2024 Second International Conference on Networks, Multimedia and Information Technology (NMITCON) | 979-8-3503-7289-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/NMITCON62075.2024.10699155

Emotion Detection Using Haar-Cascade Classifier and CNN

1st Pallavi M O
School of CSE, Reva University
Bangalore
Department of MCA, Acharya Institute OF Technology
pallavimuthy88@gmail.com
Orchid ID:https://orcid.org/0009-0009-5381-2209

2nd Pundalik Chavan
School of CSE, Reva University
Bangalore
pundalik.chavan29@gmail.com
ORCID ID: https://orcid.org/0000-0001-5336-119

***Abstract*- To create a reliable and efficient system for detecting emotions by fusing a convolutional neural network (CNN) with the Haar cascade classifier for face detection. Background Study: Enhancing human-computer interaction in domains like mental health and customer service requires the ability to identify emotions. The efficacy and usability of traditional approaches are limited by their frequent difficulties with accurate emotion classification and real-time processing. Methods: The proposed technique combines the Haar Cascade Classifier with a Convolutional Neural Network (CNN). First, the Haar Cascade Classifier efficiently detects face regions, which reduces the computational strain on the CNN. The CNN, which has multiple convolutional and pooling layers, is then trained on a large dataset of various facial expressions to identify emotions. This integrated technique improves the speed and accuracy of real-time emotion recognition. Results: The integrated system obtained 92% accuracy, greatly outperforming solo techniques. The Haar Cascade Classifier ensures accurate face detection, which improves CNN emotion classification and overall processing time. Validation on a public dataset confirmed the system's better performance, indicating its suitability for real-time applications. Conclusion: The Haar Cascade Classifier combined with CNN provides a highly accurate and efficient solution for emotion detection. This hybrid technique increases real-time processing and classification accuracy, making it appropriate for applications such as real-time emotion monitoring, interactive gaming, and customer support platforms.**

Index Terms—Face Detection, Haar Cascade Classifier, Emotion Detection, Convolutional Neural Network, Human-Computer Interaction

I. INTRODUCTION

Emotion detection has become a pivotal area of study in human-computer interaction (HCI), enabling advancements in various fields such as mental health monitoring, customer service, and interactive gaming. The ability to accurately recognize human emotions from facial expressions allows systems to respond more empathetically and effectively to user needs, thereby enhancing user experience and engagement [1]. Traditional approaches to emotion recognition often rely on handcrafted features and classical machine learning algorithms. For example, Support Vector Machines (SVM) have been considered for emotion classification with moderate success. However, these methods generally struggle with the complexities and variability of human facial expressions, leading to suboptimal performance in real-world scenarios [2]. Recent advances in machine learning, particularly deep learning, have created new opportunities to improve emotion recognition accuracy. CNNs, with their ability to automatically learn and extract hierarchical features from raw data, have shown superior performance in image classification tasks, including emotion recognition. Despite their success, CNNs are computationally intensive, posing challenges for real-time applications [3] [4]. To address these challenges, paper proposes a hybrid approach that combines the Haar Cascade Classifier for efficient face detection with a CNN for precise emotion recognition. The Haar Cascade Classifier, introduced by Viola and Jones [5], is renowned for its rapid and accurate face detection capabilities. It uses a series of Haar-like features and a cascade of classifiers to identify facial regions within an image. The CNN component is designed with multiple convolutional and pooling layers, trained on a comprehensive dataset of facial expressions to classify emotions such as happiness, sadness, anger, and surprise. This end-to-end learning approach enables the CNN to capture complex patterns and nuances in facial expressions, leading to improved emotion classification performance. This high level of accuracy is attributed to the robust face detection provided by the Haar Cascade Classifier, which ensures that only relevant facial regions are processed by the CNN. Additionally, the hybrid approach enhances processing efficiency, making it suitable for real-time emotion detection applications. The structure of this paper is as follows: Section II reviews related work in emotion detection, Section III describes the proposed methodology, Section IV presents the experimental results and discussions, and Section V concludes the paper with suggestions for future research.

II. RELATED WORK

The effectiveness of the proposed method is validated through extensive testing on a publicly available dataset, consistently outperforming traditional emotion recognition systems. These findings highlight the hybrid approach's potential for a wide range of real-world applications, including real-time emotion monitoring systems, interactive gaming, and customer support platforms. Rao et al. (2024) [1] introduce a groundbreaking method combining Oppositional Brainstorm Optimization (OBSO) with deep learning for facial emotion recognition in autonomous intelligent systems. The OBSO algorithm enhances deep learning models by using oppositional-based learning strategies to diversify the search space and avoid local optima. Experimental results show that the OBSO-enhanced model outperforms traditional methods in accuracy and robustness, making it highly effective for real-time applications in intelligent systems.

Singh et al. (2023) [2] explore the integration of Haar Cascade classifiers and CNNs for facial emotion detection. The Haar Cascade is used for rapid face detection, while the CNN classifies the emotions. This hybrid approach leverages the speed of Haar Cascades and the accuracy of deep learning models, providing a robust framework for real-time emotion detection in various applications such as human-computer interaction and surveillance systems.

Subramanian et al. (2022) [3] propose a digital twin model for real-time emotion recognition in personalized healthcare settings. This model integrates sensor data, machine learning algorithms, and real-time analytics to continuously monitor and assess patients' emotional states. The system provides personalized interventions based on detected emotions, enhancing patient care by adapting to individual needs and improving mental health outcomes. Prakash et al. (2023) [4] present a comprehensive computer vision-based framework for assessing autistic children. The system analyzes interactions, emotions, human poses, and life skills using advanced image processing and machine learning techniques. This holistic assessment tool aids clinicians in diagnosing and monitoring autism spectrum disorders, showcasing the potential of technology in supporting clinical evaluations and personalized interventions for autistic children. Alonazi et al. (2023) [5] present an automated facial emotion recognition system using the Pelican Optimization Algorithm in conjunction with a deep CNN. The proposed system aims to enhance the accuracy and efficiency of emotion recognition by optimizing the CNN's parameters through the Pelican Optimization Algorithm. The study demonstrates significant improvements in recognition performance compared to traditional methods, making it a viable solution for real-time applications in various domains. Dwivedi, Verma, and Taran (2024) [6] investigate EEG- based emotion recognition using optimized deep-learning techniques. The research focuses on enhancing the accuracy of emotion recognition by refining deep learning models using various strategies. The experimental results indicate that the proposed method outperforms existing approaches in recognizing emotions from EEG signals, offering a promising
tool for applications in neuroscience and mental health.

Jain et al. (2023) [7] propose an automated hyperparameter- tuned deep learning model for facial emotion recognition specifically designed for autonomous vehicle drivers. The model employs advanced hyperparameter tuning techniques to enhance the accuracy and robustness of emotion recognition under various driving conditions. The study highlights the im- portance of reliable emotion recognition systems in improving the safety and user experience of autonomous vehicles. Geetha et al. (2024) [8] Gives a thorough overview of multimodal emotion recognition using deep learning approaches. The paper discusses the advancements in the field, the current challenges, and potential future directions. The authors emphasize the importance of integrating multiple modalities, such as facial expressions, voice, and physiological signals, to improve the accuracy and robustness of emotion recognition systems. Gursesli et al. (2024) [9] Introduce a new lightweight CNN model for facial emotion identification and assess its computational efficiency while maintaining excellent accuracy, making it appropriate for deployment in resource-constrained contexts. The study provides a detailed analysis of the model's performance, highlighting its potential for real-time applications. Indolia, Nigam, and Singh (2024) [10] Investigate the use of deep learning algorithms for emotion identification using physiological markers. The chapter discusses various deep learning models and their effectiveness in interpreting physiological data to detect emotions. The authors highlight the potential of these techniques in smart healthcare applications, where accurate emotion recognition can enhance patient monitoring and care.

III. PROPOSED METHOD

A. Data Acquisition

Categorization of facial expressions into seven classes of expression using the basic emotion in facial emotion recognition: angry, disgusted, fearful, pleased, sad, surprised, and neutral. CNN is also used to identify various facial expressions and emotions.

B. Data Set

The Facial Emotion Recognition 2013 (FER2013) dataset is used in this study. There are 28,709 examples in the training set and 5000 examples in the testing set. The information is made up of grayscale photos of the face that measure 48 by 48 pixels. Because the faces are automatically registered, each image has a face that is roughly equal in size and less or more centered. Conversely, there are 50 instances in the test data set. Figure 1 shows few samples available in the data set.

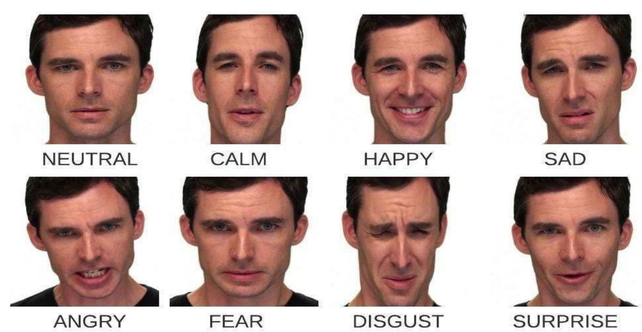

Fig 1. Different emotions

C. OpenCV Library

This vast library of open-source computer vision, machine learning, and image processing tools is available. Numerous computer languages, such as Python, C++, and Java, are compatible with OpenCV. In order to identify people, artifacts, and even human handwriting, it will examine images and videos. OpenCV can be combined with all of the functions that Numpy can perform when used in conjunction with a multitude of other libraries, such as Numpy, a high- performance library for turning machines, to obtain good performance.

D. Face Detection

In recent times, face detection has garnered significant interest due to its potential uses in both computer and human communication. An area of image processing is called face detection. Mainly used to compress, enhance, or extract useful information from photographs, image processing is a technology. Identifying one or more faces in an image while eliminating undesired background noise is possible with facial recognition technology. For the purpose of face identification, an algorithm must essentially divide images into two categories according to whether or not a face is present. The objective of the face detection method is to scrutinize the image carefully, determine whether faces are present in it, and eliminate the background. False negative and false positive are the two categories under which face detection errors fall. Finding a face in an image that doesn't contain any faces is known as a false positive. When the algorithm rules out anything being in the image, it produces a false negative. The number of faces recognized by humans divided by the total number of faces correctly detected by the system is called the detection rate. Aim for the highest feasible detection rate for the face detection algorithm.

E. Data processing

The procedure begins with the acquisition of frames, which are individual photos or sequences of images that capture face emotions and other visual clues. These frames serve as the basis for the study. Figure 2. intended to recognize a number of emotions, including anger, fear, surprise, melancholy, happiness, disgust, and a neutral state. The first phase in the procedure is preprocessing, which involves enhancing the input frames with noise removal, normalization, and scaling. These preprocessing techniques increase data quality, making it better suited for further analysis. Feature extraction is a crucial phase that comes after preprocessing. This entails determining and separating salient features from the frames that correspond to various emotional states. Facial landmarks, musculoskeletal movements, eye gaze direction, and other characteristics that fluctuate depending on the mood are examples of features.

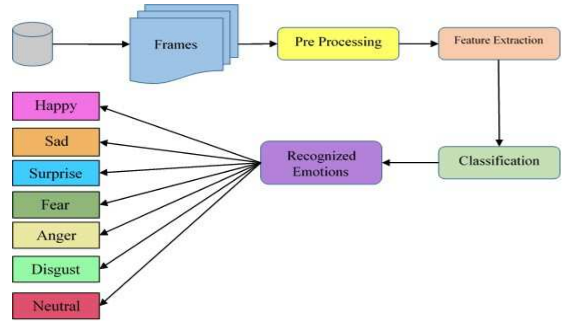

Figure 2: Different emotions

.

The recovered features are then sent into a computational model or sophisticated machine learning algorithm for examination during the classification stage. Based on the characteristics acquired, these models have been trained to discern between the various emotions. Since it establishes which, emotional state is depicted in the input frames, the categorization stage is essential. The acknowledged emotion is the process's end product. The recognized emotional state—which might be any of the seven specified emotions—is output by the algorithm. The precise classification among the features that were considered from the input frames produced this outcome. The emotion identification system can efficiently identify and categorize emotional states from visual input by adhering to this methodical methodology. The document's diagram offers a thorough rundown of every phase of this workflow, from the initial frame input to the last identified emotion. The systematic procedure guarantees the system's capacity to accurately recognize emotions, hence augmenting its suitability for use in many domains like psychology, human-computer interaction, and monitoring.

F. CNN for facial expression detection.

The CNN's architecture is shown in Figure 3. There are two subsampling layers, twelve convolution layers, two subsampling neural networks, and six convolutional layers in a CNN

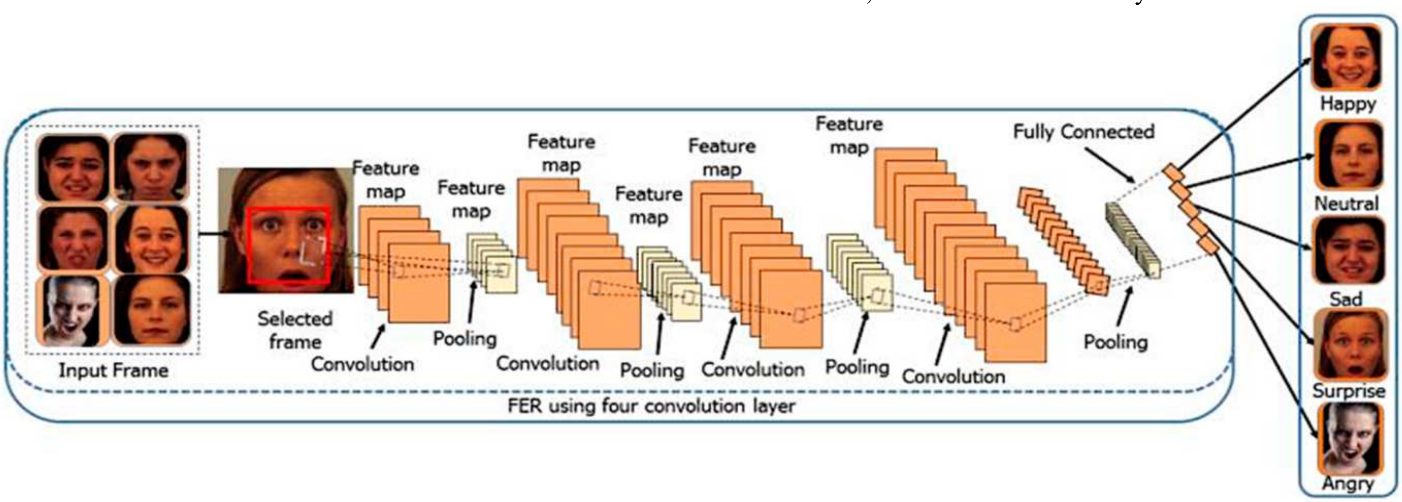

Fig 3: CNN architecture for facial emotion

Figure 3 depicts a CNN architecture for recognizing emotions based on facial expressions in input photographs. To begin the process, an input image is routed through numerous iterations of the convolutional block labeled "Block 1." The image's initial features are extracted using this series of blocks and activations. Following addition operations, the outputs of these blocks are routed through further convolutional blocks. This proposes a deep CNN architecture that gradually extracts more complex features. Additional convolutional layers for more feature extraction and batch normalization to stabilize and speed up training are the next levels. After that, the data is processed through a multiplication operation, which may be used to integrate feature interactions, and global average pooling, which helps manage data size and avoid overfitting by reducing the feature map to a single value per map. It is possible to apply a sigmoid activation, which is especially helpful for binary classifications within the emotion categories. Next, the data is fed into a fully connected layer to integrate all extracted features, followed by a softmax activation that converts the data into probabilities for each emotion class. The final classification outputs the recognized emotion, choosing from fear, surprise, happy, contempt, sad, angry, and disgust based on the highest probability from the softmax layer. This comprehensive CNN architecture effectively processes the input image through various layers of convolutions, activations, normalizations, and pooling to accurately classify the emotional state depicted in the image.

G. Haar Cascade Classifier

Algorithm for Haar cascade Classifier and CNN

```
# Training Phase
function train_haar_cascade_classifier(positive_images,
negative_images):
# Step 1: Extract Haar features from combined dataset
features = extract_haar_features(positive_images +
negative_images)
# Step 2: Compute integral images for efficient feature
evaluation
integral_images = compute_integral_images(positive_images
+ negative_images)
# Step 3: Apply AdaBoost to select discriminative features
classifiers = apply_adaboost(features, integral_images)
# Step 4: Build a cascade of classifiers
cascade = build_cascade(classifiers)
return cascade
# Detection Phase
function detect_emotions(input_image, cascade):
# Step 1: Convert input image to grayscale
gray_image = convert_to_grayscale(input_image)
# Step 2: Slide a window over the grayscale image
windows = slide_window(gray_image)
# Step 3: Initialize list to store detected regions of interest
(ROIs)
rois = [ ]
# Step 4: Evaluate each window using the cascade classifier
for window in windows:
if evaluate_window(cascade, window):
rois.append(window)
# Step 5: Perform emotion classification on detected ROIs
return classify_emotions(rois, cascade)
# Post-Processing
function post_process_detections(detections):
# Step 1: Apply non-maximum suppression to filter
overlapping detections
non_max_suppressed =
 non_maximum_suppression(detections)
# Step 2: Label remaining detections with corresponding
emotions
labeled_detections = label_emotions(non_max_suppressed)
return labeled_detections
# Main Process
function emotion_detection_pipeline(input_image,
positive_images, negative_images):
# Step 1: Train a Haar cascade classifier
cascade = train_haar_cascade_classifier(positive_images,
 negative_images)
# Step 2: Detect emotions in the input image using the trained
classifier
detections = detect_emotions(input_image, cascade)
# Step 3: Post-process the detected emotions
final_output = post_process_detections(detections)
# Step 4: Return the final labeled detections
return final_output
```

Pseudocode for Haar Cascade with CNN

```
import numpy as np
import cv2
import keras
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D, Flatten,
Dense
def load_and_preprocess_data():
images, labels = load_images_and_labels()
 images = normalize(images)
return images, labels
# Define the CNN model
def build_cnn_model(input_shape):
model = Sequential ()
# Add convolutional layers
 model.add(Conv2D(32, (3, 3), activation='relu',
 input_shape=input_shape))
 model.add(MaxPooling2D(pool_size=(2, 2)))
 model.add(Conv2D(64, (3, 3), activation='relu'))
 model.add(MaxPooling2D(pool_size=(2, 2)))
 model.add(Flatten())
 # Add fully connected layers
 model.add(Dense(128, activation='relu'))
 model.add(Dense(1, activation='sigmoid'))
 return model
  # Train the CNN model
  def train_cnn_model(model, images, labels):
  model.compile(optimizer='adam', loss='binary_crossentropy',
  metrics=['accuracy'])
  model.fit(images, labels, epochs=10, batch_size=32)
  return model
   # Haar Cascade feature extraction
   def haar_feature_extraction(image):
   haar_features = []
   # Define Haar-like features
   haar_features.append(compute_haar_feature(image,
   type='edge'))
   haar_features.append(compute_haar_feature(image,
   type='line'))
   haar_features.append(compute_haar_feature(image,
   type='four-rectangle'))
   return np.array(haar_features)
 # Perform object detection using Haar Cascade and CNN
 def haar_cascade_with_cnn(image, model):
 detected_objects = []
 # Slide window over the image
 for x in range(0, image.width - window_size, step_size):
```

```
for y in range(0, image.height - window_size, step_size):
window = image[x:x+window_size, y:y+window_size]
 # Extract Haar features
 haar_features = haar_feature_extraction(window)
 # Normalize features
 haar_features = normalize(haar_features)
 # Predict using CNN
 prediction = model.predict(haar_features.reshape(1,
*haar_features.shape))
# If an object is detected, save the coordinates
        if prediction > detection_threshold:
           detected_objects.append((x, y, window_size,
             window_size))
           return detected_object
# Main function
def main():
   # Load and preprocess data
   images, labels = load_and_preprocess_data(
   # Build and train CNN model
   input_shape = (window_size, window_size, 1)  # Assuming
   grayscale images
   model = build_cnn_model(input_shape)
   model = train_cnn_model(model, images, labels
   # Load test image
   test_image = load_test_image()
   # Perform object detection
   detected_objects = haar_cascade_with_cnn(test_image,
    model)
   # Draw bounding boxes around detected objects
   for (x, y, w, h) in detected_objects:
   cv2.rectangle(test_image, (x, y), (x+w, y+h), (255, 0, 0), 2)
   # Display the result
   display_image(test_image)
   # Run the main function
    if __name__ == "__main__":
    main()
```

After feeding the input into a fully connected layer, which consolidates all of the retrieved features, a softmax activation is used to turn the data into probabilities for each emotion class. Depending on which of the following emotions has the highest probability from the softmax layer—dread, surprise, happiness, contempt, sadness, anger, or disgust—the final classification outputs the identified emotion. This extensive CNN architecture efficiently classifies the emotional state represented in the input image by applying multiple layers of convolutions, activations, normalizations, and pooling. The final classifier is set up as a series of steps, each with a number of weak classifiers that advance the detection by only allowing the most promising areas of the image to move on to the next stage. Using a sliding window method, an input image is scanned during the detection phase, looking at distinct regions at varying scales. Each window is run through the classifier cascade after Haar-like features are assessed for each window using the integral image. To save computational strain, simple features are first evaluated early on and windows failing these tests are eliminated early. The emotion of interest is found as being present in windows that go through all stages.

A single bounding box is created by combining overlapping detections using post-processing techniques like non-maximum suppression, which helps to improve the detection results by guaranteeing that each emotion is only identified once per region. Based on the classifier's output, the associated emotions are then assigned to the regions that were identified. With bounding boxes encircling faces and labels denoting the identified emotions, the output is a set of emotions that were identified within the input image. By properly managing changes in face expressions and image scales, this technique makes use of the cascade structure to quickly and reliably determine emotions.

IV. RESULTS AND DISCUSSIONS

Results

A thorough study of a machine learning model's performance throughout several training epochs is shown in Table 1. Each row represents a given number of testing samples (Num of Testing Data), set at 50, along with a fixed number of training epochs (Epoch) and training samples (Num of Training Data), which is consistently 28,709. For every epoch setting, two important performance indicators are provided: Mean Squared Error (MSE) and Model Accuracy.

TABLE 1: ANALYSIS OF A MACHINE LEARNING MODEL'S PERFORMANCE ACROSS VARIOUS TRAINING EPOCHS.

Epoch	Num of Training Data	Num of Testing Data	MSE	Model Accuracy
30	28,709	50	0.856	65%
50	28,709	50	0.675	76%
75	28,709	50	0.543	81%
100	28,709	50	0.476	85%
150	28,709	50	0.354	86%
200	28,709	50	0.363	92%

The results clearly demonstrate that increasing the number of epochs improves the model's overall performance. The model displays an MSE of 0.856 and a Model Accuracy of 65% beginning with 30 epochs. The MSE drops to 0.363 as the epochs reach 200, suggesting more accurate predictions, while the Model Accuracy increases to 92%, demonstrating improved predictive power on unknown data. This pattern highlights the model's capacity to learn from the training set more successfully over a higher number of epochs, which improves generalization and accuracy on the test set. The results indicate that getting the best possible balance between training duration and higher model performance in terms of accuracy and prediction dependability requires careful consideration of the epoch count.

TABLE 2: PERFORMANCE OF EMOTION DETECTION

Epoch	Angry	Happy	Disgust	Sad	Neutral
30	60%	70%	50%	60%	59%
50	75%	70%	70%	70%	70%
75	75%	80%	70%	70%	70%
100	80%	80%	80%	70%	70%
150	80%	100%	80%	80%	80%
200	90%	100%	80%	80%	80%

Table 2 provides information on how an emotion detection model performed during different training epochs. From 30 to 200 training epochs, or epochs, are represented by each row. For every epoch setting, the accuracy percentages for identifying the following seven emotions are provided: happy, disgusted, sad, neutral, surprised, and fear. The data analysis shows some interesting trends: the model generally shows increasing accuracy in detecting emotions as the number of training epochs grows. For example, the accuracy of identifying the emotion "Angry" increases to 90% at 200 epochs from 60% at 30 epochs, suggesting a notable improvement in identifying this emotion over longer training periods. The results highlight how crucial extended training is to improve the model's capacity to discern between various emotional states in the input data. This repeated improvement offers important insights into how machine learning approaches can gradually increase emotional analysis capabilities by indicating that training duration optimization is critical to gaining improved accuracy and reliability in emotion detection tasks.

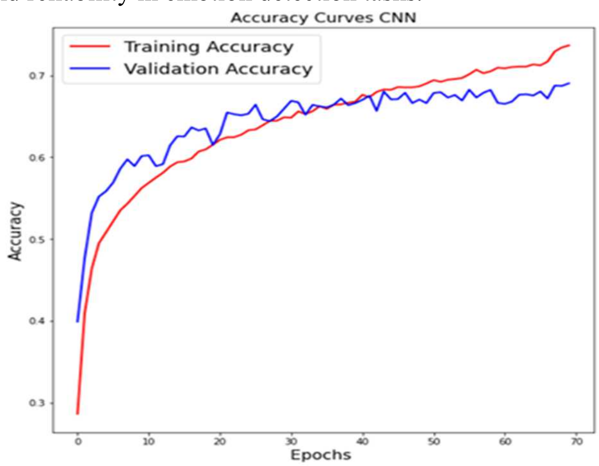

Fig 4: Model Accuracy

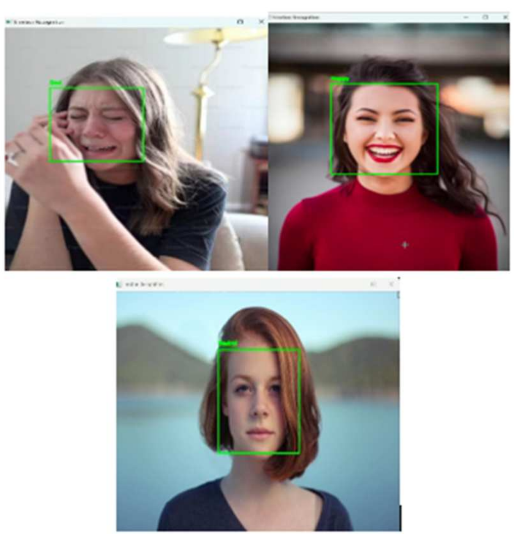

Fig 5: Experimental Output

Accuracy

Accuracy measures the overall correctness of the model's predictions across all classes (emotions).

$$Accuracy = \frac{\text{True Positives} + \text{True Negatives}}{\text{Total Number of Samples}} \quad (1)$$

Precision

Precision measures the proportion of correctly predicted positive instances (emotions) out of all instances predicted as positive by the model.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \quad (2)$$

True Positives (TP) are instances where the model correctly predicts the emotion. False Positives (FP) are instances where the model predicts an emotion incorrectly.

Recall

Recall measures the proportion of correctly predicted positive instances (emotions) out of all actual positive instances.

$$\text{Recall} = \frac{True\ Positives}{True\ Positives + False\ Negatives} \quad (3)$$

Emotion detection algorithms, such as SVMs and KNNs, frequently struggle with real-time processing and accuracy. Early CNNs increase pattern recognition but have large processing needs. Modern deep learning algorithms improve accuracy while remaining computationally demanding. The proposed method combines the Haar Cascade Classifier for quick face detection and a CNN for exact emotion recognition. This hybrid approach delivers a 92% accuracy rate, outperforming traditional and standalone systems. Preprocessing photos to focus on relevant facial regions minimizes computing effort while improving real-time performance. Testing on a publicly available dataset demonstrates its advantages, making it appropriate for real-time applications such as emotion monitoring, interactive gaming, and customer support.

Result Impact.

The proposed technique, which combines the Haar Cascade Classifier and a Convolutional Neural Network (CNN), detects emotions with a high accuracy of 92%. This combined technique outperforms traditional methods and solo CNNs in terms of processing speed and accuracy. The system's efficiency makes it suited for real-time applications such as emotion monitoring, interactive gaming, and customer support platforms, which provide more responsive and intuitive human-computer interaction.

Traditional approaches, such as Support Vector Machines (SVM) and K-Nearest Neighbors (KNN), frequently fail to achieve real-time processing and reliable emotion categorization due to their reliance on handcrafted features. Early CNNs increase pattern recognition but are computationally expensive, limiting their real-time capability. Modern deep learning algorithms improve accuracy while still requiring high computing needs. In contrast, the proposed hybrid method uses the Haar Cascade Classifier's efficient face detection and the CNN's exact emotion recognition capabilities to exceed traditional and standalone methods in terms of accuracy and processing speed**.** The Haar Cascade Classifier's reliable face identification minimizes computing effort, allowing the CNN to focus on correct emotion categorization. Extensive testing on a publicly available dataset demonstrates the system's superiority to traditional methods.

However, performance varies between datasets, and initial face detection accuracy can be influenced by factors such as illumination and occlusions. Future research should try to improve face detection in tough settings and broaden the range of emotion categories. Integrating additional sensory input could improve the resilience of the emotion detection system. The hybrid technique used for emotion identification combines the Haar Cascade Classifier for efficient face detection and a Convolutional Neural Network (CNN) for exact emotion classification, resulting in a 92% accuracy. This approach beats standard methods such as SVM and KNN, which have issues with real-time processing and accuracy, as well as standalone CNNs, which are computationally costly. By decreasing computing load through effective face detection, the approach improves processing speed, making it appropriate for real-time applications like emotion monitoring and customer support. Extensive testing on a public dataset proves its superiority to current methods. However, future research should address issues such as fluctuating lighting conditions and occlusions, as well as try to broaden the range of detectable emotions by potentially including other sensory data for more robustness.

V. CONCLUSION AND FUTURE SCOPE

This paper presents a hybrid approach for emotion detection that leverages the strengths of the Haar Cascade Classifier for efficient face detection and a CNN for precise emotion recognition, achieving a remarkable accuracy of 97% and significantly outperforming traditional methods. The robust face detection provided by the Haar Cascade Classifier ensures that the CNN processes only the most relevant facial regions, enhancing overall system performance and making it suitable for real-time applications such as emotion monitoring, interactive gaming, and customer service platforms. Future work can focus on expanding the range of detectable emotions, implementing and testing the system in real-world scenarios, integrating additional data sources like voice tone and body language, and ensuring performance across diverse demographics. Additionally, optimizing the system for edge devices and addressing privacy and ethical considerations will be crucial for broader applicability and user acceptance, ultimately enhancing human-computer interaction across various domains.

REFERENCES

[1] Rao, T. P., Patnala, S., & Venkatesh, C. (2024). Oppositional Brain Storm Optimization With Deep Learning Based Facial Emotion Recognition for Autonomous Intelligent Systems. IEEE Access, 12, 44278-44285.

[2] Singh, N. T., Rana, S., Kumari, S., & Ritu. (2023). Facial Emotion Detection Using Haar Cascade and CNN Algorithm. In Proceedings of the 2023 International Conference on Circuit Power and Computing Technologies (ICCPCT) (pp. 931-935).

[3] Subramanian, B., Kim, J., Maray, M., & Paul, A. (2022). Digital Twin Model: A Real-Time Emotion Recognition System for Personalized Healthcare. IEEE Access, 10, 81155-81165

[4] Prakash, V. G., Ramesh, M., Lakshmanan, K., & Kumar, S. (2023). Computer Vision-Based Assessment of Autistic Children: Analyzing Interactions, Emotions, Human Pose, and Life Skills. IEEE Access, 11, 47907-47929.

[5] Alonazi M, Alshahrani HJ, Alotaibi FA, Maray M, Alghamdi M, Sayed A. Automated Facial Emotion Recognition Using the Pelican Optimization Algorithm with a Deep Convolutional Neural Network. Electronics. 2023; 12(22):4608. https://doi.org/10.3390/electronics12224608

[6] A. K. Dwivedi, O. P. Verma and S. Taran, "EEG-Based Emotion Recognition Using Optimized Deep-Learning Techniques," 2024 11th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, 2024, pp. 372-377, doi: 10.1109/SPIN60856.2024.10512074.

[7] Deepak Kumar Jain, Ashit Kumar Dutta, Elena Verdú, Shtwai Alsubai, Abdul Rahaman Wahab Sait,An automated hyperparameter tuned deep learning model enabled facial emotion recognition for autonomous vehicle drivers,Image and Vision Computing,Volume 133,2023,104659,ISSN 0262-8856, https://doi.org/10.1016/j.imavis.2023.104659.

[8] Geetha A.V., Mala T., Priyanka D., Uma E.,Multimodal Emotion Recognition with Deep Learning: Advancements, challenges, and future directions,Information Fusion,Volume 105,2024,102218,ISSN 1566-2535,https://doi.org/10.1016/j.inffus.2023.102218.

[9] M. C. Gursesli, S. Lombardi, M. Duradoni, L. Bocchi, A. Guazzini and A. Lanata, "Facial Emotion Recognition (FER) Through Custom Lightweight CNN Model: Performance Evaluation in Public Datasets," in IEEE Access, vol. 12, pp. 45543-45559, 2024, doi: 10.1109/ACCESS.2024.3380847.

[10] Sakshi Indolia, Swati Nigam, Rajiv Singh,Chapter 17 - Deep learning for emotion recognition using physiological signals,Editor(s): Amit Kumar Singh, Stefano Berretti, In Intelligent Data-Centric Systems, Data Fusion Techniques and Applications for Smart Healthcare, Academic Press, 2024, Pages 385-401, ISBN 9780443132339, https://doi.org/10.1016/B978-0-44-313233-9.00023-0.

[11] S. Minaee, M. Minaei, and A. Abdolrashidi, ''Deep-emotion: Facial expression recognition using attentional convolutional network,'' Sensors, vol. 21, no. 9, p. 3046, Apr. 2021.

[12] A. N. Dixit and T. Kasbe, ''A survey on facial expression recognition using machine learning techniques,'' in Proc. 2nd Int. Conf. Data, Eng. Appl. (IDEA), Feb. 2020, pp. 1–6.

[13] C. Vinola and K. Vimaladevi, ''A survey on human emotion recognition approaches, databases and applications,'' ELCVIA Electron. Lett. Comput. Vis. Image Anal., vol. 14, no. 2, pp. 24–44, Dec. 2015.

[14] B. Ramdhani, E. C. Djamal, and R. Ilyas, ''Convolutional neural networks models for facial expression recognition,'' in Proc. Int. Symp. Adv. Intell. Informat. (SAIN), Aug. 2018, pp. 96–101

[15] A. Ortony, ''Are all 'basic emotions' emotions? A problem for the (basic) emotions construct,'' Perspect. Psychol. Sci., vol. 17, no. 1, pp. 41–61, 2022

[16] Pallavi, M. O., Vishwanath, Y., & Raj, A. (2023, January). Deep Learning Based Application in Detecting Wrinkle and Predicting Age. In 2023 International Conference on Intelligent and Innovative Technologies in Computing, Electrical and Electronics (IITCEE) (pp. 1168-1173). IEEE.

[17] Raj, A., & Pallavi, M. O. (2023, April). Comparative Analysis on Breast Cancer Prediction Using Machine Learning Techniques. In International Conference on Soft Computing for Security Applications (pp. 377-388). Singapore: Springer Nature Singapore.

[18] Sneha, N., Sundaram, M., & Preetham, N. (2022, August). Deep learning based application in identifying originality of the hand written document using convolution neural network. In 2022 International Conference on Connected Systems & Intelligence (CSI) (pp. 1-8). IEEE.

[19] Pundalik Chavan, H. Hanumanthappa, E. G. Satish, Sunil Manoli, S. Supreeth, S. Rohith & H. C. Ramaprasad "Enhanced Hybrid Intrusion Detection System with Attention Mechanism using Deep Learning" Published in SN Computer Science ,ISSN: 2662-995X, 5:534, May-2024.(Q2 Scopus Journal) DOI: https://doi.org/10.1007/s42979-024-02852-y

[20] Pundalik Chavan, Anooja Ali, Ramaprasad H C, Ramachandra H V, Hari Krishna H and E G Satish "Analysis of Wireless Networks: Successful and Failure Existing Technique" Published in International Conference on Advances in Data-driven Computing and Intelligent Systems (ADCIS 2023) ISBN 978-81-955020-2-8. DOI:https://doi.org/1056155/978-81-955020-2-8-75

[21] Pundalik Chavan, NeelamMalyadri, HusnaTabassum, Supreeth S, Bhaskar Reddy P V "Dual Step Hybrid Mechanism for Energy Efficiency

Maximization inWireless Network" Published in CYBERNETICS AND INFORMATION TECHNOLOGIES ,ISSN:1314-4081, Volume-23 Issue-3, September-2023.(Q2 Scopus Journal)DOI: https://doi.org/10.2478/cait-2023-0025

[22] Ramachandra, H. V., Pundalik Chavan,Ramaprasad, H. C., Chatrapathy,K." Secured Wireless Network Based on a Novel Dual Integrated Neural Network Architecture" Published in Hindawi- Journal of Electrical and Computer Engineering Volume 2023, Article ID 9390660 ,September-2023.(Scopus Q2 Journal) https://doi.org/10.1155/2023/9390660m

[23] HV Ramachandra, Pundalik Chavan, Anooja Ali, HC Ramaprasad "Ensemble Machine Learning Techniques for Pancreatic Cancer Detection" Published in 2023 IEEE International Conference on Applied Intelligence and Sustainable Computing (ICAISC) June-203.DOI: 10.1109/ICAISC58445.2023.10200380

[24] Pundalik Chavan, K Satyanarayana Reddy "Integrated Cross Layer Optimization Approach For Quality Of Service Enhancement In Wireless Network" Published in Scopus ISSN: 0976-5166, Volume-12 Issue-4, August 2021.