Comparative Analysis of Facial Expression Recognition Algorithms

Ujjwal MK1, Sunil Parameswaran1 , V. Guna Chowdary1, Varun Bharadwaj1 and Dayananda P2

1 Department of Computer Science and Engineering, PES University Bangalore, India

{ujjwalmk16, sunil05.parameswaran, v.gunachowdary,varunbwaj}@gmail.com

2 Dept. of Information Technology Manipal Institute of Technology Bengaluru, India

{[dayananda.p](mailto:dayananda.p@manipal.edu)}[@manipal.edu](mailto:dayananda.p@manipal.edu) (Correspondence Author)

**Abstract.** Facial expression recognition has emerged as a significant area of research in computer vision and pattern recognition due to its wide range of applications which include emotion analysis, human-computer interaction among many others. In recent times multiple facial expression recognition algorithms have come into light. Each of these techniques employ different techniques and principles. This paper aims to provide a comprehensive analysis and comparison of these algorithms by evaluating their performance, strengths, capabilities and limitations to provide an overall understanding of the algorithm. By doing the comparative analysis this paper aims to facilitate better understanding of these algorithms.

**Keywords:** Convolutional Neural Networks(CNNs), Artificial Neural Networks(ANNs), Recurrent Neural Networks(RNNs), Adaboost, XGBoost, Haar-Cascade Classifier, PCA, LDA, Binary Neural Networks(BNN).

**1 Introduction**

Emotions play an important role in human brain dynamics. Facial expressions are the most prominent and explicit emotional features, due to their communicability. Hence, there is a crucial need to be able to understand and interpret the expressions for communications and interactions. The corresponding facial expressions for moods are given in Table 1 [1]. This task is simple and easy for the human brain. However, building a computer system to do the same is a challenging task. Each person’s face is different and their facial features differ from one human to another, even though they are quite similar. Hence, there is a need for a deep-learning model to develop its own non-deterministic algorithm for this type of problem.

Nowadays, with the development of computer technology, facial expression recognition is widely applied in different realms. Facial Expression Recognition is a technique that extracts and identifies the expressions from a person’s face. In this technique, the extraction of expression features is the most important task. It is a complicated problem that involves steadily extracting the expression features even from an image that contains plenty of noise. There are various algorithms available that can be used to detect the expressions on a face. But, the algorithms are not perfectly accurate and do have a scope for improvement.

There are two types of feature extraction methods:

* Geometric Based method: These algorithms primarily emphasize on lifelong facial features like eyes, eyebrows, nose, forehead and mouth to determine the position and shape of the face. These distinct facial components are used to feature vectors that accurately represent the face.
* Appearance Based method: These algorithms emphasize short-lived features like bulges, wrinkles, and other features that reflect the changes in face texture, pixel values and intensity. These algorithms analyze and extract information related to the changing patterns, surface details, color intensity, and pixel values in order to capture and represent the evolving appearance of the face.

This paper primarily focuses on describing and comparing the various facial expression recognition algorithms available by examining their principles, working and their performance. Hence, there is a need to highlight the strengths and weaknesses of the algorithms in real-world applications. To achieve this, a selection of an extensive range of algorithms are needed.

By doing a comparative analysis of various facial expression recognition algorithms, the paper aims to contribute to the ongoing efforts of increasing the performance of these algorithms. This paper also serves as a reference for developers looking to explore this domain and it can also help in deciding the appropriate algorithm when building a real-world application. Happiness, sadness, surprise, fear, disgust, and anger are six basic facial expressions defined by modern psychology the following table shows some sample facial features for the expressions.[2]

**Table 1.** Emotions and their corresponding facial expressions

| Emotion | Prominent Facial Features |
| --- | --- |
| Anger | Furrowed brows, tightened jaw, intense or hostile gaze |
| Disgust | Wrinkled nose, raised upper lip, squinted eyes |
| Fear | Widened eyes, raised eyebrows, open mouth |
| Happy | Smiling mouth, raised cheeks, crinkled eyes |
| Sad | Downturned mouth, drooping eyebrows, teary eyes |
| Surprise | Raised eyebrows, widened eyes, open mouth |
| Neutral | Relaxed facial muscles, neutral expression |

# 2 Various Facial Expression Recognition Algorithms

**2.1 ANN**

Artificial Neural Networks are widely used in expression detection and are known for their ability to simulate the human brain. ANNs are computational models that possess unique capabilities like adaptability, data clustering and characterization [3]. They require labeled data to work with. The ANN framework typically consists of four stages: detection and pre-processing, training, ANN implementation, and testing. The model’s performance is evaluated using fitness functions with the aim being to minimize the deviations as much as possible.

**2.2 ANN utilizing Harmony Search Algorithm**

The performance of ANN is highly dependent on the configuration of the hidden layers and the learning rate. Determining the optimal values of these is usually done by using trial-and-error method, which is time-consuming and may not guarantee finding the best values. To overcome this drawback, Harmony Search Algorithm(HSA) can be applied [4]. This is a meta-heuristic algorithm which works by evaluating a fitness function. HSA is preferred over other algorithms since it is simpler to implement and finds the optimum solution is fewer iterations. HSA utilizes two key parameters : Harmonic Memory Consideration Rate(HMCR) and Pitch Adjusting Rate(PAR). The optimization process is divided into four stages :

**2.2.1 Initializing the Harmony Memory.** HSA utilizes a population matrix, where each harmony corresponds to an input vector in the search space with m dimensions. Input Vector Z can be expressed by :

Zij=[Xij1Xij2...Xijm] **(1)**

**(2)**

Where, i = 1,2…HMS j =1,2….m and are upper and lower bounds of the element p.

**2.2.2 Improvising the new harmony.** By taking into consideration the pitch, memory consideration and randomization, a new harmony input vector Yij will be constructed. It can be modified by the existing harmony vectors using PAR.

**2.2.3 Updating harmony memory.** After creation of a new vector in the second stage, it is evaluated using the fitness function f(Yij). If the fitness function value is better than the fitness function of the worst vector Zijworst, then it will be replaced by Yij. Else, the new vector will be ignored.

**2.2.4** **Stopping Condition.** When the maximum number of improvisations,NI, is reached, HSA will be terminated. Otherwise, steps 3 and 4 will be repeated.

## 2.3 AdaBoost with Haar Cascade Classifier and PCA along with LDR

The task of real-time face recognition and expression detection is highly challenging due to factors like complex background subtraction and the inherent variability of human faces. To address these challenges, machine learning techniques can be effectively employed [5].

AdaBoost, an adaptive algorithm, can be utilized in conjunction with other classifiers to prioritize previously misclassified instances. However, it is important to note that AdaBoost may exhibit sensitivity to noise and outliers present in the data.

Another valuable approach is the Haar Cascade Classifier, which leverages Haar-like features and integral images to swiftly reject(crop) unlikely object regions.[6]

To further enhance performance, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) techniques can be incorporated.

PCA facilitates dimensionality reduction by transforming correlated variables into uncorrelated ones, while LDA identifies optimal feature combinations for effective class separation.

Combining PCA and LDA not only improves execution time and memory requirements but also enhances overall accuracy.

By employing these advanced techniques, real-time face recognition and expression detection can be accomplished with increased efficiency and accuracy.

Another valuable approach is the Haar Cascade Classifier, which leverages Haar-like features and integral images to swiftly reject(crop) unlikely object regions.[6]

To further enhance performance, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) techniques can be incorporated.

PCA facilitates dimensionality reduction by transforming correlated variables into uncorrelated ones, while LDA identifies optimal feature combinations for effective class separation.

Combining PCA and LDA not only improves execution time and memory requirements but also enhances overall accuracy.

By employing these advanced techniques, real-time face recognition and expression detection can be accomplished with increased efficiency and accuracy.

## 2.4 Binary Neural Networks (BNN)

Binary Neural Networks (BNN) represent a unique form of Artificial Neural Networks (ANNs) where the weights are encoded as binary values (0s and 1s) rather than real numbers. In order to extract meaningful features from the dataset, two algorithms, namely Local Binary Pattern Rotation Invariant (LBPROT) and Viola-Jones, are employed. LBPROT captures local binary information, while Viola-Jones intercepts the facial image.[7] A key advantage of BNNs is the compression of traditional 32-bit activations found in Convolutional Neural Networks (CNNs) into just 1 bit. This compression significantly reduces computation requirements, making BNNs suitable for resource-constrained devices.

The Binary Convolutional Activation can be mathematically represented as follows:

**(3)**

Forward Propagation can be re-expressed for binary weights as:

**(4)**

Here, "Qw" and "Qa" are scale factors, contributing to the transformation. To ensure effective backpropagation, a clip function is utilized:

clip(x,-1,1) = max(-1,min(1,x)) **(5)**

The model's framework draws inspiration from RESNET and incorporates additional route and reorg layers. The reorg layer extracts the input feature map and samples it based on position within the neighborhood. A combination of convolution, batch normalization, and pooling layers is referred to as a "Block." The routing layer concatenates the feature maps after all "Block" operations to form the input features for the subsequent layer. The final output is obtained using the popcount function. By cascading these "Blocks," the model achieves a feature-rich input map and attains higher accuracy while maintaining resource efficiency [7]. This model is specifically designed for edge computing scenarios, as it outperforms traditional CNNs in terms of speed and computational requirements.

## 2.5 Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) are widely used for image classification tasks. They operate by converting input images into arrays of pixels, which represent the image's pixel values. A regular VGG based model has several layers, these layers are combination of multiple convolution layer(ex: Conv2D), activation layer and MaxPooling layers, these layers help the model in feature extraction[8].Through a series of such layers, these networks extract features from the input image by applying different filters to identify patterns. To introduce non-linearity, the Rectified Linear Unit (ReLU) activation function is commonly employed.[9] Pooling layers are then used to reduce parameter count and spatial dimensions in the feature maps.

Finally, fully connected (FC) layers flatten the feature vectors, allowing for the final classification. In the literature, various CNN models have been proposed, such as LeNet, AlexNet, VGG-Net, GoogleNet, and ResNet. Each model has its own input image size, including 227x227 for LeNet, 224x224 for AlexNet, 229x229 for VGG-Net, and 224x224 for both GoogleNet and ResNet.[10]

## 2.6 CNN, XGBoost and Model-Fusion

XGBoost functions as an ensemble tree model, aiming to combine lower-accuracy models into a single high-accuracy model. It stands out by fully utilizing multi-core CPUs, unlike traditional boosting algorithms [11]. The integration of XGBoost and CNN enhances feature extraction capabilities while simultaneously improving the accuracy of XGBoost.

The objective function, denoted as F(x), involves the number of samples (n), classification and regression trees (K), a loss function , and regularization function is

**(6)**

By employing the second-order Taylor expansion and adding regularization to prevent overfitting, the optimal solution is obtained. Additionally, the importance of each feature is determined based on the frequency of tree node splits.

Model Fusion technology, with voting and learning strategies, combines multiple learners to achieve superior generalization performance [11].

In the voting method, the recognition probability for the jth face is calculated as a weighted sum of recognition probabilities from individual learners. The recognition result for each face is determined by selecting the category with the highest probability .

**(7)**

## 2.7 Recurrent Neural Network (RNN)

To extract high-level information, the estimation of model parameters that accurately describe a face within an image is necessary. Model fitting tackles this task by minimizing an objective function that measures the fit between the model parameterization and the given image. Traditionally, the objective function is manually designed, often based on pixel error between the rendered model and the underlying image content. This model primarily focuses on a different approach by learning the objective function from annotated example images.

The objective function, denoted as f(I,p), quantifies the accuracy of how a parameterized model p fits to an image. The fitting algorithm then searches for the optimal model parameters that maximize the objective function. However, the specific algorithms used for model fitting are beyond the scope of this paper.

Designing the objective function manually is a time-consuming process that relies on the designer's domain knowledge and subjective evaluation of its performance on example images and parameters. To address this, learning the objective function instead can be used. Properties of ideal objective functions, can be used, where the global minimum corresponds to the correct contour point position, and no other local minima exist [12].

fn⋆(I,u) = |u−xˆ⋆n|  **(8)**

By learning the objective function, this approach considers multiple image features simultaneously, enabling efficient and accurate model fitting. The learned objective function incorporates the desirable properties of an ideal objective function, leading to improved runtime performance and accuracy in contour point localization.

To enhance recurrent neural networks (RNNs), the long short-term memory (LSTM) architecture is commonly employed. LSTM cells utilize multiplicative gates to effectively retain information over extended time periods, thereby increasing the network's temporal context.[13]

## 2.8 Support Vector Machine

SVM is a binary classification technique that seeks optimal linear decision boundaries by minimizing structural risk[14]. The decision boundary is a weighted combination of training set elements known as support vectors, which delineate class boundaries. In cases where linear separation is not feasible, SVM employs kernel functions to map data into higher-dimensional spaces for improved separation. The input consists of labeled training data {(xi, yi)}, where xi represents the data and yi denotes the label (-1 or 1). The output includes support vectors si, coefficient weights i, class labels yi, and a constant term b.

w.z + b = 0 **(9)**

The above equation represents the decision boundary, where w is computed as the weighted sum of support vectors i multiplied by their corresponding coefficients si and labels yi. The goal is to maximize the margin, i.e., the distance between the hyperplane and the nearest data points, thereby achieving robust generalization.

**(10)**

().z + b = 0  **(11)**

To represent a facial image, it is converted into a vector P in an N-dimensional face space (RN). Face space can be defined as the vectorized original pixel values or another feature space. The objective is to maximize the margin, ensuring a larger separation between facial features in the face space, leading to more robust and accurate classification.

**3. Comparative Analysis**

**Table 2.** Summary of the findings.

| Sl No. | Technique  used | Features | Model Statistics | Dataset Used |
| --- | --- | --- | --- | --- |
| 1 | ANN | Adapt, characterize and cluster data.  Non-linear Mapping.  Hierarchical feature extraction.  Four stages : detection and pre-processing, training, implementation, testing. | Model Accuracy :  82%[4] | MATLAB environment  of 10 people |
| 2 | ANN utilizing Harmony Search Algorithm | HSA optimizing weights and biases.  Implemented based on fitness  function evaluation.  Enhanced global search.  Improved convergence speed  Robust and stable. | Model Accuracy:  94%[4] | MATLAB environment of 10 people |
| 3 | AdaBoost with Haar Cascade Classifier and PCA along with LDR | Subsequent classifiers are built in favor of previously misclassified instances. AdaBoost is highly sensitive to outliers. Uses “Haar-like” features to identify the region in which a face is present. PCA transforms many potentially correlated variables into uncorrelated ones. LDR identifies feature combination that distinguishes multiple classes | Model Accuracy : 88%[3] | FERET |
| 4 | BNN | Represents the weights in 0s and 1s. Uses bitwise XOR and bit counting operations which are implemented efficiently. Cannot achieve high accuracy for complex tasks and the sensitivity to noise in binary computations. Used mostly for edge deployment and resource-constrained devices. | Model Accuracy: 83%[7] | FER 2013[15] |
| 5 | CNN | Convolution layers apply learnable filters for feature extraction. Pooling layers down sample spatial dimensions. Non-linear activation functions introduce non linearities. Interpretability and visualization aid in understanding model decisions | Model Accuracy: 91%[19] | FER 2013 |
| 6 | CNN, XGBoost and Model-Fusion | XGBoost is an ensemble tree model that combines multiple other models to boost accuracy. Can utilize multi-core CPUs. Combination of XGBoost with CNN increases XGBoost’s accuracy and CNN’s feature extraction capability  Model-Fusion combines multiple learners and obtains better generalization | Model Accuracy: 72.54% [16] | FER 2013 |
| 7 | RNN | Network Layers are connected to itself  More robust to shifts and distortions  Can make flexible use of temporal context. LSTM extends range of temporal context | Model Accuracy : 41% [17] | FER 2013 |
| 8 | SVM | Finds optimal linear decision surfaces by minimizing structural risks. Support Vectors characterize boundaries between classes. Can handle binary and multi classification problems. Widely used in outlier detection and anomaly detection tasks | Model Accuracy : 77%[18] | FER 2013 |

# 4. Conclusion

Facial expression recognition has evolved into a pivotal realm of investigation, finding applications across emotional analysis, human-computer interaction, and beyond. This study delves comprehensively into the landscape of facial expression recognition algorithms, aiming to illuminate their distinct strengths, limitations, and underlying mechanisms that shape their performance. Through systematic comparative analysis, a range of prominent algorithms, including Artificial Neural Networks (ANNs), Binary Neural Networks (BNNs), ANNs augmented by the Harmony Search Algorithm, AdaBoost combined with Haar Cascade Classifier, PCA integrated with LDA, Convolutional Neural Networks (CNNs), CNNs enriched by XGBoost and Model-Fusion, Recurrent Neural Networks (RNNs), and Support Vector Machines (SVMs), are dissected. Each algorithm's unique principles and methodologies are evaluated in the context of facial expression recognition. This examination reveals ANNs' adeptness at learning from labeled datasets, the efficacy of AdaBoost and PCA with LDA for real-time expression detection, and CNNs' consistent accuracy. Furthermore, the collaborative potential of CNNs with XGBoost and Model-Fusion is explored, tapping into their collective strength. The capacity of RNNs to capture temporal context and SVMs' proficiency in delineating optimal linear decision surfaces are also unveiled. This paper delves into the significance of feature extraction techniques, spanning geometric-based and appearance-based methods. This study contributes to the wealth of knowledge on facial expression recognition algorithms, equipping researchers, practitioners, and developers with insights essential for the design and implementation of real-world facial recognition applications.

**References**

1. S. A. M. Silveira and V. P. Mishra, "Development of Simulator to Recognize the Mood using Facial Emotion Detection," 2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM), Gautam Buddha Nagar, India, 2022, pp. 488-490, doi: 10.1109/ICIPTM54933.2022.9754012.

2. M. Kumar and S. Srivastava, "Emotion Detection through Facial Expression using DeepLearning," 2021 5th International Conference on Information Systems and

3. Thai Hoang Lee, “Applying Artificial Neural Networks for Face Recognition”, 2011, pp. 1-15, doi:10.1155/2011/673016

4. Maryam Mahmood Hussein, Ammar Hussein Mutlag, Hussain Shareef, “An Improved Artificial Neural Network Design for Face Recognition utilizing Harmony Search Algorithm”, IOP Conf. Series: Materials Science and Engineering 745 (2020) 012038, pp. 1-9, doi:10.1088/1757-899X/745/1/012038

5. K. Susheel Kumar, Shitala Prasad, Vijay Bhaskar Semwal, R C Tripathi, “REAL TIME FACE RECOGNITION USING ADABOOST IMPROVED FAST PCA ALGORITHM”, International Journal of Artificial Intelligence & Applications (IJAIA), Vol.2, No.3, July 2011, pp. 45-56 Computer Networks (ISCON), Mathura, India, 2021, pp. 1-4, doi: 10.1109/ISCON52037.2021.9702451.

6. Seville Anna Maria Silveria, Ved P Mishra, “Development of Simulator to Recognize the Mood using Facial Emotion Detection ”, 2022 2nd International Conference on Innovative Practices in Technology and Management (ICIPTM), pp. 488-490, DOI: 10.1109/ICIPTM54933.2022.9754012

7. G. Zhao, W. Wei, X. Xie, S. Fan and K. Sun, "An FPGA-Based BNN Real-Time Facial Emotion Recognition Algorithm," 2022 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA), Dalian, China, 2022, pp. 20-24, doi: 10.1109/ICAICA54878.2022.9844526.

8. S. K. LokeshNaik et al., "Real Time Facial Emotion Recognition using Deep Learning and CNN," 2023 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2023, pp. 1-5, doi: 10.1109/ICCCI56745.2023.10128259.

9. L. S. K. Dasari, F. Baig, D. J. Sai, G. K. Michael, A. Laxman and A. Madhuri, "CNN based Mood Detection using Facial Expression," 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2023, pp. 1457-1462, doi: 10.1109/ICSSIT55814.2023.10060991.

10. R. A. Sukamto, Munir and S. Handoko, "Learners mood detection using Convolutional Neural Network (CNN)," 2017 3rd International Conference on Science in Information Technology (ICSITech), Bandung, Indonesia, 2017, pp. 18-22, doi: 10.1109/ICSITech.2017.8257079

11. Xianghai Xu et al 2021 IOP Conf. Ser.: Earth Environ. Sci. 645 012054

12. Alex Graves, Ju ̈rgen Schmidhuber, 2008, “Facial Expression Recognition with Recurrent Neural Networks”, Technische Universita ̈t Mu ̈nchen, Germany

13. M. Khazbak, Z. Wael, Z. Ehab, M. Gerorge and E. Eliwa, "MindTime: Deep Learning Approach for Borderline Personality Disorder Detection," 2021 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), Cairo, Egypt, 2021, pp. 337-344, doi: 10.1109/MIUCC52538.2021.9447620.

14. Shah, Parin M., "Face Detection from Images Using Support Vector Machine" (2012). Master's Projects. 321.

15. Manas Sambre, FER-2013, <https://www.kaggle.com/datasets/msambare/fer2013>.

16. Punuri, S.B.; Kuanar, S.K.; Kolhar, M.; Mishra, T.K.; Alameen, A.; Mohapatra, H.; Mishra, S.R. Efficient Net-XGBoost: An Implementation for Facial Emotion Recognition Using Transfer Learning. Mathematics 2023,11,776. https:// doi.org/10.3390/math11030776

17. R. Qayyum et al., "Android based Emotion Detection Using Convolutions Neural Networks," 2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), Dubai, United Arab Emirates, 2021, pp. 360-365, doi: 10.1109/ICCIKE51210.2021.9410768.

18. Saeed, Saeeda & Baber, Junaid & Bakhtyar, Maheen & Ullah, Ihsan & Sheikh, Naveed & Dad, Imam & Ali, Anwar. (2018). Empirical Evaluation of SVM for Facial Expression Recognition. International Journal of Advanced Computer Science and Applications. 9. 10.14569/IJACSA.2018.091195.

19. Zahara, Lutfiah & Musa, Purnawarman & Prasetyo, Eri & Karim, Irwan & Musa, Saiful. (2020). The Facial Emotion Recognition (FER-2013) Dataset for Prediction System of Micro-Expressions Face Using the Convolutional Neural Network (CNN) Algorithm based Raspberry Pi. 1-9. 10.1109/ICIC50835.2020.9288560.