IJWR

A Scalable Method for Real-Time Facial Emotion Recognition using an Artificial Neural Network and Polynomial Equation

Omid Ghadami, Alireza Rezvanian*
Department of Computer Engineering, University of Science and Culture, Tehran, Iran; omidghadami@stu.usc.ac.ir, rezvanian@usc.ac.ir

ABSTRACT

Facial emotion recognition has recently attracted considerable interest due to its wide range of applications. It plays a crucial role in supporting individuals with autism spectrum disorders and improving interactions between humans and computers. The ability to execute these applications in real-time is essential. The architecture of the model and the computational resources available are the key determinants of inference time. Consequently, the development of a real-time solution requires a concentrated effort on these elements. In this paper, we present a scalable approach that utilizes EfficientNetV2, chosen for its operational efficiency. Our methodology involves resolution scaling based on a polynomial equation, which ensures real-time performance across various computational resources and model configurations. This scalable technique employs a polynomial equation to identify the optimal resolution for designated inference times, specifically adapted to our hardware and model specifications. By implementing the polynomial equation for resolution scaling, we created two variants of EfficientNetV2. Our findings from the KDEF dataset indicate that the proposed EfficientNetV2 can accurately classify images in real time on our hardware.

Keywords*— Real-time facial emotion recognition, Deep Learning, EfficientNetV2, Imbalanced Datasets, Resolution scaling.*

1. Introduction

Emotion recognition can be approached through various modalities, such as facial expressions, textual content, vocal signals, and more [1]. 1. It is considered one of the most significant elements in the realm of human-computer interaction [2]. Furthermore, it may assist individuals with Autism Spectrum Disorders (ASDs) in enhancing their ability to recognize the facial emotions of others, thereby facilitating improved social interactions [3]. Individuals with Autism Spectrum Disorder (ASD) frequently encounter obstacles in both comprehending and articulating emotions, resulting in complications in their social interactions and relationships. Consequently, the ability to recognize emotions can aid those with ASD in enhancing their understanding and interpretation of emotional cues, thereby promoting more effective communication, social engagement, and emotional expression. Additionally, the application of emotion recognition extends to other areas, including the assessment of customer satisfaction [4]. Emotion recognition technology enables organizations to assess facial expressions and emotional states, thereby offering a more profound comprehension of customer sentiments and levels of satisfaction. By utilizing advanced deep learning methodologies for facial emotion detection, companies can obtain immediate insights into customer emotions, allowing them to customize their products, services, and interactions to better align with customer requirements. This improved grasp of customer emotions can result in heightened customer satisfaction, as businesses can modify their strategies in response to customer feedback and emotional indicators, ultimately enriching the overall customer experience and fostering loyalty. According to [5], Facial expressions serve as the primary indicators for interpreting human emotions, surpassing the significance of auditory messages and verbal communication. Consequently, the examination of facial images enhances the comprehension of emotional states. In the context of facial emotion

http://dx.doi.org/10.22133/ijwr.2024.459595.1224
Citation O. Ghadami, A. Rezvanian, " A Scalable Method for Real-Time Facial Emotion Recognition using an Artificial Neural Network and Polynomial Equation", *International Journal of Web Research*, vol.7, no.4, pp.39-49, 2024, doi: http://dx.doi.org/ 10.22133/ijwr.2024.459595.1224.

*Coressponding Author
Article History: Received: 26 May 2024; Revised: 12 September 2024; Accepted: 16 September 2024.

recognition, the process involves inputting a facial image, with the resulting output categorizing the emotion into one of several distinct types. According to [3], There exist two primary methodologies for facial emotion recognition. The first methodology is known as static facial emotion recognition [4], while the second is referred to as dynamic facial emotion recognition [3]. Furthermore, the efficacy of emotion recognition in the aforementioned applications is predominantly contingent upon its implementation in real-time. Consequently, the primary focus of this paper is on real-time facial emotion recognition.

A significant obstacle in developing a high-performance real-time emotion recognition system is the trade-off between model accuracy and inference time; as the model's accuracy improves, the inference time tends to increase correspondingly. Real-time execution necessitates that the inference time remains below 40 milliseconds, as presented by [3]. Nevertheless, the majority of highly accurate methods tend to exhibit longer delays. Conversely, many techniques that achieve lower delays often lack precision. This issue is particularly challenging to address within Deep Learning frameworks, which typically incur substantial computational expenses and require robust computational resources to facilitate real-time performance. In reference [2], the authors utilized a range of Convolutional Neural Networks (CNNs), applying fine-tuning methods in conjunction with various classification components. Additionally, reference [4] introduced a lightweight CNN that leverages the inception module concept and incorporates Global Average Pooling in place of fully connected layers, aiming to decrease inference time for emotion recognition across multiple datasets, such as FER2013 and JAFFE. Additionally, in [6], the implementation of depth-wise separable convolution has been adopted, resulting in the removal of fully connected layers to enhance inference speed. Two primary determinants of real-time execution are computational capacity and the model itself. Prior research has predominantly focused on enhancing the model with respect to its computational efficiency. As a result, these models may experience a decline in performance when subjected to varying levels of computational power. In certain instances, the models may fail to achieve real-time performance due to the high demand for computational resources. In such scenarios, the implementation of a scalable approach becomes essential to maintain real-time execution capabilities.

In this paper, we examined emotion recognition through the lens of classification and aimed to introduce a scalable, high-performance method for real-time emotion detection utilizing the KDEF dataset. EfficientNetV2 stands out as one of the most efficient neural networks available [7]. This model emphasizes the creation of smaller architectures that facilitate quicker training and reduced inference times. Building on the foundations of the EfficientNet model, it seeks to improve both efficiency and speed in the training of deep learning frameworks. The design of the EfficientNetV2 architecture prioritizes compactness and efficiency while ensuring high performance, rendering it applicable across a range of computer vision and machine learning tasks. Consequently, it is capable of delivering commendable performance with minimal inference time compared to other convolutional neural networks (CNNs). EfficientNetV2 employs both MBConv (Mobile Inverted Bottleneck Convolution) and Fused-MBConv blocks extensively, integrating convolution and pointwise convolution into a unified operation to enhance efficiency. The architecture favors smaller 3x3 kernel sizes within the convolutional layers, compensating for the diminished receptive field by incorporating additional layers. Notably, EfficientNetV2 eliminates the final stride-1 stage found in the original EfficientNet, addressing concerns related to its substantial parameter size and memory access overhead. Additionally, reducing the resolution proves advantageous for decreasing inference time and facilitating real-time processing. However, this reduction may lead to a decline in accuracy, necessitating preprocessing techniques to mitigate performance loss. Conversely, there are instances where the neural network can operate in real time without the need for resolution scaling, allowing for potential model scaling to enhance performance. It is important to note that arbitrary scaling does not inherently ensure optimal performance; thus, resolution should be adjusted according to the model's requirements for real-time execution on specific computational resources. Furthermore, the model itself is not the sole determinant of real-time performance; computational power plays a crucial role and can significantly influence inference time, potentially hindering real-time execution. To address these challenges, we propose a scalable approach tailored to both the model and the available computational power.

The remaining sections of this paper are organized into four sections. Section 2 presents an overview of the previous works on facial emotion recognition. Also, section 3 explains our proposed method, and section 4 covers the experimental settings. Section 5 demonstrates outcomes and analyzes them. Finally, section 6, concludes the paper.

2. Related Work

Facial emotion recognition has been the subject of numerous studies aimed at various applications.

As previously noted, there are two primary methodologies for facial emotion recognition: dynamic and static approaches. Reference [3] exemplifies research in dynamic emotion recognition, specifically designed to assist individuals with autism spectrum disorders (ASD) who often struggle with social interactions due to difficulties in interpreting facial expressions. This study seeks to develop efficient, low-latency systems capable of recognizing facial expressions in real-time video contexts. The proposed model utilizes a deep time windowed convolutional neural network (TimeConvNets) trained on the CK+ dataset. The TimeConvNets architecture employs time windowing techniques within a convolutional neural network framework, enabling the capture of temporal dependencies in facial expressions across video sequences. Conversely, various initiatives have concentrated on static emotion recognition, which will be elaborated upon in subsequent sections.

Previous studies have introduced specific techniques for the recognition of static facial emotions. Nonetheless, a notable drawback of many current approaches is their prolonged inference time, particularly when utilizing standard computational resources. For instance, in [8], the authors aimed to develop a model employing convolutional neural networks (CNN) alongside various preprocessing strategies, including intensity normalization, down-sampling, image cropping, and spatial normalization, to mitigate the challenges posed by limited data availability in facial expression recognition and enhance the model's efficacy. Additionally, they investigated the importance of the order of training samples in CNN models for facial expression recognition, emphasizing how the arrangement of these samples can affect both the learning process and the overall performance of the model. In [9], the authors review and analyze current techniques for automatically identifying human emotions through facial expressions, focusing on both traditional feature-based methods and modern deep-learning approaches. They highlight the superiority of Convolutional Neural Networks (CNNs) over traditional methods in handling variations and achieving higher accuracy, particularly when fine-tuned on large datasets. Also, they identified key challenges such as the need for diverse datasets, real-time processing capabilities, and robustness against occlusions and cultural variations.

In [10], the authors employ a three-dimensional convolutional neural network (3DCNN) alongside a Convolutional-Long-Short-Term-Memory (ConvLSTM) neural network to address the task of facial emotion recognition. Their objective is to accurately model the dynamic nature of human emotional behavior by integrating both spatial and temporal data. Although deep learning techniques have demonstrated efficacy in recognizing emotions within video sequences, they often fall short in effectively capturing spatiotemporal interactions and detecting nuanced emotional variations. This study seeks to enhance the precision and overall performance of emotion recognition systems by leveraging the spatial and temporal dimensions of emotional experiences through the use of 3DCNN. The findings have potential implications for diverse applications, including human-computer interaction, social media analytics, and mental health assessment. In [11], the authors propose an advanced method for recognizing facial emotions by integrating image segmentation with the VGG-19 deep learning architecture. This approach leverages the powerful feature extraction capabilities of VGG-19, enhanced by pre-processing steps involving facial region segmentation to improve focus on pertinent areas of the face. The model demonstrates improved accuracy and robustness compared to conventional methods, effectively handling variations in facial expressions, lighting, and occlusions.

The significance of real-time execution is exacerbated by constraints in computational resources. Under these circumstances, it is essential to employ methods that incur low computational costs to facilitate real-time performance. Numerous studies have sought to develop lightweight deep-learning techniques to address this issue; however, many of these approaches fall short in terms of accuracy. For example, [6] and [12] proposed light CNNs and Support Vector Machine (SVM) on FERC and FER2013 datasets, respectively. In [6], the focus is on deep learning methodologies, particularly Convolutional Neural Networks, which are evaluated against traditional techniques for the task of facial expression recognition. Also, in [12], the authors focused on the application of Convolutional Neural Networks (CNNs) for the task of facial expression recognition, emphasizing the advantages this methodology offers in diverse fields such as human-computer interaction, emotion analysis, and affective computing. Furthermore, authors in [2] introduced CNN-based techniques that leverage fine-tuning, utilizing pre-trained weights from the ImageNet dataset as a foundational step. This approach aims to enhance the effectiveness and naturalness of human-computer interaction by integrating the capability to discern human emotions through facial expressions. Additionally, an active learning strategy was proposed to identify the most relevant segments of the CK dataset within the training set, as noted in [13]. Subsequently, action units and Support Vector Machines (SVM) were employed for the static facial emotion recognition task utilizing this dataset. The researchers in this study sought to tackle the issues of diminished

recognition accuracy and inadequate robustness in automated facial expression analysis. The experimental findings indicate that the proposed algorithm successfully mitigates correlated noise and achieves superior recognition rates when compared to principal component analysis and human evaluators across seven distinct facial expressions. Besides, the study presented in [14], different configurations of the web-shaped structure are explored to find the optimal one for the emotion recognition task, and the K-nearest neighbor classifier on the CK+ and KDEF datasets was utilized. This method does not require a training phase as it analyzes the position of facial reference points on the web, adapting to various face sizes and types without the need for specific training data.

In earlier research, both the accuracy and inference time of models have been examined. For example, studies [4] and [15] introduced real-time approaches for facial emotion recognition in robotic systems. These investigations focused on minimizing the number of parameters to enhance inference speed. Specifically, the study [4] employed a model inspired by the inception module, achieving a tenfold reduction in parameters. This research presents findings on the effectiveness of the model, its resilience in interpreting diverse facial expressions, and its potential applications in human-computer interaction. Furthermore, recognizing that a significant portion of CNN parameters resides in the classification layer, the authors of [4] implemented Global Average Pooling and utilized various datasets to bolster the model's robustness. Additionally, a study [15] proposed two CNN-based architectures that leverage depth-wise separable convolution and eliminate connected layers for real-time facial emotion recognition, tested on the FER2013 and IMDB datasets. The computational efficiency of the model is noteworthy, with a processing time of under 0.008 seconds on a Core i7 CPU, rendering it suitable for real-time applications. The authors have made their open-source code and pre-trained models available through a GitHub repository.

In [16], the authors present a novel system that combines facial emotion recognition with personalized music recommendations. The system employs Convolutional Neural Networks (CNNs) to accurately detect and classify emotions from facial expressions. Once the user's emotional state is identified, the system recommends music tracks that align with the detected mood, enhancing the user experience. The study highlights the effectiveness of CNNs in handling diverse facial expressions and varying conditions, achieving high accuracy in emotion recognition. Additionally, the integration of emotion recognition with music recommendation offers a seamless and intuitive application, demonstrating significant potential in fields such as entertainment, mental health, and user experience personalization. Scaling methods serve as effective strategies to achieve a balance between inference time and accuracy, thereby enhancing overall efficiency. These methods can be employed to either decrease inference time or improve accuracy. Numerous studies have explored three primary scaling approaches: width (neural network channel), depth (neural network layer), and resolution (neural network input size).

The literature indicates that facial expression recognition is applicable across a wide range of contexts. Furthermore, real-time performance is critical in most applications; however, this aspect has often been overlooked in prior research. The computational cost becomes particularly relevant in systems with limited processing capabilities. Conversely, while some earlier studies have addressed the need for real-time execution, many have not achieved high-performance outcomes. Therefore, there is a pressing need for a method that combines real-time processing with high performance in the domain of facial emotion recognition to meet the demands of various applications.

3. Methodology

In this section, we address our proposed method beginning with the definition of the facial emotion recognition task, followed by our proposed method in detail.

3.1. Task Definition

In the task of facial emotion recognition, the model necessitates a facial image as input, subsequently producing a label that indicates the individual's emotional state from a predefined set of categories. Furthermore, the model needs to operate in real-time with a high degree of accuracy to facilitate various applications, including the enhancement of human-computer interaction. Additionally, the model must be designed to be scalable, ensuring its effectiveness across diverse computational environments. Therefore, the primary aim of this paper is to propose a scalable, high-performance method for real-time facial emotion recognition.

3.2. Methodology Overview

Our proposed methodology encompasses five primary stages: preprocessing, resolution scaling, evaluation of inference time, resolution tuning, and training. Upon completion of these stages, our models are capable of identifying emotions in facial images within a timeframe of 40 milliseconds on our computational resources. The stages of our proposed approach are illustrated in Figure 1. Furthermore,

each stage comprises multiple steps, which will be elaborated upon in detail.

Preprocessing

The preprocessing phase encompasses data augmentation, normalization, and the management of the imbalanced dataset (KDEF). To mitigate the risk of overfitting, data augmentation is widely recognized as an effective strategy. This technique, prevalent in both machine learning and deep learning, serves to artificially expand the training dataset by generating modified versions of existing data or creating new data points derived from the original dataset. The augmentation process typically involves applying minor alterations or transformations to the data, which may include operations such as flipping, rotating, cropping, adjusting brightness, introducing noise, or implementing various other modifications. The application of data augmentation contributes to enhanced model performance, a reduction in overfitting, improved accuracy, and better generalization capabilities of machine learning models. This approach is particularly advantageous in situations where acquiring large volumes of diverse training data is either difficult or financially prohibitive. In this study, we employed techniques such as zoom, horizontal flipping, shifting, shearing, and rotation as methods of data augmentation.

Normalization is widely recognized as a fundamental preprocessing procedure that must be tailored to the specific requirements of the neural network, as each neural network operates with distinct input value ranges. In the context of Convolutional Neural Networks (CNNs), four common types of normalization are identified: normalization range, centering, standardization, and per-channel normalization. In this study, we employed the normalization range, which involves scaling pixel values to a defined interval, such as -1 to +1.

The preprocessing phase encompasses data augmentation, normalization, and the management of imbalanced datasets, specifically the KDEF dataset. A critical aspect of this phase is addressing the issue of dataset imbalance. The KDEF dataset exhibits significant imbalance, and training models on such datasets can adversely affect their performance in practical applications. Models trained on imbalanced datasets tend to exhibit bias towards classes with a higher number of samples. In this research, we implemented a method for managing imbalanced datasets as outlined in reference [17]. This method involves assigning weights to classes under the quantity of their samples; classes with a greater number of samples receive lower weights.

Consequently, this approach preserves the dataset's integrity while effectively mitigating the challenges posed by imbalance.

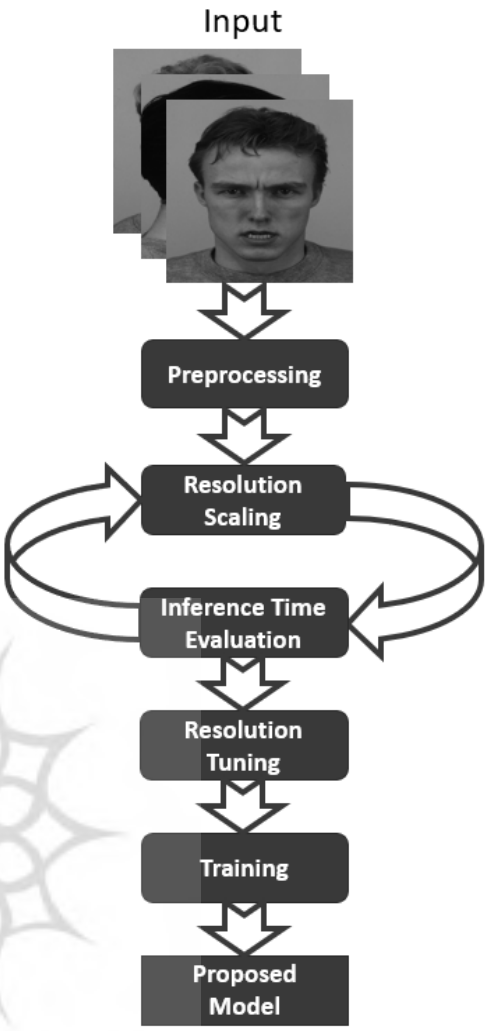

Figure. 1. The overall structure of the proposed method for scalable real-time emotion recognition

Resolution Scaling

A prevalent method for balancing inference time and accuracy is resolution scaling [7]. This technique involves modifying the resolution of input images and the initial layer of the neural network under the dimensions of the new input images. In this study, we examine n distinct input sizes or resolutions (both width and depth of the input images) to evaluate the inference time associated with each of these resolutions.

Inference Time Evaluation

Following the assignment of distinct numerical values to various resolutions, it is essential to assess the model's inference time or latency corresponding to each specific resolution. This evaluation is determined through forward propagation using a

single image sample to identify the emotion depicted in the image.

Resolution Tuning

At this stage, we have n distinct resolutions paired with n corresponding inference times. We denote these resolutions as x and the inference times as *y*. This results in n pairs of (*x*, *y*), which can be utilized to formulate a polynomial equation derived from these data points. Essentially, this polynomial equation serves to identify the optimal resolution that yields a specific inference time given a particular computational capacity and model. Furthermore, an increase in the value of *n* enhances the accuracy of the polynomial equation.

Various methodologies exist for deriving a polynomial equation that intersects a specified set of points, including polynomial regression, Lagrange interpolation, and Newton's divided difference interpolation. In this research, we employed Lagrange interpolation. Although initially discovered by Edward Waring in 1779, the method is named after Joseph-Louis Lagrange, who published it in 1795. This technique is effective for constructing a polynomial that accurately represents a collection of discrete data points, thereby facilitating the estimation of function values at intermediate locations based on the available data. Equ(1) represents the formula of Lagrange interpolation where x_0, x_1, x_2, ... are distinct resolutions and y_0, y_1, y_2, ... are distinct corresponding numbers for inference times. In addition, Equ (2) represents l_i which is a part of the *P(x)* equation.

$$P(x) = \sum_{i=0}^{n} y_i l_i(x) \quad (1)$$

$$l_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j} \quad (2)$$

The optimal resolution for real-time execution on our designated computational resource and model has been determined following the evaluation of the polynomial equation, which was based on varying resolutions and inference times.

Training

Following the assessment of inference time and the optimization of resolution derived from the polynomial equation obtained through Lagrange interpolation, it becomes evident which resolution is suitable for real-time operation on the designated model and available computational resources. Subsequently, it is necessary to modify the image resolutions and the input dimensions of the neural networks following the values determined by the polynomial equation in the preceding phase. At this point, the model will be prepared for the training phase.

4. Experiments

In this section, we address the experimental setting, which includes the dataset and the metrics used for evaluation, baselines, and implementation in detail.

4.1. Dataset and Evaluations

We evaluate our proposed models utilizing the Karolinska Directed Emotional Faces (KDEF) dataset, which comprises annotated images of human faces. The KDEF dataset features a variety of facial expressions and serves as a significant resource for research in affective neuroscience, psychology, and computer vision. It is extensively employed in studies focused on emotion recognition, facial expression analysis, and human-computer interaction. The dataset is available for non-commercial research purposes and has contributed to over 1500 scholarly publications, highlighting its importance in the academic community. Established in 1998, KDEF has remained a widely used resource. It contains a total of 4900 images representing seven distinct emotional categories: fear, anger, disgust, happiness, neutrality, sadness, and surprise. The dataset includes photographs of 70 individuals, evenly split between 35 females and 35 males, with each expression captured from five different angles.

We employed accuracy, loss, and F1 score as metrics for model evaluation. Additionally, inference time serves as an indicator of the model's capability for real-time processing. As noted in [3], a model is deemed to operate in real time if its inference time does not exceed 40 milliseconds. Therefore, we regard this threshold as the criterion for real-time execution.

4.2. Baselines

Based on prior related works in facial emotion recognition, we have selected two distinct models to compare with our proposed model.

BDF-InceptionV3 is a convolutional neural network (CNN) that draws inspiration from the InceptionV3 architecture. InceptionV3, developed by Google, is a member of the Inception model family and is specifically tailored for tasks related to image classification and object detection. This architecture is recognized for its remarkable efficiency and precision in addressing intricate visual recognition challenges. A notable characteristic of InceptionV3 is its incorporation of inception modules, which consist of convolutional components utilizing multiple filter sizes within a single layer. The model has found extensive application in the field of computer vision, excelling in areas such as image recognition, object detection, and facial expression analysis. In the case of BDF-

InceptionV3, the standard classifier of InceptionV3 is substituted with additional layers, which include batch normalization, dropout, a fully connected layer, and softmax activation. Additionally, BDF-InceptionV3 leverages pre-trained weights from InceptionV3; however, it is important to note that only the final layers are subject to training, while the parameters associated with the feature extraction component remain fixed or non-trainable.

BDF-MobileNet [2] is a convolutional neural network (CNN) that draws inspiration from the MobileNet [18] architecture. MobileNet is specifically engineered for efficient performance in mobile and embedded vision applications. Its lightweight structure enables deployment on devices with limited resources, such as smartphones, Internet of Things (IoT) devices, and embedded systems. The architecture is optimized to deliver high accuracy in image classification tasks while minimizing both computational demands and model size. A key feature of MobileNets is the use of depthwise separable convolutions, which significantly decrease the number of parameters and computations in comparison to conventional convolutional layers, thereby enhancing suitability for real-time applications on devices with restricted processing power. In BDF-MobileNet, the standard classifier of MobileNet is substituted with additional layers, including batch normalization, dropout, a fully connected layer, and softmax. Furthermore, BDF-MobileNet utilizes pre-trained weights from MobileNet, although only the final layers are subject to training, with the parameters of the feature extraction component being fixed or rendered non-trainable.

4.3. Implementation Details

In this part, our implementation details have been divided into several steps, representing different stages of our proposed method. In addition, the hardware involved in the computation of the inference time is the 11th Gen Intel core i5-11400H 2.70GHz CPU. It is a 6-core, 12-thread processor that belongs to Intel's 11th-generation Core i5 lineup. The base clock speed of the i5-11400H is 2.70GHz, with the ability to boost up to 4.40GHz.

The first step of the preprocessing stage is data augmentation. Table 1 demonstrates all values of data augmentation methods, which have been employed.

Normalization is the next step. All EfficientNetV2 models need [-1, +1] range of pixel values as their inputs. Hence, all pixel values need to change to this domain.

The second stage is resolution scaling. In this state, it is required to consider *n* different resolution sizes. In this study, we considered *n* equal to 10.

Table 1. Amounts for Data Augmentation Methods

Augmentation Method	***Value***
Rotaion_range	40
Width_shift_range	0.25
Height_shift_range	0.25
Shear_range	0.25
Zoom_range	0.25
Horizental_flip	True
Fill_mode	nearest

However, the higher the *n* is, the more accurate the polynomial equation becomes. Also, we utilized 10 more common resolution sizes while it is possible to specify these values randomly. These common resolutions are (299, 299, 3), (224, 224, 3), (162, 162, 3), (143, 143, 3), (128, 128, 3), (100, 100, 3), (84, 84, 3), (66, 66, 3), (48, 48, 3), and (12, 12, 3).

Following the evaluation of inference time and the adjustment of resolution through the polynomial equation derived from Lagrange interpolation, we proceeded to train the models for 120 epochs on the KDEF dataset, incorporating pre-trained weights from the ImageNet dataset. Additionally, we utilized the Adam optimizer with a dynamic learning rate.

5. Results and Analysis

In this section, we will first examine the plot of the polynomial equation related to our proposed models. Subsequently, we will conduct a comprehensive assessment of the performance of our proposed models in comparison to the baselines on the KDEF dataset.

5.1. Resolution Tuning

To achieve resolution tuning, it is essential to first determine the inference time associated with various resolutions or input sizes. This information is necessary to derive the polynomial equation utilizing Lagrange interpolation, as outlined in the methodology section. This approach facilitates the identification of an optimal resolution for real-time processing. Figures 2 and 3 illustrate the polynomial equations corresponding to the EfficientNetV2-B0 and EfficientNetV2-S models, respectively. In these figures, the green boxes indicate the real-time operational area, which is confined between 0 and 40 milliseconds on the y-axis. As indicated in Figure 2, the EfficientNetV2-B0 model can operate in real-time with a resolution of up to (302, 302, 3) on our hardware. Conversely, Figure 3 shows that the EfficientNetV2-S model can function in real-time with a resolution of up to (94, 94, 3) on the same hardware. These findings suggest that the resolutions of both models need to be fine-tuned for subsequent processes. Following the resolution tuning, the Scalable-ENV2B0 and Scalable-ENV2S

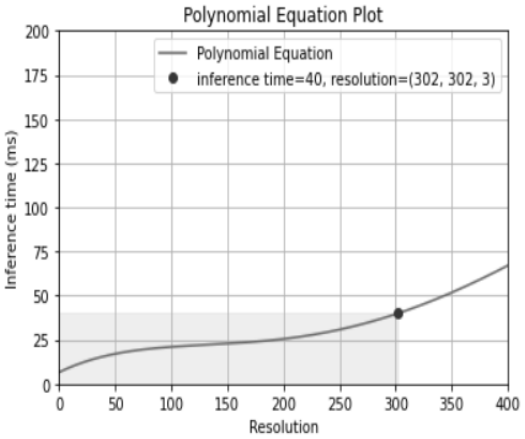

Figure. 2. The polynomial equation plot for EfficientNetV2B0

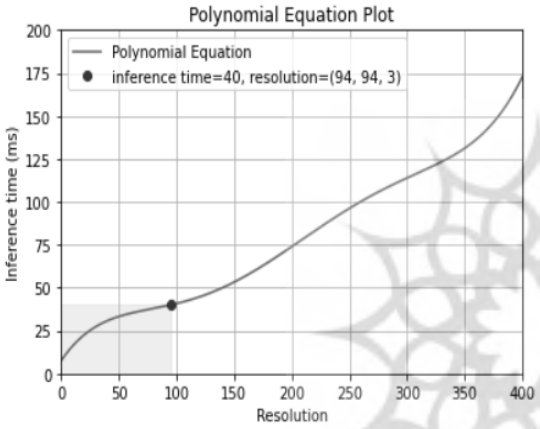

Figure. 3. The polynomial equation plot for EfficientNetV2S

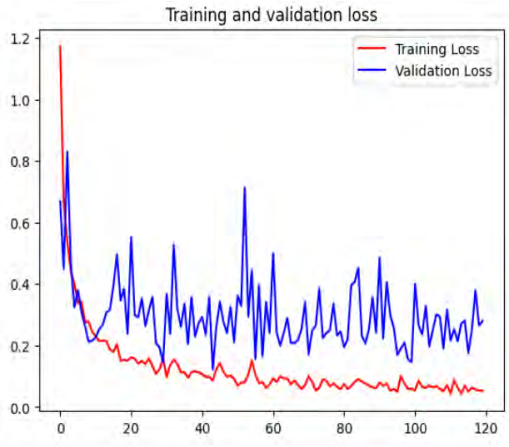

Figure. 4. Loss plot of Scalable-ENV2B0 in 120 epochs

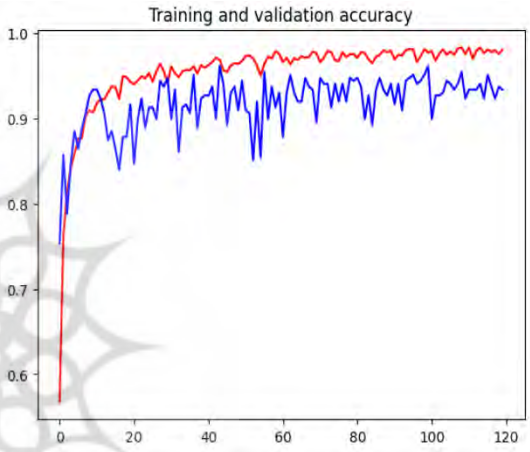

Figure. 5. Accuracy plot of Scalable-ENV2B0 in 120 epochs

models are developed based on the EfficientNetV2-B0 and EfficientNetV2-S models, respectively, incorporating additional preprocessing steps and varying resolutions tailored to the available computational resources.

5.2. Training

Figures 4 and 5 illustrate the variations in loss and accuracy for both training and validation datasets about the Scalable-ENV2B0 model applied to the KDEF dataset. Similarly, Figures 6 and 7 depict the alterations in loss and accuracy for the Scalable-ENV2S model on the same dataset. In all four figures, the blue lines indicate validation loss and validation accuracy, while the red lines represent training loss and training accuracy. The data presented in these figures reveal a notable decrease in validation loss and a significant increase in validation accuracy up to approximately the 60th epoch, after which minor fluctuations are observed.

Conversely, Figures 8 and 9 present the changes in loss and accuracy for the BDF-InceptionV3 model over 120 epochs on the KDEF dataset.

Additionally, Figures 10 and 11 illustrate the corresponding changes for the BDF-MobileNet model over the same number of epochs. In these figures, blue lines again denote validation loss and validation accuracy, while red lines signify training loss and training accuracy. Analysis of Figures 9 and 11 indicates that the most substantial increases in both training and validation accuracy occurred before the 100th epoch. Furthermore, Figures 8 and 10 reveal that the most significant reductions in training and validation loss transpired before the 80th epoch.

According to Figures 5 and 7, the peak validation accuracy achieved by the Scalable-ENV2B0 and Scalable-ENV2S models was 95% and 94%, respectively. In contrast, Figures 9 and 11 show that the highest validation accuracy for the BDF-InceptionV3 and BDF-MobileNet models reached 65% and 66%, respectively.

5.3. Results and discussions

The findings derived from our proposed models and baseline models utilizing the KDEF dataset are

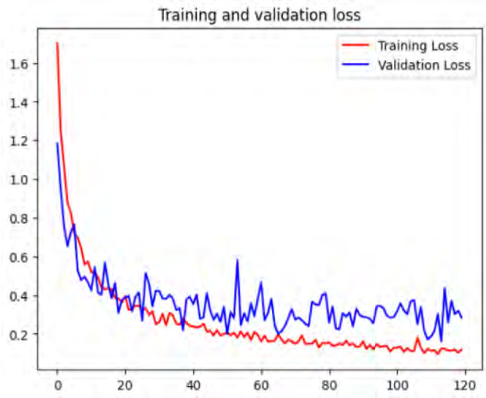

Figure. 6. Loss plot of Scalable-ENV2S in 120 epochs

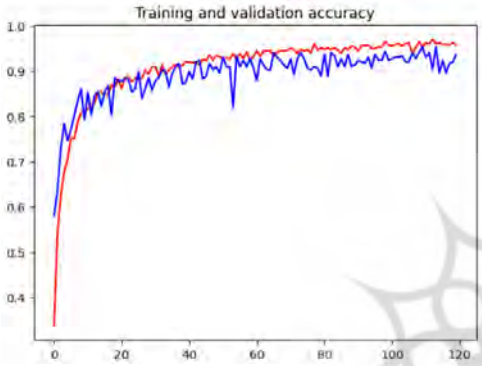

Figure. 7. Accuracy plot of Scalable-ENV2S in 120 epochs

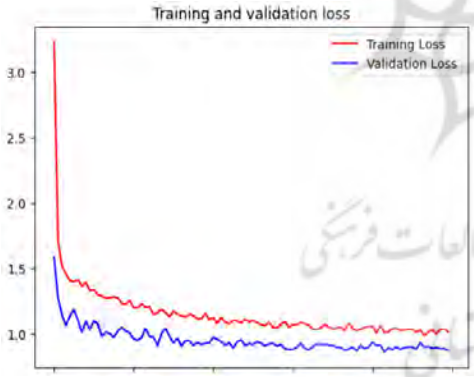

Figure. 8. Loss plot of BDF-InceptionV3 in 120 epochs

Figure. 9. Accuracy plot of BDF-InceptionV3 in 120 epochs

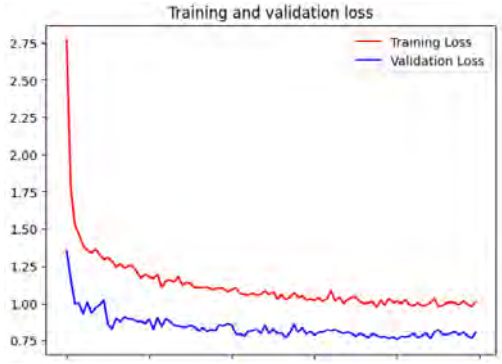

Figure. 10. Loss plot of BDF-MobileNet in 120 epochs

Figure. 11. Accuracy plot of BDF-MobileNet in 120 epochs

summarized in Table 2. The inference time indicated in this table represents the average duration from ten separate executions, while the metrics for accuracy, loss, and F1-score (macro, micro, and weighted) are calculated based on the dataset's test set. For the KDEF dataset's test set, we divided the validation set into two equal segments. As illustrated in the table, all models, except for BDF-Inception, operate in real time on our hardware. Notably, BDF-MobileNet recorded the shortest inference time. In terms of accuracy, our proposed models surpassed the other models by approximately 20 percent. Scalable-ENV2B0 achieved the highest accuracy at 96%, along with the best macro, micro, and weighted metrics. Scalable-ENV2S followed closely in second place with an accuracy of 92%. These results indicate that a significant reduction in resolution can drastically impact accuracy, as evidenced by EfficientNetV2S, which, at a resolution of (384,384,3), outperformed EfficientNetV2B0 at (224,224,3) in numerous prior studies across various datasets, including ImageNet [7]. Furthermore, regarding the loss metric, Scalable-ENV2B0 and Scalable-ENV2S recorded values of 0.16 and 0.26, respectively.

Figure 12 illustrates the performance metrics of the models in terms of accuracy and inference time when evaluated on the KDEF dataset. This visual representation encapsulates the results presented in

Table 2. The Performance of Different Models On KDEF

	Scalable-ENV2B0	*Scalable-ENV2S*	*BDF-InceptionV3*	*BDF-MobileNet*
Resolution	(302,302,3)	(94,94,3)	(299,299,3)	(224,224,3)
Accuracy	0.96	0.92	0.69	0.74
Loss	0.16	0.26	0.81	0.76
Macro	0.95	0.92	0.70	0.74
Micro	0.96	0.93	0.70	0.73
Weighted	0.97	0.92	0.68	0.73
Inference Time (ms)	40	40	54	16

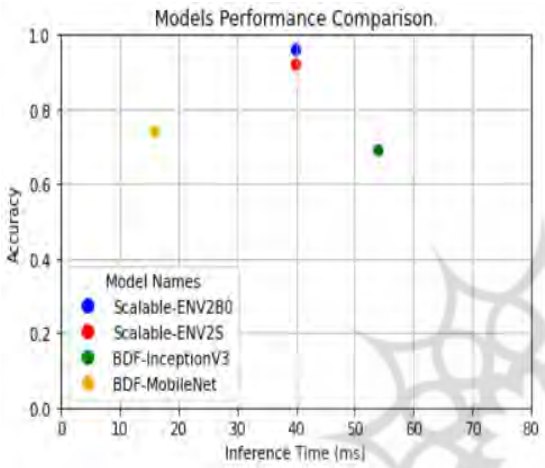

Figure. 12. Models' performance comparison based on inference time and accuracy.

Table 2, highlighting the efficacy of the models in emotion recognition during the dataset tests. The use of different colors in the charts signifies individual models, thereby offering a comprehensive overview of their respective performances.

6. Conclusions

This article presents a scalable, high-performance model designed for real-time facial emotion recognition, utilizing resolution scaling in conjunction with EfficientNetV2. The resolution scaling technique is formulated through a polynomial equation, which determines the optimal resolution concerning computational resources and the model itself. This polynomial equation is derived using Lagrange interpolation. Consequently, we introduce the Scalable-ENV2B0 and Scalable-ENV2S models. Our experimental findings indicate that the Scalable-ENV2B0 model, operating at a resolution of (302, 302, 3), achieves an accuracy of 96% on the KDEF test set, with an inference time of 40 milliseconds on our hardware. While the primary benefit of our proposed approach lies in its scalability, the performance metrics obtained surpass those of previous related studies, as evidenced by our experiments and insights. In this study, as a limitation, one can say its dependency on high-quality video inputs, which may not perform well under low-light or occluded conditions. Future work will focus on enhancing robustness in diverse environments and exploring multimodal approaches to improve accuracy and applicability across various real-world scenarios.

Declarations

Funding

This research did not receive any grant from funding agencies in the public, commercial, or non-profit sectors.

Authors' contributions

OG: Study design, acquisition of data, interpretation of the results, statistical analysis, drafting the manuscript;

AR: Study design, interpretation of the results, drafting the manuscript, revision of the manuscript.

Conflict of interest

The authors declare that no conflicts of interest exist.

References

[1] O. Ghadami, A. Rezvanian, and S. Shakuri, "Scalable Real-time Emotion Recognition using EfficientNetV2 and Resolution Scaling," in *2024 10th International Conference on Web Research (ICWR)*, Tehran, Iran, IEEE, 2024, pp. 7–12. https://doi.org/10.1109/ICWR61162.2024.10533360

[2] M. K. Chowdary, T. N. Nguyen, and D. J. Hemanth, "Deep learning-based facial emotion recognition for human–computer interaction applications," *Neural Comput & Applic*, vol. 35, no. 32, pp. 23311–23328, Nov. 2023. https://doi.org/10.1007/s00521-021-06012-8

[3] J. R. H. Lee and A. Wong, "Timeconvnets: A deep time windowed convolution neural network design for real-time video facial expression recognition," in *2020 17th Conference on Computer and Robot Vision (CRV)*, Ottawa, ON, Canada, IEEE, 2020, pp. 9–16, https://doi.org/10.1109/CRV50864.2020.00010.

[4] 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. https://doi.org/10.1007/s00521-019-04564-4

[5] B. Kratzwald, S. Ilić, M. Kraus, S. Feuerriegel, and H. Prendinger, "Deep learning for affective computing: Text-based emotion recognition in decision support," *Decision support systems*, vol. 115, pp. 24–35, 2018. https://doi.org/10.1016/j.dss.2018.09.002

[6] X. Wang, J. Huang, J. Zhu, M. Yang, and F. Yang, "Facial expression recognition with deep learning," in *Proceedings of the 10th International Conference on Internet Multimedia Computing and Service*, Nanjing China: ACM, Aug. 2018, pp. 1–4. https://doi.org/10.1145/3240876.3240908

[7] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International Conference on Machine Learning*, PMLR, 2021, pp. 10096–10106. https://proceedings.mlr.press/v139/tan21a.html

[8] A. T. Lopes, E. De Aguiar, A. F. De Souza, and T. Oliveira-Santos, "Facial expression recognition with convolutional neural networks: coping with few data and the training sample order," *Pattern recognition*, vol. 61, pp. 610–628, 2017. https://doi.org/10.1016/j.patcog.2016.07.026

[9] Z.-Y. Huang *et al.*, "A study on computer vision for facial emotion recognition," *Scientific Reports*, vol. 13, no. 1, p. 8425, 2023. https://doi.org/10.1038/s41598-023-35446-4

[10] D. Al Chanti and A. Caplier, "Deep learning for spatio-temporal modeling of dynamic spontaneous emotions," *IEEE Transactions on Affective Computing*, vol. 12, no. 2, pp. 363–376, 2018. https://doi.org/10.1109/TAFFC.2018.2873600

[11] S. Vignesh, M. Savithadevi, M. Sridevi, and R. Sridhar, "A novel facial emotion recognition model using segmentation VGG-19 architecture," *Int. j. inf. tecnol.*, vol. 15, no. 4, pp. 1777–1787, Apr. 2023. https://doi.org/10.1007/s41870-023-01184-z

[12] C. Pramerdorfer and M. Kampel, "Facial Expression Recognition using Convolutional Neural Networks: State of the Art," Dec. 09, 2016, *arXiv*: arXiv:1612.02903. https://doi.org/10.48550/arXiv.1612.02903

[13] L. Yao, Y. Wan, H. Ni, and B. Xu, "Action unit classification for facial expression recognition using active learning and SVM," *Multimed Tools Appl*, vol. 80, no. 16, pp. 24287–24301, Jul. 2021. https://doi.org/10.1007/s11042-021-10836-w

[14] P. Barra, L. De Maio, and S. Barra, "Emotion recognition by web-shaped model," *Multimed Tools Appl*, vol. 82, no. 8, pp. 11321–11336, Mar. 2023. https://doi.org/10.1007/s11042-022-13361-6

[15] O. Arriaga, M. Valdenegro-Toro, and P. Plöger, "Real-time Convolutional Neural Networks for emotion and gender classification," in *27th European Symposium on Artificial Neural Networks, ESANN 2019, Bruges, Belgium, April 24-26, 2019*, 2019, pp. 221–226. https://www.esann.org/sites/default/files/proceedings/legacy/es2019-157.pdf

[16] B. Bakariya, A. Singh, H. Singh, P. Raju, R. Rajpoot, and K. K. Mohbey, "Facial emotion recognition and music recommendation system using CNN-based deep learning techniques," *Evolving Systems*, vol. 15, no. 2, pp. 641–658, Apr. 2024. https://doi.org/10.1007/s12530-023-09506-z

[17] A. Rosenbrock, *Deep Learning for Computer Vision with Python Practitioner bundle*, PyImageSearch. com, 2018.

[18] A. G. Howard *et al.*, "MobileNets: efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, vol. 126, 2017.
https://doi.org/10.48550/arXiv.1704.04861

Omid Ghadami received the Master's degree in Computer Engineering (Software Engineering) from the Department of Computer Engineering at the University of Science and Culture (USC), Tehran, Iran, in 2023. His research focuses on deep learning, computer vision, and medical applications of machine learning.

Alireza Rezvanian received the Ph.D. Degree in Computer Engineering (Artificial Intelligence) from Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran, in 2016. Currently, he is an assistant professor at the Department of Computer Engineering, University of Science and Culture (USC), Tehran, Iran. Prior to his current position, he worked from 2016 to 2020 as a researcher at the School of Computer Science at the Institute for Research in Fundamental Sciences (IPM), Tehran, Iran. He is an associate editor of human-centric computing and information sciences, CAAI Transactions on Intelligence Technology (Wiley), The Journal of Engineering (Wiley), and Data in Brief (Elsevier). He was a guest editor of the special issue on new applications of learning automata-based techniques in real-world environments for the Journal of Computational Science (Elsevier). His research activities include soft computing, learning automata, complex networks, social network analysis, data mining, data science, machine learning, and evolutionary algorithms.