Fusion of Deep Learning Features and Histogram of Oriented Gradients for Facial Expression Recognition

Table of Contents

List of Figures

List of Tables

Acknowledgement

This research paper is made possible through the help and support of many people. Please allow me to dedicate my acknowledgment of gratitude, especially toward the significant advisors and contributors:

First and foremost, to Dr. Tina Baker for her support and encouragement. Her guidance and advice carried me through all the stages of this research. In addition, I would like to express my profound thanks to Dr. Bashir Dodo, Dr. Ronakben Bhavsar, Dr. Godswill Lucky, Dr. Waseem Ahmad, Mr. Saul Cross, Dr. Piyush Dhawankar, and Dr. Claire Ingram.

Finally, I would like to give special thanks to my family, friends, and colleagues for their continuous support and understanding. Your support and love for me sustained me this far.

Dr. Sam Khoze

Executive Summary

Facial changes in reaction to an individual's internal emotional states, goals, or social communications are referred to as facial expressions. Behavioral scientists have been analyzing facial expressions for a long time. Based on facial video processing, this innovative research detects the face and extracts features like the mouth, left eye, right eye, and nose. These features and Histogram of Oriented Gradients (HOG) features are merged to create a new dataset called HOGFER. This research proposes a model that combines a Convolutional Neural Network (CNN) and HOG features. Facial Expression Recognition (FER) analysis goes through face detection, facial expression detection, and classification of an emotional state.

The research utilizes live stream videos as input to the system. The video is converted to frames. The selected frames are fed into the Multi-task Cascaded Convolutional Neural Network (MTCNN) model for face detection. DR-GAN were considered for face rotation; however, in a few cases, they could not maintain the original expression in the occluded face part while converting it to a frontal face view. The output of MTCNN is fed to the trained 2-CNN model. As there was skewness for a few classes, Random Oversampling has been employed to augment skewed classes.

The model classifies the input frame as one of the seven basic expressions from the dataset. The model is trained on FER2013 and the HOGFER dataset. The experimental results show that the model gives reasonable accuracy, recall, precision, and F1-score for the FER2013 and the HOGFER datasets. The facial expression recognition rate increases to 83.46% on average (up to 84.9%) for the HOGFER dataset as compared to 83.17% on the FER2013 dataset, respectively, as validation accuracy. The model gives a training accuracy of 98.80% and 98.67% for HOGFER and FER2013, respectively. The proposed model performs well for all classes after data augmentation has compensated for the fewer samples in those classes.

The proposed research has given a reasonable solution to facial expression recognition by selecting an efficient model MTCNN for face detection, applying Random Sampling as a data augmentation technique for skewed classes, and choosing a deep convolutional neural network for facial expression identification.

This set of model selection and hyperparameter tuning has given results that outperform the existing solutions for facial expression recognition.

Chapter 1: Introduction

1.1 Introduction

Face detection algorithms are integral to all facial detection and recognition systems [1]. There are two general types of facial recognition algorithms: feature-based and holistic. The feature-based algorithms focus on facial landmarks, spatial parameters, and their correlation to other features, while holistic methods view the human face as a whole unit.

According to Borkar et al. [2], face detection and recognition are used in various parts of the industry. The smartphone, for example, is set up by the owner; the face of the owner is detected and recorded to unlock the smartphone. The same is true for the automation industry, where face recognition can help restrict access to certain areas of buildings in data centers and server rooms. Security concerns are addressed by facial recognition, which is highly reliable and accurate. Gender Classification, Landmark Detection, Attendance, Snapchat/Instagram camera filters, and Crowd Analysis are some eminent face detection and recognition applications from the industry. Robotics instructors are anticipated to deliver online lectures in a revised format to represent their students' personalities better and adapt to their emotional states [3]. Figure 1 explains the process of face detection.

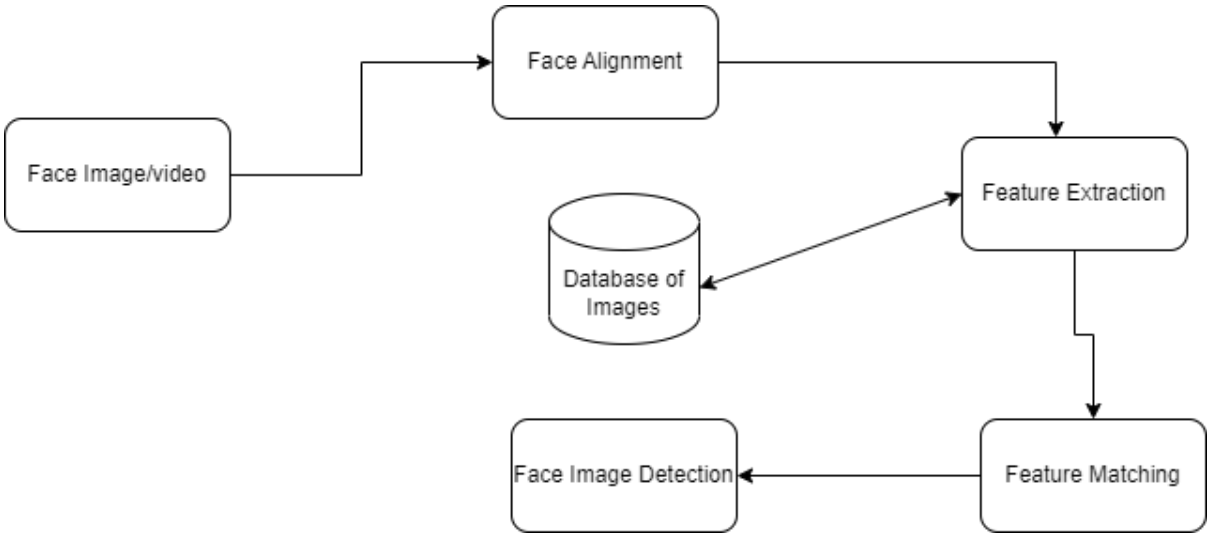


Figure 1 Face detection process

1.2 Challenges of Face Detection Techniques:

Numerous factors hinder the performance of a Face Detector. These include the following:

1.2.1 Occlusion

Occlusion of the face is considered one of the most intractable problems because we need to predict the location or size of the occluded part in a face image before it happens [4],[5]. The effect of occlusion on the ability to detect the face is significant as only a portion of the face is visible, making it difficult for a system to detect the face accurately, as shown in Figure 2.

Figure 2 Occluded face images

The Occluded Face Recognition (OFR) issue is resolved by querying a gallery of occlusion-free faces and probing another dataset of occluded faces.

1.2.2 Lighting

Illumination is a critical factor in the design of face recognition applications. However, a systematic solution to illumination challenges has become possible in the last few years. The approaches like illumination cones and spherical harmonics have helped address this issue. Each image in the top row was taken from the same viewpoint but under different external illumination conditions. The images at the bottom are taken under the same lighting conditions and viewpoint [6],[7],[8]. Figure 3 shows various illumination effects.

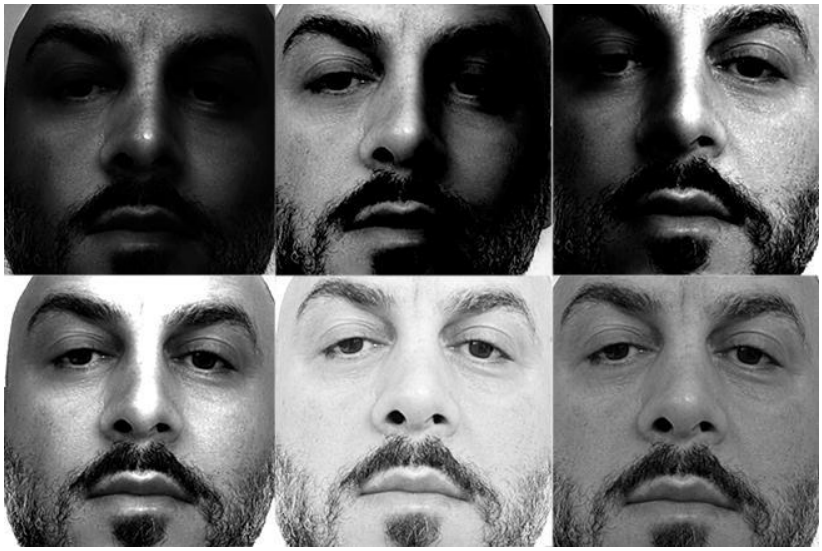


Figure 3 Impact of illumination: top row has the same viewpoint, but different illumination bottom row has the same illumination and viewpoint

1.2.3 Pose

The varying pose is another challenge affecting the face recognition process. The variation in pose makes one face appear different. Most face detection algorithms can detect faces captured from the front only wherever a face is sideways or turned to a side only. Figure 4 shows the face rotated in different directions.

Figure 4 Varying Pose conditions

1.2.4 Accessories/Makeup/Facial Hair

When designing or training the Face Detection system, accessories, facial hair, or modifications might also affect the detection system's performance. Masks, Beards, Sunglasses, Tattoos, and too many layers of makeup are a few examples of this challenge to face detection, as shown in figure 5.

Figure 5 A few examples of Accessories/Makeup/Facial Hair (images generated by AI)

1.2.5 Scale of Face

Scale poses a vexing problem in unconstrained environments; faces captured at large distances are significantly harder to recognize than faces captured at close ranges. Deep learning models like CNN-based face detectors are inefficient in handling faces of varying scales[9]. The scale of the face might change for the image/video frame, sometimes becoming so low that the face gets too small to be detected [10], as shown in figure 6.

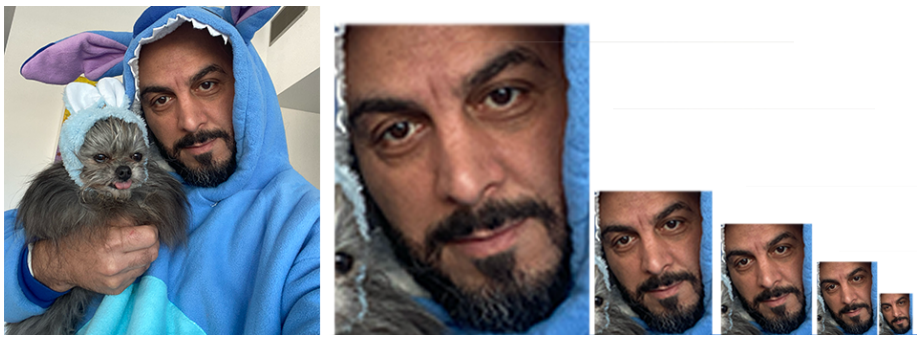


Figure 6 Scaling issues and challenges with face recognition in uncontrolled environments

1.2.6 Tiny Face Detection

An accurate and reliable algorithm to detect tiny faces in images, videos, and other real-life situations has always been the focus of research. A Single Shot MultiBox Detector (SSD) framework is an efficient framework with balanced performance for tiny face detection. A Single Shot MultiBox Detector (SSD) framework is an efficient framework with balanced performance for tiny face detection. The framework

provides a one-stage, end-to-end training that can be embedded in other systems. It offers hierarchical training by adding different scaled features. The framework performs well in real-life situations, even for multiple objects at a trade-off between speed and accuracy [11].

Emotions can be described by dimensional space called emotion dimensions using measures like valence (a measure of pleasantness), arousal (a measure of activation), etc. [12], [13]. Various metrics such as facial expressions, voice, body gestures, touch gestures, and biosensors can help identify emotions; however, the most significant is facial expression, as it clearly represents emotion [14].

Emotion recognition has been the focus of research in numerous fields in the contemporary world: Marketing, Psychology, Virtual Reality, Smart Surveillance Systems, and Entertainment are some domains with a strong base [15]. Facial Emotion Recognition (FER) can be accomplished with the help of non-AI-based solutions or AI-based approaches. Any solution for FER must be invariant to lightning, positioning, rotation, mirroring, occlusion, complex background, and scaling as is in the approaches based on Machine Learning (ML) and Deep Learning (DL) [15], [16]. The FER process is shown below in figure 7:

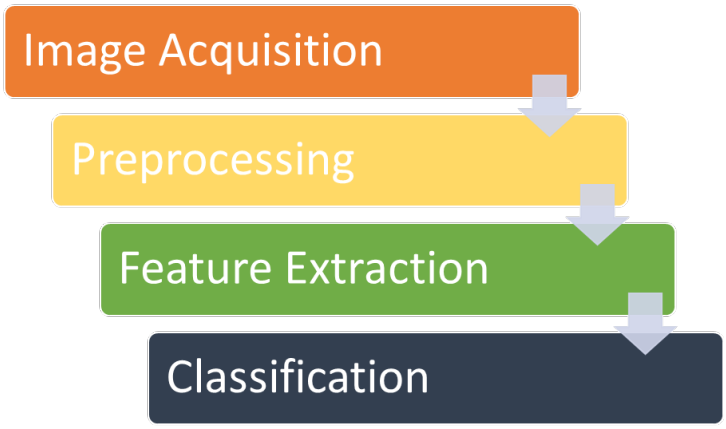


Figure 7 The Facial Recognition Process

Numerous applications require face detection to be used as input of the FER system [17]. A few of the reliable face detection algorithms are from domains like Active Shape Model, Low-Level Analysis, Feature Analysis, Linear Subspaces, Eigenfaces

algorithm with Principal Component Analysis (PCA), and Haar-Cascade Classifiers Viola Jones [18], [19].

Ekman and Friesen [19] emphasized seven basic emotions: Fear, Anger, Sadness, Happiness, Surprise, Disgust, and Neutral. FER systems can be classified based on handcrafted features or the DL-based approach.

The FER algorithms have been used in education to help teachers assess student learning capabilities and monitor the examination surroundings. It includes a Student Behavior Monitoring System, Intelligent Learning Environment (ILE) processing, and a Coaching System. FER diagnoses mental and psychiatric disorders in the medical field, assisting doctors in making timely treatment decisions for children with autism and depression [20], [21]. Internet technology companies mostly own FER systems to solve problems like the face unlock function of iPhones.

1.3 Research Objectives

This research hypothesizes that combining traditional features with DL techniques can improve performance. The specific objectives of this research are as follows:

- Creating a robust and efficient model to detect base emotions from video streams in a constraint-free environment
- Improving the emotion recognition performance in terms of time and precision
- Merging handcraft features with DL networks to boost the performance

1.4 Research Methodology

This study will work with the following research questions:

Q1: Is GAN a full replacement of the traditional pre-processing step?

Q2 Is rotated face input in any perspective returning the face with maintaining facial expressions as in the original image?

Q3: Is there any substantial accuracy improvement by merging certain efficient handcrafted features with DL?

1.5 Dataset

The proposed study uses FER2013, an open-source dataset, for development purposes. The dataset consists of 48x48 pixel grayscale images of faces. The dataset contains approximately 35.9K images [22], [23]. The images are distributed into seven categories((0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). The training set consists of 28,709 examples, and the public test set consists of 3,589 examples covering image variations in pose, illumination, orientation, rotation, size, and scale differences. The Facial Expression Recognition 2013 (FER2013) dataset was first made available through a competition. Pierre Luc Carrier and Aaron Courville are the authors of FER2013. It's a component of a bigger, continuing project at that time. The dataset was created by searching through 184 emotion-related keywords for photographs of faces that matched them using the Google image search API, such as "blissful," "enraged," etc. Nearly 600 strings were used as facial picture search queries. The queries were created by combining these keywords with phrases relating to gender, age, or ethnicity.

1.6 Ethical considerations

Due to the nature of this project, working with facial information, this research raises potential ethical implications that must be considered. The proposed method decided to use data already in the public domain. No private users' data will be collected and stored by third parties servers, and everything will be run on the machines owned by the end users. For ethical approval, the authorities acknowledged the sensitivity of the research subject, submitted the ethics form in advance, and addressed all the concerns.

Chapter 2: Literature Review

According to Mehrabian, spoken words transmit 7% of communication, voice intonation transmits 38% of the message, and facial emoticons transmit 55% of the message [24]. Detect facial emotions from a face; various features of a face make it possible. However, feature extraction from a video can be time-consuming and laborious. Researchers extracted various traditional features from images [25], [26], [27], [28], [29], [31] that are effective with traditional ML algorithms as well as DL methods. Figure 8 elaborates the overall flow of the proposed research:

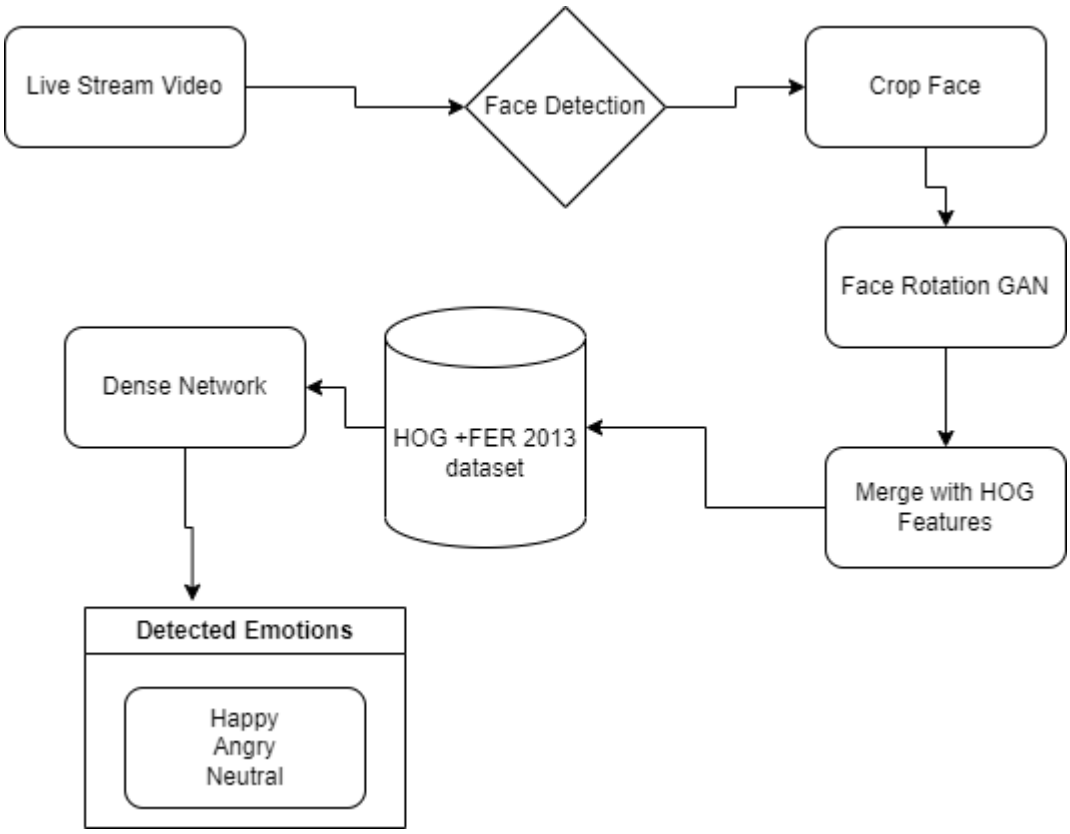


Figure 8 Overall Flow of The System

2.1 Algorithms for Face Detection

Several technological advancements have been made in face detection, involving computer vision and machine learning (ML) based techniques. According to Yang et al. [32], these algorithms can be classified as shown in figure 9:

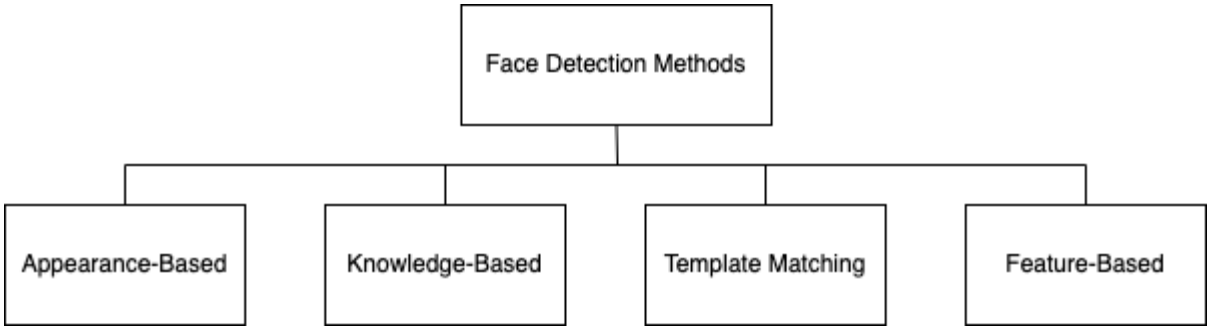


Figure 9 Face Detection Approaches

2.1.1 Knowledge-based Face Detection.

These methods use the rules derived by the researchers based on their awareness of a human face. These rules define the relationship between the different features of a face. One example of such rules is the relationship between two eyes, a nose, and a mouth. Relative distance and position are candidates for such relationships. The challenge in this approach is to map human knowledge to accurate rules. The generalized rules may generate too many wrong results. This approach is not pose-invariant. Rules are coded based on human understanding of the properties of the facial regions, such as the intensity distribution and differences.

2.1.2 Template Matching

Template matching uses predefined face patterns (usually frontal) that are manually created. Each contour of the mouth, the eyes, the nose, and the face are checked independently for correlations with the standard patterns. The correlation value decides whether a face exists or not. The template matching is not invariant to scale, pose, and shape variation, which can be achieved through multiresolution, multiscale, sub-templates, and deformable templates.

Face detection occurs if adequate facial features satisfy their proportions, ratio tests based on a face template. Lanitis et al. [33] worked on training sets of images in which they manually annotated sampled contours such as the eye boundary, nose, etc. A set of sample points represent the shape, and the Point Distribution Model (PDM) represents the shape's normalized intensity appearance. Active Shape Model (ASM) helps to find faces in new images using face-shape PDM. The deformation of

the image is mapped to the average shape and intensity parameters. The approach is easy to implement but inefficient for facial recognition.

2.1.3 Appearance-Based Methods

The “templates” in appearance-based methods do their learning from sample images. These methods are based on statistical analysis and machine learning to find the relevant features of the face and nonface images. The dimensionality reduction in these methods leads to improved computational efficiency and detection accuracy.

This approach also uses a discriminant function like separating hyperplane, decision surface, and threshold function to distinguish between face and non-face classes. These approaches, meanwhile, could suffer when there are big variations in lighting, facial expression, and other elements. This is because face patterns in high-dimensional space are found on a complicated non-linear, and non-convex basis.

2.1.4 Feature-Based Face Detection Methods

Compared to a knowledge-based top-down approach, researchers have tried to find invariant features of faces. The features like the nose, mouth, eyes, and eyebrows are commonly extracted using edge detectors. These extracted features define their inter-relationship to detect a face.

Gogic et al. [28] proposed a method for facial expression recognition using a Local Binary Pattern (LBP) and a Shallow Neural Network (SNN). This architecture can optimize the performance of time and accuracy.

Moreover, it can adjust the weights from a small dataset.

Christou et al. [29] proposed a Dense Neural Network based method for facial recognition, which has greatly improved recognition accuracy in the final layer and provides a clear separation of classes.

Furthermore, the authors used the FER 2013 dataset and found better accuracy for the proposed method compared to existing studies conducted on the same dataset.

Reddy et al. [30] also proposed a hybrid technique with hand-crafted (facial landmark point) with XceptionNet. They compared the model’s performance with existing DL

Networks, including DenseNet, and AffectNet. Their research proved that the hybrid models based on the customized features with DL methods significantly improved overall performance.

Georgescu et al. [34] have combined CNN and handcrafted features created using the Bag of Visual Words (BOVW) model. A local learning model (LLM) has been applied after feature vector concatenation and L2-normalisation. The LLM uses the k-nearest neighbors model to find the best samples for a test data sample. A Support Vector Machine (SVM) classifier predicts the class label for the test images. According to their proposed model, the accuracy of FER2013 was 75.42%. Illumination, noise, shadows, and occlusion may affect the image features. Space Gray-Level Dependence Matrix (SGLD) [35].

Shan et al. [36] have used “Local Binary Patterns (LBP)” for FER. They have used Boosted-LBP to get important LBP features using Support Vector Machine classifiers on these features. Their LBP features-based technique works well for compressed low-resolution images captured in live streaming video sequences; their approach needs improvement for high-resolution images.

Shi et al. [37] proposed a traditional Scale-Invariant Feature Transform (SIFT) feature extraction method; the method identifies the face’s boundary and generates a vector. Extracted features were scale, rotation, and illumination invariant to images. However, the accuracy of the proposed method highly depends on the detection of face shapes.

Dalal et al. [38] have used Histograms of Oriented Gradient (HOG) descriptors to detect humans. They proved that reasonable orientation binning, fine-scale gradients, relatively coarse spatial binning, and high-quality local contrast normalization in overlapping descriptor blocks equally participate in finding results. Their proposed solution worked well on the MIT pedestrian database.

Lokku et al. [39] have worked on the Optimised Scale-Invariant Feature Transform (OSIFT). These features are optimized by Hybrid Metaheuristic Algorithms like Spotted Hyena Optimizer (SHO) and Beetle Swarm Optimization (BSO) to generate the proposed Spotted Hyena-Based BSO (SH-BSO). In addition, the local tri-directional pattern (LTriDP) is generated and combined with optimized SIFT. This

helps minimize the hidden neurons in the Convolutional Neural Network (CNN), resulting in increased accuracy. This is important because less neurons mean less computation.

Kalsum et al. [40] exploited a fusion of Spatial Bag of Features (SBoFs) and Spatial Scale-Invariant Feature Transform (SBoF-SSIFT) to get an efficient FER. K-nearest neighbor (KNN) and SVM with linear, polynomial, and radial basis function kernels are applied for the classification of FER. SBoFs descriptors are size invariant, while Spatial SIFT and SURF features are scaling, rotation, translation, and Projective Invariant and are less affected by illumination variation. The proposed model has been tested on extended Cohn–Kanade (CK+) and Japanese female facial expression (JAFFE) data sets, giving an accuracy of 98.5% on CK+ and 98.3% on the JAFFE data set.

Hosseini et al. [41] have demonstrated that using the proper feature along with CNN enhances the system's accuracy. They used Gabor filter bank output, fed to CNN as Tensor. They also fused these results as images by taking a weighted sum of images and Gabor responses. Their research has shown that these hybrid methods give better results than simple CNN-based methods.

2.2 Multi-task Cascade Convolutional Neural Network (MTCNN)

Detecting small objects is challenging, and any method we apply must give good results related to image resolution, scale invariance, and contextual data. The model trains detectors based on the scales using features from multiple layers. To train models for small objects, a foveal descriptor captures details to detect small objects employing high-resolution image features. The frontal face view can be taken with the help of DR-GAN [42]. The collected data, related to resolution, scale and context, helps design an efficient face detector that outperforms other algorithms on benchmark databases [43].

The proposed MTCNN takes some time to train but gives highly accurate predictions. The time taken by MTCNN can be reasonably reduced by using pre-trained models. This is why MTCNN is a reliable model for performing real-time face detection. The MTCNN is robust and detects faces of variable size, illumination, and rotations.

2.3 Haar Feature Selection

Alfred Haar proposed Haar features as a collection of rescaled square shape functions. These Haar features work on parts of the face to detect the human face. In Haar-like features, adjacent rectangular regions within a detection window are compared, their pixel intensities are summed, and the neighbors at each site are considered. The two adjacent Haar features are rectangles above the eye and the cheeks [43]. Two adjacent rectangles above the eye and the cheek region are common Haar features for face detection. Haar features are best suited for detecting edges and lines, as shown in figure 10.

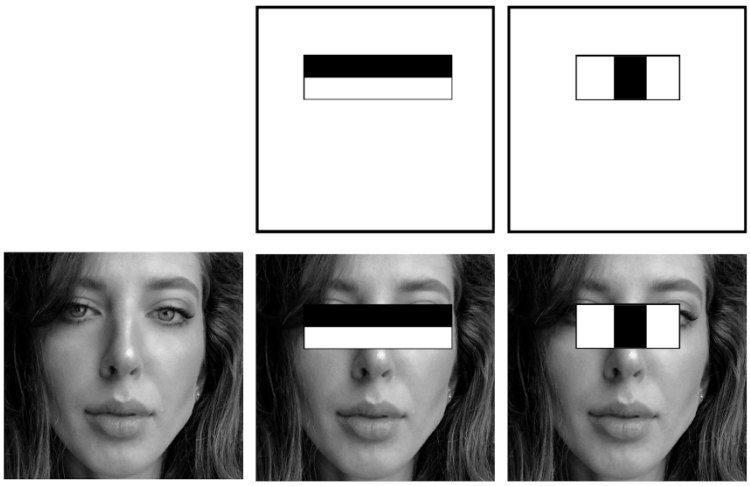


Figure 10 Face Detection with HaaR-like features

The research by Viola Jones has elaborated a new image representation known as an integral image. This image presentation does rapid calculations for features. The research also proposed a classifier that selects features using AdaBoost. The number of Haar-like features exceeds the total Pixels in an image. Having fewer but critical features helps in faster and more accurate classification. The modified AdaBoost model helps in Feature selection. The weak learners are designed so that every weak classifier only depends on one feature. The combination of consecutive combined classifiers boosts the detector's speed by giving attention to particular segments of an image. Each frontal face has 38 layers of cascaded classifiers, resulting in 6061 features [44].

2.4 Histogram of Oriented Gradients (HOG)

The histograms of oriented gradients (HOG) feature extractor calculates the distribution (histograms) of directions of gradients of the image for detecting objects [45]. The HOG algorithm breaks an image into squared cells, calculates an oriented gradient histogram for each cell, and normalizes the result using a block-wise pattern, calculating a descriptor for each cell.

This algorithm is not a rotation-invariant method. As a result, it can only be used with objects observed in the correct orientation when applied to object detection tasks. Stefanou et al. have used Particle Swarm Optimization (PSO) model with a variant of the HOG descriptor. This variant of the HOG model uses a sliding window approach and calculates all permutations of windows in an image. Their proposed method can easily construct a tracking framework, which detects objects in the vicinity of the estimated solution in the previous frame [45].

This research uses a HOG descriptor to fuse handcrafted features with the original fer2013 dataset. Figure 11 shows the HOG features for the given image.

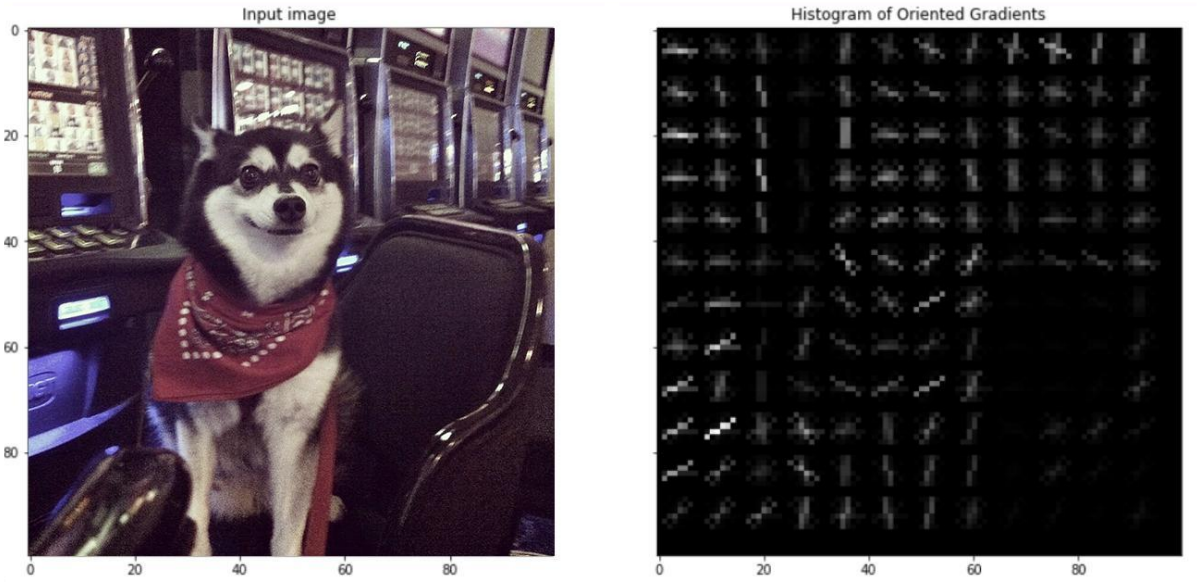


Figure 11 HOG Applied to an Image

2.5 Benchmarking Datasets for Face Detection

The advances in the Computer Vision (CV) domain have created solutions to outperform humans in facial recognition and other related applications [46]. The following is a list of a few valuable and popular datasets for human face recognition and detection [47]: AffectNet, Aff-Wild, Belfast Database, DISFA, Extended Cohn-Kanade Dataset (CK+), FERG (Facial Expression Research Group Database)-DB, IMPA-FACE3D, FEI Face Database, Indian Spontaneous Expression Database (ISED), Japanese Female Facial Expressions (JAFFE), MMI Database, Multimedia Understanding Group (MUG), Indian Semi-Acted Facial Expression Database (iSAFE), Real-world Affective Faces Database (RAF-DB), Oulu-CASIA NIR-VIS database, Radboud Faces Database (RaFD), Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS).

Chapter 3: Methodologies and Research Design

The literature review demonstrates an understanding of the problem. By summarising and critically evaluating existing work in the field, the study offers a basis for how other scientists have approached the technology.

To answer the research questions listed in the introduction, it is evident that the research methodology is quantitative due to the nature of the data and analysis techniques. Each research question can be answered using a quantitative approach using metrics. In this situation, qualitative methods are inappropriate as the data analyzed is numerical.

As the nature of the data manipulation uses supervised machine learning algorithms, a positivist approach has been adopted. This approach ensures that the researchers' views are not biased and that the values and beliefs held by the researcher do not influence the main objectives of the work [48]. Using a deductive approach analysis [49], methods have been designed to test the research hypothesis. Analysis of the data is shown in graphs, tables, and this analysis is used as a measure of the credibility of the work.

The analysis section of this chapter focuses on the models used during the development of facial expression classification at the stages of frame selection, face detection, and facial emotion identification. Although the processing cost is substantially higher for video sequences than still images, the association between consecutive image frames provides more information for facial expression recognition.

The process starts with live streaming input video converted to frames. A few selective frames are fed into the developed model, moving through phases of face detection and face rotation to get the frontal view, then feeding into a simple CNN model and feeding to a fusion-based CNN with a handcrafted feature model.

3.1 Data Pre-processing

The pre-processing starts with defining the frame size. The videos are converted to frames. In the case of videos, many frames are repeated before a frame can give a better entropy measure than other frames giving the same information, as shown in figure 12.

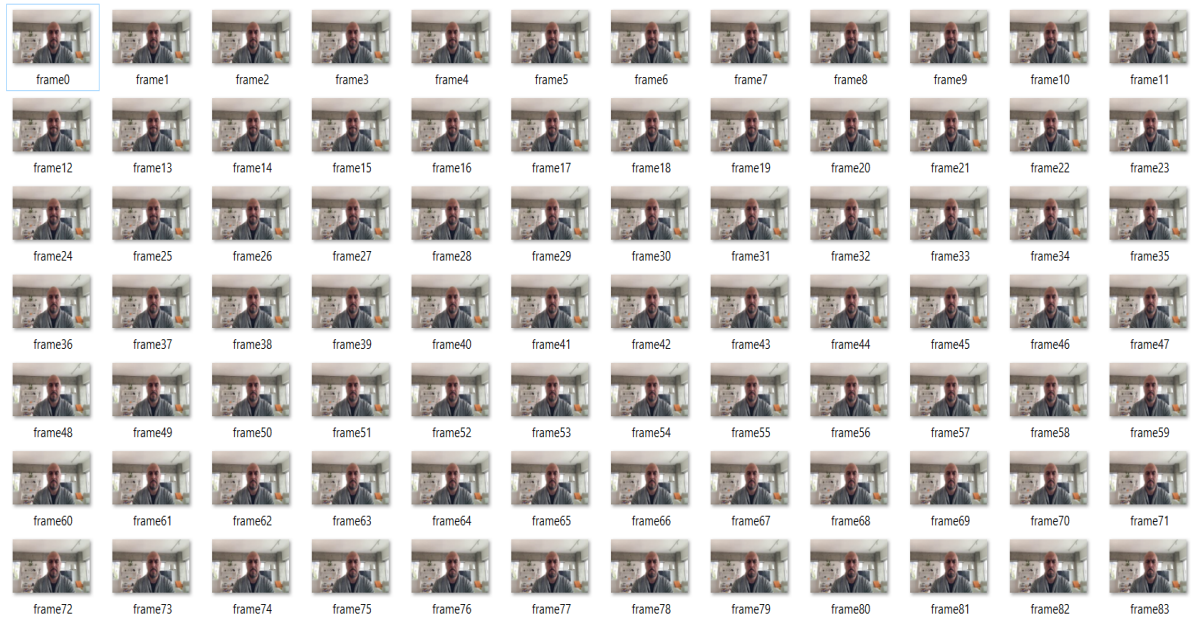


Figure 12 Facial Expressions Captured in 2.3 Second Long Video

Figure 12 shows two emotions, anger, and happiness, recorded in 2.3 seconds. A reasonable interval is chosen to avoid repeating frames. Likewise, the interval is not large enough to lose any important posture.

The dataset contains 547 disgust, 4953 anger, 4002 surprises, 6077 sadness, 6198 calm, 8989 happiness, and 5121 fear images. The system has assigned the following labels to emotions for ease of computation, as shown in figure 13:

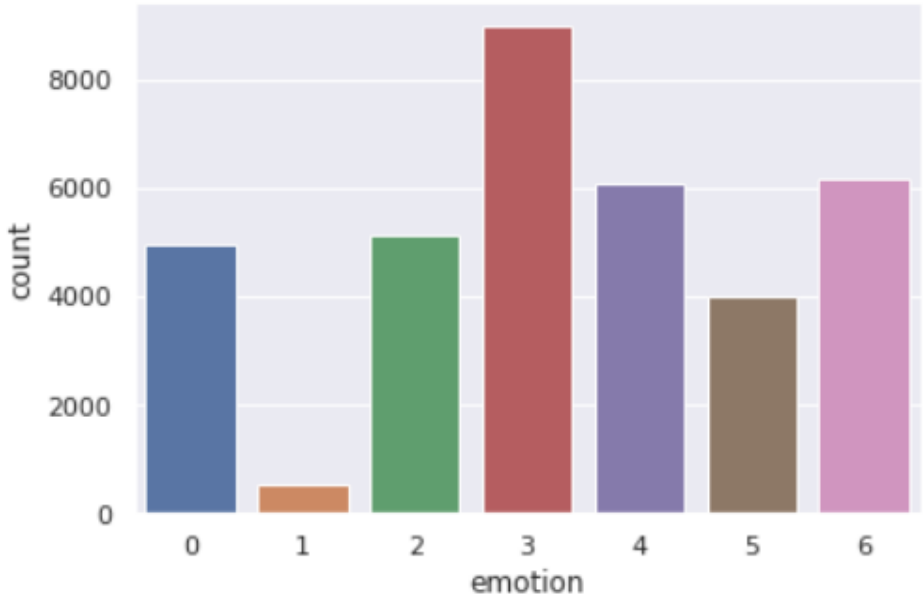


Figure 13 Facial Expression Strength in the Database

The label 0 represents 'anger', 1 represents 'disgust', label 2 represents 'fear', Label 3 represents the 'happiness' class, label 4 is for the 'sadness' class sadness, 5 is for 'surprise' expression, and 6 represents the expression 'neutral'.

The graph above shows that the data is skewed, showing biases for certain classes, especially for class 1 (disgust) and class 5 (surprise), so random oversampling based on the majority class has been adopted to make up for the skewed data [50].

After data augmentation with Random Oversampling, the number of samples for the input dataset along with the number of samples is:

Total Samples after Oversampling: **62923**

Training Samples: **56631**

Testing Samples: **6292** (However, the experiment was repeated for 10% 20% and 30% samples as testing samples)

The class prediction is more accurate for a higher number of samples in a class, deeper model and a higher number of dense layers [51]. An evenly distributed class structure helps in enhancing accuracy.

The live stream is converted to frames. The average expression of a temporal series of various faces has a time constant of about 800 ms. It is independent of the temporal frequency or the collection size [52], so two samples were taken in frames generated in 1 second, assuming 30 frames in a 1-second video. Iterate through a video, converting it to frames. The deep learning model resizes the frames/image to the required size.

Both the algorithms, HaaR Cascade Classifier and MTCNN, have been tested; both have detected faces with reasonable accuracy. However, MTCNN has been preferred over the HaaR classifier due to its invariance to face rotation to a great extent. Figure 14 shows this process:

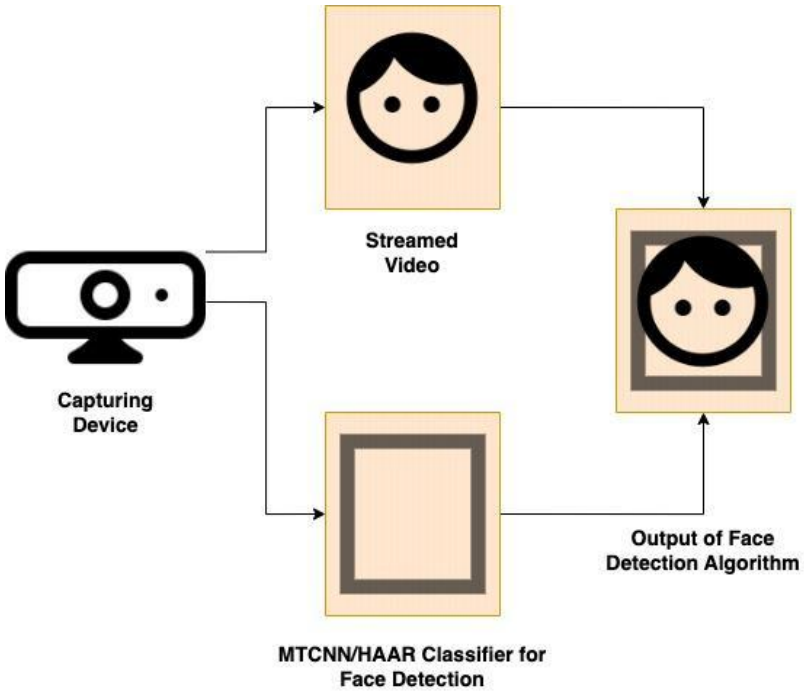


Figure 14 The Process of MTCNN/HaaR Based Face Detection

3.2 Face Detection Based on MTCNN

The overall process of MTCNN is shown in the figure below. The original image is fed to MTCNN after being extracted from the input stream. The model detects all the faces and features of the faces, as shown in figure 15.

Figure 15 Face Detection With MTCNN

The process of face detection by MTCNN goes through the following stages:

Stage 1: To combine highly overlapping candidates, the cascaded framework proposition network (P-Net), a CNN, provides the necessary facial windows and the vectors for bounding box prediction based on non-maximum suppression (NMS).
Stage 2: Refine network (R-Net); a different CNN decreases anticipated bounding boxes and performs NMS.

Stage 3: This stage is identical to step 2, except that MTCNN now predicts the positions of the five facial landmarks, namely the left eye, right eye, nose, mouth left, and mouth right coordinates [11].

3.3 GAN for Face Rotation

A face with any posture, can be frontal or rotated using the encoder-decoder structured generator used by DR-GAN [42]. The pose code in the generator and the pose estimation in the discriminator decouple our learned representation from the pose variation [11]. For **Single-image**, the encoder-decoder structured generator is used by DR-GAN to learn an identity representation for a face image, where the representation is the encoder's output and the decoder's input. It is a generative representation since it serves as the decoder's input to create different faces of the same subject by virtually rotating their face.

Second, a face's appearance is influenced by various distracting factors, including position, lighting, and expression, in addition to its identity. As a result, the distracting side variants would always be present in the identity representation that the encoder learned. To fix this, we intentionally untangle these variations to train a discriminative representation using class labels and side information like position and illumination. The following images in figures 16, 17, 18, and 19 show that sometimes DR-GAN do not retain the original face or original expression.

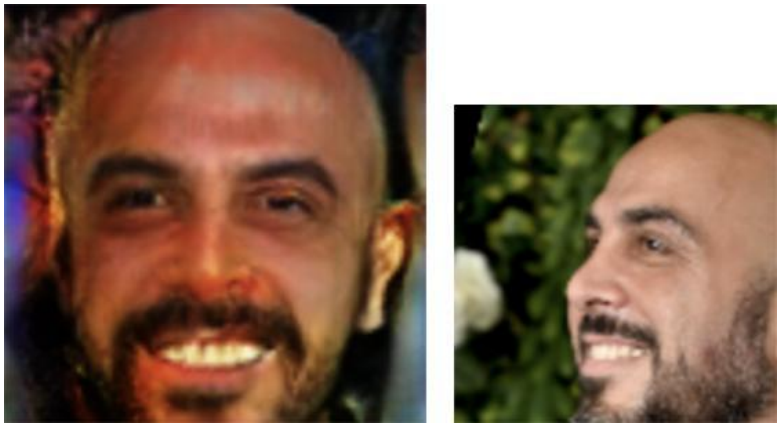


Figure 16 face rotation using DR-GAN

Figure 17 Face Rotation using DR-GAN

In another case, the image produced is entirely different from the original image, as shown in the figure below.

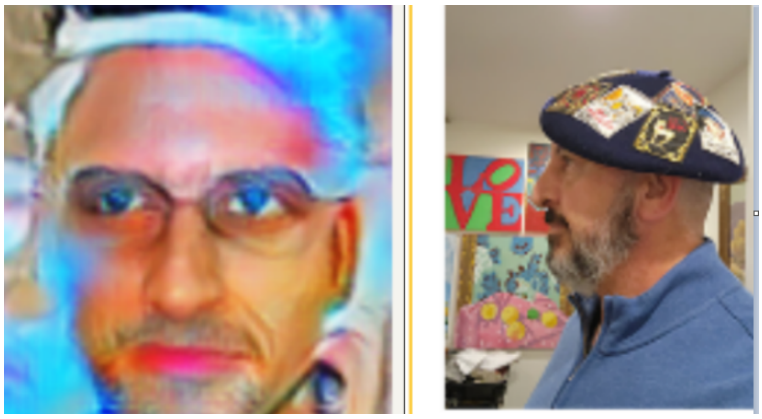


Figure 18 Face Rotation using DR-GAN

The actual rotated face is shown in figure 19.

Figure 19 Actual Face vs. Rotated face with Real Camera

Applying DR-GAN for face rotation, it was observed that in a few cases, the original face expression could not be retained, as shown in Figure above.

3.4 Model Selection

Several deep learning models have been applied for facial recognition, like Region Attention Networks (RAN), Transfer Learning, GAN, etc. However, CNNs have been employed effectively in the domain of image recognition. The problem of facial expression recognition lies in the domain of image classification. To solve this problem, 2-CNN has been used in the proposed model. A CNN model is a sequence of Convolution layers, a Max-pooling layer, and Dense layers. The Dropout layer has been used to control the number of neurons growing beyond a certain level.

3.5 Data Augmentation

There are two versions of this dataset, the CSV format and the Images based dataset format. Both datasets convey the same information but use different formats for processing. The process of data augmentation is also different. The CSV-based dataset uses Oversampling technique, while the images-based FER2013 uses traditional data augmentation techniques based on shear, translation, resizing, rotation, etc. The proposed research is based on RandomOverSample using a sampling strategy set to **auto.**

3.6 Shallow Neural Networks

Due to the nonavailability of relevant, accurate data, these models overfit on small datasets. The deep learning models use optical flow images rather than classifying the images directly due to micro expressions (short facial expressions that disappear within a fraction of a second), reflecting a person's genuine emotions.

A shallow neural network consists of an **Input layer** (All inputs enter the model here), a **Hidden layer** (can be one or more. All processing is done here using linear/non-linear functions, passing results to the output layer), and the Output layer (the layer which provides us with the output) [5]. The research performed experiments to analyze the performance of the model. However, the results obtained were poor, giving very low accuracy, so the idea of using shallow neural networks was dropped.

3.7 Dense Neural Networks

A Dense Neural Network (DNN) is a biologically inspired computational model simulating the human brain processes information. The neurons in DNN use learning to self-optimize. The densely connected neurons receive input from all the neurons from the previous layer, which becomes an input for all the neurons in the following layers [51]. The proposed research is based on the power of DNNs. Figure 20 shows the fully connected neural network.

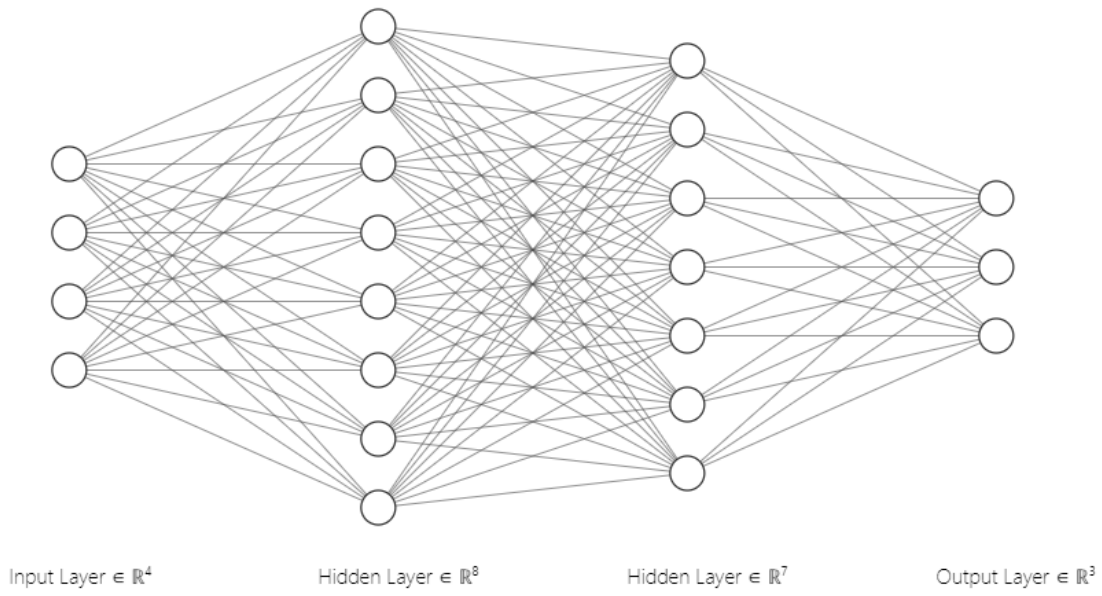


Figure 20 A Dense Neural Network (Image created in: NN-SVG)

3.8 Tools Used

To test the proposed model's performance, Python has been used to create the project in a Google Colab setting. CV2, Matplotlib, Numpy, Pandas, tqdm, Sklearn, and Tensorflow are a few of the libraries. A few Javascript libraries have also been utilized to enable live video input streaming. However, with slight modification, the project can run in VS Code Pycharm and other similar IDEs.

CHAPTER 4: Results, Analysis & Evaluation

This chapter discusses the results produced by the research at different stages of its life cycle, starting from video input to the identification of facial expressions through the proposed research.

4.1 Data Acquisition

This stage starts with acquiring selected frames from the video input stream. The faces are detected with the help of the MTCNN model. For face detection, MTCNN employs multiple CNNs, but with fewer filters in each layer. The model uses filters of size 3 x 3 for more basic computation. A sample output from the face recognition module is shown in figure 21 for the MTCNN model:

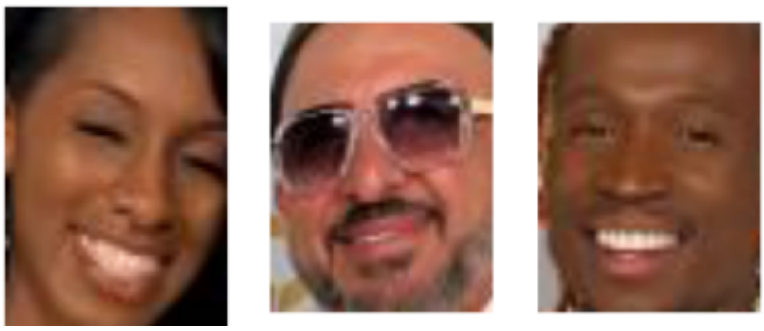


Figure 21: Output of the MTCNN Model for figure 15

[{'box': [62, 55, 43, 62], 'confidence': 0.9999982118606567, 'keypoints': {'left_eye': (80, 78), 'right_eye': (99, 85), 'nose': (90, 94), 'mouth_left': (74, 96), 'mouth_right': (94, 102)}}, {'box': [254, 43, 53, 69], 'confidence': 0.9996803998947144, 'keypoints': {'left_eye': (266, 70), 'right_eye': (290, 66), 'nose': (278, 82), 'mouth_left': (270, 95), 'mouth_right': (292, 91)}}, {'box': [148, 85, 46, 60], 'confidence': 0.9992173910140991, 'keypoints': {'left_eye': (159, 108), 'right_eye': (181, 108), 'nose': (165, 118), 'mouth_left': (157, 128), 'mouth_right': (179, 129)}}]

This output shows that there are coordinates for 3 boxes along with the keypoints like eyes, nose and mouth. One box corresponds to one image, which means tree faces were part of the original input to the MTCNN model. The entire process is shown in figure 22:

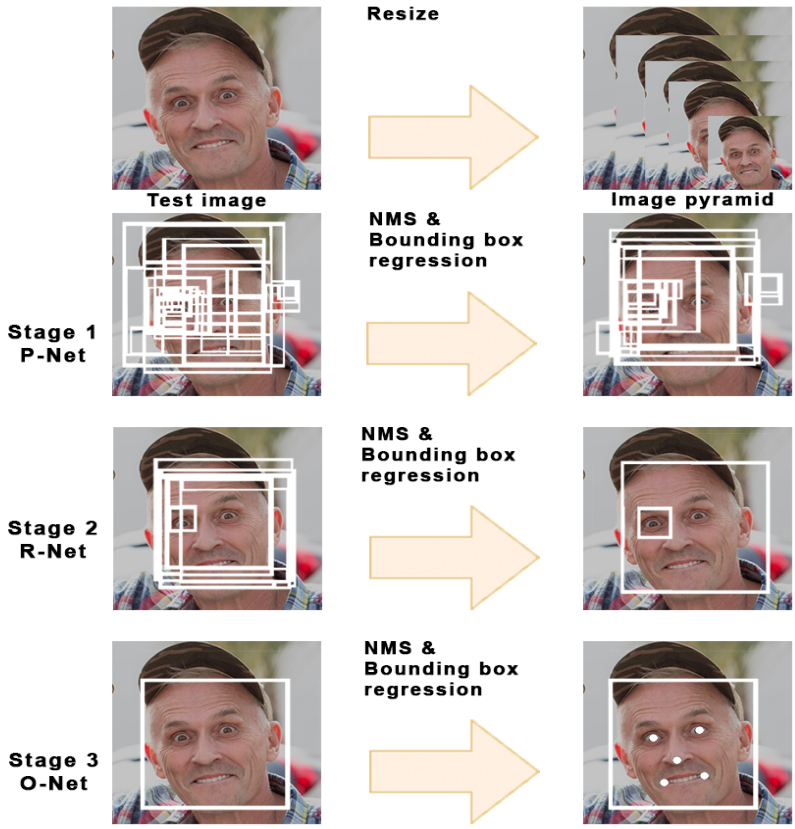


Figure 22 The Working of MTCNN

Figure 22 shows that MTCNN goes through phases of resizing the image, P-Net stage 1, R-Net stage 2, and finally, O-Net stage 3, which generates the final output of the faces detected and their key points. The output of the MTCNN network focuses on features like left eye, right eye, mouth left, mouth right, and nose coordinates.

4.2 Facial Recognition Using Shallow Networks

Different models were applied, starting with shallow neural networks. As shown in Appendix C, a shallow network was tested for accuracy before applying fusion with HOG features. The model stopped after 18 epochs due to EarlyStopping. The model gives a low accuracy as shown below in table 1:

Table 1: Accuracy of the Shallow Neural Network

No of Epochs	Training Accuracy	Validation Accuracy

18	19.65%	27.16%

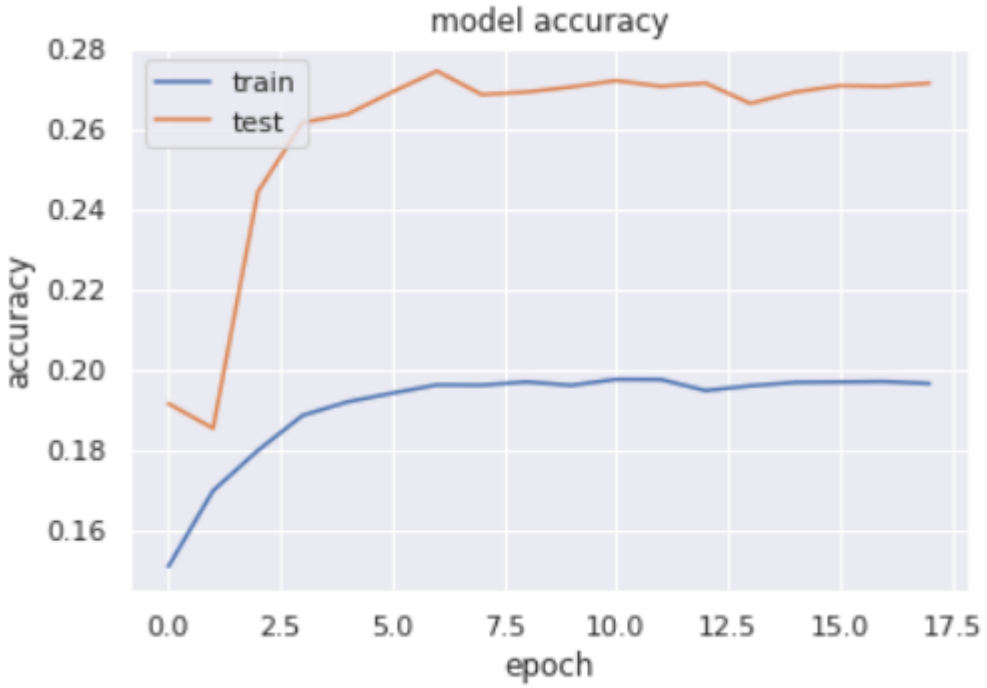


Figure 23 Training and Test Accuracy for Shallow Networks

Figure 23 shows the graph of training and testing accuracy against epochs. The figure shows that the model gives very low accuracy, which can not be helpful for any application. So the solution based on shallow networks was dropped and was not selected for final model implementation.

4.3 Facial Recognition Using Deep Learning Networks

Different models and approaches based on CNN have been applied to test the performance of the original Fer2013 dataset. The models tested were deep, as well as shallow networks. Based on their performance, the CNN with the layers and hyperparameters shown in Appendix A was used for the final performance measurement, The system was trained with different numbers of samples and hyperparameters.

At 100 epochs, the systems give reasonable accuracy, as shown below in table 2:

Table 2: Accuracy of the FER2013 based Deep Neural Network

No of Epochs	Training Accuracy	Validation Accuracy
50	92.50%	80.60%
87	98.67%	83.17%

Going beyond a certain accuracy, even adding more epochs results in accuracy oscillation in a specific range. That is the reason, after several evaluations, EarlyStopping has been applied, which stops further training. It allows you to choose a lot of training epochs at random and to terminate training whenever the model's performance on a hold-out validation dataset stops increasing. In the selected model, the training stopped at the 87th epoch.

Figure 24 shows the model training and validation/test accuracy, which are far better than the other state of art models used for this purpose.

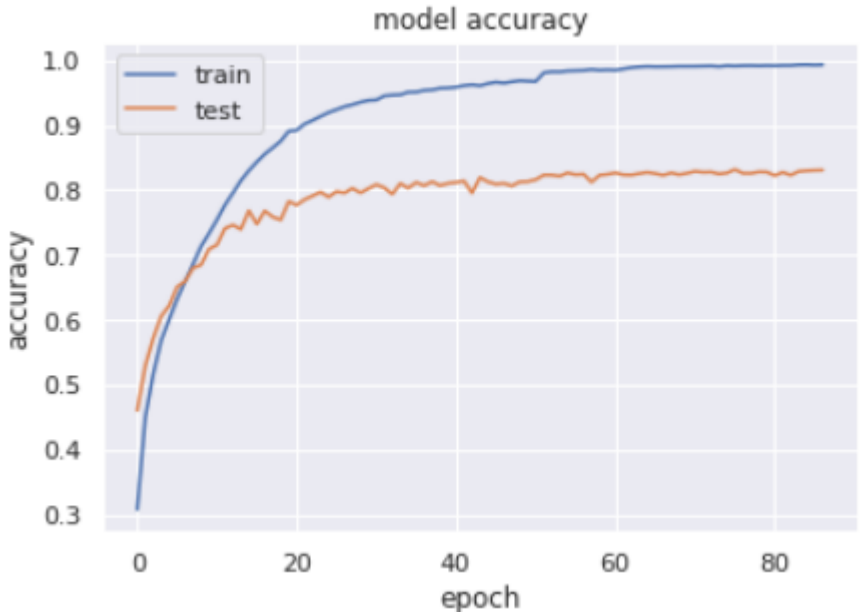


Figure 24 Model Accuracy for FER2013

Figure 25 shows training and testing loss, a testing loss of 2.0% shows a very well-designed model.

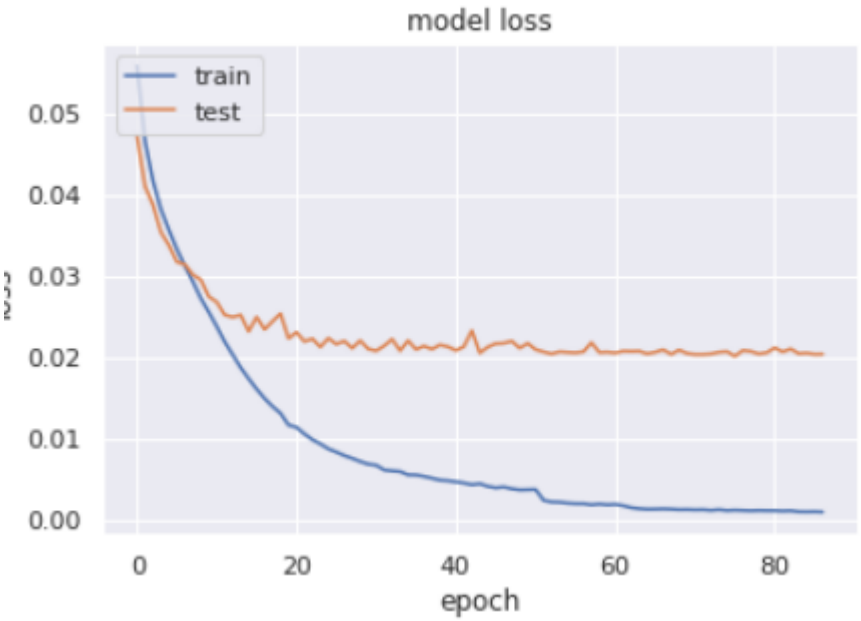


Figure 25 Validation Loss Function for FER2013 dataset

The following table shows the Precision, Recall, and F1-Score for the basic FER2013 CSV dataset showing the Evaluation Metrics for the Proposed CNN. Table 3 gives an account of precision, recall, and f1-score by the 7 classes of facial expression recognition.

Table 3: Performance Evaluation of the Deep Neural Network for FER2013

	precision	recall	f1-score	support
0	0.84	0.81	0.82	1770
1	0.99	1.00	1.00	1778
2	0.76	0.80	0.78	1775
3	0.80	0.81	0.81	1872
4	0.75	0.71	0.73	1825

5	0.93	0.94	0.93	1755
6	0.76	0.76	0.76	1810
accuracy			0.83	12585
macro avg	0.83	0.83	0.83	12585
weighted avg	0.83	0.83	0.83	12585

The confusion matrix for the 7 basic expressions are shown in figure 26 below, the diagonal entries show the correctly identified images:

Figure 26 Confusion Matrix for FER2013 dataset

As shown in the confusion matrix, the class with label 1 gives the best accuracy while class 4 gives the lowest accuracy.

4.4 HandCrafted Features Merged With Original Dataset

Different models and approaches based on CNN have been applied to the dataset created by applying handcrafted features to the images and adding the resultant images to the corresponding dataset images. The models tested were deep as well as shallow networks. Comparing their performance, the CNN with the same layers and hyperparameters was used as has been used for the basic fer2013 CSV dataset. Table 4 shares the accuracy of the model after training:

Table 4: Accuracy of the HOGFER based Deep Neural Network

No Of Epochs	Training Accuracy	Validation Accuracy
50	96.83%	81.75%
78	98.80%	83.46%

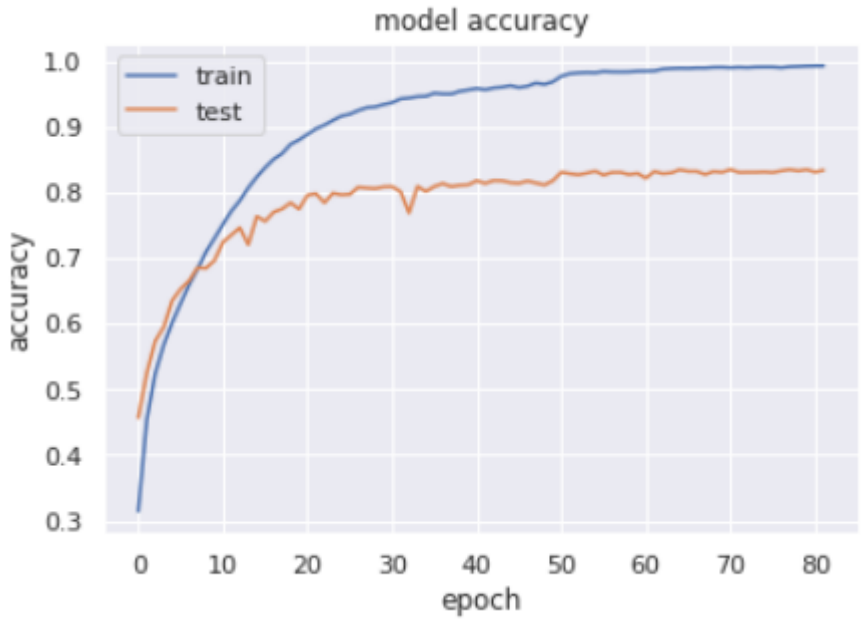


Figure27 Validation Accuracy for HOGFER Dataset

Figure 28 shows a testing/validation loss of 2.0% for the HOGFER fusion dataset. There is a training error of 0.1% on the above mentioned dataset.

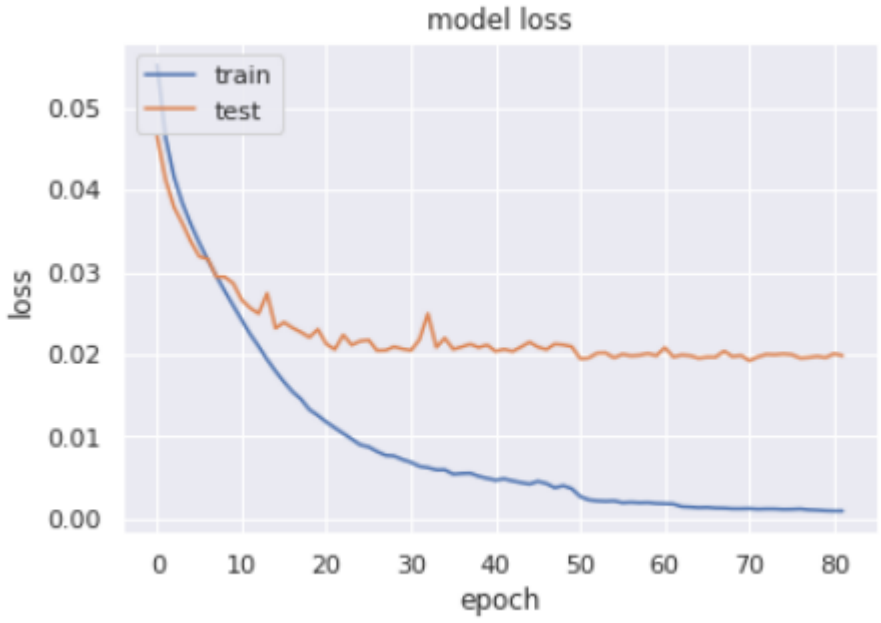


Figure 28 Validation Loss Function for HOGFER dataset

Table 5 shows the results of precision, recall, and F1-score for the HOGFER dataset based model.

Table 5: Performance Evaluation of the Deep Neural Network for HOGFER

	precision	recall	f1-score	support
0	0.80	0.82	0.81	1770
1	0.99	1.00	1.00	1778
2	0.82	0.79	0.80	1775
3	0.82	0.80	0.81	1872
4	0.74	0.74	0.74	1825
5	0.90	0.95	0.93	1755
6	0.76	0.76	0.76	1810

accuracy			0.83	12585
macro avg	0.83	0.84	0.83	12585
weighted avg	0.83	0.83	0.83	12585

The figure 27 and the figure 28 show a plot of accuracy vs. epochs and validation error vs. epochs. The model has given a very promising accuracy. At the same time, the model generates a very low validation error. These results can be further improved with the help of hyper-tuning. The following is the confusion matrix for the HOG-based handcrafted features fused with original Fer2013 dataset images elaborated in figure 29.

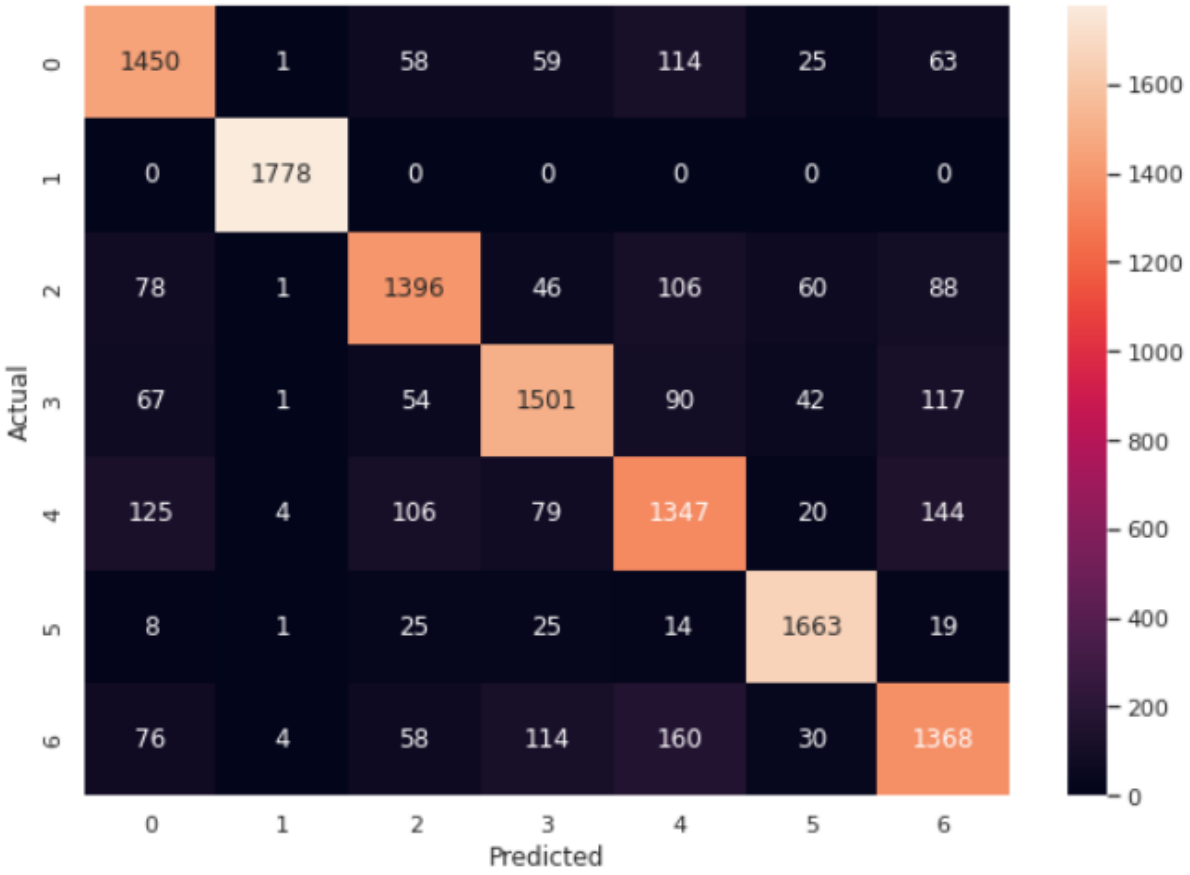


Figure 29 Confusion Matrix for the 7 Classes of HOGFER

The confusion matrix shows that emotions:” disgust” and “surprise” have been best classified as these class samples were originally lower in number. Oversampling added more samples to improve accuracy in these classes. The class “sadness” gives the lowest accuracy.

CHAPTER 5: Discussion

In this chapter, we give a parametric evaluation of the suggested model and discuss how different parameter choices affect its effectiveness. We further compare our proposed strategy to the research carried out by others to show the effectiveness of key elements of our proposed strategy.

5.1 Experimental Setup

There are three major stages in this research.

5.1.1 Processing live stream and selecting frames for further processing

The input to the model is in the form of live stream video. This live-stream video processing has been done with the help of the CV2 library in Python. The video must be converted to images so that it can be processed. Sometimes there is a jitter in the video, and conversion in frames gets slower. This is usually due to network bandwidth limitations. To resolve this issue, one solution to it is to limit the buffer size in the cv2.videoCapture() object. This will lower the elements in the buffer, giving us the latest frames. Processing all frames will lead to a very high computational cost.

Figure 12 has been generated from a 2.3 second long video. It shows two basic emotions labeled 0 and labeled 3. Label 3 represents happiness, while label 0 represents the emotion of anger. Almost 30 frames are generated in one second. On average, in a 2.3 second long video, in one second, 2 emotions can be easily expressed by the generated frames.

5.1.2 Face Detection

MTCNN algorithm has been used for face detection. This research showed promising face detection results with this model's help. The MTCNN model is fast and accurate. It covers face rotation to some extent. DR-GAN were tested for face rotation; however, as shown in chapter 3, sometimes they fail to retain the original expression. For this reason, rotation with DR-GAN has been left as future work for better face rotation.

Although GAN can add more training data by creating augmented images as is the requirement of deep neural networks, however, GAN occasionally produces a picture that is impossible and unnatural. Although the present scope of the work is limited to 2-D images only, GAN models are efficient is in handling 1-D images. Due to these factors, this research does not recommend GAN as full replacement of preprocessing phase.

5.1.3 Facial expression detection

Different models have been tested to detect facial expressions.

- Starting from the images based on FER2013, The model gave an accuracy in the range of 65% to 70%. After several repetitions of the experiment, with different hyperparameters, this research was stopped, and a different approach was adopted.
- Research based on the FER2013 CSV dataset was carried out, and it gave a very promising performance.

These methods have been discussed in detail in the next section.

5.2 Model Evaluation using the Database FER2013 CSV format

This variation of FER2013 differs from the images-based FER2013 in that it uses a CSV format using three columns. The first column represents the label of the class, the next column contains an image converted to a string, and the last column shows how this image will be used (for training, testing, etc.).

The data is divided into an 80:20 ratio, 80% of images are used for training, while 20% are used for testing. RandomOverSampler is used for data augmentation raising each class strength to approximately 8989.

The loss function initially used was categorical_crossentropy. It gave a reasonable accuracy of 84% approximately; however, it gave a very high validation error tending towards overfitting. The loss function was changed to MeanSquaredLogarithmicError

This gives a loss of 2.0% and an accuracy of 83.17% approximately. The classes which were initially skewed, have now outperformed other classes in the dataset.

5.3 Model Evaluation using the HandCrafted Features Merged With Original Dataset

The research used HOG features to test if they impacted the performance of the proposed model. For this, an early fusion of the images with their HOG was combined to create a new dataset called HOGFER with the same number of images. The experiment carried out had the following details.

The data was divided into 80%:20%, and the epochs used are 100, although the model stopped after 82 epochs due to EarlyStopping.

The loss function initially used was categorical_crossentropy. It gave approximately a reasonable accuracy of up to 85%; however, it gave a very high validation error tending towards overfitting. The loss function was later changed to MeanSquaredLogarithmicError.

This gives a loss of 2.0% and an improved accuracy of 83.46% approximately. The classes, which were initially skew classes, have again outperformed other classes in the dataset, as in the case of the base FER2013 dataset. The average precision, recall, and F1-score values are also 83%,84%, and 83%, respectively.

The above results show that handcrafted features merged with original data can improve model performance by at least 0.5%.

5.4 Comparison with Existing Techniques.

The research has been compared with other no-constraint techniques as shown in table 6 below:

Table 6: Comparison with Existing Techniques

Rank	Model	Accuracy(%)	Paper Title

1	**Ensemble ResMaskingNet with 6 other CNNs**	76.82	Challenges in Representation Learning: A report on three machine learning contests[54]
2	**Local Learning Deep+BOW**	75.42	Local Learning with Deep and Handcrafted Features for Facial Expression Recognition[53]
3	**LHC-Net**	74.42	Local Multi-Head Channel Self-Attention for Facial Expression Recognition[55]
4	**VGGNet**	73.28	Facial Emotion Recognition: State of the Art Performance on FER2013[56]
5	**CNN Hyperparameter Optimisation**	72.16	Convolutional Neural Network Hyperparameters optimization for Facial Emotion Recognition[57]

The results from above researches show that the proposed solution outperforms all the solutions as mentioned earlier. The proposed solution performs well mainly due to the CNN model structure and the type of data augmentation used. The facial expression recognition rate increases to 83.46% on average (up to 84.9%) for the HOGFER dataset compared to 83.17% on the FER2013 dataset, respectively, as validation accuracy. The CNN model structure gives an average accuracy of 84.5% when the loss function is categorical_crossentropy. However, it gives a very high validation error. To reduce validation error, two loss functions, mean_squared_error,

and MeanSquaredLogarithmicError, have been applied both have given reasonable error rates; however, the results by the MeanSquaredLogarithmicError are even better.

The results of the research prove that the accuracy of FER can be substantially improved by merging certain efficient handcrafted features with the deep learning model. There are many handcrafted features, but for this research, only HOG was considered. An early fusion of the feature with the image has shown an average increase of 0.5%.

.

CHAPTER 6: Conclusion

This research focuses on recognizing facial expressions in a constraint-free environment. A successful solution would benefit various industries and stakeholders involved in monitoring the emotional well-being of individuals. It would need to be an efficient and cost-effective solution due to the availability of imaging devices. Moreover, the solution may benefit governments and decision-makers in developing and/or refining policies that ensure public health in all dimensions.

The existing solutions suffer from the following limitations:

- The existing solutions suffer from low accuracy. The metrics of performance give poor results for most of these solutions.
- The existing models work on fewer face expression classes, limiting their scope of adaptability as a real-life solution.
- The number of samples in some categories is far below compared to other categories, creating bias in the proposed model.
- The videos/images may need pre-processing before they can be used for training.
- These solutions are affected by occlusion and face rotation. The use of MTCNN handles face rotation to some extent.
- The majority of current research uses datasets of facial expressions that have been carefully created in predetermined circumstances or are completely frontal faces; therefore, facial expression recognition on such datasets becomes a lot simpler than for real-time facial expression datasets.

6.1 Conclusion

The proposed model performs well for most classes, as few classes were skewed; data augmentation has compensated for the fewer samples in those classes. The model's performance can be improved by using denoising and sharpening filters.

The evaluation of the results shows that there is room for improving performance. To utilize FER effectively, there is a need to improve accuracy. This can be achieved with improved model design, better hyperparameter tuning and improved data

augmentation. There is a delay caused due to the conversion of a video into frames. There is no algorithm for calculating the frame rate of video in this research. If we can form a mechanism for selecting frames, we might be able to recognize facial expressions more accurately.

The model can be improved so that it detects facial expression in a complex scenario as well. In a scene, where there is occlusion, rotation, scaling, illumination etc., the proposed FER solutions can be improved sothat the proposed model can perform well..

The proposed research has tried to improve accuracy by using a fusion of handcrafted features HOG with the basic FER2013 dataset. The dataset is analyzed, and HOG is calculated for each sample; both the original sample and the output of the handcrafted feature HOG are combined and saved for model training in the form of the HOGFER dataset.

- The proposed model's structure exhibits good generality and classification performance. To cut down on information, the facial region in the input stream is identified, clipped, and converted into grayscale images. Augmentation increases the quantity and variety of training pictures and addresses fewer samples, especially in class. The use of hyperparameter optimization attained a state-of-the-art classification performance accuracy up to 83.46%.
- Repeated experiments showed that accuracy and other metrics are higher when the FERr2013 dataset combines handcrafted features. Accuracy increased by 0.5% or more.
- The validation loss is very low for the proposed model, it was minimized by using the dropout layer or trying different loss functions.
- Reducing the number of layers in the model leads to a reduction in accuracy. So a shallow network can not give higher accuracy or reduce validation loss.
- There is a trade-off between validation loss and accuracy in the proposed solution, but still accuracy of this model is higher than other presented solutions.

6.2 Future Work

The recognition of human facial expressions helps in many applications. In many real-life scenarios, automated facial expression recognition may help determine the

human's emotional state. There is a need to use models like Faster R-CNN, HOG + Linear SVM, and YOLO. The following areas need attention for improvement in the domain.

- The proposed research supports the fusion of handcrafted features at multiple stages. The fusion process can also be done during model training or at another stage. The efficacy of these fusion methods at a particular stage of model creation can be the focus of research in the future.
- To compensate for a higher validation loss during the initial stage of the research, the skewed classes are augmented with the help of Random Oversampling.
- The use of Random Oversampling as a data augmentation technique should be analyzed in more detail to improve the system performance further.
- In addition to data augmentation, the dataset needs improvement for the skewed classes. The dataset FER2013 can be improved by adding more samples in the skewed classes.
- The model's performance can be improved by using denoising and sharpening filters.

References:

[1] S. Karamizadeh, S. M. Abdullah & M. Zamani, . An overview of holistic face recognition. IJRCCT, 2(9), 738-741, 2013

[2] Y. Borkar, R. Mascarenhas, S. Tambadkar, & J. P. Gawande, Comparison of Real-Time Face Detection and Recognition Algorithms. In ITM Web of Conferences (Vol. 44, p. 03046). EDP Sciences, 2022.

[3] P. Xie, G. Ma, T. Feng, Y. Yan, , & X. Han, Behavioral feature and correlative detection of multiple types of node in the internet of vehicles. CMC-Comput Mater Cont, 64(2), 1127-1137, 2020.

[4] S. Singh & S. V. A. V. Prasad, Techniques and challenges of face recognition: A critical review. Procedia computer science, 143, 536-543, 2018.

[5] D. Zeng, R. Veldhuis & L. Spreeuwers, A survey of face recognition techniques under occlusion. IET biometrics, 10(6), 581-606, 2021.

[6] J. Ho & D. Kriegman, On the effect of illumination and face recognition. Face Processing: Advanced Modeling and Methods, 2005.

[7] C. K. Tran, C. D. Tseng, L. Chang & Lee, T. F. Face recognition under varying lighting conditions: improving the recognition accuracy for local descriptors based on weber-face followed by difference of Gaussians. Journal of the Chinese Institute of Engineers, 42(7), 593-601, 2019.

[8] M. Singh, & A. S. Arora, Varying illumination and pose conditions in face recognition. Procedia Computer Science, 85, 691-695, 2016

[9] Z. Hao, Y. Liu, H. Qin, J. Yan, X. Li, & X. Hu, Scale-aware face detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 6186-6195), 2017.

[10] A. Sapkota, B. Parks, W. Scheirer, & T. Boult, Face-grab: Face recognition with general region assigned to binary operator. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops (pp. 82-89), Jun, 2010.

[11] P. Hu, & D. Ramanan, Finding tiny faces. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 951-959), 2017.

[12] A. Mehrabian, “Framework for a comprehensive description and measurement of emotional states.,” Genet. Soc. Gen. Psychol. Monogr., vol. 121, no. 1978, pp. 339–361, 1995.

[13] E. P. Scilingo, “The role of nonlinear dynamics in affective valence and arousal recognition,” IEEE Trans. Affect. Comput., vol. 3, no. 2, pp. 237–249, 2012.

[14] L. F. Barrett, “Discrete Emotions or Dimensions? The Role of Valence Focus and Arousal Focus,” Cogn. Emot., vol. 12, no. 4, pp. 579–599, 1998.

[15] D. Canedo, & A. J. Neves, Facial expression recognition using computer vision: A systematic review. Applied Sciences, 9(21), 4678, 2019.

[16] S. Zafeiriou, C. Zhang, & Z. Zhang, A survey on face detection in the wild: past, present and future. Computer Vision and Image Understanding, 138, 1-24, 2015

[17] M. K. Hasan, , M. S. Ahsan, S. S. Newaz, & G. M. Lee, Human face detection techniques: A comprehensive review and future research directions. Electronics, 10(19), 2354, 2021.

[18] S. Paul, & S. K. Acharya, A Comparative Study on Facial Recognition Algorithms. In e-journal-First Pan IIT International Management Conference–2018, Dec. 2020.

[19] P. Ekman & W. V. Friesen, Constants across cultures in the face and emotion. Journal of personality and social psychology, 17(2), 124, 1971.

[20] H. H. Wang, & J. W. Gu, The applications of facial expression recognition in human-computer interaction. In 2018 IEEE international conference on advanced manufacturing (ICAM) (pp. 288-291), Nov., 2018

[21] R Stathacopoulou, M Grigoriadou, G D Magoulas, et al. A neuro-fuzzy approach in student modeling[C] International Conference on User Modeling. Springer-Verlag:337-341, 2003

[22] Google, FER2013, Kaggle: 2013. [Online]. Available: https://www.kaggle.com/datasets/msambare/fer2013.

[23] L. Zahara, P. Musa, E. P. Wibowo, I. Karim,, & S. B. Musa, The facial emotion recognition (FER-2013) dataset for prediction system of micro-expressions face using the convolutional neural network (CNN) algorithm based Raspberry Pi. In 2020 Fifth international conference on informatics and computing (ICIC) (pp. 1-9). Nov. 2020.

[24] A, Mehrabian, M. Wiener Decoding of Inconsistent Communications. Journal of Personality and Social Psychology. 6 (1), 109–114. Mehrabian, A., Ferris, S.R. (1967), Inference of Attitudes from Nonverbal Communication in Two Channels. Journal of Consulting Psychology. 31 (3): 248–252, 1967

[25] P. Liu, S. Han, Z. Meng & Y. Tong, Facial expression recognition via a boosted deep belief network. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1805-1812), 2014.

[26] T. Zhang, W. Zheng, Z. Cui, Y. Zong & Y. Li, Spatial–temporal recurrent neural network for emotion recognition. IEEE transactions on cybernetics, 49(3), 839-847, 2018.

[27] Y. H. Lai & S. H. Lai, Emotion-preserving representation learning via generative adversarial network for multi-view facial expression recognition. In 2018, 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018) (pp. 263-270). May, 2018.

[28] I. Gogić, M. Manhart, I. S. Pandžić, , & J. Ahlberg, Fast facial expression recognition using local binary features and shallow neural networks. The visual computer, 36(1), 97-112, 2020.

[29] N. Christou & N. Kanojiya, Human facial expression recognition with convolution neural networks. In Third International Congress on Information and Communication Technology, Springer, Singapore (pp. 539-545). 2019.

[30] G. V. Reddy, C. D. Savarni, & S. Mukherjee, Facial expression recognition in the wild, by fusion of deep learnt and hand-crafted features. Cognitive Systems Research, 62, 23-34, 2020.

[31] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, , & I. Matthews, The extended cohn-kanade dataset (ck+): A complete dataset for action unit and

emotion-specified expression. In 2010 ieee computer society conference on computer vision and pattern recognition-workshops (pp. 94-101). Jun. 2010.

[32] M. H. Yang, D. J. Kriegman, & N. Ahuja, Detecting faces in images: A survey. IEEE Transactions on pattern analysis and machine intelligence, 24(1), 34-58, 2002

[33] A.Lanitis, C. J. Taylor & T. F. Cootes, Automatic face identification system using flexible appearance models. Image and vision computing, 13(5), 393-401, 1995.

[34] M. I. Georgescu, R. T. Ionescu, & M. Popescu, Local learning with deep and handcrafted features for facial expression recognition. IEEE Access, 7, 64827-64836, 2019.

[35] I. J. Goodfellow, D. Erhan, P. L.Carrier, A.Courville, M. Mirza, B.Hamner & Y. Bengio, Challenges in representation learning: A report on three machine learning contests. In International conference on neural information processing, Springer, Berlin, Heidelberg (pp. 117-124), Nov. 2013.

[36] C. Shan, S. Gong, P.W. McOwan, Facial expression recognition based on local binary patterns: A comprehensive study. Image Vis. Comput. 27, 803–816, 2009.

[37] Y. Shi, Z. Lv, N. Bi & C. Zhang, An improved SIFT algorithm for robust emotion recognition under various face poses and illuminations. Neural Computing and Applications, 32(13), 9267-9281, 2020.

[38] D. Navneet, T. Bill, Histograms of oriented gradients for human detection. In Proceedings of the IEEE 2005 Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), San Diego, CA, USA, 20–26 Volume 2, pp. 886–893, Jun. 2005.

[39] G. Lokku, G. H. Reddy & Prasad, M. G. Optimized scale-invariant feature transform with local tri-directional patterns for facial expression recognition with deep learning model. The Computer Journal, 65(9), 2506-2527, 2022.

[40] m Kalsum, T. Anwar, S. M., Majid, M. Khan, B., & S. M. Ali, Emotion recognition from facial expressions using hybrid feature descriptors. IET Image Processing, 12(6), 1004-1012, 2018.

[41] S.Hosseini, S. H. Lee & N. I. Cho, Feeding hand-crafted features for enhancing the performance of convolutional neural networks. arXiv preprint arXiv:1801.07848, 2018.

[42] L.Tran, X. Yin & X. Liu, Disentangled representation learning gan for pose-invariant face recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1415-1424), 2017

[43] A. B. Shetty & J. Rebeiro, Facial recognition using Haar cascade and LBP classifiers. Global Transitions Proceedings, 2(2), 330-335, 2021

[44] P. Viola & M. Jones, Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001 (Vol. 1, pp. I-I). Dec. 2001.

[45] S. Stefanou, & A. A. Argyros, Efficient scale and rotation invariant object detection based on HOGs and evolutionary optimization techniques. In International Symposium on Visual Computing (pp. 220-229). Springer, Berlin, Heidelberg, Jul. 2012.

[46] M.Binkhonain, & L. Zhao, A review of machine learning algorithms for identification and classification of non-functional requirements. Expert Systems with Applications: X, 1, 100001, 2019

[47] M. F. H. Siddiqui, P. Dhakal, X. Yang, & A. Y. Javaid, A Survey on Databases for Multimodal Emotion Recognition and an Introduction to the VIRI (Visible and InfraRed Image) Database. Multimodal Technologies and Interaction, 6(6), 47, 2022

[48] A.J. Bingham & Witkowsky, Deductive and inductive approaches to qualitative data analysis. In C. Vanover, P. Mihas, & J. Saldaña (Eds.), *Analyzing and interpreting qualitative data: After the interview* (pp. 133-146), 2022, SAGE Publications.

[49] M. Khaliluzzaman, S. Pervin, M. R. Islam, & M. M. Hassan, Automatic Facial Expression Recognition using Shallow Convolutional Neural Network. In 2019 IEEE International Conference on Robotics, Automation, Artificial-intelligence and Internet-of-Things (RAAICON) (pp. 98-103), 2019.

[50] M. Hayaty, S. Muthmainah, & S. M. Ghufran, Random and synthetic over-sampling approach to resolve data imbalance in classification. International Journal of Artificial Intelligence Research, 4(2), 86-94, 2020.

[51] F. Nazari, & W. Yan, Convolutional versus Dense Neural Networks: Comparing the Two Neural Networks Performance in Predicting Building Operational Energy Use Based on the Building Shape. arXiv preprint arXiv:2108.12929, 2021.

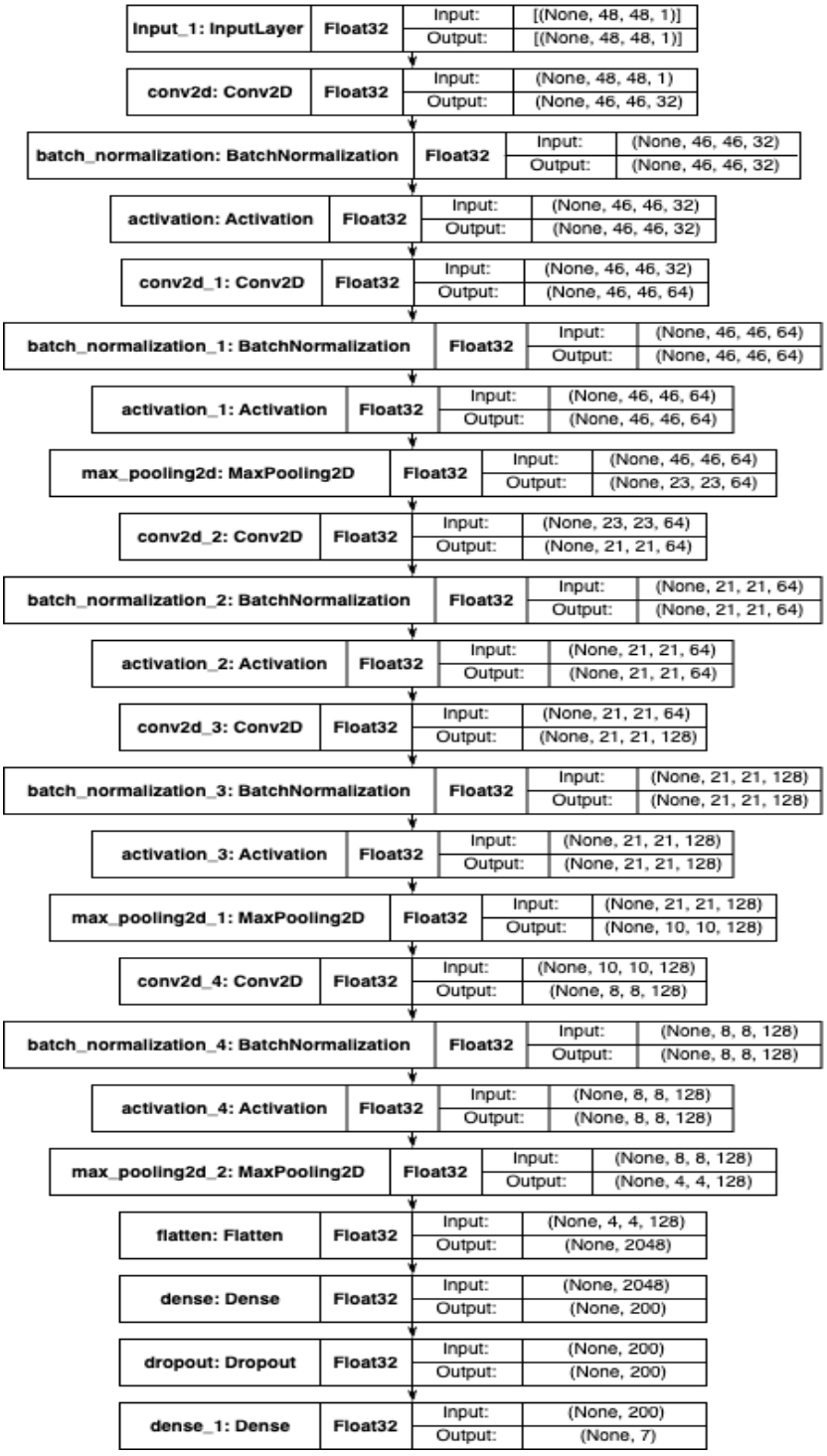
[52] J. Haberman, T. Harp, & D. Whitney, Averaging facial expression over time. Journal of vision, 9(11), 1-1, 2009.

[53] R. Pecoraro, V. Basile, & V. Bono, Local multi-head channel self-attention for facial expression recognition. Information, 13(9), 419, 2022

[54] Y. Khaireddin, & Z. Chen, Facial emotion recognition: State of the art performance on FER2013. arXiv preprint arXiv:2105.03588, 2021

[55] Vulpe-Grigoraşi, A., & Grigore, O. Convolutional neural network hyperparameters optimization for facial emotion recognition. In 2021, 12th International Symposium on Advanced Topics in Electrical Engineering (ATEE) (pp. 1-5), Mar. 2021.

Appendix A: Deep Neural Network Model



Overall Structure of the Proposed Model

Appendix B: Facial Recognition Using Deep Learning Networks

Model: "sequential"

_____

Layer (type) Output Shape Param #

==========

conv2d (Conv2D) (None, 46, 46, 32) 320

batch_normalization (BatchNo (None, 46, 46, 32) 128

activation (Activation) (None, 46, 46, 32) 0

conv2d_1 (Conv2D) (None, 46, 46, 64) 18496

batch_normalization_1 (Batch (None, 46, 46, 64) 256

activation_1 (Activation) (None, 46, 46, 64) 0

max_pooling2d (MaxPooling2D) (None, 23, 23, 64) 0

conv2d_2 (Conv2D) (None, 21, 21, 64) 36928

_____

batch_normalization_2 (Batch (None, 21, 21, 64) 256

activation_2 (Activation) (None, 21, 21, 64) 0

conv2d_3 (Conv2D) (None, 21, 21, 128) 73856

batch_normalization_3 (Batch (None, 21, 21, 128) 512

activation_3 (Activation) (None, 21, 21, 128) 0

max_pooling2d_1 (MaxPooling2 (None, 10, 10, 128) 0

conv2d_4 (Conv2D) (None, 8, 8, 128) 147584

batch_normalization_4 (Batch (None, 8, 8, 128) 512

activation_4 (Activation) (None, 8, 8, 128) 0

max_pooling2d_2 (MaxPooling2 (None, 4, 4, 128) 0

flatten (Flatten) (None, 2048) 0

dense (Dense) (None, 200) 409800

dropout (Dropout) (None, 200) 0

dense_1 (Dense) (None, 7) 1407

===

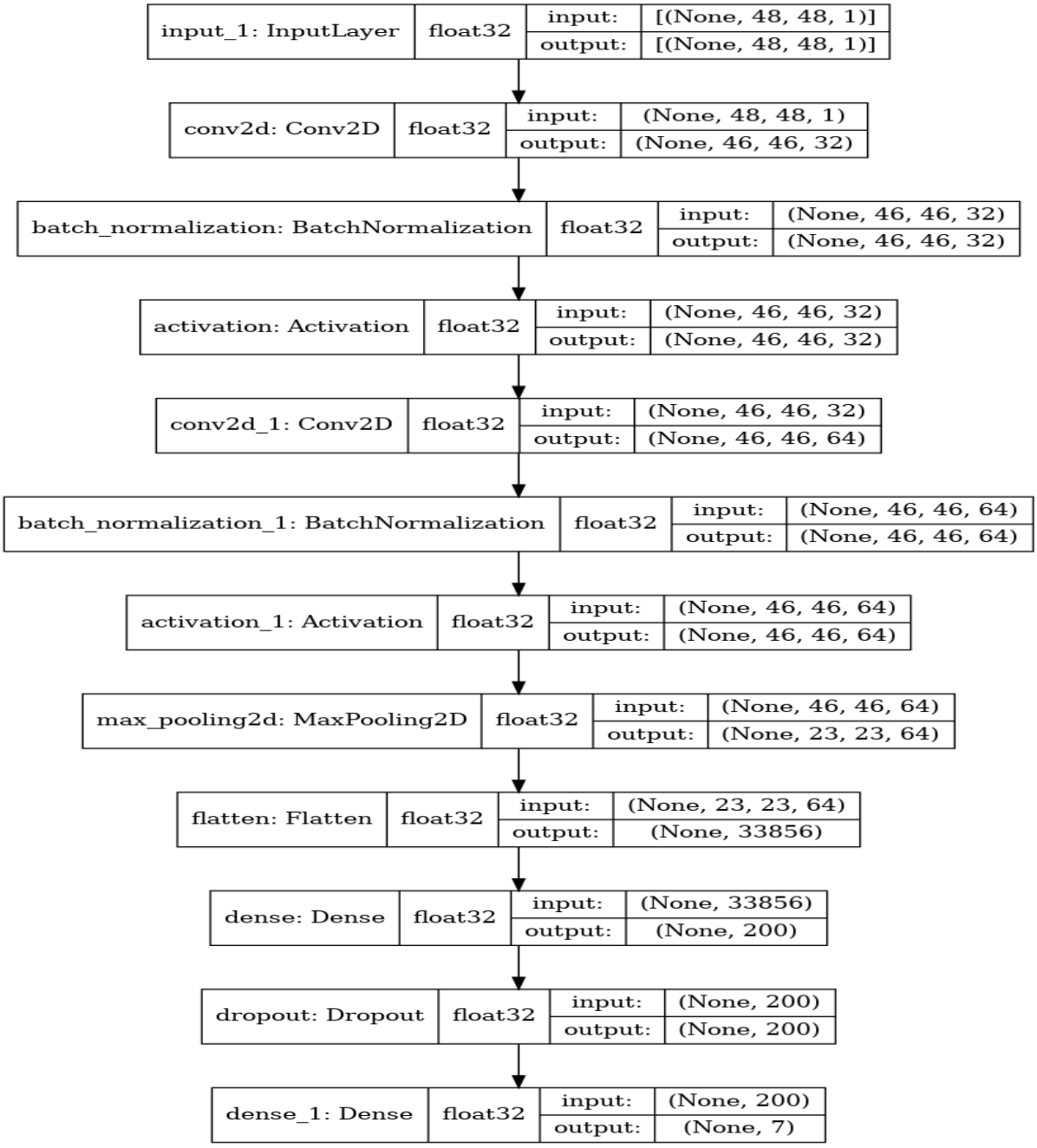
Total params: 690,055

Trainable params: 689,223

Non-trainable params: 832

_____

Appendix C: Shallow Networks Model



Layers in a Shallow Neural Network

Appendix D: Facial Recognition Model for Shallow Networks

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 46, 46, 32)	320
batch_normalization	(None, 46, 46, 32)	128
activation (Activation)	(None, 46, 46, 32)	0
conv2d_1 (Conv2D)	(None, 46, 46, 64)	18496
batch_normalization_1	(Batch (None, 46, 46, 64)	256
activation_1 (Activation)	(None, 46, 46, 64)	0
max_pooling2d	(None, 23, 23, 64)	0
flatten (Flatten)	(None, 33856)	0
dense (Dense)	(None, 200)	6771400
dropout (Dropout)	(None, 200)	0
dense_1 (Dense)	(None, 7)	1407

Total params: 6,792,007

Trainable params: 6,791,815

Non-trainable params: 192

The model stops training after 18 epochs due to EarlyStopping.

Appendix E: HOGFER +FER2013 Fusion Based Model

Model: "sequential"

_____

Layer (type) Output Shape Param #

==========

conv2d (Conv2D) (None, 46, 46, 32) 320

batch_normalization (BatchNo (None, 46, 46, 32) 128

activation (Activation) (None, 46, 46, 32) 0

conv2d_1 (Conv2D) (None, 46, 46, 64) 18496

batch_normalization_1 (Batch (None, 46, 46, 64) 256

activation_1 (Activation) (None, 46, 46, 64) 0

max_pooling2d (MaxPooling2D) (None, 23, 23, 64) 0

conv2d_2 (Conv2D) (None, 21, 21, 64) 36928

batch_normalization_2 (Batch (None, 21, 21, 64) 256

activation_2 (Activation) (None, 21, 21, 64) 0

conv2d_3 (Conv2D) (None, 21, 21, 128) 73856

batch_normalization_3 (Batch (None, 21, 21, 128) 512

activation_3 (Activation) (None, 21, 21, 128) 0

max_pooling2d_1 (MaxPooling2 (None, 10, 10, 128) 0

conv2d_4 (Conv2D) (None, 8, 8, 128) 147584

batch_normalization_4 (Batch (None, 8, 8, 128) 512

activation_4 (Activation) (None, 8, 8, 128) 0

max_pooling2d_2 (MaxPooling2 (None, 4, 4, 128) 0

flatten (Flatten) (None, 2048) 0

dense (Dense) **(None, 200)** **409800**

dropout (Dropout) **(None, 200)** **0**

dense_1 (Dense) **(None, 7)** **1407**

==========

Total params: 690,055

Trainable params: 689,223

Non-trainable params: 832

_____

A sample output of the model is as under, first output of the 10 emotions is the actual emotion while the second list shows predicted emotions. It is obvious from the output that out of 10, only one emotion has been classified incorrectly, remaining 9 has been predicted correctly.

```
##emotion_labels = ['0:angry','1:disgust','2:fear','3:Happy', '4:Sad','5:Surprise', '6:Neutral']
y_pred = model.predict(x_test)
y_result = []

for pred in y_pred:
    y_result.append(np.argmax(pred))
y_result[:10]
```

```
[6, 5, 5, 6, 1, 0, 3, 4, 1, 6]
```

```
[ ] y_actual = []

    for pred in y_test:
        y_actual.append(np.argmax(pred))
    y_actual[:10]
```

```
[6, 5, 5, 6, 1, 0, 3, 4, 1, 3]
```

Appendix F: Performance Metrics

Like speech recognition, facial recognition, and text categorization, classification is one of the most researched problems in many walks of life worldwide. The outcome of a classification algorithm can be divided into the following four categories:

True Positives (TP): These demonstrate that the model's classification of the input facial expression and the actual facial expression, which was found to belong to one class, both matched the class identified by the model

True Negative (TN): Indicates that neither the facial expression that was output by the model nor the actual expression belonged to the category. So both agreed.

False Positive (FP): When the proposed system assigns the facial expression to a class when it doesn't belong there.

False Negative (FN): The model claims that an expression does not belong to one class, but it does.

All the performance metrics evaluate the model’s performance differently. A few commonly used metrics are as below:

- Accuracy
- Confusion Matrix (can help in determining other metrics, although this is not itself a metric),
- Precision
- Recall
- F1-score

Some metrics compare discrete classes in some way because classification models produce discrete output. These metrics assess a model's performance to evaluate the performance of classification.

A tabular representation of the ground-truth labels and model predictions is called a confusion matrix. The model output/ predicted class is represented in each row of the confusion matrix, and the instances in actual classes are represented in each column(or vice versa) as shown below:

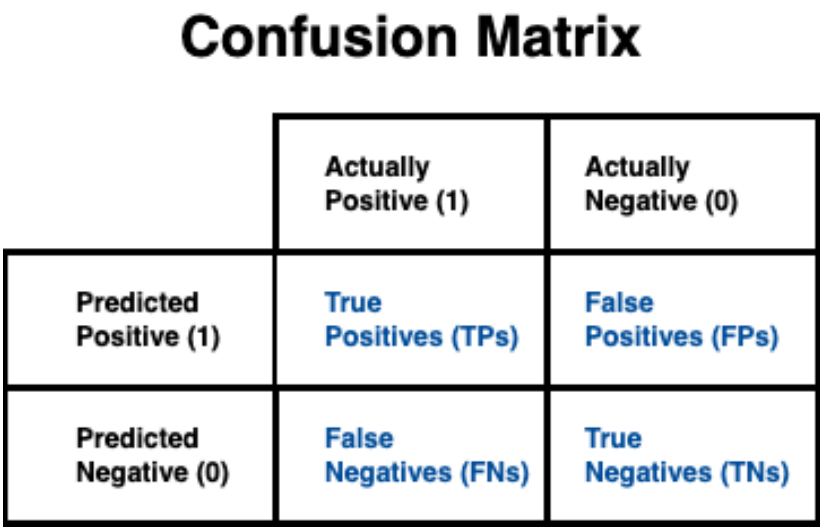


Figure 0-4 A Binary Confusion Matrix

Each of these has different significance for different problems. Precision (Pre), sensitivity (Sens), accuracy (Acc), specificity (Spec), and F1 score are all calculated using the confusion matrix data.

Accuracy reveals how frequently the ML model was overall correct. It measures total samples predicted correctly, whether positive or negative. The model's precision measures how well it can forecast a particular category. It measures all correct outcomes to a number of all positive outcomes by the model. How frequently the model identified a particular category (whether a positive or negative outcome) is indicated by the recall. These measurements have been defined below:

$$Accuracy = (TP + TN)/(TN + TP + FN + FP)$$

$$Recall = TP/(TP + FN)$$

$$Specificity = TN/(TN + FP)$$

$$Precision = TP/(TP + FP)$$

$$F1 - Score = (2 \times Precision \times Recall)/(Precision + Recall)$$

Appendix G: Artifacts Directory

Colab_Webcam_FaceDetection

ethics-fast-track

FaceDetection_MTCNN

Fast-Track Ethics Application

FER2013 dataset

Final_Result_hog+2013

frame465

Fusion_Dataset_Creation

hogfernov28

img1

readme

sk (Streamed Video)

SourceCode_without_Fusion

Weights_Model.h5