iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** XII **Month of publication:** December 2022

DOI: https://doi.org/10.22214/ijraset.2022.48474

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue XII Dec 2022- Available at www.ijraset.com

Deep Learning-Based Recognition of Facial Expressions

Prajjawal Tiwari[1], Navneet Kumar[2], Palak Singh[3], Prbhav Attray[4], Arpit Rai[5], Nizam Uddin Khan[6]
[1, 2, 3, 4, 5, 6]*IMSEC Ghaziabad*

Abstract: ***Convolutional neural networks were used in deep learning to maintain a system for recognizing face expressions of emotion. We were created as two distinct models. The first model was a suggested CNN architecture that was trained on the FER-2013 dataset. The model could classify expressions into 7 different categories with an accuracy rate of 67.18%. Using the FER-2013 dataset and a transfer learning strategy, the second model was produced. The model was able to categories the expressions into 4 groups with an accuracy of 75.55%. A mobile web application that quickly executes our FER models on a device is also provided by us. We introduce generic assessment standards, general face recognition databases, and face recognition research for real-world scenarios. We present a prospective analysis of facial recognition. Face recognition has emerged as the field's most promising area for future advancement.***

I. INTRODUCTION

The face is the most expressive and communicative portion of a human, and improving IHM to establish communication between the two entities has made it a prominent focus of recent study.

The face is the most expressive and communicative portion of a human [1], and improving IHM to allow for conversation between the two entities is a major area of current research.

Our objectives in this study were to apply emotion detection models to real-world scenarios as well as to better understand and enhance their performance. In order to increase accuracy, we adopted a number of strategies from recent papers, including transfer learning, data augmentation, class weighting, adding auxiliary data, and assembling.

We also examined our models using error analysis and various interpretability methods. In order to execute our models on a device, we also used our findings to create a mobile app.

Recently, academics have shown an interest in creating FER systems utilising machine learning (ML) and deep learning (DL) techniques[9]. This interest is paving the way for the creation of reliable FER systems as well as the discovery of novel FER parameters. Typically, visible light cameras are employed to capture the pictures needed for the categorization of facial expressions since they are widely accessible, both as standalone cameras and as an attachment for inexpensive portable devices like phones and tablets.

Despite the numerous studies on the subject, identifying facial emotions from photographs taken by cameras that use visible light remains challenging due to commonplace circumstances like shadows, reflections[10], and obscurity (or low-light). Along with the face, other features like scenery, background images, and many other things are also there.

Therefore, removing the face from the image in order to study the facial emotions becomes a burden. By taking into account the temperature distribution in face muscles and offering improved facial expression categorization, working with thermal pictures aids in resolving these problems.

The face recognition development process and related technologies, such as early algorithms, synthetic[8] features and classifiers, deep learning, and other stages, will be discussed in this study. Next, we'll discuss the studies on facial recognition in realistic settings. Finally, we introduce the general assessment standards and facial recognition databases.

II. LITERATURE SURVEY

The registration, feature extraction, and classification processes are typically the three key components of automated FER algorithms. Facial localization[10] , also known as "face detection," or "face detection," is the process of first locating faces in a picture using a series of landmark points.

This method is known as "facial registration," and it involves geometrically normalizing the detected faces to fit a template image.

The standard methodology for researchers exploring deep learning and vision is a subset of deep neural network topologies known as "convolutional neural networks" (CNNs).

The top three finishers in the 2014 Image Net object identification contest all employed a CNN strategy, with the GooLeNet architecture attaining an astounding 6.66% error rate in classification.

To study the FER problem, a brand-new deep neural network design known as a "AU-Aware" architecture was put forth in [24]. Convolution layers and max-pooling layers make up the bottom of the layer stack in an AU-Aware architecture, which is used to create a comprehensive representation of the face.

The Japanese Female Facial Expression (JAFFE) Database is one database, while the Cohn Cade Database is the other. A multi-step, two-class facial expression classification issue was devised by Kyperountas[11] et al.15 and results were published using the JAFFE and MMI databases.

The best two-class classifier is chosen from a large pool of classifiers at each stage of the procedure. This aided the authors in developing a more effective FER system. A two-step strategy for categorizing facial expressions was suggested by Ali et al.

A histogram of oriented gradients (HOG) was utilized to extract the face characteristics, and a sparse representation classifier (SRC) was employed to identify the facial emotions.

To learn hierarchical features, a multilayer Restricted Boltzmann Machine (RBM) is utilized. The network's outputs are then combined into characteristics that are used to train a linear SVM classifier to recognize the six fundamental phrases.

FER2013 was designed by Goodfellow et al. as a Kaggle competition to promote researchers to develop better FER systems. The top three teams all used CNNs trained discriminatively with image transformations [3].

The winner, Yichuan Tang[13] , achieved a 71.2% accuracy by using the primal objective of an SVM as the loss function for training and additionally used the L2-SVM loss function [4].

III. DATASETS

With a wide variety of available datasets, FER is a well-researched area. We used FER2013 as our primary dataset and CK+ and JAFFE as auxiliary datasets to increase accuracy on its test set. In order to fine-tune our models so they perform better in real-world circumstances, we also generated our own web app dataset.

A. FER2013 Dataset

48*48 pixel pictures of human facial expressions in grayscale make up the FER-2013 dataset. It classifies the photos into 7 different categories of emotion. They are ecstatic, depressed, angry, disgusted, shocked, afraid, or neutral. There are 35887 photos in all.

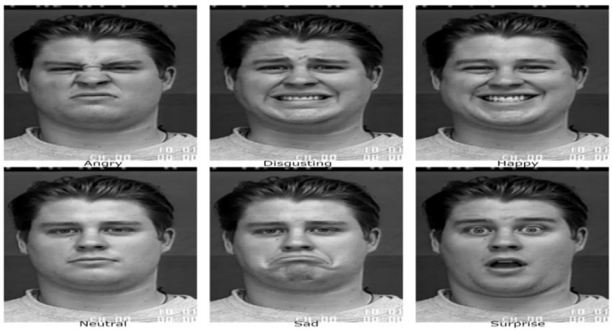


Figure 1 Images from each of the FER2013's emotion classes

.

B. CMU MultiPIE

The CMU MultiPIE[12] face database [15] has almost 750,000 pictures of 337 people taken from various angles and with various lighting effects. In four recording sessions, participants were taught to make a variety of facial expressions (i.e. Angry, Disgust, Happy, Neutral, Surprise, Squint, and Scream). Only the five frontal angles (from -45 to +45) were chosen, giving us a total of almost 200,000 pictures.

C. MMI

More than 20 subjects of either a European, Asian, or South American ancestry (44% female), ranging in age from 19 to 62, are included in the MMI [35] database. Subjects were told to present 79 series of facial expressions, six of which constitute prototypical emotions, and a picture sequence with neutral faces was acquired at the start and conclusion of each session. From each sequence, we took static frames to create 11,500 photos.

Figure 2: Our web app dataset contains images of each emotion class

D. Dataset of Japanese Women's Facial Expressions.

A tiny dataset called Japanese Female Facial Expression (JAFFE) has 213 photos of 10 Japanese female models. Similar to FER2013, the photos are captioned with 7 different face emotions.

E. Comprehensive Cohn-Kanade Dataset

123 people between the ages of 18 and 50 are represented by 593 picture sequences in the enlarged Cohn-Kanade dataset (CK+). Each image in the series consists of between 10 and 60 frames showing a subject changing from neutral to the desired mood. Each frame is approximately 640x480 and has either gray scale or color values [9].

IV. ARCHITECTURE

This model comprises a softmax output layer, an FC layer with a 1024 by 1024 pixel size, and three stages of convolutional and max-pooling layers. The sizes of the 32, 32, and 64 filters used by convolutional layers are 5x5, 4x4, and 5x5.

The max-pooling layers employ 3x3 kernels with a stride of 2.

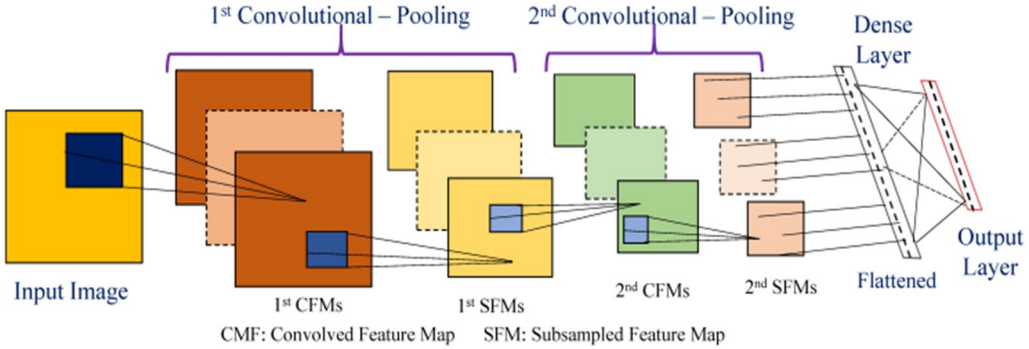
ReLU served as the activation function. We used these features, together with batchnorm at each layer and 30% dropout after the final FC layer, to improve speed. In order to optimize the cross-entropy loss, we trained the model for 300 iterations using stochastic gradient descent with a momentum of 0.9.

Initial learning rates are specified at 0.1, 128 for batch sizes, and 0.0001 for weight decay. If the validation accuracy does not increase after 10 epochs, the learning rate is cut in half.

Subsampling layers come after convolution layers in the traditional convolutional neural network topologies. The size of the cards is decreased by the sub-sampling layer, which also introduces (poor) rotation and translation invariance as input.

A. Transfer Learning

The FER2013 dataset is small and unbalanced, thus we found that using transfer learning significantly improved our model's accuracy. We looked into transfer learning using the pre-trained models ResNet50, SeNet50, and VGG16 [14] together with the Keras VGG-Face library. In FER2013, we reduced the size and changed the hue of the 48x48 gray scale pictures to match the RGB images no less than 197x197 that these new networks anticipated as input[12].



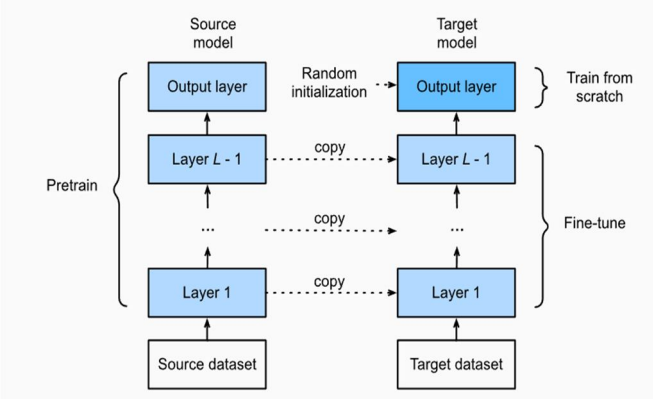
B. Fine Tuning Techniques

The first pre-trained model we looked at was ResNet50. A deep residual network with 50 layers is called ResNet50. It has 175 layers in Keras, where it is specified. We began by re-creating the work of Brechet et al. [10]. With two FC layers of 4,096 and 1,024 pixels each, as well as a softmax output layer with 7 emotion classes, we took the role of the original output layer. ResNet's top 170 layers were frozen, while the remaining layers remained trainable.

ResNet50 was the first pre-trained model we examined. ResNet50 is the name of a deep residual network with 50 layers. In Keras, where it is stated, it has 175 layers. We started by copying Brechet et alwork .'s [10]. We assumed the function of the original output layer and used two FC layers with 4,096 and 1,024 pixels each, as well as a softmax output layer with seven emotion classes. The top 170 layers of ResNet were frozen, while the lower levels were still trainable.

We utilized SGD as our optimizer, using a learning rate of 0.01 and a batch size of 32. Even when we tried to freeze the whole pre-trained network and simply train the FC layers and output layer, the model failed to fit onto the training set in the first 20 epochs despite our repeated attempts to alter hyper parameters[18]. Due of our limited computational resources, we decided to abandon further study along this road.

SeNet50 was another pre-trained model we examined. It is a deep residual network with 50 layers. Given that SeNet50 and ResNet50 have identical structural designs, we didn't put much work into optimizing this model.



Despite having just 16 layers, which is significantly shallower than ResNet50 and SeNet50, VGG16 is more complex and has many more parameters. We kept the pre-trained layers completely frozen and added two FC layers of size 4096 and 1024, respectively, and a 50% dropout.

C. Mobile Web App

We believed it would be hard and interesting to bring our research to the actual world by creating a mobile web app to run our model rather than taking a purely theoretical approach.

Creating a mobile web application to run our model in real-time on the device. We carefully analyzed the appropriate assessment criteria for our model given the memory, disc, and computational constraints of mobile devices.

We came to the conclusion that low memory/disk requirements and quick prediction speeds were considerably more essential than slight accuracy gains.

Our satisficing parameter was on-device recognition speed, and our optimizing metric was accuracy preservation. Due to this, we looked at smaller networks and eventually used the five-layer CNN model created by B.-K. Kim et al.

The biggest challenge in getting the model tuned to function well with our app was dataset mismatch. Contrary to the images in our training datasets, those captured via the web app usually displayed poor lighting and tilted angles.

We get around this by retaining 20% of the data from our web app in the test set and randomly distributing the other 80% into the training set along with all of the other dataset’s images.

After training for 120 epochs without altering the hyper parameters, we obtained an accuracy of 69.8% on the web app test set with a 40ms recognition speed, which was sufficient for our assessment criterion.

The user's face is recognized, cropped, and resized in our web app's architectural framework using TensorFlow.js, React.js, and face-api.js before being sent as a 48x48 picture with a single gray scale channel to our model. Additionally, model weights are compressed using tensor flows converter before being downloaded to the user's device to reduce their memory and disc footprint.

D. Android Application

Technology advancement has led to an increase in mobile device usage in recent years. As a consequence, deep learning and machine learning models may be installed on mobile devices. Due to the storage restrictions of mobile devices, deep learning models should be optimized before being used on those devices. As a consequence, the optimized model (.pb) for the original model is generated first. Here, the transfer learning model was improved and made available as a mobile Android application.

The Android application has a camera activity that snaps a photo of the other person. After picture capture, the color image is converted into a gray scale image with a size of 48*48 pixels. Following that, the programmer will predict which expression class each image falls into. The textual form of the anticipated statement will then be converted into audio in order to assist persons who are blind.

E. Tools and Frameworks →

1) *Google Collaborator:* The Google Collaborator is a free cloud-based Jupyter notebook. It aids in the development and operation of several deep learning models. Support for CPU, GPU, and TPU[12] is provided. The majority of machine learning and deep learning projects frequently use it.
2) *Android Studio:* A free integrated development environment (IDE) for creating android applications is called Android Studio. It offers support for a variety of programming languages, including java, kotlin, and others. Additionally, it supports running the programme on emulators.
3) *Java:* The most popular programming language for creating Android applications is Java. Java is effective for creating Android applications because of its many qualities, including its simplicity, dependability, independence from platforms, and others.
4) *Python:* The majority of deep learning and machine learning models are built on the Python programming language. This general purpose language is useful for building different machine learning and deep learning models due to its simplicity and simple syntax.
5) *Tensorflow:* The open source library Tensor flow is used for many different things, including classification and prediction. This framework aids in the development of sophisticated[13] deep learning and machine learning applications. It is a Google invention that aids with symbolic computation.
6) *Keras:* On top of Tensor flow, the Keras deep learning API runs. It is a straightforward Python package that makes it easier to import photos.

V. MOBILE WEB APP

Instead of adopting a purely theoretical approach, we felt it would be interesting and difficult to apply our research to the actual world by creating a mobile web app that would allow users to run our model in real-time on their device.

We carefully analyzed the appropriate assessment criteria for our model given the memory, disc, and computational constraints of mobile devices. We came to the conclusion that low memory/disk needs and quick prediction rates were considerably more essential than little accuracy increases. We retained accuracy as our optimizing goal and settled on a satisfying metric of 100 ms recognition speed on-device. This led us to investigate shorter networks and subsequently adopt the five-layer CNN model developed by B.-K. Kim et al. [6]. There were a number of difficulties in tuning the model to work properly with our app, most notably dataset mismatch. Images taken by the web app frequently exhibited poor lighting and slanted angles, unlike those in our training datasets. We overcome this by retaining 20% of our web app dataset in the test set and randomly distributing 80% of it into the training set along with all the photographs from the other datasets. Without changing the hyper parameters during training, we were able to attain an accuracy of 69.8% on the 40ms web app test set.

VI. PROPPOSED

Although there are several face Expression Reorganization datasets accessible online, their image size, color, and particularly their format, as well as their labeling and directory structures, all differ significantly. FER projects may be divided into two categories based on their methodology.

A. Appearance Based on Images

The first method uses the Point Contour Detection Method (PCDM) to increase the accuracy of the mouth and eyes in the submitted picture for recognition. It uses the Rough Contour Estimation Routine (RCER) to extract features from the uploaded image's eyebrows, eyes, and mouth.

For the purpose of easily checking the dataset's structure, they identified more than 30 face characteristic points for the image's eye, mouth, and brow in order to distinguish facial expression.

For that, they used 80 face photos from the pre-dataset with 128 × 128 pixel resolution and equal lighting, distance, and backdrop[17] settings. After putting this theory into practice, they discovered that this method produces an output with a 92.1% recognition rate.

Improve method for Face PCA (Principal Component Analysis) is used to recognize a picture from a digital facial image. In this study, they break the picture down into tiny tuples of feature images or Eigen faces..

They first generate a training dataset of the more than 30 different types of photos stated above in order to compare the results. Once the face picture that was uploaded had been pre-processed, it was compared to training data that had previously been added into the dataset.

When numerous face photos are supplied, the success rate is highest, but processing time is lengthy. For this study, they used the FACE94[14] database, which resulted in a 35% reduction in processing time compared to PCA's initial processing time. With this new technique, they also achieved a 100% recognition rate.

B. Techniques for Model-based Recognition

Using the PCA[15] methodology, a method for Eigen faces-based facial expression identification extracts the features from the input picture and tests them using training data.

Based on universal expressiveness, they organized the training set into six fundamental groups. A linear filter for edge detection in image processing is called a Gabor filter after Dennis Gabor.

Gabor filters[16]' representations of frequency and orientation are comparable to those of the human visual system and have been shown to be very useful for representing and differentiating textures. A 2D Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave in the spatial domain.

VII. CONCLUSION

At the conclusion of this project, I would say that I learnt a lot from several sources to finish it. I used a variety of techniques to finish this project successfully. When we started this project, we wanted to apply FER models to the actual world and first attain the maximum accuracy possible.

Next, we looked at a number of models, such as shallow CNNs and pr-trained networks based on SeNet50, ResNet50, and VGG16[18]. We used class weights, data augmentation, and supplementary datasets to reduce the FER2013's innate class imbalance. The maximum accuracy we were able to attain was 75.8% by combining the seven models. Accuracy this is the highest accuracy we achieved.

Additionally, we discovered that our models trained to concentrate on essential face cues for emotion recognition through network interpretability. Additionally, by creating a mobile web application with real-time recognition speeds, we showed that FER models could be used in the actual world.

By creating our own training datasets and fine-tuning our architecture, we were able to overcome data mismatch concerns and run on-device with low memory, storage, and compute needs.

VIII. RESULT ANALYSIS

The proposed approach using DenseNet-161 is taken into account for performance comparison because it demonstrated the greatest accuracy among the eight DCNN models taken into consideration. The KDEF and JAFFE datasets show that the suggested technique outperforms any traditional feature-based strategy.

For JAFFE, the feature-based approaches that first used face recognition [8] still have the best recognition accuracy, with a recognition accuracy of 98.98% for equally dividing the training and test sets.

The technique with feature representation using DWT with 2D-LDA and classification using SVM [9] has the greatest accuracy of 95.70% in the 10-Fold CV example.

In contrast, the suggested technique outperformed all existing feature-based methods with an accuracy of 99.52% in the 10-Fold CV on JAFFE and 100% on randomly chosen 10% test samples.

When applied to the KDEF dataset, the suggested technique outperformed the existing ones and attained an accuracy of 98.78% (on randomly chosen 10% test samples).

Notably, accuracy on 10% test samples is 82.40% by [7], taking just chosen 980 frontal photographs into account, and the effectiveness is lower than the suggested technique. The proposed FER's performance should be compared to other deep learning techniques, particularly CNN-based techniques, because it is built on the DCNN model through TL.
When simply taking into account frontal pictures, the study using SCAE plus CNN [3] demonstrates an accuracy of 92.52% on the KDEF dataset. On the JAFFE dataset, the hybrid CNN and RNN approach [56] has an accuracy of 94.91%.

IX. FUTURE WORK

A. Analysis using Principal Components (FACE 94): Our experiment will be repeated on bigger and diverse databases.
B. Eigen faces plus PCA (CK, JAFFE) Future work will focus on creating a facial expression detection system that integrates the user's body motions and facial expressions to recognize 83% Surprise in CK and 83% Happiness in JAFFE.
C. They focus on incorporating global and local color histograms as well as factors related to the forms of objects in photos when using the 2D Gabor filter (Random Images)
D. PCA + AAM [24] (FG-NET consortium image sequences) Expand your efforts to recognize faces and their expressions in 3D photos.
E. A large, publicly accessible AU database containing singly-occurring AUs will be created using Gabor + SVM using HAAR + Adaboost (Cohn-Kanade database) to aid in future study
F. Features similar to dynamic HAAR [26] (CMU expression Database + Own Database):
Face recognition based on video may be implemented using this technique.

X. CODE

We have tried to develop the code on the basis of the requirement of the project so mentioned repository show how actually code will be of the facial expression recognitions using deep learning.
https://github.com/PrajjawalTiwari29/FER-code

XI. CONTRIBUTIONS

A. Prajjawal Tiwari.
1) Managing project .
2) Dataset .

B. Palak Singh.
1) Mobile web app development .
2) Future work.

C. Navneet Kumar.
1) Preprocessing of auxiliary data and datasets.
2) Interpretability of networks and error analysis.

D. Arpit Rai.
1) Methods.
2) Conclusion.

E. Prbhav Attray.
1) Models.
2) Mobile web app.

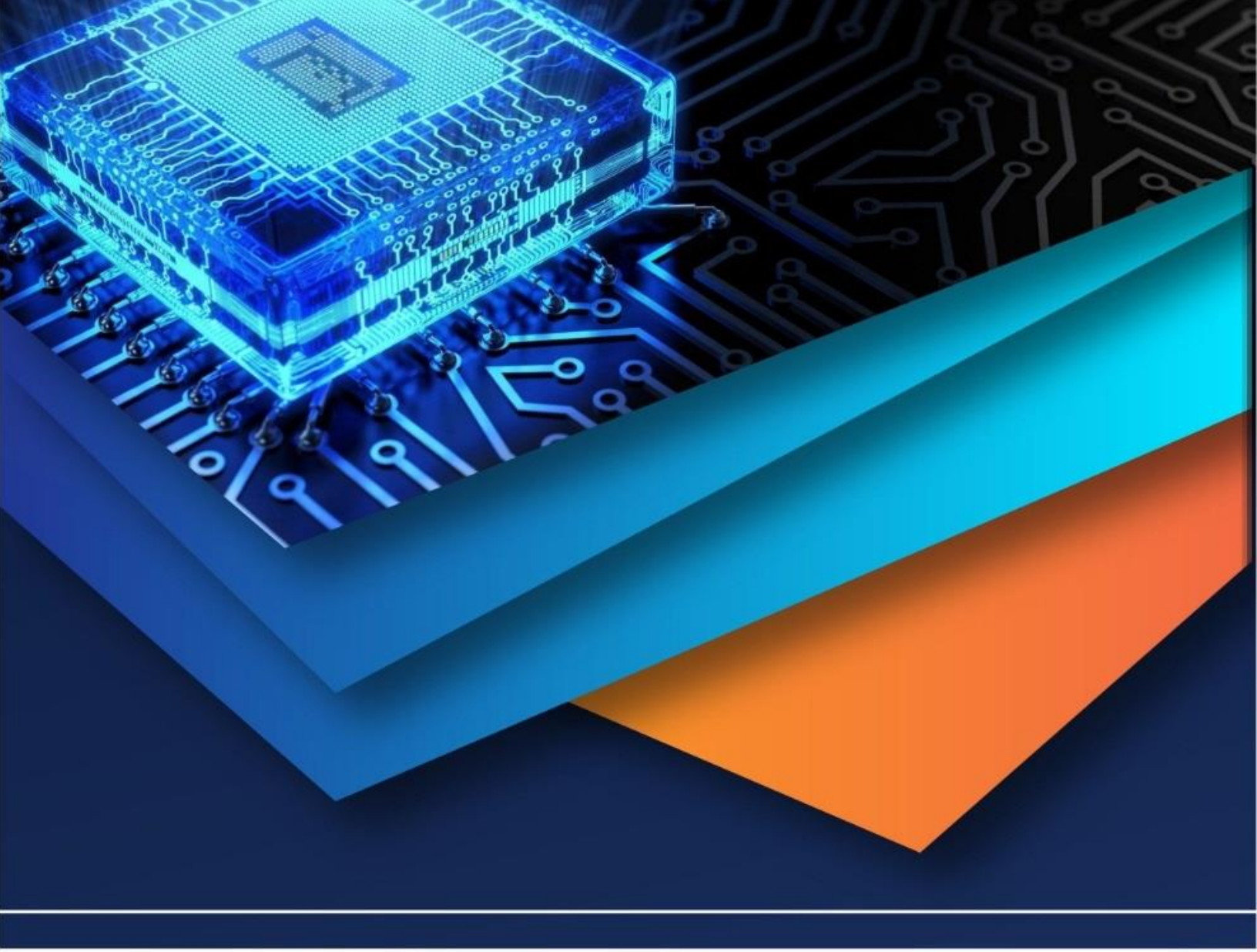
XII. ACKNOWLEDGEMENTS

We are really thankful to Assistant Professor Mr. Nizam Uddin Khan from the IMS Engineering College in Ghaziabad's Computer Science and Engineering department for his assistance in assisting us with the application of our research to the real world.
Its our privilege to express our sincere regards to our project guide, Prof. Mr. Nizam Khan for his valuable inputs, able guidance, encouragement, cooperation and constructive criticism throughout the duration of our project.

We sincerely thank the Project Assessment Committee members for their support and for enabling us to present the project on the topic.
" Recognizing Facial Expressions Using Deep Learning."

REFERENCES

[1] Y. Tang, "Deep Learning using Support Vector Machines," in International Conference on Machine Learning (ICML) Workshops, 2013.
[2] Quinn M., Sivesind G., and Reis G., "Real-time Emotion Recognition From Facial Expressions", 2017.
[3] Wang J., and Mbuthia M., "FaceNet: Facial Expression Recognition Based on Deep Convolutional Neural Network", 2018.
[4] Challenges in representation learning: Facial expression recognition challenge http://www.kaggle.com/c/challengesin-representation-learning-facial-expression recognition.
[5] H. Jung et al., "Development of deep learning-based facial expression recognition system", Proc. 21st Korea-Jpn. Joint Workshop Frontiers Comput. Vis. (FCV), pp. 1-4, 2015.
[6] Martina Rescigno, Matteo Spezialetti1 and Silvia Rossi1, "Personalized models for facial emotion recognition through transfer learning",Springer,2020.
[7] A. Sehgal and N. Kehtarnavaz, "Guidelines and benchmarks for deployment of deep learning models on smartphones as real-time apps", Machine Learning and Knowledge Extraction, vol. 1, no. 1, pp. 450-465, 2019.
[8] I. M. Revina and W. S. Emmanuel, "A survey on human face expression recognition techniques" in Journal of King Saud UniversityComputer and Information Sciences, 2018.
[9] Lei Xu, Minrui Fei, Wenju Zhou and Aolei Yang, "Face expression recognition based on convolutional neural network", Australian & New Zealand Control Conference (ANZCC), pp. 115-118, 2018.
[10] M. Banerjee, S. Bose, A. Kundu and M. Mukherjee, "A Comparative Study: Java Vs Kotlin Programming in Android Application Development", International Journal of Advanced Research in Computer Science, vol. 9, no. 3, pp. 41-45, 2018
[11] 2018; Multimedia Tools and Applications, vol. 77, pp. 22821-22839; C. Tang, "Twelve-layer deep convolutional neural network with stochastic pooling for tea category categorization on GPU platform."
[12] "A pansharpening strategy employing spectral graph wavelet transformations and convolutional neural networks," International Journal of Remote Sensing, vol. 42, pp. 2898-2919, April 2021. N. Saxena and R. Balasubramanian
[13] Ensemble of Deep Neural Networks with Probability-Based Fusion for Facial Expression Recognition. Wen, G.; Hou, Z.; Li, H.; Li, D.; Jiang, L.; Xun, E. 2017's Cogn. Comput. 9, 597–610 Using Google Scholar [CrossRef].
[14] Evaluation of Data Augmentation Techniques for Facial Expression Recognition Systems by Porcu, Floris, and Atzori. 2020, 9; Electronics; 1892. Using Google Scholar [CrossRef]
[15] Visualizing Deep Convolutional Neural Networks Using Natural Pre-images. Mahendran, A.; Vedaldi, A. Journal of Computer Vision, 2016, 120, 233–255. Using Google Scholar [CrossRef] [Version Green]
[16] The Japanese Female Facial Expression (JAFFE) Database. Lyons, M.J.; Akamatsu, S.; Kamachi, M.; Gyoba, J.; Budynek, J. http://www.kasrl.org/jaffe download.html is a website where it is accessible. (viewed on February 1, 2021).
[17] Information Processing and Management, vol. 58, article ID: 102439, 2021. D. S. Guttery, "Improved Breast Cancer Classification Through Combining Graph Convolutional Network with Convolutional Neural Network,"
[18] S. Sharma, S. Kumar, and R. Mehra (2021). For the multi-classification of breast cancer, an optimised CNN in combination with a successful pooling method was used. Early Access Article on IET Image Processing. IPR2.12074, doi: 10.1049



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY