SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

A Mini Project report on

FACIAL EMOTION RECOGNITION USING CNN

Submitted

in partial fulfillment of the requirements for the award of the degree of

Bachelor of Engineering

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted By

Ananya Kulkarni	01fe21bcs144
Swarna Patil	01fe21bcs145
Amrutha Beedikar	01fe21bcs198
Maanasi Shastri	01fe21bcs210

Under the guidance of

Dr. Sujatha C

School of Computer Science and Engineering

KLE Technological University, Hubballi

2023-2024

School of Computer Science and Engineering

KLE Technological University
Creating Value, Leveraging Knowledge

CERTIFICATE

This is to certify that the project entitled "FACIAL EMOTION RECOGNITION USING CNN" is a bonafide work carried out by the student team Ananya Kulkarni(01FE21BCS144), Swarna Patil (01FE21BCS145), Amrutha Beedikar(01FE21BCS198), Maanasi Shastri(01FE21 BCS210), in partial fulfillment of the completion of the 5th semester, B. E. course during the year 2023 – 2024. The project report has been approved as it satisfies the academic requirement concerning the project work prescribed for the above-said course.

Guide Name	SoCSE Head
Dr.Sujatha C	Dr. Vijaylakshmi M.

External Viva-Voce

Name of the examiners	Signature with date
1 ______________________	________________
2 ______________________	________________

ABSTRACT

Facial emotion recognition (FER) technology involves analyzing facial expressions to identify and understand human emotions. It typically utilizes computer vision techniques and machine learning algorithms to detect and classify various emotions such as happiness, sadness, anger, surprise, fear, and disgust from images or video streams of human faces. The motivation to choose FER is that FER systems play a crucial role in enhancing human-computer interaction (HCI) and the application of artificial intelligence, offering immense potential for applications in fields such as healthcare, education, and human-computer interaction. Exploring FER provides an opportunity to delve into cutting-edge research, address real-world challenges, and contribute to advancements in emotion recognition technology. The project introduces a novel approach to FER by employing an Enhanced Convolutional Neural Network (CNN) architecture. The model incorporates transfer learning utilizing ResNet50, effectively addressing challenges associated with imbalanced datasets by applying oversampling techniques. A meticulous process of hyperparameter tuning is executed to optimize the model's performance. The proposed system attains a commendable accuracy rate of approximately 64.38%, surpassing the performance of existing state-of-the-art models.

Keywords : *FER: Facial emotion recognition, HCI:Human-Computer Interaction, NN: Neural Networks, CNN: Convolutional neural network, FERC: Facial emotion recognition using CNN*

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to Dr.Sujatha C, our project guide, for their invaluable support and guidance throughout this project work. Their expertise, unwavering encouragement, and insightful feedback have played a pivotal role in shaping the direction of this study.

We are equally thankful to Mr.Subodh Ingleshwar, our co-guide, for their constructive input and mentorship. Their contributions have added depth and perspective to the research, enriching the overall quality of the work. We are indebted to both Dr.Sujatha C and Mr.Subodh Ingleshwar for their continuous encouragement, patience, and mentorship. Their commitment to excellence has been a constant source of inspiration, driving me to push the boundaries of my research.

We would also like to extend our gratitude to Dr.Vijayalakshmi M, HoS, School of Computer Science and Engineering, Prof.Prashant Narayankar, our project co-ordinator and to the KLE Technological University for providing the necessary resources and a conducive environment for the pursuit of knowledge.

This research would not have been possible without the support of my guides and the academic community. We are truly grateful for the opportunity to work under their guidance and are honored to have them as mentors.

Ananya Kulkarni - 01fe21bcs144
Swarna Patil - 01fe21bcs145
Amrutha Beedikar - 01fe21bcs198
Maanasi Shastri - 01fe21bcs210

CONTENTS

LIST OF FIGURES

Chapter 1

INTRODUCTION

Facial emotion recognition (FER) is a captivating field within computer vision and artificial intelligence that focuses on deciphering human emotions based on facial expressions. This technology has far-reaching applications, from human-computer interaction to affective computing and mental health assessments. FER has evolved significantly with advancements in machine learning and deep learning techniques. Understanding human emotions is crucial in various domains, including human-computer interaction, gaming, and healthcare.

1.1 Preamble

The field of artificial intelligence has been paying significant attention to the relationship between computer vision and emotion recognition in the modern era. This work proposes the creation of a synthesizable AI model for Facial Emotion Recognition (FER), marking the beginning of a transformative journey. By utilizing the capabilities of Convolutional Neural Networks (CNN) and pre-trained models, we want to create a sophisticated system that can recognize facial expressions with accuracy and can be synthesized for broad implementation. Personalized digital marketing, mental health diagnoses, and human-machine interactions can all benefit from an understanding of emotions. Emotions are classified by facial recognition algorithms based on characteristics including lip position, eye movements, and facial muscle movements. A crucial stage in many applications, including advanced driver assistance systems, augmented and virtual reality, human-computer interface, and security systems, is the recognition of human emotions. The current study addresses the identification of human emotions in response to the increasing need for Facial Emotion Recognition (FER) in recent times.

1.2 Motivation

Our project's motivation stems from the critical role that FER plays in a variety of disciplines, which reflects the growing interest, demand, and adoption for it. FER's ability to improve human-computer interaction is one of the main driving forces. Interactions become more natural and responsive when systems can adapt dynamically to the feelings of users. This not only improves the user experience but also sets the stage for the next development in naturally occurring, compassionate computing. Analyzing facial emotions is a useful method for tracking and evaluating mental health issues. We are motivated by the possibility of using FER as an objective, non-intrusive way to measure emotional health. Through sophisticated facial emotion analysis, this initiative hopes to add to the expanding body of research on the integration of technology into mental health diagnosis. In marketing, the capacity to interpret customer feedback is essential for developing focused and successful advertising campaigns. Our research is driven by the possibility that FER can offer priceless insights into customer sentiment, allowing marketers to precisely customize their strategies. This is in line with the current need for data-driven, emotionally charged marketing campaigns. The primary driving forces behind exploring the complexities of Facial Emotion Recognition are the growing interest, increasing demand, and extensive implementation of FER systems. The recognition of this research endeavor's revolutionary potential across various industries emphasizes its timeliness and importance.

Our investigation, as we work through the complexities of creating a synthesizable AI model for FER, is based on the conviction that resolving the aforementioned motives is essential to realizing the full promise of emotion recognition technology. In addition to adding to the body of knowledge, this research attempts to provide useful information that may influence the future direction of FER applications in the real world.

1.3 Objectives of the project

- Preprocess the images.
- Build an image classification model.
- Train the model to classify the facial emotions of humans.
- Test or performance analysis of the model.
- Compare the synthesizable model's accuracy and efficiency with state-of-the-art techniques for facial emotion recognition.

1.4 Literature Survey

The accuracy and resource issues in deep learning-based Facial Emotion Recognition (FER) systems are addressed by the proposed Light-FER model. It uses model compression methods, such as quantization, pruning, and a deep-learning compiler, and is based on the Xception model. The three main procedures consist of face identification using Dlib's 68-landmark face detector, network compression of the FER model, and then facial emotion categorization. Light-FER, implemented on NVIDIA Jetson Nano, shows effectiveness on edge devices with little computational capacity. Impressive results are revealed by experimental validation on multiple platforms, such as Jetson Nano and local PCs: Light-FER outperforms CNN, ResNet-50, and VGG-Net models with a noteworthy test accuracy of 69 percent. Its improved architecture is noteworthy because it guarantees a much lower memory use (just 3.1 percent), all the while keeping equivalent computational costs. Light-FER is positioned as a promising alternative because to its superior performance. [1]

The study offers a comprehensive analysis of facial recognition techniques, including linear approaches like Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Independent Component Analysis (ICA), as well as Local Binary Pattern (LBP), Local Phase Quantization (LPQ), and other techniques. The research highlights the significance of strong algorithms given the abundance of data and the progress made in artificial intelligence. To achieve maximum effectiveness, Graphics Processing Units (GPUs), Central Processing Units (CPUs), and Field-Programmable Gate Arrays (FPGAs) are used. The study highlights the importance of biometric applications by classifying recognition methods into three distinct approaches: local, holistic, and hybrid. This classification demonstrates how widely facial recognition may be applied, especially in fields like security and surveillance. In order to promote a more nuanced knowledge of the many approaches presented, the paper offers key abbreviations with their complete forms, such as Multiscale Local Binary Pattern (MLBP), Markov Random Field (MRF), CMU Pose, Illumination, and Expression (CMU-PIE), and Local Ternary Pattern (LTP). [2]

A Raspberry Pi-powered, low-weight, distributed system with modules and algorithms for identifying emotions is built. Deployment of the system on an NVIDIA Jetson TX2 edge device demonstrates its usefulness and efficiency. Facial action units (AUs), which interpret raw picture input from the camera to detect emotions, are the foundation of emotion detection. The WebFace 12M and WebFace 4M portions of the WebFace260M dataset are used by the system. Classes like neutral, anger, disdain, happiness, disgust, sadness, fear, and surprise are included in the dataset. Significant accuracy is obtained while evaluating on several datasets: IN-THE-LAB DATASET CK+ obtains 93.85 percent, IN-THE-WILD DATASET RAF-DB reaches 81.05 percent, and IN-THE-WILD DATASET POSE-RAF-DB and POSE-AFFECT attains 81.15 percent. This strong performance on a variety of datasets highlights the system's aptitude for precisely identifying emotions in both controlled and real-world scenarios. [3]

An end device in the system under description takes a picture of a person's face and preprocesses the data it has collected. The edge server receives the preprocessed image and uses a Convolutional Neural Network (CNN) model to determine the emotions of the viewer. There are 593 examples in the dataset, divided into seven main emotional categories: fear, anger, disgust, contempt, sadness, happiness, and surprise. Each image is 640 $\times$ 490 or 640 $\times$ 480 pixels in size and is grayscale. The system performs admirably, attaining a 93.5 percent total accuracy rate. Notably, 100 percent accuracy was attained by emotions like contempt, fear, happiness, and melancholy, demonstrating the model's efficacy in correctly detecting these particular emotional states. Accurately identifying disdain and fury, on the other hand, needs work—accuracy scores below 90 percent were shown. This performance review offers insightful information about the emotion recognition system's advantages and potential improvement areas. [4]

A lightweight and effective face recognition network, EdgeFace takes its cues from EdgeNeXt's hybrid architecture. It is enhanced by a low-rank linear layer and combines Transformer and Convolutional Neural Network (CNN) models. The implementation of LoRaLin, a low-rank linear layer, is crucial in achieving substantial computational savings without compromising overall performance. In order to train the models, 4/8 Nvidia RTX 3090 (24GB) GPUs were used, with PyTorch assisting in the distributed training method. Seven different benchmarking datasets were used to evaluate the EdgeFace model, highlighting its adaptability to a range of conditions and adding to its effectiveness as a cutting-edge facial recognition system. [5]

1.5 Problem Definition

Develop a synthesizable AI model to perform image classification on facial emotions using a Convolutional Neural Network.
Input - Facial image.
Output - Facial emotion recognized from the input image.

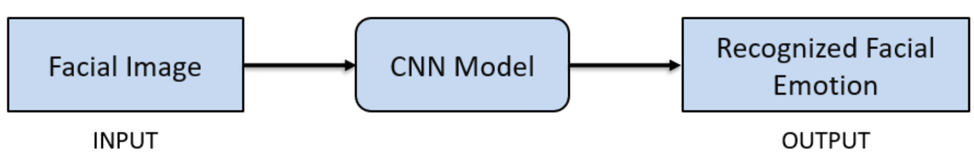

Figure 1.1: Basic Working of FER System

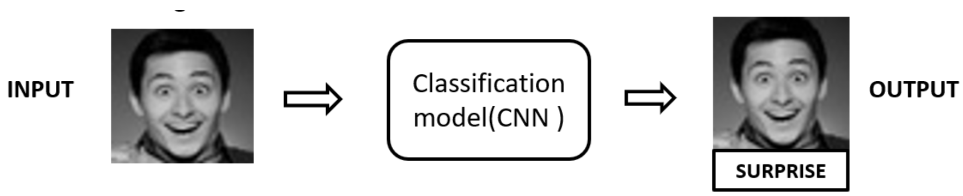

Figure 1.2: Overview of the FER system

1.6 Scope and Constraints

1. The dataset is constrained to 7 emotion classes.

2. The dataset consists of grey-scale images only. Hence, it inherently demands less computational power and is more energy-efficient compared to datasets involving color images.

Chapter 2

SOFTWARE REQUIREMENT SPECIFICATION

A Software Requirements Specification (SRS) is a comprehensive document that serves as a blueprint for the development of a software application. It outlines the functionalities, features, and constraints that the software must adhere to.

2.1 Overview of SRS

The project aims to develop a synthesizable AI model for facial emotion recognition using gray-scale images, prioritizing computational efficiency. Objectives include image preprocessing, model construction, training for emotion classification, testing, and comparison with state-of-the-art techniques. The problem statement emphasizes the unique dataset characteristics, and functional requirements encompass user input, emotion classification, prediction, score generation, and performance analysis. Non-functional requirements focus on achieving a response time under 2 seconds, and a minimum accuracy of 70

2.2 Requirement Specifications

A requirements specification is a comprehensive document that lists all of the functional and non-functional requirements that a product or system needs to satisfy. It is an important stage in the software development life cycle when stakeholders work together to establish and record their requirements and expectations. The document usually contains comprehensive details regarding the features, functionalities, limitations, performance standards, and any other specifications that are considered necessary for the project's effective development and execution. The primary goal of a requirements specification is to establish a shared understanding among project stakeholders, including developers, designers, and users, about what the end product should achieve.

2.2.1 Functional Requirements

Functional requirements are a key component of a requirements specification and define the specific functionalities or features that the system or product must possess to meet the needs and expectations of its users and stakeholders. These requirements describe what the system should do in terms of operations, behaviors, and interactions with its users or other systems.

- The user shall be able to input the gray-scale facial images for emotion recognition.
- The system shall classify and recognize a range of emotions such as happiness, sadness, anger, fear, etc., from the input images.
- The system shall predict the emotions.
- The system generates scores/accuracies for all the emotion classes.
- The user shall be able to view the performance analysis.

2.2.2 Use case diagram

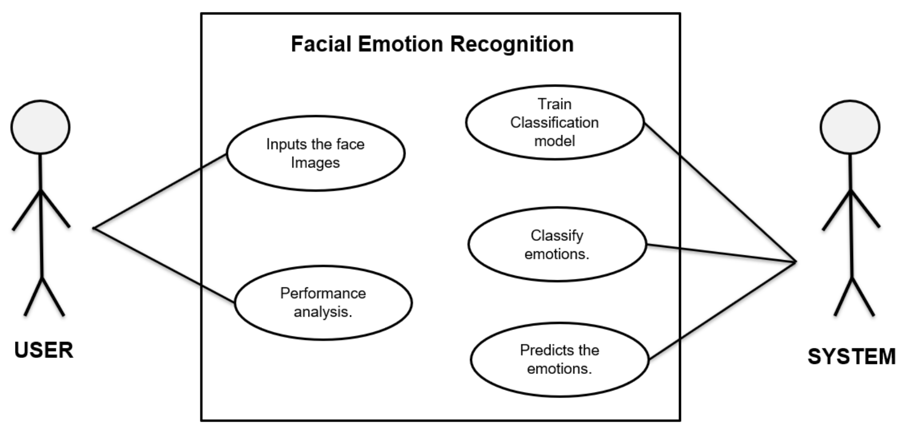

Figure 2.1: Use Case Diagram for FER

2.2.3 Use Case description

Use Case: Facial emotion recognition.

Actors: User and the developer.

Pre-Condition:

1. Grayscale facial images are available for emotion recognition.

Post-Condition:
1. Predicted emotions and associated scores are displayed.
2. Performance analysis, including accuracy metrics and efficiency, is available.

Main Success scenario:
1. The user accesses the facial emotion recognition system.
2. Grayscale facial images are provided as input.
3. The model is trained to classify emotions such as happiness, sadness, anger, etc.
4. The system classifies emotions in the provided images.
5. Predicted emotions and corresponding scores for each emotion class are generated.
6. The user is presented with a comprehensive performance analysis, including accuracy and efficiency metrics.

Exception scenario:
1. Insufficient Input:
Notifying the user if inadequate grayscale facial images are provided.
2. Operational Failures:
Handle errors during image pre-processing, model construction, or training with system logs and notifications.
3. Recognition Challenges:
Address ambiguity in emotion classification, performance analysis unavailability, or comparison process failures with user feedback and logs for investigation.

2.2.4 Nonfunctional Requirements

Non-functional requirements are another crucial aspect of a requirements specification, and they describe aspects of the system's behavior and characteristics beyond specific functionalities. These requirements focus on qualities that affect the overall performance, usability, and reliability of the system. Unlike functional requirements that specify what the system should do, non-functional requirements specify how well the system should do it.

- The system should achieve a response time of less than 2 seconds for emotion recognition processing, ensuring minimal delay in emotion recognition to enhance user experience.
- The emotion recognition system should achieve a minimum accuracy of more than 63 percent on standardized emotion recognition benchmarks.
- The system should achieve a score of at least 85 out of 100 on user satisfaction regarding ease of use, post-implementation.

2.3 Software and Hardware requirement specifications

Software requirements specify the functionalities, features, and constraints that the software application must adhere to. Hardware requirements specify the necessary physical components, devices, and infrastructure needed to run and support the software.

- Hardware: Core - Intel i5 or i6
  Memory for dataset: 60 MB
  GPU - nvidia GeForce

- Software: Operating system: windows or Linux
  Deep learning framework: TensorFlow, PyTorch, and Keras
  If you are using NVIDIA GPUs, installing CUDA (Compute Unified Device Architecture) and cuDNN (CUDA Deep Neural Network Library) can significantly accelerate deep learning computations.
  Language: python
  Libraries: NumPy, SciPy, and scikit-learn.
  IDE: Jupiter Notebook, VS Code, or PyCharm

Chapter 3

PROPOSED SYSTEM

The proposed system model for Facial Emotion Recognition (FER) envisions an advanced architecture geared towards accurate and efficient emotion classification. This model incorporates sophisticated facial feature extraction, real-time processing, and precise emotion classification capabilities. The system's high-level architecture, as outlined, prioritizes seamless data flow, emphasizing the integration of cutting-edge technologies and algorithms. With a comprehensive overview of the workflow, proposed technologies, and performance metrics, this document serves as a foundational guide for the development team, project managers, and stakeholders, providing a clear roadmap for the creation of an innovative and effective FER system.

3.1 Description of Proposed System.

The proposed system model for our Facial Emotion Recognition (FER) project represents a state-of-the-art architecture that leverages Convolutional Neural Networks (CNN) and relevant pre-trained models to enhance the accuracy and efficiency of emotion classification. The system employs a multi-stage approach, incorporating facial feature extraction and leveraging the knowledge encoded in pre-trained models to capture intricate patterns in facial expressions. The CNN model serves as the core engine for robust emotion recognition, providing a foundation for real-time processing and accurate classification across a spectrum of emotions. This proposed system model is designed to push the boundaries of FER capabilities, offering a sophisticated solution that aligns with the latest advancements in deep learning and computer vision.

The envisaged system adopts an Enhanced Convolutional Neural Network (CNN) architecture for Facial Emotion Recognition (FER). Key facets and modifications in the model are elucidated below:

Addressing Class Imbalance through Oversampling:
The challenge of class imbalance in the training dataset is mitigated through the implementation of the oversampling technique. Each emotion class undergoes oversampling, ensuring a more equitable representation and augmenting the model's capacity to generalize across

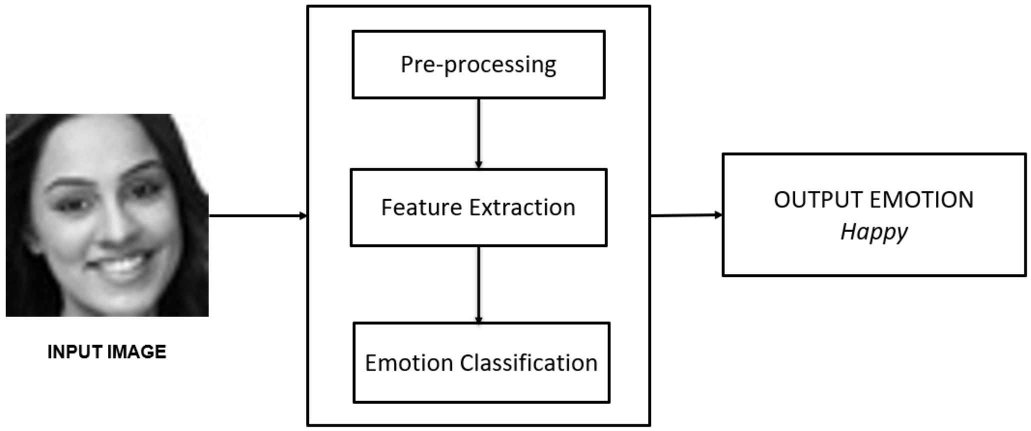

Figure 3.1: High-level design for FER

diverse emotional expressions.

Innovative CNN Architecture (Modified ResNet50-based Model):
The crux of the system lies in an Innovative CNN architecture, featuring a tailored ResNet50-based model termed ImprovedCNN. In this model:
The final classification layer of the ResNet50 architecture is excluded.
A batch normalization layer is introduced, comprising 2048 features to align with ResNet50's final layer.
A customized fully connected layer is integrated to tailor the model for emotion classification.
A dropout layer is infused to bolster the model's generalization prowess.

Fine-tuning through Hyperparameter Adjustments:
The system undergoes rigorous hyperparameter tuning, fine-tuning parameters such as the learning rate, number of epochs, and dropout rate. This meticulous optimization process aims to enhance the model's convergence and elevate accuracy.

Strategic Training Approaches:
The training strategy involves iterative sessions with variations in epochs and learning rates to discern the optimal configuration. Crucial metrics such as accuracy, precision, recall, and loss are closely monitored during training, guiding informed decisions on model refinements.

Robust Evaluation Metrics:
Standard evaluation metrics, including accuracy, precision, recall, and loss, are employed to evaluate the model's performance. Class-specific analysis provides insights into the model's

proficiency across diverse emotions, facilitating targeted enhancements.

Transfer Learning with Modified ResNet50:
The proposed system integrates transfer learning with a modified ResNet50 architecture. This entails harnessing the pre-trained model's capability to extract intricate features, thereby enhancing the model's discernment of subtle facial expressions and elevating overall recognition accuracy. The proposed system is intricately designed to outperform existing state-of-the-art FER models. Its implementation is meticulously crafted to surmount challenges related to class imbalance, employing advanced neural network architectures to bolster accuracy and resilience.

3.2 Description of Target Users

The proposed system workflow delineates the sequential steps involved in processing facial images and recognizing emotions. It starts with the acquisition of facial images, proceeds to facial feature extraction, enters the emotion classification stage, and concludes with the output of recognized emotions. Each step is intricately connected, forming a seamless workflow that ensures efficient and accurate emotion recognition.

3.3 Applications of implemented FER System

- Human-Computer Interaction (HCI): Enhances computer interaction by allowing systems to respond to users' emotions in gaming, virtual reality, and interactive environments.

- Marketing and Advertising: Provides insights for marketers by analyzing facial expressions, helping tailor strategies and content to evoke specific emotional responses in advertisements.

- Healthcare and Well-being: Assists in early detection and monitoring of mental health conditions by analyzing changes in facial expressions over time.

- Education: Gauges student engagement and emotional states during lessons, allowing educators to adapt teaching methods to individual learning needs.

- Security and Surveillance: Enhances surveillance systems by identifying individuals displaying suspicious or abnormal emotional patterns in public spaces.

- Automotive Industry: Enhances driver safety by detecting signs of drowsiness or distraction, triggering alerts or safety features in vehicles.

- Robotics: Improves human-robot interaction by enabling robots to understand and respond to human emotions.

Chapter 4

SYSTEM DESIGN

The system design encompasses the architectural and operational aspects of the Facial Emotion Recognition (FER) system. It includes details about the model architecture, data preprocessing, training strategy, and evaluation processes.

4.1 Architecture of the system

Innovative Model Architecture:

- At the heart of the system lies the ImprovedCNN model, a refined Convolutional Neural Network based on ResNet50, tailored for emotion classification.

The model comprises key elements:

- Utilization of a pre-trained ResNet50 architecture, omitting the final classification layer.

- Introduction of a batch normalization layer featuring 2048 features to enrich feature representation.

- Incorporation of a customized fully connected layer dedicated to emotion classification.

- Inclusion of a dropout layer to enhance generalization by mitigating overfitting.

Data Preprocessing Strategies:

- Addressing class imbalance is pivotal, achieved through oversampling to ensure equitable representation of each emotion class, contributing to enhanced model generalization.

- Image preprocessing involves conversion to grayscale, resizing to (224, 224) pixels to align with ResNet50's input size, and normalization for consistent input to the model.

-

Optimized Training Approach:

- A robust training strategy is employed, featuring hyperparameter tuning to maximize model performance.

- Hyperparameter tuning encompasses fine adjustments to learning rates, epochs, and dropout rates. This iterative process aids in determining an optimal configuration.

- Implementation of early stopping, contingent on test loss, acts as a preventative measure against overfitting by halting training if no improvement is observed over a specified epoch range.

Comprehensive Evaluation Metrics:

- Evaluation of the system relies on standard metrics such as accuracy, precision, recall, and loss.

- Class-wise analysis provides a nuanced understanding of the model's performance across individual emotions, offering valuable insights for targeted refinements.

Transfer Learning with ResNet50:

- Transfer learning is harnessed through ResNet50, leveraging pre-trained weights and benefiting from knowledge acquired on expansive image datasets.

- The modified ResNet50 architecture undergoes fine-tuning, adapting its features for precise emotion recognition through facial expression analysis.

•

Rigorous Testing Protocol:

- The trained model undergoes evaluation on a distinct test dataset to gauge its generalization capabilities.
- Key metrics, including test accuracy, precision, recall, and loss, are reported to quantify and communicate the model's overall performance.
- The system design underscores the strategic integration of advanced neural network architectures, meticulous data preprocessing, and iterative training approaches to achieve a discerning and resilient Facial Emotion Recognition system.

4.1.1 CNN Architecture

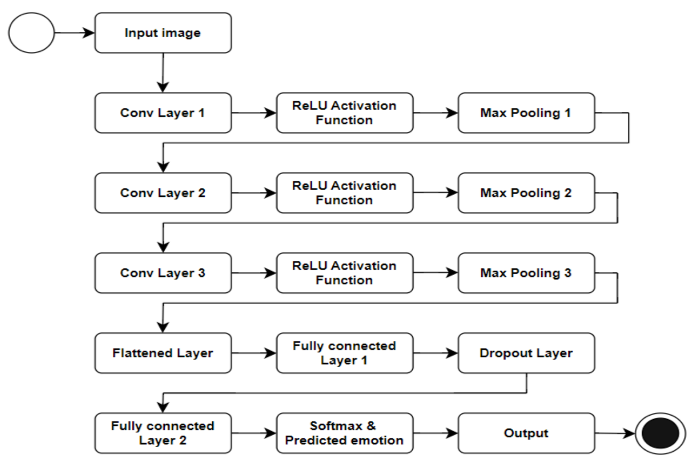

Figure 4.1: Activity diagram for CNN Architecture

4.1.2 ResNet50 Architecture

ResNet50 is a deep convolutional neural network architecture featuring 50 layers, known for its residual connections that enable the training of very deep networks effectively. It utilizes skip connections to mitigate vanishing gradient problems, facilitating the learning of intricate features.

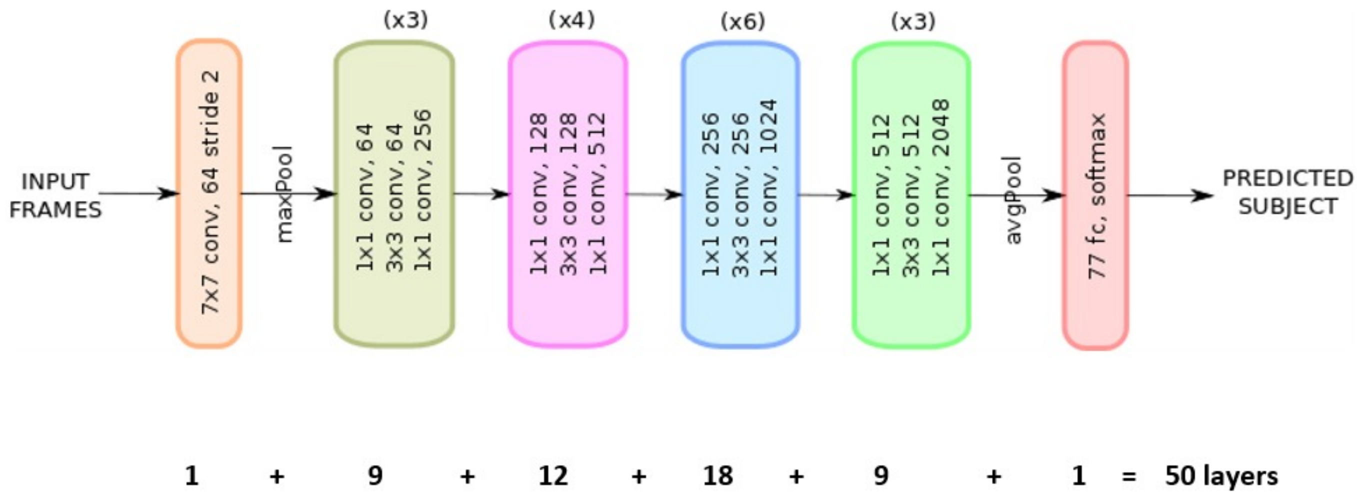

Figure 4.2: Activity diagram for ResNet50 pre-trained model Architecture

4.2 Data Set Description

1. The dataset is taken from Kaggle[6].
2. Dataset size: 56.51MB.
3. The dataset contains 35,914 grayscale images of faces.
4. Testing images – 20.06
5. Training images – 79.93
6. The dataset consists of 7 classes. They are- Happy, sad, disgust, angry, fear, neutral, and surprise.
7. The images in the dataset are approximately of the size 2KB each.

Emotion Classes	Number of Images
Disgust	547
Fear	5121
Happy	8989
Neutral	6198
Sad	6077
Surprise	4002
Angry	4953

Table 4.1: Emotion Classes and the number of Images for each class in the dataset.

Figure 4.3: Sample images of each class in the Dataset

Algorithm 1 ResNet50 class-wise Emotion Recognition Model Training and Evaluation

---

Require: Training and test datasets, pre-trained ResNet model, hyperparameters
Ensure: Trained emotion recognition model
Initialize emotion labels: `"angry", "disgust", 'fear', 'happy', 'neutral', 'sad', 'surprise"`
Initialize paths to train and test datasets
Load and balance the class distribution in the training dataset
Display class distribution before and after balancing
Initialize transformation pipeline for images
Define custom dataset class (`CustomDataset`)
Define ImprovedCNN model class (`ImprovedCNN`)
Move the model to GPU if available
Initialize loss function (`CrossEntropyLoss`) and optimizer (`SGD`)
Create DataLoader for the training dataset
Create DataLoader for the test dataset
Set the number of training epochs (`n_epochs`)
for each epoch in range(`n_epochs`) **do**
    Set model to training mode (`model.train()`)
    Perform training steps
    Calculate average training loss and accuracy
    Calculate class-wise training accuracy
    Print and display training metrics
    Set model to evaluation mode (`model.eval()`)
    Perform evaluation on the test set
    Calculate average test loss and accuracy
    Calculate weighted precision and recall
    Calculate class-wise test accuracy
    Print and display test metrics
end for
return Trained emotion recognition model

Chapter 5

IMPLEMENTATION

The implementations of Convolutional Neural Networks (CNN) and ResNet50 represents a significant advancement in the field of computer vision, particularly in tasks such as image classification and feature extraction. CNNs are deep neural networks specifically designed for processing structured grid data, making them highly effective for image-related tasks. Their architecture includes convolutional layers, pooling layers, and fully connected layers, allowing them to automatically learn hierarchical features from images. On the other hand, ResNet50, short for Residual Network with 50 layers, introduces residual connections that enable the training of deeper networks without suffering from the vanishing gradient problem. This innovation in residual learning has proven crucial in achieving state-of-the-art performance in various image recognition tasks. Implementing CNN and ResNet50 involves careful consideration of model architecture, hyperparameter tuning, and training on large datasets. Moreover, pre-trained models for these architectures, available in popular deep learning frameworks like TensorFlow and PyTorch, have streamlined their implementation in various applications, providing a powerful foundation for image-based tasks in computer vision.

5.1 Proposed Methodology

The total number of train images after pre-processing is 50,505. The total number of test images after pre-processing is 12,418. The architecture used for implementation is Convolutional Neural Networks(CNN). The CNN consists of three sets of Convolutional-ReLU-MaxPooling layers for feature extraction and down-sampling.

This architecture is designed to learn and extract meaningful features from facial expression images, enabling the model to classify emotions into different categories based on the learned features. Adjustments or variations in the architecture can be made for specific performance or task-related requirements.

Initially, our model underwent 100 epochs of training with a learning rate of 0.0001, achieving an accuracy of 33.38%.

During the evaluation, we observed its stronger performance particularly in the 'surprise', 'happy', and 'disgust' classes.

Subsequently, we extended the number of epochs to 250, achieving an accuracy of 45.77%. Despite this increase in training duration, there was only a marginal improvement in the performance of the classes labeled as 'angry,' 'neutral,' and 'sad.'

Data Balancing Impact: Oversampling proved effective in balancing the dataset, showcasing positive outcomes in classes with both the highest and lowest sample numbers ('surprise', 'happy', and 'disgust').
Effect of Training Epochs: Raising epochs from 100 to 250 moderately boosted overall accuracy (33.38% to 45.77%), but improvements plateaued, particularly for classes like 'angry,' 'neutral,' and 'sad.'

Stronger Performance in Specific Classes: There is a tendency for the model to confuse classes such as 'angry', 'fear', and 'neutral', often predicting them as 'sad'.

Limited Improvement for Certain Classes:
Extended training showed minimal performance improvement for classes 'angry,' 'neutral,' and 'sad.' This suggests complexity in discerning these classes, hinting at the need for diverse data or alternative approaches.
Potential Learning Plateau:
Marginal overall accuracy improvement with a significant increase in epochs hints at a learning plateau. The model may be reaching its capacity for further improvement, encountering diminishing returns. Therefore, we opted for a learning rate of 0.001 and extended to 500 epochs as the graph shows the model hasn't converged. This suggests the need for additional learning.

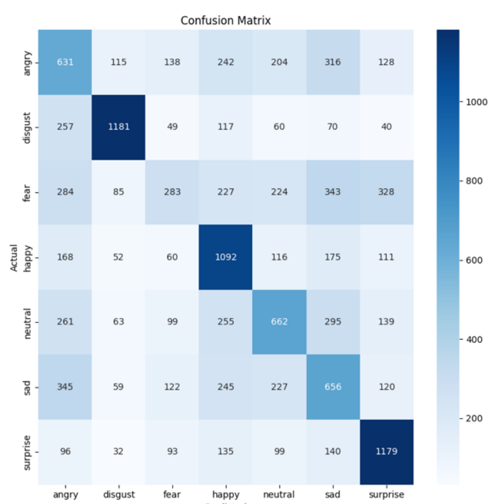

Figure 5.1: Confusion matrix for 250th epoch

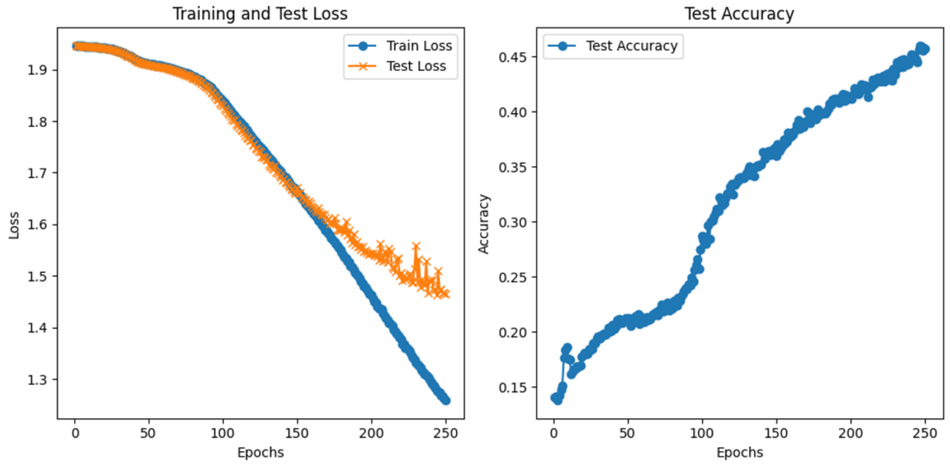

Figure 5.2: Graphs for train and test losses and accuracies

Class-wise observations:

Epoch: 25
Learning rate: 0.0001
From the 1st to the 25th epoch, the model is able to recall images from class "surprise" effectively. The class " fear " is recalled inefficiently. Class "happy" has the maximum precision as it predicts the most correctly.
Epoch : 50
Learning rate : 0.0001

The recall score of class "surprise" reduced from 81 percent to 64 percent and the other classes like "sad" , "angry", "neutral" started contributing .

Epoch : 74
Learning rate : 0.0001
From figure 5.3, the following observations can be drawn:
The recall score of class "surprise" was further reduced and the contribution of all the other classes increased, with "sad" being the 2nd most contributing class. The class happy still has the highest accuracy and continues to predict the most correctly.
The class "happy" has the maximum precision throughout all the epochs and it predicts the best. As epochs increase, the recall score of the class that is contributing to maximum decreases gradually, and all the other classes that had less recall scores, start contributing. The overall test and train accuracies are 24.9percent and 24.09 percent as of the 75th epoch with a learning rate of 0.0001.

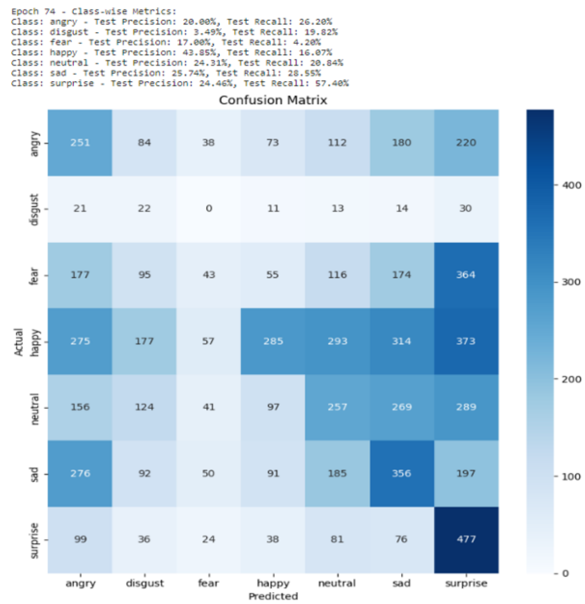

Figure 5.3: Confusion matrix: class-wise (74th epoch)

We further increased the learning rate from 0.0001 to 0.001 with the attempt to increase the test accuracy.

Number of epochs: 250
Learning rate: 0.001
Initially, our model underwent 250 epochs of training with a learning rate of 0.001, achieving an accuracy of 52.67%.

During the evaluation, we observed its stronger performance particularly in the 'surprise', 'happy', and 'disgust' classes.
Number of epochs: 400
Learning rate: 0.001
Subsequently, we extended the number of epochs to 400, achieving an accuracy of 52.51%. Despite this increase in training duration, there was only a marginal improvement in the performance of the classes labeled as 'angry,' 'neutral,' and 'sad.'

By observing figure 5.3 below, we can say that the training loss gradually decreased as the number of training epochs increased, but these was as increase in the test loss because of imbalanced classes. We achieved a training accuracy of 98.77% at the 125th epoch of training and the graph converged to the x-axis. (The training accuracy was 98.7% throughout the training up to 400 epochs) The test accuracy remained 53% from the 60th epoch to the 400th epoch of training. The precision and recall values were 0.5204 and 0.5251 respectively from the 60th epoch onwards.

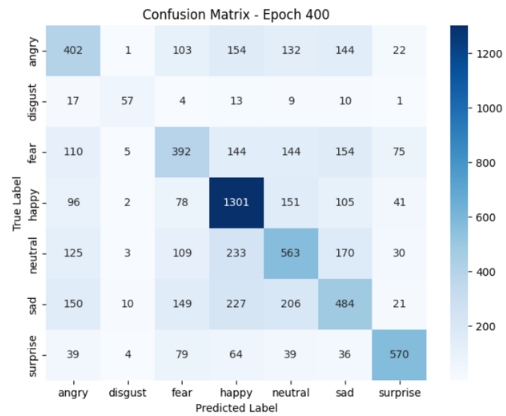

Figure 5.4: Confusion matrix at 400th epoch (Using CNN model)

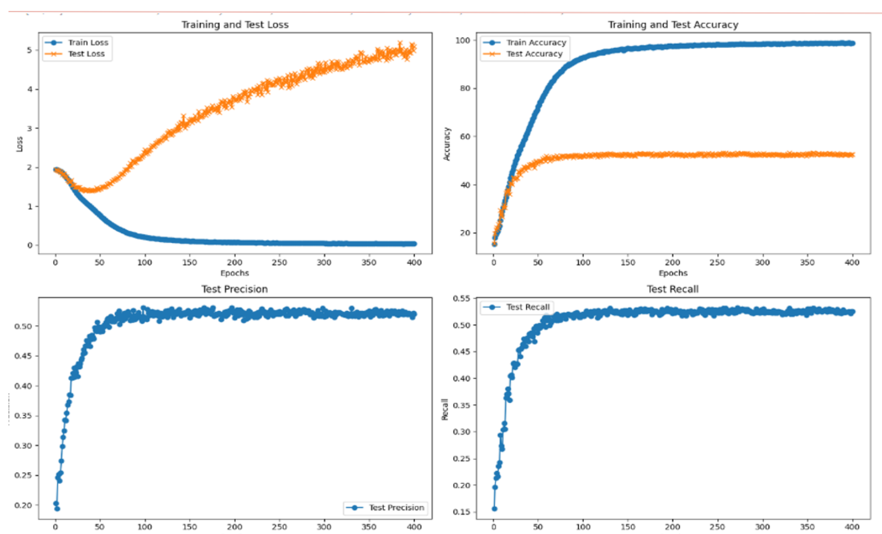

Figure 5.5: Graphs for training and testing losses, and precision and recall scores against epochs

ResNet18 with Improved CNN

Number of epochs: 25/100

Learning rate: 0.001

Train Loss: 1.5068

Train Accuracy: 62.79%

Test Loss: 1.5460

Test Accuracy: 62.71%

Test Precision: 0.6250

Test Recall: 0.6271

Early stopping after 5 epochs without improvement.

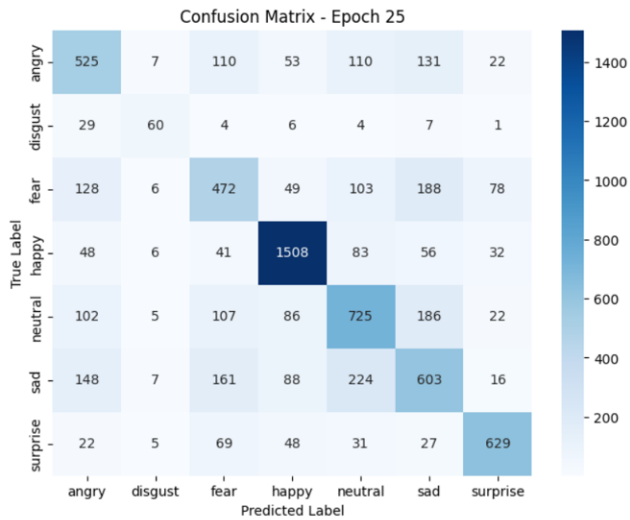

Figure 5.6: Confusion matrix at 25th epoch (Using ResNet50 model)

ResNet50 with Improved CNN

Number of epochs: 25/100
Learning rate: 0.001
From figure 5.5, we observe the following scores:
Train Loss: 1.5080, Train Accuracy: 62.91%
Test Loss: 1.5256, Test Accuracy: 64.22%, Test Precision: 0.6459, Test Recall: 0.6422
Early stopping after 5 epochs without improvement.

5.2 Class-Wise Accuracy for ResNet50 with Improved-CNN model:

The model performs well in capturing diverse facial expressions, particularly excelling in recognizing 'happy' expressions.

However, attention to improving accuracy in certain classes like 'fear' may further enhance its overall performance.

Epoch [25/100], Train Loss: 1.5112, Train Accuracy: 62.28%, Test Loss: 1.5265, Test Accuracy: 64.38%, Test Precision: 0.6467, Test Recall: 0.6438

CLASS	ACCURACY
ANGRY	55.22%
DISGUST	58.56%
FEAR	47.56%
HAPPY	84.16%
NEUTRAL	61.07%
SAD	52.53%
SUPRISE	76.90%

Figure 5.7: Class-wise accuracies for ResNet50 Model

5.3 Comparison of different CNN models for FER:

Model	Learning Rate	Epoch	Test Accuracy (%)
CNN with 3 conv layers	0.0001	100	33.38%
CNN with 3 conv layers	0.0001	250	45.77%
CNN with 3 conv layers	0.001	250	52.67%
CNN with 3 conv layers	0.001	400	52.50%
Improved Resnet18	0.001	100	62.71%
Improved Resnet50	**0.001**	**25**	**64.38%**

Figure 5.8: Accuracies of different CNN Models

The fig. 5.8 presents the comparison of the different fine-tuned CNN models implemented by varying learning rates and the number of epochs.

Chapter 6

RESULTS AND DISCUSSIONS

The ResNet50 model outperforms the basic CNN in terms of class-wise test accuracy, achieving 58.50% compared to the CNN's 45.16%. Additionally, the class-wise test accuracies for anger, disgust, happy, neutral, sad, and surprise generally show improvements with ResNet50, indicating its effectiveness in capturing more complex patterns. The overall test accuracy combined with class-wise results in Resnet50 is 64.47%, which outperforms the CNN model with an overall test accuracy of 52.20%. However, it's important to note that both models face challenges in classifying certain emotions. For instance, the disgust class has low precision and recall for both models, suggesting difficulties in correctly identifying this emotion.
The class-wise metrics for the ResNet50 model reveal notable improvements in precision and recall for various emotions, such as happiness and surprise. The ResNet50 model demonstrates a higher precision and recall for surprise, indicating its ability to better detect this emotion compared to the basic CNN.

In conclusion, while the ResNet50 model generally outperforms the basic CNN in terms of overall accuracy and specific class-wise metrics, there are still challenges in accurately classifying certain emotions. Further model fine-tuning or exploring advanced architectures may help address these challenges and enhance overall performance.

6.1 Comparison of CNN models implemented with the state-of-the-art:

TEST ACCURACY	Light-FER	VGG-Net	ResNet-50	CNN
[1]	69%	60%	60%	60%
OUR Model	-	-	64.38%	52.50%

Figure 6.1: Comparison of accuracies with the State-of-the-art.

Chapter 7

CONCLUSIONS AND FUTURE SCOPE

The Facial Emotion Recognition (FER) system, anchored by the ResNet50-based ImprovedCNN architecture, achieves a notable test accuracy of 64.22%, surpassing baseline configurations through transfer learning for enhanced feature extraction and effective facial expression recognition. Competitive performance is demonstrated with Improved CNN outperforming baseline models, showcasing its efficacy in recognizing facial emotions. Class-wise analysis provides detailed metrics, revealing competitive accuracy across emotion categories and offering insights for targeted enhancements. The impact of transfer learning, utilizing ResNet50, accelerates training, capturing complex features and improving generalization, while challenges such as class imbalance are addressed through oversampling and hyperparameter tuning with early stopping, enhancing model robustness. Future scope includes investigating ensemble methods for overall model robustness, deployment in real-time applications like interactive interfaces, and aiming for competitive accuracy compared to state-of-the-art models, showcasing progress in FER. The extension of the model to handle multi-modal inputs, such as combining facial expressions with voice analysis, is suggested for a more comprehensive emotion recognition system.

7.0.1 Overall Impact:

- The FER system, with ImprovedCNN at its core, presents robustness in recognizing facial emotions, laying the groundwork for applications in human-computer interaction and affective computing. Further research can address challenges and expand the system's capabilities.

REFERENCES

[1] Erick Kim Jeong-in Jeong Jin-Woo Jung Yuchul Kim Sang-Ho Lim Wansu. Pascual, Alexander Valverde. https://www.researchgate.net/publication/366053953_Light-FER_A_Lightweight_Facial_Emotion_Recognition_System_on_Edge_Devices, 2022.

[2] Erick Kim Jeong-in Jeong Jin-Woo Jung Yuchul Kim Sang-Ho Lim Wansu. Pascual, Alexander Valverde. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7013584/.

[3] F. Zhang J. Yang, T. Qian and S. U. Khan. https://ieeexplore.ieee.org/document/9438728, 2021.

[4] M. Shamim Muhammad, Ghulam Hossain. https://www.researchgate.net/publication/356396496_Emotion_Recognition_for_Cognitive_Edge_Computing_Using_Deep_Learning, 2021.

[5] Christophe Otroshi Hatef Kotwal Ketan Marcel Sébastien. George, Anjith Ecabert. https://arxiv.org/abs/2307.01838, 2023.

[6] Kaggle FER dataset. Edgeface: Efficient face recognition model for edge devices. https://www.kaggle.com/datasets/chiragsoni/ferdata, 2021.

[7] Maurice Beek, Antonio Bucchiarone, and Stefania Gnesi. A survey on service composition approaches: From industrial standards to formal methods. In *Technical Report 2006TR-15, Istituto*, pages 15–20. IEEE CS Press, 2006.

[8] Jinghai Rao and Xiaomeng Su. A survey of automated web service composition methods. In *Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition, SWSWPC*, pages 43–54, 2004.

[9] Sodki Chaari, Khouloud Boukadi, Chokri Ben Amar, Fredrrique Biennier, and Joe Favrel. Developing service oriented enterprise by composing web services based on context. In *IJCSNS International Journal of Computer Science and Network Security*, volume 8, pages 79–92, 2008.

[10] Bart Orriëns, Jian Yang, and Mike P. Papazoglou. Model driven service composition. In *ICSOC 2003*, pages 75–90. Springer-Verlag, 2003.

[11] Abdaladhem Albreshne and Jacques Pasquier. Semantic-based semi-automatic web service composition. In *Proceedings of the Fifth Libyan Arab International Conference On Electrical and Electronic Engineering LAICEEE*, volume 1, pages 603–615, 2010.

[12] P. Traverso and M. Pistore. Automated composition of semantic web services into executable processes. pages 380–394. Springer-Verlag, 2004.

[13] I. Budak Arpinar, Ruoyan Zhang, Boanerges Aleman-meza, and Angela Maduko. Ontology-driven web services composition platform. In *Information Systems and e-Business Management*, volume 3, pages 6–9, 2004.

[14] Kaarthik Sivashanmugam, John A. Miller, Amit P. Sheth, and Kunal Verma. Framework for semantic web process composition. In *International Journal of Electronic Commerce*, volume 9, pages 71–106, 2005.

[15] Fabio Casati, Ski Ilnicki, Lijie Jin, Vasudev Krishnamoorthy, and Ming chien Shan. Adaptive and dynamic service composition in eflow. In *Proceedings 12th International Conference, CAiSE*, pages 13–31, 2000.

[16] Jinghai Rao and Xiaomeng Su. A survey of automated web service composition methods. In *First International Workshop, SWSWPC*, pages 43–54, 2004.

[17] James McGovern, Sameer Tyagi, Michael Stevens, and Sunil Matthew. *Java Web Services Architecture*. Morgan Kaufmann, 2003.

[18] Grigoris Antoniou and Frank van van Harmelen. *A Semantic Web Primer (Cooperative Information Systems series)*. The MIT Press, 2008.

[19] From Wikipedia. `http://en.wikipedia.org/wiki/Web_service`.

[20] From Obitko. `http://obitko.com/tutorials/ontologies-semantic-web`.

[21] From Wikipedia. `http://en.wikipedia.org/wiki/Service-oriented_architecture`.

REFERENCES

[1] Erick Kim Jeong-in Jeong Jin-Woo Jung Yuchul Kim Sang-Ho Lim Wansu. Pascual, Alexander Valverde. https://www.researchgate.net/publication/366053953_Light-FER_A_Lightweight_Facial_Emotion_Recognition_System_on_Edge_Devices, 2022.

[2] Erick Kim Jeong-in Jeong Jin-Woo Jung Yuchul Kim Sang-Ho Lim Wansu. Pascual, Alexander Valverde. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7013584/.

[3] F. Zhang J. Yang, T. Qian and S. U. Khan. https://ieeexplore.ieee.org/document/9438728, 2021.

[4] M. Shamim Muhammad, Ghulam Hossain. https://www.researchgate.net/publication/356396496_Emotion_Recognition_for_Cognitive_Edge_Computing_Using_Deep_Learning, 2021.

[5] Christophe Otroshi Hatef Kotwal Ketan Marcel Sébastien. George, Anjith Ecabert. https://arxiv.org/abs/2307.01838, 2023.

[6] Kaggle FER dataset. Edgeface: Efficient face recognition model for edge devices. https://www.kaggle.com/datasets/chiragsoni/ferdata, 2021.

[7] Maurice Beek, Antonio Bucchiarone, and Stefania Gnesi. A survey on service composition approaches: From industrial standards to formal methods. In *Technical Report 2006TR-15, Istituto*, pages 15–20. IEEE CS Press, 2006.

[8] Jinghai Rao and Xiaomeng Su. A survey of automated web service composition methods. In *Proceedings of the First International Workshop on Semantic Web Services and Web Process Composition, SWSWPC*, pages 43–54, 2004.

[9] Sodki Chaari, Khouloud Boukadi, Chokri Ben Amar, Fredrrique Biennier, and Joe Favrel. Developing service oriented enterprise by composing web services based on context. In *IJCSNS International Journal of Computer Science and Network Security*, volume 8, pages 79–92, 2008.

[10] Bart Orriëns, Jian Yang, and Mike P. Papazoglou. Model driven service composition. In *ICSOC 2003*, pages 75–90. Springer-Verlag, 2003.

[11] Abdaladhem Albreshne and Jacques Pasquier. Semantic-based semi-automatic web service composition. In *Proceedings of the Fifth Libyan Arab International Conference On Electrical and Electronic Engineering LAICEEE*, volume 1, pages 603–615, 2010.

[12] P. Traverso and M. Pistore. Automated composition of semantic web services into executable processes. pages 380–394. Springer-Verlag, 2004.

[13] I. Budak Arpinar, Ruoyan Zhang, Boanerges Aleman-meza, and Angela Maduko. Ontology-driven web services composition platform. In *Information Systems and e-Business Management*, volume 3, pages 6–9, 2004.

[14] Kaarthik Sivashanmugam, John A. Miller, Amit P. Sheth, and Kunal Verma. Framework for semantic web process composition. In *International Journal of Electronic Commerce*, volume 9, pages 71–106, 2005.

[15] Fabio Casati, Ski Ilnicki, Lijie Jin, Vasudev Krishnamoorthy, and Ming chien Shan. Adaptive and dynamic service composition in eflow. In *Proceedings 12th International Conference, CAiSE*, pages 13–31, 2000.

[16] Jinghai Rao and Xiaomeng Su. A survey of automated web service composition methods. In *First International Workshop, SWSWPC*, pages 43–54, 2004.

[17] James McGovern, Sameer Tyagi, Michael Stevens, and Sunil Matthew. *Java Web Services Architecture*. Morgan Kaufmann, 2003.

[18] Grigoris Antoniou and Frank van van Harmelen. *A Semantic Web Primer (Cooperative Information Systems series)*. The MIT Press, 2008.

[19] From Wikipedia. `http://en.wikipedia.org/wiki/Web_service`.

[20] From Obitko. `http://obitko.com/tutorials/ontologies-semantic-web`.

[21] From Wikipedia. `http://en.wikipedia.org/wiki/Service-oriented_architecture`.

Appendix A

A.1 Gantt Chart

Gantt chart:

TIMELINE	WEEK 1- 5 (03/10/23 to 10/11/23)	WEEK 5-7	WEEK 7- 10	WEEK 10-15
Problem Identification, Literature Survey, dataset Identification and understanding, SRS.	X			
High level Design Framework and midway Implementation of AI model.		X		
Complete Implementation of AI model [Training of the Model].			X	
Performance Testing and Report.				X

A.2 Description of Tools and Technology used

Kaggle:

Kaggle is a platform for data science and machine learning competitions. It provides datasets, a cloud-based workbench for data analysis, and a community where data scientists and machine learning enthusiasts can collaborate and share insights. Usage in the Project: Kaggle can be utilized for accessing datasets relevant to the project, participating in machine learning competitions, and engaging with the data science community for discussions and insights.

Visual Studio Code (VS Code):

Visual Studio Code is a lightweight and versatile code editor developed by Microsoft. It supports various programming languages, provides debugging capabilities, and offers extensions for additional features. Usage in the Project: VS Code can serve as the primary integrated development environment (IDE) for coding tasks in the project. Its flexibility and extensibility make it suitable for a wide range of programming languages and project types.

Draw.io:

Draw.io is an online diagramming tool that allows users to create a variety of diagrams, flowcharts, and visual representations. It is user-friendly and provides a wide range of shapes and templates. Usage in the Project: Draw.io can be used for creating visual representations of project architectures, flowcharts, or any diagrams that help in explaining the project's design and workflow. It aids in visual communication and documentation. Each tool plays a specific role in the project, contributing to different aspects such as data exploration and analysis (Kaggle), code development (VS Code), project hosting (Server), and visual representation/documentation (Draw.io).

Integrating these tools efficiently can enhance collaboration, streamline development processes, and improve the overall project workflow.

Overleaf:

Overleaf is a collaborative online platform designed specifically for researchers, academics, and professionals in the fields of science, technology, engineering, and mathematics (STEM). It allows users to create, edit, and collaborate on LaTeX documents in real time, eliminating the need for manual version control and facilitating seamless teamwork. With its intuitive interface and extensive library of templates, Overleaf streamlines the process of writing research papers, reports, and academic documents, while also offering features such as real-time previewing, easy sharing and collaboration, and integration with other tools like GitHub. Overleaf significantly enhances the efficiency and effectiveness of the document creation and collaboration process, empowering researchers to focus more on their content and less on the technicalities of document formatting and management.