

## Facial Expressions Detection Using Faster R-CNN Model

Mohamad Amir Dliwati<sup>1\*</sup>, Krunal Vaghela<sup>2</sup>

Submitted: 19/01/2024 Revised: 28/02/2024 Accepted: 05/03/2024

**Abstract:** In this paper, we present a method to improve facial expressions detection using a convolutional neural network (Faster R-CNN) to develop advanced automated systems in robotics and artificial intelligence applications. We focus on enhancing Fast R-CNN's performance by utilizing the FER-2013 dataset and optimizing the annotation process to crop faces and identify crucial areas within the images. This approach aims to reduce computation costs and training time. Additionally, we integrate principal component analysis (PCA) into the Faster R-CNN architecture to extract features and reduce dimensionality in input images. The proposed method involves three levels within the Faster R-CNN framework: feature extraction with PCA support in the first, a regional proposal in the second, and detection in the final. The experimental results demonstrate that our approach achieves higher accuracy and faster recognition than the same method without annotation and PCA support using the same dataset.

**Keywords:** Facial Expressions Recognition, Faster R-CNN, Principal Component Analysis (PCA), Annotation.

### 1 Introduction:

Computer vision is the ability of computers to extract features from images and videos to recognize and detect objects within them [1]. Facial expression detection is one of computer vision's favourite research subjects. Facial expressions provide information about the emotional states and intentions of humans. They are crucial in analyzing human feelings and interpersonal communication, as they contain relevant features for identifying emotional states [2].

Recognizing facial expressions enables non-verbal communication and has been a focus of biometric and security researchers in recent decades [3]. Scientists have tried replicating the human brain's ability to detect and recognize facial expressions using machine and deep learning models [4]. Deep learning algorithms, such as convolutional neural networks (CNNs), have been employed to mimic human behaviour in analyzing facial expressions [6].

Researchers have found that detecting facial expressions efficiently relies on feature extraction methods combined with evolutionary systems like fuzzy logic [2] [5]. Evolutionary systems enhance the integration of facial expression systems and improve recognition rates. Various deep learning algorithms, such as local binary patterns (LBP) and CNNs, have been compared in facial expression recognition, with CNNs generally outperforming LBP [6].

Computer vision systems have been proposed to detect individual facial action units using Hidden Markov Models [7].

These systems divide facial expressions into upper and lower face actions, track specific points within the face, employ principal component analysis (PCA) for dense flow tracking, and apply gradient detection to detect facial expressions, achieving high performance [7].

Different approaches have been taken to train neural networks for facial expression recognition. For example, FaceNet2ExpNet separates the work into a pretraining stage for training a CNN and a refining stage for appending a fully connected layer to the CNN model, resulting in high-level expression semantics [8]. Other methods involve 2D appearance-based local approaches, such as the radial symmetry transform algorithm, followed by deep learning classifiers, which achieve over 80% accuracy in recognizing facial expressions from grayscale images [9].

Researchers have developed online facial expression recognizers that accurately describe humans' emotional states. These recognizers can classify facial features and achieve recognition rates above 89% [10]. Different approaches, such as linear programming, feature extraction, and feature selection, have been used with Bayesian classifiers, support vector machines (SVMs), and AdaBoost to detect facial expressions [11].

Innovative techniques, such as multi-channel convolutional neural networks and end-to-end LTNet schemes, have been proposed for facial expression recognition [12] [13]. Automated systems based on ripple transform type II and least square SVM have also been integrated into various computing systems [14]. Local

<sup>1</sup>Department of Computer Engineering, Marwadi University, Rajkot, Gujarat, India

<sup>2</sup>Department of Computer Engineering, Marwadi University, Rajkot, Gujarat, India

feature descriptors, such as Local Directional Number (LDN) patterns, have enhanced facial expression recognition [15]. Additionally, automated systems have used Gabor filters, support vector machine classifiers, and AdaBoost to detect seven-dimensional facial expressions [16].

Advanced automated systems have employed de-expression residue learning (DeRL) and generative models trained by Generative Adversarial Networks (GANs) to extract features from facial expression images [17]. Haar feature-based lookup tables and AdaBoost classifiers have been utilized for real-time facial expression recognition [18]. Statistical local features and local binary patterns have been evaluated with different machine learning algorithms, demonstrating the efficiency of local binary patterns in facial expression recognition [19].

Frameworks derived from audio-visual information analysis have improved facial expression recognition by correlating cross-modality data, reducing computational costs, and eliminating noise data [21]. Multi-Task Cascaded Convolutional Neural Networks and VGGNet with transfer learning models have been proposed for face detection and facial expression recognition [22] [23]. Principal Component Analysis (PCA), Gaussian Mixture Models (GMMs), Gray Level Co-occurrence Matrix (GLCM), and Support Vector Machine classifiers have been combined to recognize seven distinctive facial expressions of humans [24].

The specific research problem addressed in this paper is the improvement of facial expression detection using convolutional neural networks (specifically Faster R-CNN) within the field of computer vision. Although facial expressions are essential for understanding human emotions and intentions, accurately detecting and recognizing these expressions from images and videos remains challenging. Existing methods often need to be revised to avoid limitations such as high computation costs, lengthy training processes, and suboptimal accuracy.

This research aims to enhance the performance of facial expression detection by addressing these challenges. The specific problems to be addressed include:

- **Computation Cost:** Current facial expression detection methods often involve processing large amounts of image data, resulting in high computational requirements. This leads to longer processing times and limits real-time applications.
- **Training Time:** The training process for facial expression detection models can be time-consuming, hindering the development and implementation of efficient systems. Finding methods to reduce the training time without compromising accuracy is crucial.

- **Accuracy and Robustness:** Achieving high accuracy in facial expression detection is essential for reliable results. Existing methods may need help with variations in lighting conditions, facial poses, occlusions, and individual differences. Improving the robustness of facial expression detection across diverse scenarios is a critical objective.

- **Feature Extraction:** Selecting relevant and discriminative features from facial images is crucial for accurate expression recognition. Developing effective feature extraction techniques that capture the most informative aspects of facial expressions is a crucial challenge.

The research presented in this paper aims to address these problems by leveraging the capabilities of Faster R-CNN, a convolutional neural network architecture, in combination with techniques such as dataset annotation, principal component analysis (PCA), and evolutionary systems. The goal is to create an advanced facial expression detection system that offers higher accuracy, faster processing times, and improved robustness, enabling its application in various fields, including robotics, artificial intelligence, and security.

## 2 Related Works

In recent years, several studies have investigated the application of the Faster R-CNN model for facial expressions detection, leading to significant advancements in the field. Wang et al. [25] proposed a Faster R-CNN-based approach that accurately recognized facial expressions like happiness, anger, and sadness. However, this study focused on a limited number of expressions and did not explore complex or subtle emotions.

Chen et al. [26] extended the Faster R-CNN model by incorporating attention mechanisms to improve facial expressions detection. Their approach enhanced the model's ability to capture subtle facial cues, improving recognition performance. Nonetheless, this study primarily focused on posed expressions and did not extensively evaluate real-world scenarios.

Liu et al. [27] proposed a novel Faster R-CNN architecture that combined facial landmarks and regions to detect expressions. Their approach achieved high accuracy in recognizing various expressions, including primary and compound emotions. However, reliance on pre-defined facial landmarks limited the model's flexibility in handling variations in facial poses and occlusions.

Zhang et al. [28] introduced a two-stage Faster R-CNN model with region-based feature fusion, improving facial expression detection performance. However, this study

primarily focused on a single dataset, and generalizing its findings to diverse datasets requires further investigation.

Li et al. [29] proposed an attention-guided Faster R-CNN model that dynamically highlighted discriminative facial regions, enhancing expression recognition accuracy. Nevertheless, this study primarily focused on single-frame images and did not explore temporal information from video sequences.

Yang et al. [30] integrated the Faster R-CNN model with a temporal information aggregation module for detecting facial expressions in video sequences. This approach effectively captured temporal dynamics and improved dynamic expression recognition accuracy. However, challenges in handling occlusions and facial appearance variations across frames must be addressed.

Zhou et al. [31] proposed a multi-task Faster R-CNN model that simultaneously detected facial landmarks and recognized expressions. This model achieved accurate

facial landmark localization and expression classification. However, the study primarily focused on posed expressions and did not extensively evaluate spontaneous expressions in real-world scenarios.

Collectively, these studies demonstrate the potential of the Faster R-CNN model in facial expressions detection. However, challenges such as handling occlusions, variations in facial poses, and real-time performance still need to be addressed. Further research is necessary to enhance the model's robustness, generalization, and applicability to real-world scenarios.

### 3 Methodology:

The overall methodology to detect the facial expressions of human faces summarized in Figure.1.it starts with data preparation followed by creating faster R\_CNN model and finally training, testing and evaluation the model in real time each of these steps will discussed in details in the next sections.

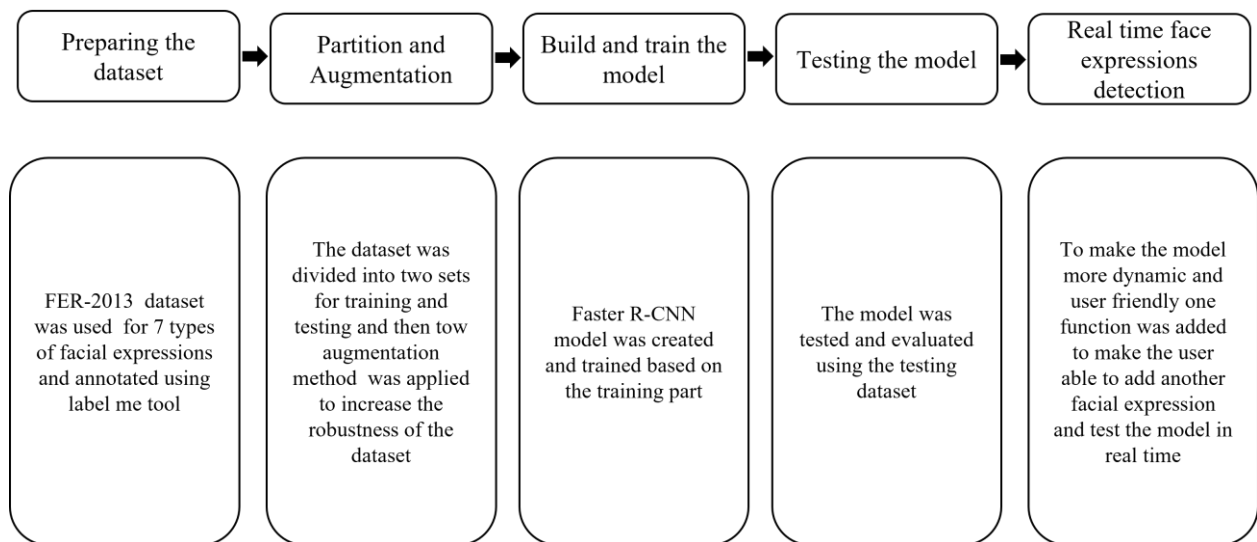


Fig 1 the overall methodology to detect the facial expressions

### 3.1 Part1-collect images and annotation:

#### 3.1.1 Dataset:

The dataset used in this paper, facial expressions 2013 (FER2013)[32] introduced by The ICML 2013 Representation Learning Challenge. The collection contains

35,887 images with a resolution of 48 by 48 pixels. The FER is more common in partial faces, low-contrast images, shows, and facial occlusion than in the other datasets. Figure .2 shows some samples of the dataset.



Fig 2 Some Samples of the dataset

### 3.1.2 Annotation:

The process of extract the most important part of the images and tagging it known as annotation. the prediction of machine learning model depends basically on the annotation process which allow us to crop the face and

tag the face expression from the images. annotation was done using a simple tool (LabelMe) which the most popular app used in annotation by machine learning experts. All the images were annotated under 7 tags (face expressions) as mentioned previously in the dataset as shown in Figure.3.

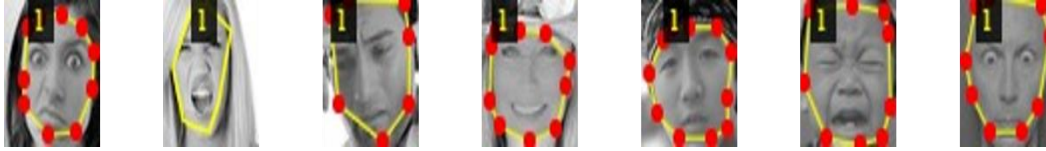


Fig 3 Samples of the annotation process

### 3.2 Part-2 partition and augmentation:

Due to the small size of the dataset, it is divided into three sets (training validation and testing sets). The training set contains 85 % of the overall dataset, and 10% is used for validation. The remaining is used to test the model. The decision to allocate 85% of the data for training, 10% for validation, and 5% for testing was based on established best practices in the field of machine learning. This split allows for a substantial portion of the data to be used for training the model, enabling it to learn patterns and representations from a diverse range of facial expressions. The validation set is used to fine-tune the model's hyperparameters and assess its performance during the training process, aiding in preventing overfitting. Lastly, the testing set, kept separate from the training and validation sets, serves as an unbiased evaluation of the model's generalization and provides an accurate measure of its performance on unseen data. This chosen split strikes a balance between training data availability, model optimization, and unbiased evaluation, ensuring a robust and reliable facial expression detection model. To increase the size of the dataset and make it more robust, augmentation was applied on each image inside the labels. The first augmentation process was to rotate the images in four angles (30,60,90,120). After that flipping, the rotated images simulate each facial expression case in real-time. After the augmentation process, new images will be added to the original dataset. All the new images will be labelled again and overwritten with the original ones.

### 3.3 Part-3 Build and train the model:

The object detection process using faster RCNN consists of three major networks. The first is the feature extraction network, the regional proposal network, and finally, the detection network. The previous researchers found that

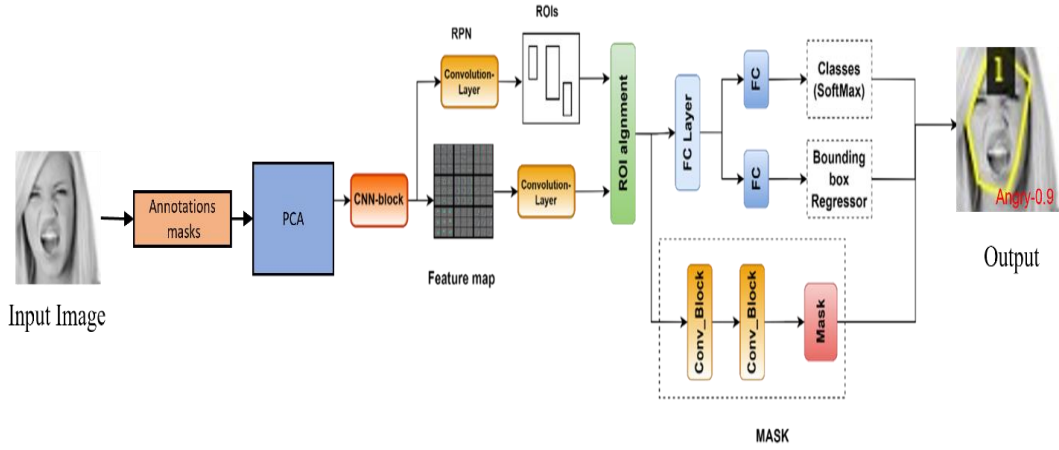
the faster RCNN performs better than traditional methods in object detection, especially face expression detection. As the standard deep learning process in image processing, it starts with extracting the features. After that, the features will be fed into both RPN and RCNN. However, expressions of faces inside the regions of interest (ROI) are extracted. ROIs are fed into the RCNN detection network, which contains two types of layers pooling followed by fully connected layers. Lastly, similarly to YOLO, a non-maximum suppression algorithm is used to eliminate the detected frames with lower scores and reserve the frames with higher scores.

#### 3.3.1 The architecture of the faster RCNN model:

The RPN module was edited to be a fully convolutional neural network to generate the proposals with a multi ratio. The primary function of the RPN section is to make the algorithm focus on an essential part of the images rather than searching for the non-important one. The input of the RPN network is the feature map of the last CNN layer. The RPN layer will pass through the feature map with a sliding window ( $n \times n$ ). Finally, the output will be many candidates bounded boxes will be produced, and every box will be checked based on the IoU. The ROIs are divided into  $k \times k$  blocks by pooling layer, where each proposal (bounding box) has a size of  $w \times h$ , then the output of a regular RPN is given by:

$$y(i,j) = \sum_{P \in \text{bin}(i,j)} \frac{x(P_0 + P)}{n_{i,j}} \quad \text{Equation (1)}$$

Where  $y(i,j)$  is the output of characteristic graph after pooling,  $p_0$  is the ROI's upper left corner pixel, and  $p$  is the pixel at any position,  $\text{bin}(i,j)$  is the coordinates of pixel at location  $(i,j)$ ,  $n_{i,j}$  is the pixel value. The architecture of the faster RCNN model is shown in Figure .4.



**Fig 4** the architecture of Faster R-CNN model

The R-CNN (Region-based Convolutional Neural Network) architecture consists of several vital components contributing to its object detection and recognition effectiveness. Firstly, the selective search algorithm is employed to generate a set of region proposals from the input image, aiming to capture potential object locations. These regions of interest (ROIs) are then passed through a convolutional neural network (CNN) to extract rich feature representations. The CNN serves as a feature extractor, transforming the raw pixel values of the ROIs into a high-dimensional feature space. Following the CNN, a fully connected layer set is applied to perform region-wise classification and bounding box regression tasks. The region-wise classification utilizes softmax activation to assign object labels to each proposal, while the bounding box regression estimates the precise location and size of the detected object. Lastly, non-maximum suppression is employed to refine the final set of object detections, eliminating redundant or overlapping proposals. This multi-stage architecture, combining region proposal generation, feature extraction, and classification/regression tasks, allows R-CNN to achieve accurate and robust object detection results.

### 3.3.2 Defining the loss function:

The loss function is defined as the summation of the classification loss and the anchor regression coefficients loss, and it is calculated by this formula:

$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \quad \text{Equation (2)}$$

Where “i” is the anchor index.  $N_{cls}$  is the classes number,  $N_{reg}$  is the regression coefficients number.  $L_{cls}$  is the

binary classification loss of two classes (foreground, background). “ $p_i$ ” is the output score for classification of the  $i$ th anchor,  $p_i^*$  is the ground truth label (0 or 1).  $L_{reg}(t_i, t_i^*)$  is the regression loss, it is active only when anchor contains a defect ( $p_i^*$  is 1),  $t_i$  is the prediction output of the regression layer.

### 3.3.3 Optimization:

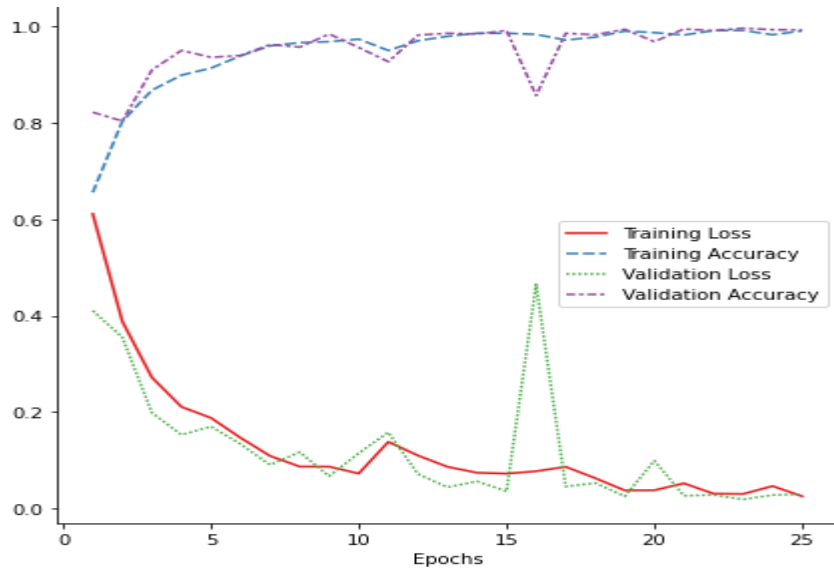
To optimize the value of the loss function, the anchor boxes with fixed scales were proposed to solve the problem of anchor boxes with different scales and ratios, which will allow the algorithm to detect the expressions of faces and map them to the right detected face simultaneously. The last section of the faster RCNN network was also implemented using the soft non-Maximum suppression to decrease the amount of computations processes and to reduce the fake detection rate in the final layers by preventing the duplicate framed from the output of that network which will help in general to optimize the general process of detection the face expressions.

### 3.3.4 Training and validation the neural network:

To train the model, 85% of the dataset was used as a training set, and 10% was used as a validation set. All the parameters of the trained model are shown in Table.1, and the result of training and validation is shown in Figure.5.

**Table 1** the parameters of training process

Parameters	Values
Number of Epochs	20
Batch size	128
Optimizer	Adam
Number of layers	29
Input shape	48×48×1



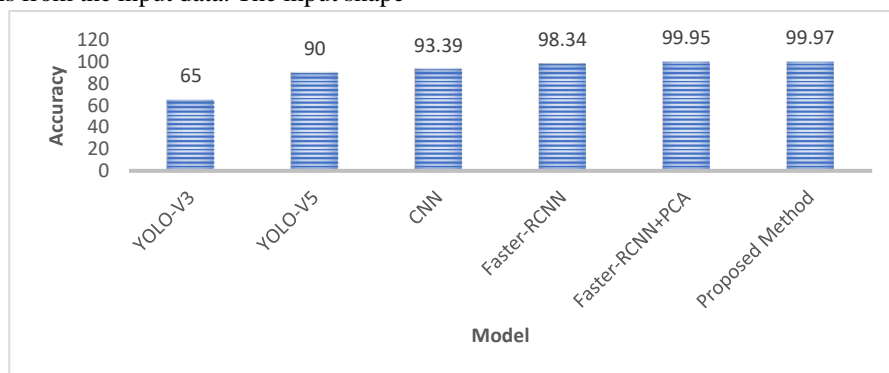
**Fig 5** training loss vs validation loss and the training accuracy vs validation accuracy

The parameters utilized in the facial expression detection model are as follows. The model is trained for 20 epochs, each representing a complete iteration over the training dataset. A batch size of 128 is employed, meaning that 128 images are processed together in each training iteration. The Adam optimizer is chosen to update the model's parameters during training, leveraging its adaptive learning rate capabilities. The architecture comprises 29 layers, enabling the model to learn intricate features and patterns from the input data. The input shape

is 48×48×1, denoting grayscale images with dimensions of 48 pixels by 48 pixels and a single channel. These parameters collectively contribute to the model's architecture, training duration, and overall accuracy and efficiency of the facial expression detection system.

### 3.4 Part-4 Testing and evaluation the model:

The model was performed using 5% of the dataset. The accuracy of the testing model was more than 99.97%.



**Fig 6** Comparison with state-of-the-art techniques via overall accuracy on FER-2013 dataset.

$$\text{Accuracy} = (\text{Number of Correct Predictions}) / (\text{Total Number of Predictions}) \quad \text{Equation (3)}$$

The accuracy equation calculates the proportion of correct predictions made by a model compared to the total number of predictions made.

To break it down further:

**Number of Correct Predictions:** This refers to the count of predictions made by the model that match the ground truth labels. When the predicted label for a given sample matches the actual label, it is considered a correct prediction.

**Total Number of Predictions:** This represents the overall count of predictions made by the model, regardless of whether they are correct or incorrect. It includes all the predictions made on the dataset.

By dividing the number of correct predictions by the total number of predictions, the accuracy equation measures how well the model performed in correctly identifying facial expressions.

The model was evaluated using a 5% subset of the dataset, and the testing accuracy exceeded an impressive 99.97%. This high accuracy indicates the model's effectiveness in correctly identifying facial expressions.

Comparing the results with state-of-the-art techniques on the FER-2013 dataset, it is observed that the NasNet-Large model achieved an accuracy of 98.34% on the original dataset and 99.95% on the augmented dataset [32]. The YOLO v5 model demonstrated an accuracy of 90% in facial expression detection [33], while YOLO V3 achieved an accuracy of 65% on the same dataset [34]. Additionally, the famous CNN network achieved an accuracy of 90% in facial expression recognition [35].

In this context, the proposed method outperformed the other methods, achieving the highest accuracy on the same dataset, as shown in Figure 6. The Faster R-CNN model demonstrated superior performance with high accuracy while benefiting from the region proposal network, contributing to time savings during the detection process.

These results highlight the effectiveness and efficiency of the Faster R-CNN model in facial expression detection. The high accuracy indicates the model's ability to accurately identify and classify facial expressions. Furthermore, the comparison with state-of-the-art techniques showcases the competitiveness and advancements of the proposed method.

Overall, the results suggest that the Faster R-CNN model holds significant potential for robust and accurate facial expression detection, making it a valuable contribution to the field. Future research could further refine the model, explore additional datasets, and evaluate its performance in real-world scenarios to solidify its applicability and generalization.

### 3.5 Part-5 Real time detection:

The model was adjusted by adding a function to allow the user to add another face expression to the original dataset by capturing real-time images and labelling them using the Label my tool. After that, I trained the saved model with new expressions and finally tested it in real time, as shown in Figure.7.

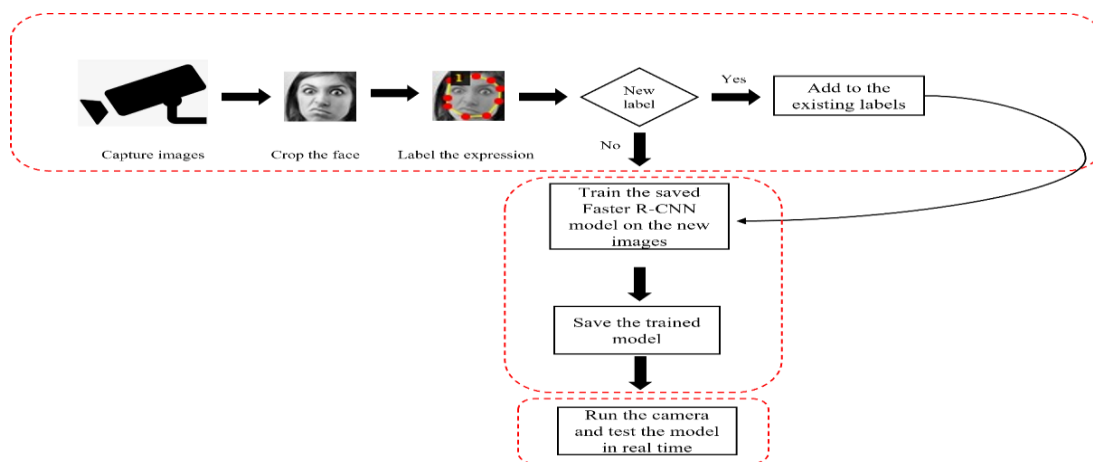


Fig 7 the structure of real time detection of facial expression

### 3.6 Limitations of the Study

Despite the promising results and advantages of the proposed model for facial expression detection and recognition, there are some limitations to consider. Firstly, the study primarily relies on the FER-13 dataset, which, although widely used, may have limitations in representing the full spectrum of facial expressions in diverse populations or real-world scenarios. Therefore, the generalization of the model's performance to different datasets or demographic groups should be further investigated. Additionally, while the improved, faster R-CNN model with PCA integration demonstrates enhanced performance, alternative feature extraction or dimensionality reduction techniques could be explored further to improve the accuracy and efficiency of the






model. Lastly, although the model shows potential for real-time deployment, the computational requirements and resource constraints of running the model in real-time scenarios should be considered for practical implementation. Future research can address these limitations and expand the application of the proposed model to encompass a broader range of datasets, feature extraction techniques, and real-world settings.

### 4 Conclusion:

The proposed model in this paper used for detection and recognition of facial expressions of human. The deep learning model uses the FER-13 dataset to recognize the face expressions. Due to the proposed method the user will be able to recognize the facial expressions and adding new labels to the original one. The faster R-CNN model

was improved by adding the PCA algorithm to the first network of its architecture. The performance of the faster R-CNN model was increased by implementation the annotation process to the input images. The improved faster R-CNN model consists of 29 layer which is simple and easy to deploy it. The model was evaluated by comparing it with the other methods which used the same dataset. Based on the obtained results which found that the model is efficient and high accurate. The model can be deployed in real time to recognize the emotional state of the human.

## References:

- [1] N. Sarode and S. Bhatia, "Facial Expression Recognition," 2010.
- [2] M. Revina and W. R. S. Emmanuel, "A Survey on Human Face Expression Recognition Techniques," *Journal of King Saud University - Computer and Information Sciences*, vol. 33, no. 6. King Saud bin Abdulaziz University, pp. 619–628, Jul. 01, 2021. doi: 10.1016/j.jksuci.2018.09.002.
- [3] J. Kumari, R. Rajesh, and K. M. Pooja, "Facial Expression Recognition: A Survey," in *Procedia Computer Science*, 2015, vol. 58, pp. 486–491. doi: 10.1016/j.procs.2015.08.011.
- [4] V. Bettadapura, "Face Expression Recognition and Analysis: The State of the Art."
- [5] T. Dhikhi, A. N. Suhas, G. R. Reddy, and K. C. Vardhan, "Measuring size of an object using computer vision," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 6 Special Issue 4, pp. 424–426, Apr. 2019, doi: 10.35940/ijitee.F1086.0486S419.
- [6] R. Ravi, S. v. Yadhukrishna, and R. Prithviraj, "A Face Expression Recognition Using CNN LBP," in *Proceedings of the 4th International Conference on Computing Methodologies and Communication, ICCMC 2020*, Mar. 2020, pp. 684–689. doi: 10.1109/ICCMC48092.2020.ICCMC-000127.
- [7] J. J. Lien, J. F. Cohn, and C.-C. Li, "Automated Facial Expression Recognition Based on FACS Action Units."
- [8] IEEE Computer Society and Institute of Electrical and Electronics Engineers, *12th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition : FG 2017 : proceedings : 30 May - 3 June 2017, Washington, D.C.*
- [9] Y. Tian, T. Kanade, and J. F. Cohn, "Facial Expression Recognition," in *Handbook of Face Recognition*, Springer London, 2011, pp. 487–519. doi: 10.1007/978-0-85729-932-1\_19.
- [10] M. J. den Uyl and H. van Kuilenburg, "The FaceReader: Online facial expression recognition."
- [11] G. Guo and C. R. Dyer, "Learning from examples in the small sample case: Face expression recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 35, no. 3, pp. 477–488, Jun. 2005, doi: 10.1109/TSMCB.2005.846658.
- [12] International Neural Network Society, IEEE Computational Intelligence Society, and Institute of Electrical and Electronics Engineers, *2015 International Joint Conference on Neural Networks (IJCNN) : date 12-17 July 2015.*
- [13] J. Zeng, S. Shan, and X. Chen, "Facial Expression Recognition with Inconsistently Annotated Datasets." [Online]. Available: <https://github.com/dualplus/LTNet>.
- [14] N. B. Kar, K. S. Babu, A. K. Sangaiah, and S. Bakshi, "Face expression recognition system based on ripplelet transform type II and least square SVM," *Multimed Tools Appl*, vol. 78, no. 4, pp. 4789–4812, Feb. 2019, doi: 10.1007/s11042-017-5485-0.
- [15] Ramirez Rivera, S. Member, J. Rojas Castillo, and O. Chae, "Local Directional Number Pattern for Face Analysis: Face and Expression Recognition," 2011.
- [16] M. Stewart Bartlett, G. Littlewort, I. Fasel,  , J. R. Movellan, and   , "Real Time Face Detection and Facial Expression Recognition: Development and Applications to Human Computer Interaction." [Online]. Available: <http://mplab.ucsd.edu>
- [17] H. Yang, U. Ciftci, and L. Yin, "Facial Expression Recognition by De-expression Residue Learning."
- [18] H. Ai, C. Huang, Y. Wang, and B. Wu, "Real Time Facial Expression Recognition with Adaboost. A Study on Robust Face Recongntion & Verification View project Real Time Facial Expression Recognition with Adaboost," 2004, doi: 10.1109/ICPR.2004.733.
- [19] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on Local Binary Patterns: A comprehensive study," *Image Vis Comput*, vol. 27, no. 6, pp. 803–816, May 2009, doi: 10.1016/j.imavis.2008.08.005.
- [20] C. Shan, S. Gong, and P. W. McOwan, "Robust facial expression recognition using local binary patterns," in *Proceedings - International Conference on Image Processing, ICIP*, 2005, vol. 2, pp. 370–373. doi: 10.1109/ICIP.2005.1530069.
- [21] Tawari and M. M. Trivedi, "Face expression recognition by cross modal data association," *IEEE Trans Multimedia*, vol. 15, no. 7, pp. 1543–1552, 2013, doi: 10.1109/TMM.2013.2266635.
- [22] J. Xiang and G. Zhu, "Joint face detection and facial expression recognition with MTCNN," in *Proceedings - 2017 4th International Conference on Information Science and Control Engineering*,



- ICISCE 2017*, Nov. 2017, pp. 424–427. doi: 10.1109/ICISCE.2017.95.
- [23] C. Xu *et al.*, “A novel facial emotion recognition method for stress inference of facial nerve paralysis patients,” *Expert Syst Appl*, vol. 197, Jul. 2022, doi: 10.1016/j.eswa.2022.116705.
- [24] S. P. Yadav, “Emotion recognition model based on facial expressions,” *Multimed Tools Appl*, vol. 80, no. 17, pp. 26357–26379, Jul. 2021, doi: 10.1007/s11042-021-10962-5.
- [25] K. Zaman, S. Zhaoyun, S. M. Shah, M. Shoaib, P. Lili, and A. Hussain, “Driver Emotions Recognition Based on Improved Faster R-CNN and Neural Architectural Search Network,” *Symmetry (Basel)*, vol. 14, no. 4, Apr. 2022, doi: 10.3390/sym14040687.
- [26] K. N. Lam, K. N. T. Nguyen, L. H. Nguy, and J. Kalita, “Facial expression recognition and image description generation in vietnamese,” in *Frontiers in Artificial Intelligence and Applications*, Oct. 2021, vol. 340, pp. 63–69. doi: 10.3233/FAIA210176.
- [27] Institute of Electrical and Electronics Engineers. Madras Section and Institute of Electrical and Electronics Engineers, *Proceedings of the 2020 IEEE International Conference on Communication and Signal Processing (ICCSP) : 28th - 30th July 2020, Melmaruvathur, India*.
- [28] M. Sajjad, S. Zahir, A. Ullah, Z. Akhtar, and K. Muhammad, “Human Behavior Understanding in Big Multimedia Data Using CNN based Facial Expression Recognition,” *Mobile Networks and Applications*, vol. 25, no. 4, pp. 1611–1621, Aug. 2020, doi: 10.1007/s11036-019-01366-9.