121 might face the problem of overfitting issues given a smaller dataset size. We employed VGG16 and VGG19, pre-trained on the ImageNet dataset, for classification purposes. In our approach, we unfroze the last 4 layers of VGG16 and the last 5 layers of VGG19, while freezing the remaining layers. This strategy allowed us to update the pre-trained ImageNet [70] weights with the weights learned from our specific datasets, enhancing the model's ability to learn more effectively.

Following the basic VGG architecture, we added a global average pooling layer, a dropout layer, a dense layer with ReLU activation, another dropout layer, and finally a dense layer with softmax activation to classify the emotions. The proposed VGG16 and VGG19 model architecture is given in Table 2 and 3 respectively.

Table 2. The proposed modified VGG16.

Layer (type)	Output Shape	Trainable Parameters	Non-Trainable Parameters
	VGG16 Base Model*	7,635,264	7,079,424
Global AveragePooling 2D	(None, 512)	0	0
Dropout	(None, 512)	0	0
(Dense+ReLU)	(None, 1024)	525312	0
Dropout	(None, 1024)	0	0
(Dense+Softmax)	(None, 7)	7175	0
Trainable Parameters		7611911	0
Non-trainable Parameters		0	7635264
Total Parameters			15247175

*The layers of the VGG16 model, excluding the last four, were frozen.

For comparison purposes, our experiment is structured in three distinct setups. Firstly, we maintain all layers of the base VGG19 and VGG16 models frozen, without applying any histogram equalization. Secondly, we fine-tune the VGG19 and VGG16 architectures by unfreezing the last 4 layers of VGG16 and the last 5 layers of VGG19, while keeping the remaining layers frozen. This setup also does not include any histogram equalization. Lastly, we incorporate histogram equalization into the final experimental setup, along with fine-tuning the models as described in the second setup. A general overview of our framework can be found in Figure 1.

Table 3. The proposed modified VGG19.

Layer (type)	**Output Shape**	**Trainable Parameters**	**Non-Trainable Parameters**
	VGG19 Base Model*	10,585,152	9,439,232
Global AveragePooling 2D	(None, 512)	0	0
Dropout	(None, 512)	0	0
(Dense+ReLU)	(None, 1024)	525312	0
Dropout	(None, 1024)	0	0
(Dense+Softmax)	(None, 7)	7175	0
Trainable Parameters		9,971,719	0
Non-trainable Parameters		0	10,585,152
Total Parameters			20,556,871

* The layers of the VGG19 model, excluding the last five, were frozen.

4.4. Hyper Parameters

For training our models on different datasets, we have used various hyperparameters to optimize performance. These hyperparameters include Input Size, Batch Size, Epochs, Learning Rate, Early Stopping, Learning Rate Scheduler, Dropout Rate, and L2 Regularization. The specifics of these parameters for different datasets are outlined in Table 4.

Table 4. Hyper parameters for the models in different datasets.

Hyper parameters	**CK+**	**KDEF**	**FER2013**
Input Size	**(224,224,3)** (48,48,3) (144,144,3)	**(224,224,3)** (48,48,3) (144,144,3)	(224,224,3) (48,48,3) **(144,144,3)**
Batch Size	16, **32**,64	16, **32**,64	16, 32,**64**
Epochs	300	300	30
Learning Rate	0.01,0.001,**0.0001**	10.01,0.001,**0.0001**	0.01,0.001,**0.0001**
Early Stop	Monitor Validation Accuracy Patience=5	Monitor Validation Accuracy Patience=5	Monitor Validation Accuracy Patience=5
Learning Rate Scheduler	-	Monitor Validation Accuracy Patience=3, Factor=0.5	Cosine Annealing
Dropout Rate	0.1,0.3,**0.5**	**0.1**,0.3,0.5	**0.1**,0.3,0.5
L2 Regularization	-	**0.01**,0.1,0.2	0.01,**0.1**,0.2

* Bold parameters are used in final models.

In the CK+ dataset, the original images were re-sized at (224, 224, 3). We applied a dropout rate of 0.5 for regularization purposes. Additionally, we implemented early stopping, which halts the training process if the validation accuracy does not improve for 5 consecutive epochs, to prevent from overfitting.

In the KDEF dataset, we employed the same data augmentation and resizing techniques. Since the images were in RGB format, we converted them to grayscale to utilize histogram equalization. After equalization, we converted them back to RGB format. We also utilized early stopping and a learning rate scheduler. If the validation accuracy did not improve for three consecutive epochs, the learning rate was multiplied by 0.5 to decrease it. If the accuracy did not increase for five consecutive epochs, the training process was stopped. The dropout rate for this dataset was set to 0.1, and l2 kernel regularization with a value of 0.01 was applied for optimal performance.

For the FER2013 dataset, we conducted extensive experimentation with various learning strategies and optimizers. Through our analysis, we discovered that employing cosine annealing with the adam optimizer yielded the most accurate results. Additionally, we explored different image sizes and were surprised to find that transfer learning models performed exceptionally well when the images were resized to (144,144,3). In terms of batch size, we used the value of 64, which differed from the batch sizes used in the other datasets. Furthermore, we also found that the optimal dropout rate and l2 regularization penalty were 0.1 for this dataset. Regarding model training strategies, we initially employed a similar approach to that used in the CK+ dataset, wherein the model training would stop if the validation accuracy did not improve for five consecutive epochs. Subsequently, we saved the models and proceeded to train them again using cosine annealing for an additional 30 epochs. The initial learning rate was set to 0.0001, and then it gradually decreased according to the equation specified in equation 3. Afterward, the model is trained using this gradually decreasing learning rate.

5. Results

In this section, an in-depth discussion of the performance of the the VGG models on each of the datasets is given followed by an overall performance comparison on all three datasets in Table 7.

5.1. KDEF

For the KDEF dataset, specifically, the hyper-tuned VGG19 model with histogram equalization achieved a commendable accuracy of 95.92%, while the fine-tuned VGG19 model without histogram lagged by just 1.7%. Similar trends were also observed in the VGG16 architectures, where histogram equalization helped the fine-tuned models achieve slightly better results. On the other hand, the models with all layers frozen and without histogram equalization, demonstrated inferior performance.

However, the most significant results are highlighted in Table 5, which presents class-wise evaluation metrics for our best model, the fine-tuned VGG19 model with histogram equalization. Notably, the highest scores were observed in the happy class, with precision, recall, and F1-score all reaching 100%. Furthermore, our models performed exceptionally when predicting classes with actual emotions (Figure 3). In both scenarios, the models with the highest accuracies showcased the correct classification of emotions.

Table 5. Performance of fine-tuned VGG19 on different classes in the KDEF dataset.

Class	Precision	Recall	F1-score
Angry	0.95	0.93	0.94
Disgust	0.94	0.94	0.94
Fear	1.00	0.90	0.95
Happy	1.00	1.00	1.00
Neutral	0.95	0.95	0.95
Sad	0.95	1.00	0.97
Surprise	0.93	1.00	0.97

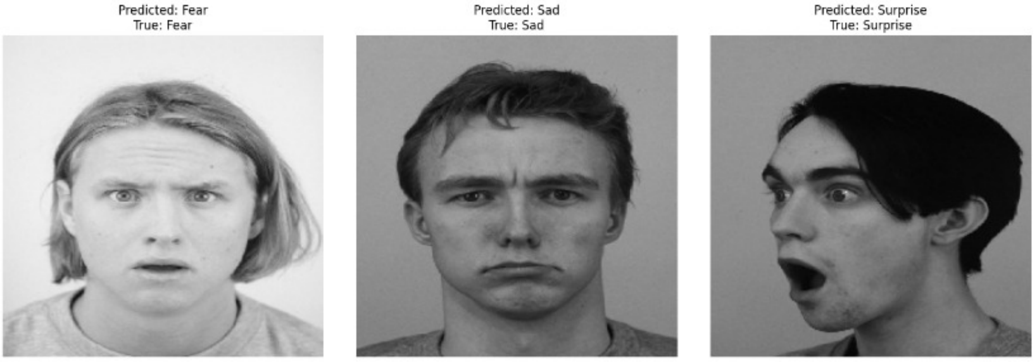

Figure 3. Predictions of VGG19 model for the KDEF Dataset.

5.2. CK+

For the CK+ dataset, our models have achieved the highest accuracies among various architectures. Specifically, the fine-tuned VGG16 and VGG19 models, both with and without histogram equalization, demonstrate exceptional performance with an accuracy of 100% and 98.99%, repectively (Table 7). Additionally, these models exhibit strong performance in terms of weighted F1, AUC-ROC, and AUC-PRC metrics. VGG16 and VGG19 models, where the layers of the base models are frozen and not fine-tuned, exhibit lower performance across all evaluation metrics, showing 96.97% and 90.91% respectively.

To gain insights into the inner workings of our CNN models, we have employed Grad-CAM (Gradient-weighted Class Activation Mapping) [71]. Grad-CAM is a visualization technique that highlights important regions of an input image used by CNN for classification. By examining the Grad-CAM visualizations (Figure 4), we observe that our models are effectively focusing on crucial areas (red marked areas) where important features are located. This suggests that our models are making informed decisions based on relevant image features.

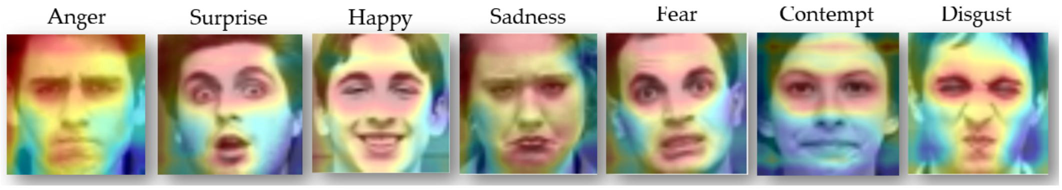

Figure 4. GradCam from CK+ Dataset.

Furthermore, the superiority of our models can be visualized when we are examining the confusion matrix. The confusion matrix (Figure 5) reveals zero misclassification rate on the CK+ datasets.

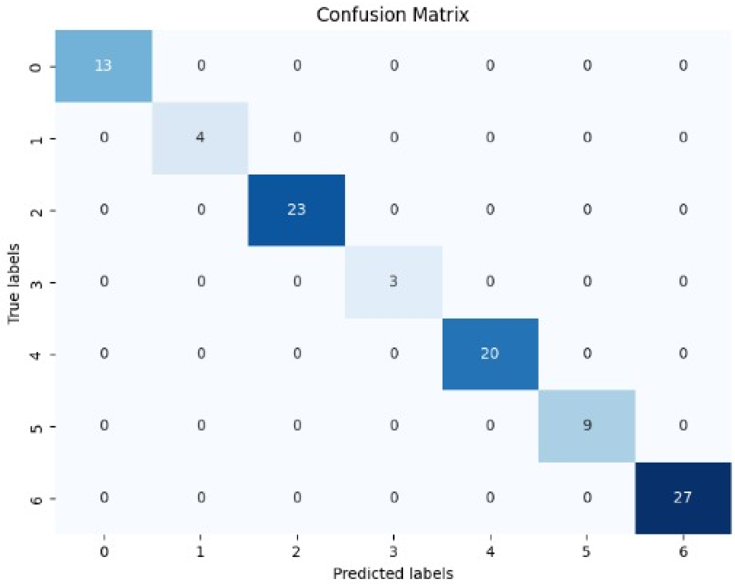

Figure 5. Confusion Matrix for Fine-tuned VGG16 Model with Histogram Equalization on CK+ Dataset.

5.3. Filtered FER2013

In the case of the Filtered FER2013 dataset, the VGG16 model with histogram equalization outperforms other models, achieving an accuracy of almost 69.65%. The performance difference was marginal in the fine-tuned VGG19 models with and without histogram, which achieved accuracies of 69.44% and 60.06%, respectively.

For instance, when we applied cosine annealing to the fine-tuned VGG16 models with histogram, the model's accuracy initially plateaued at around 67.57%. However, after reloading the model and running the algorithm for an additional 30 epochs with cosine annealing, the accuracy improved to 69.65%. This adaptive learning rate scheduling of cosine annealing allowed the model to escape local minima and explore the parameter space more effectively, ultimately leading to improved performance. The effectiveness of the models is further highlighted in Table 6, where per-class accuracy demonstrates how well the model performs across different classes with the FER2013 dataset.

Table 6. Performance of fine-tuned VGG16 model with histogram on different classes in the FER2013 dataset.

Class	Precision	Recall	F1-score
Angry	0.62	0.62	0.62
Disgust	0.81	0.71	0.76
Fear	0.58	0.51	0.54
Happy	0.87	0.88	0.87
Neutral	0.63	0.66	0.64
Sad	0.57	0.61	0.59
Surprise	0.81	0.80	0.80

Table 7. Performance comparison of different models on various datasets.

Model	Dataset	Accuracy (%)	Weighted-F1 (%)	AUC-ROC (%)	AUC-PRC (%)
VGG19 with all layer freezed + No Histogram	CK+	90.91	91.00	99.00	96.00
Finetuned VGG19 + No Histogram	CK+	97.98	98.00	100.00	99.00
Finetuned VGG19 + Histogram	CK+	98.99	99.00	100.00	100.00
VGG16 with all layer freezed + No Histogram	CK+	96.97	97.00	100.00	99.00
Finetuned VGG16 + No Histogram	CK+	97.98	98.00	100.00	100.00
Finetuned VGG16 + Histogram	CK+	**100**	100.00	100.00	100.00
VGG19 with all layer freezed + No Histogram	KDEF	54.76	53.93	86.65	59.29
Finetuned VGG19 + No Histogram	KDEF	94.22	94.20	99.64	98.25
Finetuned VGG19 + Histogram	KDEF	**95.92**	95.90	99.60	98.53
VGG16 with all layer freezed and + No Histogram	KDEF	56.80	56.62	88.97	62.80
Finetuned VGG16 + No Histogram	KDEF	92.18	92.12	99.62	98.08
Finetuned VGG16 + Histogram	KDEF	92.86	92.87	99.69	98.34
VGG19 with all layer freezed + No Histogram	FER2013	35.99	29	57.70	19.73
Finetuned VGG19 + No Histogram	FER2013	69.06	68.57	80.87	52.26
Finetuned VGG19 + Histogram	FER2013	69.44	68.34	80.20	51.58
VGG16 with all + freezed with No Histogram	FER2013	41.20	35.53	60.36	22.22
Finetuned VGG16 + No Histogram	FER2013	68.8	69.29	81.34	52.37
Finetuned VGG16 + Histogram	FER2013	**69.65**	69.65	80.75	51.83

5.4. Comparison of Methods

The comparison of methods primarily focused on the accuracy metric, as most other researchers did not employ additional metrics like the AUC-ROC or AUC-PRC. The comparison of methods based on the KDEF, CK+, and FER2013 is depicted in the Table 8.

Our model demonstrates exceptional performance on the KDEF dataset compared to state-of-the-art works. The closest contender is the study by Sahoo et al., achieving nearly 93% accuracy using a transfer learning model based on VGG19. On the CK+ dataset, our model also showcases supremacy. Although the work of Dar et al. achieved the same accuracy, their model was notably complex, with each convolutional block comprising approximately 18 layers.

Table 8. Comparison of methods.

Literature	Dataset	Type	Accuracy
Puthanidam [51]	KDEF	Hybrid CNN	89.58%
Chen et al. [72]	KDEF, CK+, FER2013	IACNN	67%, 95%, 68%
Liu et al. [73]	KDEF, CK+, FER2013	2B (N + M)Softmax	81%,87%,67%
Dar et al. [74]	KDEF, CK+, FER2013	Efficient-SwishNet	88.3%, 100%, 63.4%
Zahara et al. [75]	FER2013	Xception	65.97%
Minaaee et al. [76]	CK+, FER2013	CNN+Attention	98%, 70.02%
Fei et al. [77]	KDEF, CK+,FER2013	MobileNet + SVM	86.4%, 89.8%, 51.7%
Mahesh et al. [78]	KDEF	Feed Forward Network	88.7%
Sahoo et al. [55]	KDEF, CK+, FER2013	Pre-trained VGG19	93%,98.98%, 66.6%
Bialek et al. [22]	FER2013	4 Ensemble Model	75.06%
Proposed Model	KDEF, CK+, FER2013	Histogram + Pretrained VGG16	**95.92%****, **100%***, **69.65%***

* Fine-tuned VGG16 with histogram equalization. ** Fine-tuned VGG19 with histogram equalization.

However, the discussion primarily centers around the FER2013 dataset, where our model exhibits relatively good performance. We utilized the filtered FER2013 dataset from the work of Bialek et al., where their single 5-layer model achieved approximately 70.09% accuracy. Despite attempting to implement the same model on the filtered FER2013 dataset, we achieved only 68.27% accuracy. Therefore, we implemented our own pipeline, achieving an accuracy of almost 69.65%. Moreover, despite employing a simpler model based on pre-trained VGG16, our model compares favorably with other works that utilize more complex architectures. Notably, our model takes less time because of the lower parameters that it is using.

6. Discussion

This section of our study highlights key notations and compares our models with existing works in the field. We delve into specific notations crucial for understanding our approach and provide a thorough comparison of our models with those previously established in the literature.

The comparison between models with all layers frozen with no histogram and those with fine-tuned models with no histogram reveals significant differences in performance across datasets. For instance, in the KDEF dataset, the VGG19 with all layer freeze + No histogram achieved an accuracy of 54.76%, whereas the Finetuned VGG19 + No Histogram achieved nearly 94.22%. This variation underscores the limitations of freezing all layers, as it limits the model's adaptability to the new task/domain by preserving fixed feature extraction mechanisms. Oppositely, unfreezing the last five layers allows for fine-tuning, enabling the model to learn task-specific representations and enhance performance by using pre-trained weights while accommodating adjustments to suit the new task requirements.

Furthermore, we proceed to compare the models' performance post-histogram equalization. Notably, all hyper-tuned models exhibited slightly improved performance with histogram equalization. As indicated in Table 7, hyper-tuned models using histogram equalized images showed a 0.3-2% enhancement on average compared to their counterparts without histogram equalization. However, exceptions were observed in the CK+ dataset, where both hyper-tuned models, with and without histogram equalization, exhibited similar performance.

In the case of the Filtered FER2013 dataset, one interesting observation worth discussing is the performance of the FER2013 dataset with image size set to 144x144. Despite the original VGG16 and VGG19 models being trained on the 224x224x3 ImageNet dataset, the 144x144 image size showed promising results across all experiments. This deviation from the conventional image size might be attributed to the potential loss of information and degradation of image quality when resizing an image to larger dimensions. When resizing an image to a larger size, the existing pixels are stretched to fill the new dimensions, which can lead to blurriness or pixelation. This loss of information may have been mitigated by selecting an optimum size of 144x144, which preserved the originality of the image while maintaining a balance between image quality and resolution.

When handling a small dataset like CK+, training models like VGG16 and VGG19 can be prone to overfitting due to their extensive parameter count. However, employing regularization techniques and training techniques, coupled with data augmentation, aids in effectively training the model with this limited data. This effectiveness can be seen in Figure 4, where the model effectively identifies crucial features in facial data across various classes, showcasing its robustness even with a constrained dataset.

7. Conclusion

This paper presents a thorough comparative analysis of FER methods using CNN architectures via extensive experiments and prediction metrics such as AUC-ROC, AUC-PRC, and F1 scores for each model.. Comparing these methodologies, offers a nuanced exploration of their strengths and limitations, shedding light on areas for improvement and potential avenues for further research. This paper demonstrates that the application of histogram equalization along with data augmentation can improve accuracy in recognizing facial emotions. Another significant contribution of this paper is in optimizing the performance of pre-trained simple architectures like VGG across various dataset sizes by employing different regularization techniques, callbacks, and learning scheduler techniques on selected datasets. Instead of opting for complex models, this study demonstrates that careful fine-tuning can lead to better classification accuracy with simpler architectures.

The primary limitation of our study lies in the simplistic architecture used. Compared to the methodologies employed by most studies outlined in Table 8, our approach is notably less complex. Additionally, for the FER2013 dataset, the potential incorporation of auxiliary data could enhance model performance. Currently, facial landmark recognition and attention mechanisms are used widely in facial recognition problems. Integrating these techniques could prove beneficial for improving the performance of our models in future endeavors.

Author Contributions: The conceptualization and methodology were developed by JC, QL, and SR. JC handled data curation, software development, and validation, as well as taking the lead in preparing the original draft for writing. QL and SR conducted the review and editing of the writing, and they also provided supervision throughout the experiment. Funding for the resources utilized in the experiments were provided by SR.

Data Availability Statement: The research presented in this article was performed using the publicly available image dataset KDEF [21], CK+ [23] , and the version of FER2013 that was derived from the paper 'Efficient Approach to Face Emotion Recognition with Convolutional Neural Networks' [22].

Acknowledgments: S. Ramanna's research was supported by the NSERC Discovery Grant #194376.

Conflicts of Interest: The manuscript is approved for publication by all authors. The work described is original research that has not been published previously and is not under consideration for publication elsewhere.

References

1. Ekman, P. Cross-cultural studies of facial expression. *Darwin and facial expression: A century of research in review* **1973**, *169222*.
2. Ramsay, R.W. Speech patterns and personality. *Language and Speech* **1968**, *11*, 54–63.
3. Fast, J. *Body language*; Vol. 82348, Simon and Schuster, 1970.
4. Newmark, C. Charles Darwin: the expression of the emotions in man and animals. In *Schlüsselwerke der Emotionssoziologie*; Springer, 2022; pp. 111–115.
5. Ragsdale, J.W.; Van Deusen, R.; Rubio, D.; Spagnoletti, C. Recognizing patients' emotions: teaching health care providers to interpret facial expressions. *Academic Medicine* **2016**, *91*, 1270–1275.
6. Suhaimi, N.S.; Mountstephens, J.; Teo, J. EEG-Based Emotion Recognition: A State-of-the-Art Review of Current Trends and Opportunities. *Computational Intelligence and Neuroscience* **2020**, *2020*.

7. Fernández-Caballero, A.; Martínez-Rodrigo, A.; Pastor, J.M.; Castillo, J.C.; Lozano-Monasor, E.; López, M.T.; Zangróniz, R.; Latorre, J.M.; Fernández-Sotos, A. Smart environment architecture for emotion detection and regulation. *Journal of biomedical informatics* **2016**, *64*, 55–73.
8. Mattavelli, G.; Pisoni, A.; Casarotti, A.; Comi, A.; Sera, G.; Riva, M.; Bizzi, A.; Rossi, M.; Bello, L.; Papagno, C. Consequences of brain tumour resection on emotion recognition. *Journal of Neuropsychology* **2019**, *13*, 1–21.
9. Suja, P.; Tripathi, S.; others. Real-time emotion recognition from facial images using Raspberry Pi II. 2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN). IEEE, 2016, pp. 666–670.
10. Ojala, T.; Pietikäinen, M.; Harwood, D. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition* **1996**, *29*, 51–59.
11. Jolliffe, I.T.; Cadima, J. Principal component analysis: a review and recent developments. *Philosophical transactions of the royal society A: Mathematical, Physical and Engineering Sciences* **2016**, *374*, 20150202.
12. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *Journal of machine learning research* **2008**, *9*.
13. Cortes, C.; Vapnik, V. Support-vector networks. *Machine learning* **1995**, *20*, 273–297.
14. Breiman, L. Random forests. *Machine learning* **2001**, *45*, 5–32.
15. Payal, P.; Goyani, M.M. A comprehensive study on face recognition: methods and challenges. *The Imaging Science Journal* **2020**, *68*, 114–127.
16. O'shea, K.; Nash, R. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458* **2015**.
17. Hochreiter, S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* **1998**, *6*, 107–116.
18. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* **2014**.
19. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *CoRR* **2015**, *abs/1512.03385*, [1512.03385].
20. Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks, 2018, [arXiv:cs.CV/1608.06993].
21. Lundqvist, D.; Flykt, A.; Öhman, A. Karolinska directed emotional faces. *PsycTESTS Dataset* **1998**, *91*, 630.
22. Białek, C.; Matiolański, A.; Grega, M. An Efficient Approach to Face Emotion Recognition with Convolutional Neural Networks. *Electronics* **2023**, *12*, 2707.
23. Lucey, P.; Cohn, J.F.; Kanade, T.; Saragih, J.; Ambadar, Z.; Matthews, I. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. 2010 ieee computer society conference on computer vision and pattern recognition-workshops. IEEE, 2010, pp. 94–101.
24. Xie, Y.; Ning, L.; Wang, M.; Li, C. Image Enhancement Based on Histogram Equalization. *Journal of Physics: Conference Series* **2019**, *1314*, 012161. doi:10.1088/1742-6596/1314/1/012161.
25. Gotmare, A.; Keskar, N.S.; Xiong, C.; Socher, R. A closer look at deep learning heuristics: Learning rate restarts, warmup and distillation. *arXiv preprint arXiv:1810.13243* **2018**.
26. Hanley, J.A.; McNeil, B.J. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology* **1982**, *143*, 29–36.
27. Powers, D.M. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061* **2020**.
28. Xiao-Xu, Q.; Wei, J. Application of wavelet energy feature in facial expression recognition. 2007 International Workshop on Anti-Counterfeiting, Security and Identification (ASID). IEEE, 2007, pp. 169–174.
29. Lyons, M.; Kamachi, M.; Gyoba, J. The Japanese female facial expression (JAFFE) dataset. *The images are provided at no cost for non-commercial scientific research only. If you agree to the conditions listed below, you may request access to download* **1998**.
30. Tyagi, M. Hog (histogram of oriented gradients): An overview. *Towards Data Science* **2021**, *4*.
31. Ahonen, T.; Rahtu, E.; Ojansivu, V.; Heikkila, J. Recognition of blurred faces using local phase quantization. 2008 19th international conference on pattern recognition. IEEE, 2008, pp. 1–4.
32. Lloyd, S. Least squares quantization in PCM. *IEEE transactions on information theory* **1982**, *28*, 129–137.
33. Lee, H.; Kim, S. SSPNet: Learning spatiotemporal saliency prediction networks for visual tracking. *Information Sciences* **2021**, *575*, 399–416.

34. Yang, S.; Bhanu, B. Facial expression recognition using emotion avatar image. 2011 IEEE International Conference on Automatic Face & Gesture Recognition (FG). IEEE, 2011, pp. 866–871.
35. Dhall, A.; Asthana, A.; Goecke, R.; Gedeon, T. Emotion recognition using PHOG and LPQ features. 2011 IEEE International Conference on Automatic Face & Gesture Recognition (FG). IEEE, 2011, pp. 878–883.
36. Cootes, T.F.; Edwards, G.J.; Taylor, C.J. Active appearance models. Computer Vision—ECCV'98: 5th European Conference on Computer Vision Freiburg, Germany, June 2–6, 1998 Proceedings, Volume II 5. Springer, 1998, pp. 484–498.
37. Sharmin, N.; Brad, R. Optimal filter estimation for Lucas-Kanade optical flow. *Sensors* **2012**, *12*, 12694–12709.
38. Pu, X.; Fan, K.; Chen, X.; Ji, L.; Zhou, Z. Facial expression recognition from image sequences using twofold random forest classifier. *Neurocomputing* **2015**, *168*, 1173–1180.
39. Golzadeh, H.; Faria, D.R.; Manso, L.J.; Ekárt, A.; Buckingham, C.D. Emotion recognition using spatiotemporal features from facial expression landmarks. 2018 International Conference on Intelligent Systems (IS). IEEE, 2018, pp. 789–794.
40. Aifanti, N.; Papachristou, C.; Delopoulos, A. The MUG facial expression database. 11th International Workshop on Image Analysis for Multimedia Interactive Services WIAMIS 10. IEEE, 2010, pp. 1–4.
41. Viola, P.; Jones, M. Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2001, Vol. 1, pp. I–I. doi:10.1109/CVPR.2001.990517.
42. Freeman, W.T.; Roth, M. Orientation histograms for hand gesture recognition. International workshop on automatic face and gesture recognition. Citeseer, 1995, Vol. 12, pp. 296–301.
43. Liew, C.F.; Yairi, T. Facial expression recognition and analysis: a comparison study of feature descriptors. *IPSJ transactions on computer vision and applications* **2015**, *7*, 104–120.
44. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.
45. Thakare, C.; Chaurasia, N.K.; Rathod, D.; Joshi, G.; Gudadhe, S. Comparative analysis of emotion recognition system. *Int. Res. J. Eng. Technol.* **2019**, *6*, 380–384.
46. Goodfellow, I.J.; Erhan, D.; Carrier, P.L.; Courville, A.; Mirza, M.; Hamner, B.; Cukierski, W.; Tang, Y.; Thaler, D.; Lee, D.H.; others. Challenges in representation learning: A report on three machine learning contests. Neural Information Processing: 20th International Conference, ICONIP 2013, Daegu, Korea, November 3-7, 2013. Proceedings, Part III 20. Springer, 2013, pp. 117–124.
47. Jalal, A.; Tariq, U. The LFW-gender dataset. Computer Vision–ACCV 2016 Workshops: ACCV 2016 International Workshops, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part III 13. Springer, 2017, pp. 531–540.
48. Zhang, W.; He, X.; Lu, W. Exploring discriminative representations for image emotion recognition with CNNs. *IEEE Transactions on Multimedia* **2019**, *22*, 515–523.
49. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861* **2017**.
50. Badrulhisham, N.A.S.; Mangshor, N.N.A. Emotion Recognition Using Convolutional Neural Network (CNN). *Journal of Physics: Conference Series* **2021**, *1962*, 012040. doi:10.1088/1742-6596/1962/1/012040.
51. Puthanidam, R.V.; Moh, T.S. A hybrid approach for facial expression recognition. Proceedings of the 12th International Conference on Ubiquitous Information Management and Communication, 2018, pp. 1–8.
52. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Communications of the ACM* **2017**, *60*, 84–90.
53. Iandola, F.N.; Han, S.; Moskewicz, M.W.; Ashraf, K.; Dally, W.J.; Keutzer, K. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size, 2016, [arXiv:cs.CV/1602.07360].
54. Dhall, A.; Goecke, R.; Lucey, S.; Gedeon, T. Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. 2011 IEEE international conference on computer vision workshops (ICCV workshops). IEEE, 2011, pp. 2106–2112.
55. Sahoo, G.K.; Das, S.K.; Singh, P. Performance Comparison of Facial Emotion Recognition: A Transfer Learning-Based Driver Assistance Framework for In-Vehicle Applications. *Circuits, Systems, and Signal Processing* **2023**, pp. 1–28.

56. Chandrasekaran, G.; Antoanela, N.; Andrei, G.; Monica, C.; Hemanth, J. Visual sentiment analysis using deep learning models with social media data. *Applied Sciences* **2022**, *12*, 1030.
57. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-ResNet and the impact of residual connections on learning. Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. AAAI Press, 2017, AAAI'17, p. 4278–4284.
58. Zagoruyko, S.; Komodakis, N. Wide residual networks. *arXiv preprint arXiv:1605.07146* **2016**.
59. Subudhiray, S.; Palo, H.K.; Das, N. Effective recognition of facial emotions using dual transfer learned feature vectors and support vector machine. *International Journal of Information Technology* **2023**, *15*, 301–313.
60. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.
61. Kaur, S.; Kulkarni, N. FERFM: An Enhanced Facial Emotion Recognition System Using Fine-tuned MobileNetV2 Architecture. *IETE Journal of Research* **2023**, pp. 1–15.
62. Mollahosseini, A.; Hasani, B.; Mahoor, M.H. Affectnet: A database for facial expression, valence, and arousal computing in the wild. *IEEE Transactions on Affective Computing* **2017**, *10*, 18–31.
63. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1–9.
64. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
65. Zavarez, M.V.; Berriel, R.F.; Oliveira-Santos, T. Cross-Database Facial Expression Recognition Based on Fine-Tuned Deep Convolutional Network. 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), 2017, pp. 405–412. doi:10.1109/SIBGRAPI.2017.60.
66. Gonzalez, R.C.; Woods, R.E. *Digital Image Processing (3rd Edition)*; Prentice-Hall, Inc.: USA, 2006.
67. Zhang, C.; Shao, Y.; Sun, H.; Xing, L.; Zhao, Q.; Zhang, L. The WuC-Adam algorithm based on joint improvement of Warmup and cosine annealing algorithms. *Mathematical Biosciences and Engineering* **2024**, *21*, 1270–1285.
68. Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; Witten, I.H. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter* **2009**, *11*, 10–18.
69. Barhoumi, C.; Ayed, Y.B. Unlocking the Potential of Deep Learning and Filter Gabor for Facial Emotion Recognition. International Conference on Computational Collective Intelligence. Springer, 2023, pp. 97–110.
70. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255. doi:10.1109/CVPR.2009.5206848.
71. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *International Journal of Computer Vision* **2019**, *128*, 336–359. doi:10.1007/s11263-019-01228-7.
72. Chen, Y.; Liu, Z.; Wang, X.; Xue, S.; Yu, J.; Ju, Z. Combating Label Ambiguity with Smooth Learning for Facial Expression Recognition. International Conference on Intelligent Robotics and Applications. Springer, 2023, pp. 127–136.
73. Liu, X.; Vijaya Kumar, B.; You, J.; Jia, P. Adaptive deep metric learning for identity-aware facial expression recognition. Proceedings of the IEEE conference on computer vision and pattern recognition workshops, 2017, pp. 20–29.
74. Dar, T.; Javed, A.; Bourouis, S.; Hussein, H.S.; Alshazly, H. Efficient-SwishNet based system for facial emotion recognition. *IEEE Access* **2022**, *10*, 71311–71328.
75. Zahara, L.; Musa, P.; Wibowo, E.P.; Karim, I.; Musa, S.B. The facial emotion recognition (FER-2013) dataset for prediction system of micro-expressions face using the convolutional neural network (CNN) algorithm based Raspberry Pi. 2020 Fifth international conference on informatics and computing (ICIC). IEEE, 2020, pp. 1–9.
76. Minaee, S.; Minaei, M.; Abdolrashidi, A. Deep-emotion: Facial expression recognition using attentional convolutional network. *Sensors* **2021**, *21*, 3046.

77. Fei, Z.; Yang, E.; Yu, L.; Li, X.; Zhou, H.; Zhou, W. A novel deep neural network-based emotion analysis system for automatic detection of mild cognitive impairment in the elderly. *Neurocomputing* **2022**, *468*, 306–316.
78. Mahesh, V.G.; Chen, C.; Rajangam, V.; Raj, A.N.J.; Krishnan, P.T. Shape and texture aware facial expression recognition using spatial pyramid Zernike moments and law's textures feature set. *IEEE Access* **2021**, *9*, 52509–52522.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.