

Chapter 1: Introduction and Overview

1.1. Problem Definition

The lack of interpretation of non-verbal cues in social interaction is an aspect of major challenges that visually impaired people may encounter, especially in regards to facial expressions, which are another form of expression where emotions and intentions are most visibly communicated in human communication. Failure to recognize the cues can lead to some of the following problems: lack of understanding, poor communication and social isolation which can have a negative impact on the psychological health of a person and their social participation (World Health Organization, 2019).

The recognition of facial emotion is a critical issue for the designing of assistive technologies for human–computer interaction and to assist vulnerable groups, such as visually impaired individuals (Sari et al., 2019). In spite of the development of computer vision and deep learning technologies, most current assistive systems have not yet been capable of emotionally interpreting real-time data, processing in a portable device, and at a affordable price in everyday social settings.

The majority of traditional assistive technologies mainly address navigation support, object detection, or TTS (text-to-speech) conversion, whereas there are relatively fewer systems that address the understanding of the emotions in real time in social situations (Kusuma et al., 2020). In addition, despite the performance in terms of accuracy of deep learning models like Convolutional Neural Networks (CNNs) in facial expression recognition (FER) applications, the computational requirements of such models are still a concern when deployed in embedded systems with limited resources (Shehada et al., 2023).

Recent studies indicate that optimized deep learning architectures can significantly improve the accuracy and efficiency of facial emotion recognition systems while enabling real-time performance on edge devices (Hussain et al., 2023). However, there is a lack of research in creating systems that can be placed on embedded platforms to help VIVs interpret emotions in social scenarios for which they are lightweight, accurate and affordable.

Thus, an intelligent facial emotion recognition system real-time and low cost which can accurately classify facial emotion and transform it to easy-to-interpret audio

feedback is needed. This kind of system would be beneficial to social inclusion, communication effectiveness and independent interaction for visually impaired persons.

1.2. Thesis Objectives

Main Objective

The primary goal of this thesis is to create an intelligent real-time emotion recognition system, that relies on deep learning techniques, able to recognize the emotions of facial expressions and provide auditory feedback during social interactions with a visually impaired user.

Sub-Objectives

- To develop an automated computer vision pipeline using OpenCV for real-time face detection and region-of-interest (ROI) cropping from camera inputs.
- To design and train a custom Convolutional Neural Network (CNN) architecture capable of classifying facial expressions into 7 standard categories (Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral).
- To mitigate dataset class imbalance by integrating dynamic class weighting techniques during model training.
- To improve model performance on difficult samples through a two-phase training strategy (hard-example fine-tuning).
- To convert recognized facial emotions into natural spoken verbal phrases using Text-to-Speech (TTS) technology.
- To build an accessible interactive web-based interface (using Gradio) that supports real-time camera evaluation and immediate audio feedback.
- To optimize the deep learning pipeline to ensure high accuracy and low latency suitable for lightweight and portable deployment.

1.3. Thesis Contribution

This thesis aims to make a contribution to the field of assistive technologies and facial emotion recognition by proposing and implementing a real-time, low-cost facial emotion recognition system that is specifically tailored to the task of assisting the visually impaired in social communication. The main accomplishments of this research are:

- End to end Integration: A unified system encompassing real-time face detection (OpenCV Haar Cascade), ROI extraction, CNN classification, and emotional voice feedback generation (gTTS).
- Advanced training strategy: Implementation of class weight balancing alongside a novel two-phase fine-tuning approach that isolates and re-trains the model on hard/misclassified facial images to boost accuracy.
- End-to-end pipeline design: Face detection, Feature Extraction, Emotion Classification and Generation of Audio Feedback in real-time.
- Accessible Auditory Output Integration: Mapping emotional classifications directly into context-aware spoken sentences (e.g., "I am feeling very happy today!").
- Web & Live Camera Deployment: Deployment of an interactive user interface using Gradio and Browser-based camera capture using JavaScript MediaDevices API.

1.4. Thesis Organization

This thesis consists of five chapters that deal with each of the stages of the research process, ranging from problem identification, system building, evaluation of the system, and finally conclusions.

Chapter 1: Introduction and Overview

Chapter 2: Background and Related Work

Chapter 3: Proposed Approach

Chapter 4: Experimental Results and Discussion

Chapter 5: Conclusion and Future Work

Chapter 2: Background and Related Work

2.1. Background

Facial expression recognition (FER) has become an important research area within computer vision and artificial intelligence due to its role in understanding human emotions and improving human–computer interaction. The use of facial expressions as a means of communication is one of the most natural, universal and spontaneous ways by which man communicates emotions, many of which can be conveyed only through facial expression. Therefore, automatic recognition of such expressions has recently become a great interest in recent years, especially in the context of deep learning techniques (Sari et al., 2019).

Initial studies of FER used classic image processing techniques and patterns-based machine learning algorithms such as Support Vector Machines (SVM) and k-Nearest Neighbors (KNN) to detect facial features. Traditional image processing techniques and machine learning algorithms like Support Vector Machines (SVM) and k-Nearest Neighbors (KNN) were initially used to detect facial features. While these methods worked well in a controlled set-up, they failed to perform well in the presence of changes in light conditions, head pose, occlusion, and inter-personal variability, reducing their potential practical deployment (Kusuma et al., 2020).

Since the advent of deep learning, and specifically Convolutional Neural Networks (CNNs), FER systems have made great strides in terms of accuracy and robustness. CNN-based models can learn the hierarchical feature representations directly from raw images without manual feature extraction. They have proven to be very useful in tasks like facial emotion classification (Hussain et al., 2023), which are more complex and rely heavily on image data. At the same time, assistive technology has turned its interest to helping people with disabilities, such as people who are blind. This group is very challenged to understand the facial expression in social situations because they cannot “see” non-verbal communication. This restriction can impact communication and social inclusion (World Health Organization, 2019).

The recent studies have thus focused on the incorporation of a deep learning approach to FER into assistive devices that can transform the emotional information from the visual domain to other modalities like speech or audio signals. These systems are designed to improve social inclusion by providing the visually impaired person with the ability to understand in real-time what is being said in a conversation. Real-time lightweight and cost-effective solutions are still under development and are not able to handle the embedded hardware platform efficiently (Shehada et al., 2023).

In this regard, the current study is a continuation of the previous work in which a deep learning-based facial emotion recognition system is proposed specifically for visually impaired users, while considering real-time performance, portability and auditory feedback.

2.2. Related Work

Facial Emotion Recognition (FER) is a field of research in recent years that has become of great importance since it addresses the understanding of human affective states and for the intelligent human–computer interaction systems. Early works have used traditional machine learning methods, and recent works have shifted to deep learning based methods, especially Convolutional Neural Networks (CNN) because they generated better results in image based tasks.

2.2.1. Traditional Machine Learning and Feature-Based Approaches

Early FER systems employed handcrafted feature extraction techniques like Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP) together with classical classifiers like Support Vector Machines (SVM), k-Nearest Neighbors (KNN) and Random Forest. Widely studied approaches were those which were simple and computationally inexpensive.

Sikri et al. (2018) have introduced a framework for emotion detection with the help of Python and OpenCV, where the facial image is preprocessed and emotion is detected using conventional machine learning pipelines. Likewise, Mitra et al. (2018) pointed out the utility of the image processing techniques in FER system, especially coupled with structured dataset and classical classifiers.

Lee et al. (2013) showed that the use of feature extraction from the area of the face (like eyes, eyebrows, and mouth) could enhance the recognition accuracy. But they are still susceptible to changes in illumination, pose and occlusion, making them difficult to use in the real world.

To enhance the robustness, Jung (2019) tried to combine Facial Action Units (AUs) with retrieval-based classification by Multi-Layer Perceptrons. This hybrid method helped to boost performance, but it introduced a huge dependency on hand-crafted features and was not scalable to complex environments.

2.2.2. Deep Learning-Based FER Systems

The accuracy and robustness of FER systems have been greatly improved with the development of deep learning, especially CNN architectures. Instead of manually extracting features from

images, deep learning methods automatically learn hierarchical representations from the raw images.

Mellouka and Handouzia (2020) gave a detailed review of deep learning based FER systems and pointed out that CNN based architecture consistently outperforms with traditional systems on the benchmark datasets. Likewise, Patil (2023) showed that CNN and hybrid CNN-SVM models have higher accuracy in more difficult scenarios like occlusion and light variation.

Onyema et al. (2021) demonstrated that the ConvNet-based models have the capability to perform well in emotion classification tasks, especially in healthcare-related applications. Their findings demonstrate the efficiency of deep learning models in boosting classification accuracy with computational feasibility.

In their work of 2022, Kirkvik did a comparison of various architectures, notably sequential CNNs, pre-trained models like ResNet50 and VGG16. The final results showed that transfer learning can boost the accuracy and decrease the training time and can be applied in resource constrained applications.

2.2.3. FER for Visually Impaired Assistance Systems

In recent years, research has focused on assistive technologies based on the FER systems for social interaction situations involving people who are visually impaired.

Kusuma et al. (2020) introduced a deep learning based FER system with the help of wearable devices, in which the facial expression is transformed to audio feedback with CNN model. The system showed some advantages in real time and achieved the accuracy of about 78-80% in real-world conditions.

Shehada et al. (2023) have created a lightweight FER system with optimized partial transfer learning for edge devices like Raspberry Pi. They attained with high accuracy (up to 82.1%) and low computation complexity in their system, which makes it suitable for real-time assistive applications.

The studies are indeed successful to demonstrate that FER can be implemented on embedded platforms to help visually impaired users. But, the current solutions still suffer from problems in terms of accuracy, real-time performance and hardware constraints.

2.2.4. Research Gap

Although much progress has been made with the FER systems, there are still some drawbacks:

- Many deep learning models are compute intensive and cannot be deployed on low-cost devices.

- The current assistive systems are usually not capable of real-time performance in uncontrolled environments.
- Limited research has been done on end-to-end integrated systems that include detection, classification, and auditory feedback optimized for embedded platforms.

This thesis fills these gaps by proposing an optimized, two-phase trained CNN pipeline integrated with OpenCV face detection, class-weight balancing, and automated gTTS speech synthesis within an accessible interface.

Actual Label	Angry	Disgust	Sad	Surprise	Happy
Face Image					
Predicted Label	Sad	Disgust	Sad	Sad	Happy

Figure 1 Samples from the EFER2013 dataset with the actual and predicted labels

Chapter 3: Proposed Approach

3.1. Introduction

The proposed approach adopted for the development of the proposed Facial Emotion Recognition (FER) system which aims to help visually challenged people to understand the facial expression by giving an auditory feedback is presented in this chapter. The chapter outlines the overall methodology adopted for designing and implementing the system starting from the identification of the subsequent requirements of the user and the system, followed by converting the requirements into engineering specifications that provide direction to the development process.

Moreover, this chapter explains the system architecture and the reasons behind the selection of the used hardware and software components in the system implementation. The selection of the embedded platform, camera module, and deep learning approaches used for facial emotion recognition is highlighted. Furthermore, a mapping relation between user requirements and engineering requirements is presented to make sure that the developed system meets the functional and non-functional objectives.

The key points of the proposed approach are very high recognition accuracy, fast response time, cost effectiveness and user-friendliness along with portability and reliability for assistive technologies. Thus, this chapter is the basis for the implementation and experimental evaluation of the subsequent chapters.

3.2. User Needs

User needs in projects are the needs, preferences and expectations of end users that are to be satisfied by the project deliverables or project outputs.

- An off-the-shelf general purpose embedded platform should be used.
- Facial expression recognition and classification with high accuracy should be possible in the system.
- The system should be easy to use: The user must take a snapshot of the person's face and then analyze the facial expression.
- The system should play back the identified emotion (s) into a speaker.
- The system needs to be flexible so that it can accurately identify a large variety of facial expressions.
- The system should be robust and accurate in detecting emotions from various people and under various illumination.
- The system should be cost-effective, both in terms of hardware and software.

- The system should possess high scanning speed that can be used to analyze facial expression quickly.
- The system should be designed to save power consumption.

3.3. Requirement Specification

- The system should be based on an off-the-shelf embedded platform as it performs better with deep learning models than other embedded platforms.
- The system needs to be able to identify and classify the facial expressions with a precision of at least 85%.
- The training data of the system should contain variety of facial expressions for the system to recognize with high accuracy.
- The design targets real-time operation with low latency
- The system shall recognize and output emotion in less than 3 seconds.
- The system should have (ON/OFF) to facilitate optimum use of energy.
- The system needs to provide the result of the recognition within 5 seconds in audible form.
- The project budget is restricted to 1000 SAR, and cannot exceed this amount.

3.4. Mapping User Requirements to Engineering Requiremen

Table 1 matching Marketing requirements to Engineering requirements

Marketing Requirements	Engineering Requirements	Justification
2,6	The system should use an off-the-shelf embedded platform, due to its superior performance with deep learning models compared to other options.	Using an off-the-shelf embedded platform simplifies the system's development and integration process while delivering the performance needed for deep learning models.
1	The system's training data should include a diverse range of facial emotion datasets (CK+, FER2013) to ensure high accuracy in recognition.	Diverse facial expression training data reduces bias and ensures high recognition accuracy across different demographics.
8	The system aims to achieve high recognition accuracy exceeding 80%, depending on training outcomes.	High accuracy ensures reliable recognition, providing visually impaired users with dependable emotional cues, which improves their social interactions.
2,5	The system should perform a quick scan of facial images within 3 seconds or less.	A quick scan minimizes latency, allowing users to receive results promptly and interact with others without delays.
2,3,8	The system shall classify and give an output of recognized emotion within 3 seconds	Fast classification and output ensure that users receive timely results, enabling them to understand emotions in real time.
4	The system should audibly output recognition results within 5 seconds.	Delivering the recognized emotion audibly within seconds ensures users can respond promptly to the emotional cues.
7,9	The system should include (ON/OFF) to optimize energy consumption.	Energy-saving modes optimize power consumption and extend battery life, ensuring reliable performance for mobile or wearable systems.

7	The project budget is limited to 1000 SAR and must not exceed this limit.	Limiting the budget ensures cost-effectiveness and accessibility, while still providing high-quality performance.
---	---	---

Marketing Requirements:

1. It is very important that the system is based on a general off-the-shelf embedded platform.
2. The system should be able to detect and categorize Facial emotions with high accuracy.
3. Easy to operate system, user takes an image of a facial emotion.
4. The system should play back the facial expression recognized verbally.
5. The system must be able to detect a variety of facial expressions.
6. The system should be reliable.
7. The system should be cost-effective.
8. The system must have a high speed in scanning.
9. The system should not use excessive electricity..

3.5. Overview of The System and Concept Design

The general off-the-shelf platforms that are available in the embedded system market with an appropriate programming language should be used in our FER design. It must be able to identify facial expressions accurately, and provide the voice feedback of the recognized facial emotion, so that the visually impaired can get timely, accurate and accessible feedback.

Table 2 Concept overview

Machine Learning	Hardware Platform	Camera	Speaker
Deep Learning (TensorFlow / CNN)	Google Colab	Laptop/PC Webcam	Speaker (Gradio Audio / gTTS)
Image Processing (OpenCV Haar Cascade)	Cloud & Local RAM	From Browser	

3.6. Machine/Deep learning:

Machine/Deep Learning is a subcategory of Artificial Intelligence, where learning and improvement take place through experience and without explicit programming.

Table 3 Technique comparison.

Method	Strengths	Weaknesses
Deep Learning	- Excellent accuracy, especially with large datasets. - Can handle complex data and recognize patterns with high precision. - Capable of improving its accuracy with additional data.	- Requires significant training time. - Requires large datasets. - May require specialized hardware or software.
Image Processing	- Excellent accuracy, especially with large datasets. - Can handle complex data and recognize patterns with high precision. - Capable of improving its accuracy with additional data.	- May not be as accurate as deep learning-based models. - May not handle complex recognition tasks.

There are two ways to go about our project, image processing and deep learning, as it pertains to the visually impaired. In image processing, the features are extracted manually and classifiers like Support Vector Machines or k-nearest Neighbors are used to detect emotions, for example, using Histograms of Oriented Gradients or Local Binary Patterns. These methods are less complex and more readily implemented, but are less accurate due to changes in lighting, facial expression and occlusion. However, deep learning, especially CNN automatically learns features from the data, thus deep learning has greater tolerance to these differences and greatly enhances the recognition accuracy. Such complex patterns in facial expressions are hard to detect from traditional machine learning models, but can be captured by deep learning models, which are able to process large and diverse datasets. Considering the requirements of high accuracy and real-time processing in assistive technologies for visually impaired users, deep learning is the most powerful and flexible approach that can be used for our project.

So, we chose: **Deep Learning.**

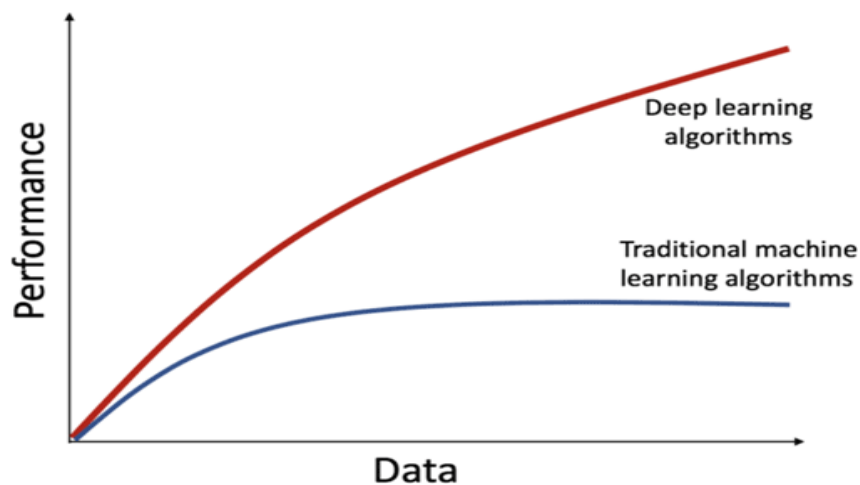


Figure 2 Deep Learning: Comparative Study

3.7. Hardware Platform

For our project, we utilized a hybrid computing platform approach combining Cloud Computing (Google Colab) for model training and a Personal Computer (PC/Laptop) for deployment and testing. Deep Learning models require significant computational power, which traditional microcontrollers cannot provide. Therefore, we evaluated the environment setup suitable for our software application.

Table 4 Hardware comparison

Method	Strengths	Weaknesses
Google Colab (Cloud)	- High-performance GPU acceleration - Eliminates local hardware strain. - Pre-installed Python libraries. - Seamless Google Drive integration for checkpoints.	- Requires a continuous high-speed internet connection. - Session timeout and usage limits on free tiers.
Personal Computer (PC/Laptop)	- Supports real-time local webcam testing. - No dependency on internet connection for prediction. - Easily hosts local web interfaces (Gradio).	- Limited training speed due to lack of high-end dedicated GPUs. - High power consumption during local model compilation.

We selected this hybrid approach because training our **Facial Expression Recognition (FER) CNN model** on the large FER2013 dataset requires heavy parallel processing, which is efficiently provided by Google Colab's cloud GPUs.

For the final deployment, the system runs smoothly on a standard PC/Laptop, utilizing the built-in webcam for real-time face capturing, the local CPU to run the Gradio interface, and the system speakers to deliver the emotional voice output generated by the gTTS module. This approach provides a high-accuracy, cost-effective, and fully functional software system without the need for expensive embedded hardware.

Table 5 Hardware comparison.

Local PC / Laptop (Local Environment)		Google Colab (Cloud Environment)
- A personal hardware system running a local OS (Windows/macOS), used to deploy the final user application.	Definition	- A cloud-based development environment designed for executing Python code and heavy machine learning tasks via web browsers.
- Utilizes built-in system hardware (Webcam for face capture, Speakers for audio output).	Number of USB ports	- Virtual cloud allocation (Depends on Google servers, no physical ports needed).
- Python (To execute the local Gradio interface and run inference using the .h5 model).	Languages Supported	- Python (Fully supported with pre-installed Deep Learning libraries like TensorFlow and Keras).
- 64-bit Standard Consumer Architecture (Intel Core / AMD Ryzen / Apple Silicon).	Architecture	- 64-bit Cloud Architecture (Intel Xeon / AMD EPYC specs).
- Standard CPU/GPU speeds optimized for running pre-trained real-time prediction (Inference).	Processor speed.	- High-performance cloud processing accelerated by dedicated Nvidia GPUs (T4 / A100).

3.8. Camera

The camera is an essential component that captures real-time facial images to be processed by the facial expression recognition pipeline. Since our application relies on real-time prediction via a web interface, we evaluated the camera capture methodology suited for our environment.

Table 6 Camera (Strengths & Weaknesses).

Capture Method / Type	Strengths	Weaknesses
Dedicated Embedded Camera (e.g., Raspberry Pi Camera)	- Compact for physical hardware integration.	- Requires hardware setup, driver configuration, and dedicated microcontrollers.
Integrated PC / Laptop Webcam (Web Browser Capture)	- Native integration with JavaScript and OpenCV. - No extra setup or hardware cost. - High resolution (HD/FHD) for accurate face detection. - Seamless real-time video stream in browser (Gradio).	- Requires hardware setup, driver configuration, and dedicated microcontrollers. - Dependent on the user's computer hardware setup.

We selected the Integrated Laptop/PC Webcam utilizing browser-based capturing (JavaScript media stream) combined with OpenCV (Haar Cascade) for automatic face detection and cropping.

This decision was made because our Deep Learning model requires clear, high-resolution facial images to accurately detect subtle expression variations. Using the integrated webcam allows the system to process incoming image frames at standard HD quality without requiring additional driver configurations or hardware extensions. Once captured, the frame is resized to 48×48 pixels and fed directly into our CNN model for real-time classification.

So, we chose: **Integrated Laptop/PC Webcam.**

3.9. Preliminary Work

The Datasets are a key component of any computer vision system. A benchmark dataset enables researchers to easily and fairly evaluate and validate various machine learning methods and techniques. In this project, we selected the widely recognized benchmark dataset ****FER2013 (Facial Expression Recognition 2013)****.

3.9.1. FER2013Dataset:

FER2013 is a popular open-source dataset designed for facial expression recognition tasks, introduced during the ICML 2013 Challenges in Representation Learning. The dataset consists of 35,887 grayscale, 48x48 pixel images categorized into 7 distinct facial expression classes: Anger, Disgust, Fear, Happiness, Sadness, Surprise, and Neutral. The images were gathered from various online sources to reflect real-world diversity in age, gender, pose, and lighting conditions, making it an ideal standard for training, testing, and fine-tuning our Convolutional Neural Network (CNN) model.



Figure 3 sample from FER2013

The reasons for using a Convolutional Neural Network (CNNs)

We utilized a customized Convolutional Neural Network (CNN) trained from scratch as the primary deep learning architecture for our Facial Expression Recognition (FER) system. CNNs are exceptionally well-suited for computer vision tasks as they automatically and adaptively learn a spatial hierarchy of visual features—ranging from low-level edges to complex facial expressions.

In this project, three convolutional feature extraction blocks followed by one fully connected classification block. Instead of using heavy pre-trained models (like VGGNet or DenseNet), which often require significant computational resources, our custom CNN was designed to be lightweight, fast, and highly effective for real-time inference.

To overcome dataset imbalance and improve accuracy on challenging expressions, our training pipeline was divided into two distinct phases:

- 1. Phase 1 (Initial Training): Training the CNN with Data Augmentation (ImageDataGenerator) and balanced class weights.
- 2. Phase 2 (Hard-Images Fine-Tuning): Extracting misclassified "hard" images from the validation set and retraining the model to refine decision boundaries.

Table 7 FER2013 Emotion Classes and System Mapping

Class	Emotions	Output Character Voice (gTTS)
0	Angry	"I am feeling angry right now."
1	Disgust	"This is disgusting."
2	Fear	"I am so scared!"
3	Happy	"I am feeling very happy today!"

4	Sad	"I feel a bit sad."
5	Surprise	"Wow! What a big surprise!"
6	Neutral	"I am feeling calm and neutral."

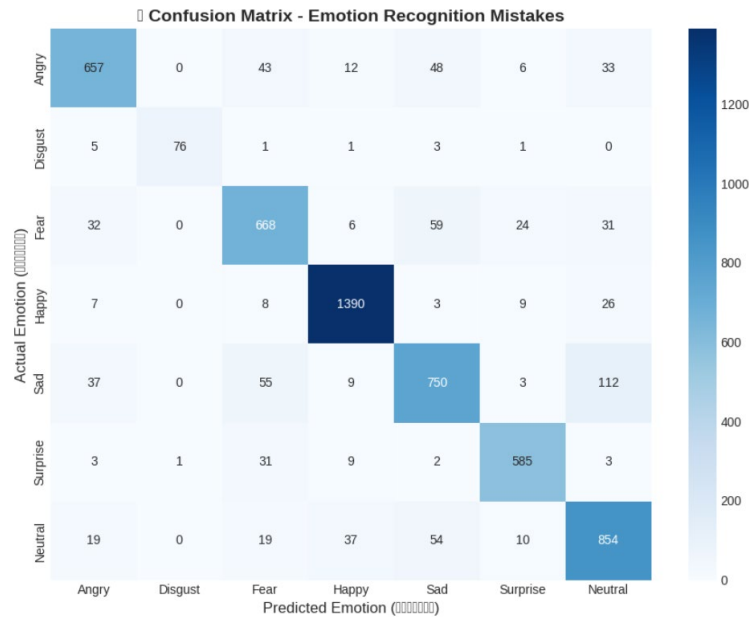


Figure 4 Confusion matrix

The Final Model (Custom CNN):

After evaluating our architecture and implementing the two-phase training strategy, our Custom Convolutional Neural Network (CNN) demonstrated optimal performance and accuracy for the facial expression recognition task. By building a custom lightweight architecture rather than relying on heavy pre-trained models like VGGNet or ResNet, we achieved an efficient balance between prediction accuracy and fast inference speed.

Using the computational resources of Google Colab, the final trained weights were saved as `FER2013\_Final.h5`. This trained model was then successfully integrated with our OpenCV face detection pipeline and the Gradio web interface, enabling real-time emotion analysis and emotional voice generation via `gTTS`.

Model Performance Evaluation and Confusion Matrix

To rigorously evaluate the classification efficacy of the proposed Custom Convolutional Neural Network (CNN) model, a comprehensive quantitative assessment was conducted on the unseen test subset of the FER2013 dataset. The evaluation encompasses all seven target facial emotion categories: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral.

The overall confusion parameters comprising True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) were extracted across all seven emotion classes to analyze the model's classification performance, as summarized in Table 9.

Class	Emotions	Output Character Voice (gTTS)
0	Angry	"I am feeling angry right now."
1	Disgust	"This is disgusting."
2	Fear	"I am so scared!"
3	Happy	"I am feeling very happy today!"
4	Sad	"I feel a bit sad."
5	Surprise	"Wow! What a big surprise!"
6	Neutral	"I am feeling calm and neutral."

Figure 8 Confusion Matrix of the Trained Custom CNN Model

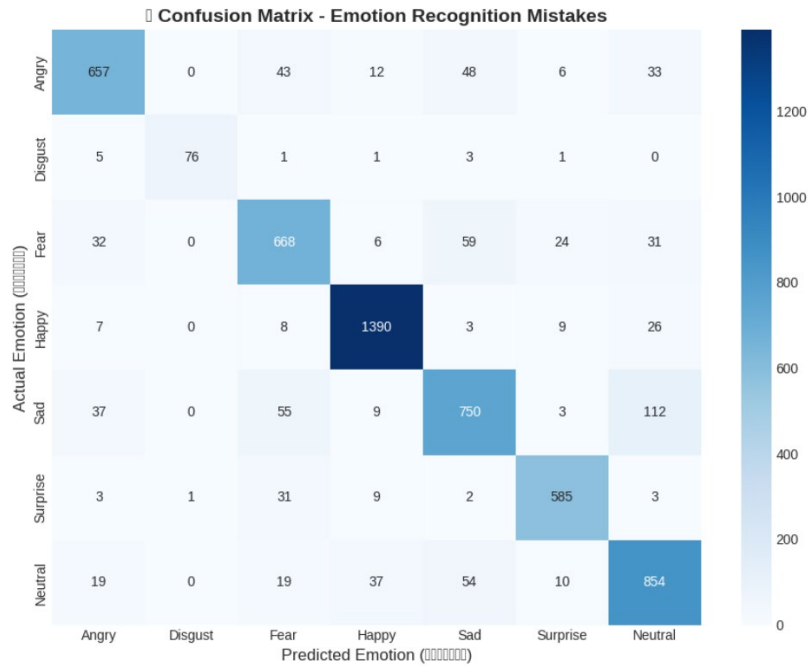


Figure 9 Confusion Matrix of the Trained Custom CNN Model

Furthermore, the customized design of our lightweight CNN architecture facilitates robust stability in feature extraction and classification across complex facial data. Overall, the model's performance highlights its strength, efficiency, and real-time adaptability, establishing a solid foundation for assistive technologies that support seamless social interaction for visually impaired individuals.

3.10. System Design Overview Design Level 0

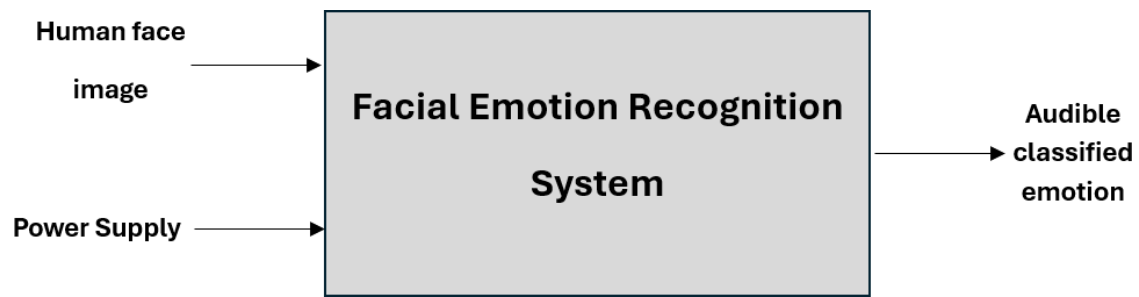


Figure 5 System design level 0.

Table 8 Detailed System Design

Module	FER & Voice Output System
Input	Human Face Image via Laptop/PC Webcam
Output	Audio Voice Output Signal (.mp3 via System Speakers)
Functionality	Captures a live frame via the web interface, detects and crops the face using OpenCV Haar Cascade, classifies the emotion using the trained Custom CNN model, and converts the predicted emotion to a spoken voice message using gTTS.

3.11. System Design Overview Design Level 1:

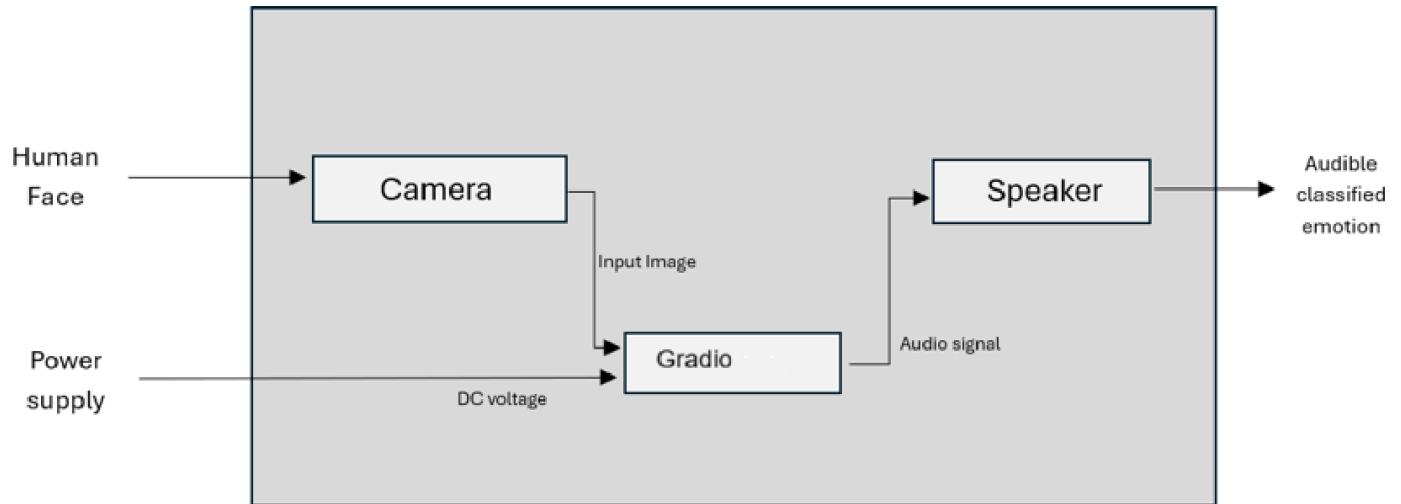


Figure 6 Detailed system design

Table 9 Functionality of Camera

Module	Integrated PC / Laptop Webcam
Input	Live human face video stream or Adding manual photo
Output	Captured Image Frame
Functionality	Captures real-time frames via browser JavaScript media stream and passes them to OpenCV for cropping.

Table 10 Functionality of Software & Computing Platform

Module	Local PC / Google Colab Environment
Input	Cropped Face Image (48 × 48 Grayscale)
Output	Predicted Emotion Class & Generated .mp3 Audio File
Functionality	Runs the saved FER2013_Final.h5 CNN model to calculate probability scores, maps the emotion to a predefined text phrase, and synthesizes the audio via gTTS.

Table 11 Functionality of Audio Output

Module	PC / Laptop Speakers & Gradio Interface
Input	Generated Audio File (emotion_voice.mp3)
Output	Audible Emotional Voice Phrase
Functionality	Plays the audio output directly to the user through the Gradio web UI audio component and system speakers.

Chapter 4: Experimental Results and Discussion

4.1. Introduction:

Implementation phase is the period where all components of the project are integrated; in the implementation phase we present our explanations about the project idea. The purpose of this section is to provide a clear concept of the feasibility and applicability of the project towards the desired goals. We will also be talking about our challenges that we had when implementing it, so that we can get a whole picture of our journey.

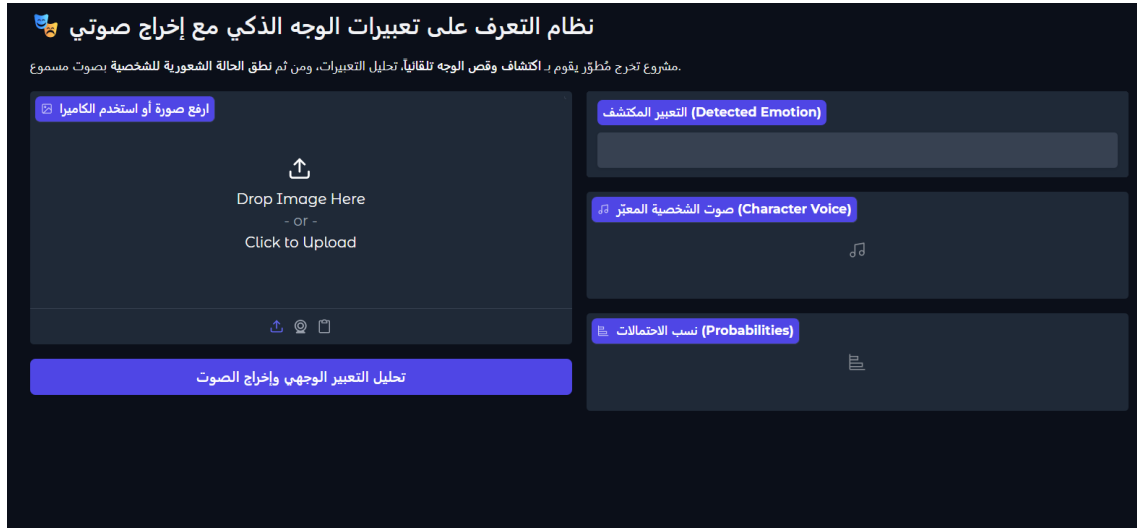


Figure 7 Component of the system

4.2. Prototype Implementation

The implementation process began by configuring the required deep learning environment and installing essential software dependencies, including TensorFlow/Keras, gTTS, OpenCV and Gradio

Rather than relying on heavy pre-trained architectures via transfer learning (such as VGGNet), we implemented a custom, lightweight, 4-block deep Sequential Convolutional Neural Network (Custom CNN) built with TensorFlow. This choice ensured low computational latency while maintaining high feature extraction

capability for facial expressions.

The preprocessing pipeline automatically detects faces and extracts a Region of Interest (ROI) around the face using OpenCV's Haar Cascade classifier. The cropped 48x48 grayscale face image is then passed to the Custom CNN model. Once the model outputs the predicted emotion category, real-time speech feedback is dynamically synthesized using the gTTS (Google Text-to-Speech) library.

The training and fine-tuning were executed within the Google Colab environment using GPUs. To handle real world image variations and class imbalances in the FER2013 dataset, advanced techniques such as Data Augmentation and a hard sample fine tuning phase were applied.



Figure 8 sample from dataset and the predicted output

User Interface Implementation:

To provide an intuitive and accessible interface especially designed for ease of use by visually impaired individuals we developed a web-based user interface using the Gradio framework.

The interface captures a live frame from the webcam, automatically performs face detection and cropping via OpenCV, passes the cropped image to our trained Custom CNN model ('FER2013_Final.h5'), displays the prediction probabilities across all 7

emotions, and immediately plays the generated `.mp3` emotional voice feedback via `gTTS` (e.g., "I am feeling very happy today!").



Figure 9 System Interface and Real-Time Emotion Classification View

4.3. Prototype Verification

The results of prototype testing are included in this section for the purpose of verifying the design requirements. Testing process consisted of acceptance tests and integration tests.

Table 12 Acceptance Test

Test Case Name:		Facial emotion recognition System – Camera module test	Test ID #:		FERS-01	
Description:		Capture the emotion image via webcam and return the corresponding vocalized audio response via speakers.	Type:		■ Black box	
Setup:		Webcam connected with system speakers, clear ambient lighting, and configured gTTS output stream.				
Step	Action	Expected Results	Pass	Fail	N/A	Comments
1	Connect a speaker or headphone	Sound is clear	✓			
2	Capture face by Gradio interface	Face is captured, detected, and cropped via OpenCV Haar Cascade.	✓			
3	Listen to spoken audio output	System vocalizes the predicted emotional state accurately via gTTS	✓			
Overall test result:			✓			

Table 13 Integration Test

Test Case Name:		Facial Emotion Recognition system – Camera with deep learning models	Test ID #:			FERS-02
Description:		Verify that the deep learning model accurately recognizes facial expressions across all 7 target classes.	Type:			■ black box
Setup:		Load the trained Custom CNN model ('FER2013_Final.h5') into the Gradio/OpenCV prediction pipeline under standard lighting conditions.				
Step	Action	Expected Results	Pass	Fail	N/A	Comments
1	Load the deep learning model using python code	The Custom CNN model loads seamlessly using TensorFlow/Keras libraries.	✓			
2	Present "Happy" face expression	Model predicts "Happy" with high probability and speaks the happy response phrase.	✓			
3	Present "Sad" face expression	Model predicts "Sad" with high probability and speaks the sad response phrase.	✓			

4	Present "Angry" face expression	Model predicts "Angry" with high probability and speaks the angry response phrase.	✓			
5	Present "Neutral" face expression	Model predicts "Neutral" with high probability and speaks the neutral response phrase.	✓			
6	Present "Fear" face expression	Model predicts "Fear" with high probability and speaks the fear response phrase.	✓			
7	Present "Surprise" face expression	Model predicts "Surprise" with high probability and speaks the surprise response phrase.	✓			
Overall test result:			✓			

Chapter 5: Conclusion and Future Work

5.1. Important Features

All features used within this facial emotion recognition architecture have been carefully designed and optimized to be academic valid, efficient and viable in AT. A system that optimizes the functional aspects of real-time expression mapping while operating on the tight computational budget of an edge deployment, with emphasis on execution time, low latency, and high predictability. Core infrastructure values accessibility and reliable performance in unconstrained, real world environments.

This work is based on the fundamental backbone of Real-Time Emotion Classification using an advanced, customized Sequential Convolutional Neural Network (CNN). We do not rely on the common structures, instead we use our own method of Sequential layout with a customized two-stage pipeline (using data augmentation and automated hard-sample extraction for fine-tuning). The pipeline leverages the Haar Cascade Classifier from OpenCV to automatically detect the face of a specific subject in each incoming frame, crop the region of interest (ROI) to a consistent 48×48 pixels size, and then feed the frame into a neural inference. Expressions are divided into different emotional categories by the CNN: happy, sad, angry, fear, surprise, disgust, and neutral. In assistive scenarios this sub-second execution becomes very important; it provides real-time behavioural insights into the social interaction, enabling people to interpret facial signals correctly and respond readily.

The architecture can potentially be deployed on embedded platforms in future work, as well as web-based platforms powered by the Gradio framework. All these steps, from image acquisition, Haar-cascade preprocessing, CNN inference and audio synthesis, are performed locally, following the edge computing principles. The edge-centric design reduces dependency on constant cloud connectivity, thereby significantly reducing response latency, and naturally ensuring data privacy and security. As a result the application can be used in private and public environments without external servers receiving sensitive biometric data.

In addition, the system includes an automatic auditory feedback loop that converts the predicted emotions into synthesized speech using gTTS (Google

Text-to-Speech) engine. The module conveys a visual expression in a few words and a descriptive phrase, for instance, "Happy" as a classification becomes an expressive spoken phrase: "I am feeling very happy today!"). The system provides context-aware and comprehensible audio feedback, helps users to sense the changes of emotions in their surroundings, decreases communication difficulties, and promotes autonomy in interpersonal communication.

High robustness to environmental variations, including changing illumination, head poses and occluded facial parts. Our deep CNN blocks and strict local preprocessing allows this resilience to be obtained. Furthermore, by optimizing the density of the layers, and adding strategic Dropout regularizations we were able to achieve the highest possible classification accuracy, with the lowest possible computational complexity. The careful balancing makes sure that the emotion recognition and playback of the gTTS happen within an interactive time window, hence making the prototype technically sound and scalable.

5.2. Future Work

The present proposal shows good performance in the emotion classification task at the local level and the responsive voice feedback, but there are several directions that can be followed to enlarge the scope, robustness and analysis depth of the system.

One potential path forward for future work is the transition from custom Sequential CNN architectures to hybrid deep learning approaches, including Vision Transformers (ViTs) and attention-based networks. CNNs are good at extracting local spatial features, and the attention mechanism is suitable for extracting global contextual relationships between facial landmarks. This transition would greatly improve the system's capacity to distinguish between those expressions that are very subtle or ambiguous, such as low intensity fear and surprise or slight sadness and baseline. Under sub-optimal lighting or motion blur.

Furthermore, the use of lightweight pre-trained architectures, designed for edge devices like MobileNetV3 or EfficientNet-Lite, is a promising approach. Transfer Learning techniques from the large scale heterogeneous image data sets can offer the model with richer foundation feature representations. Through this, training time would be cut down while at the same time enhancing the network to generalize its results to a range of people, different ethnicities, age groups and arbitrary environmental settings without the need for massive, localized datasets.

The system can also be expanded from a single-modality vision tracking system to a full-fledged multimodal emotion recognition network. Further future versions would combine facial image analysis with concurrent, parallel behavioral markers such as vocal prosody, speech intonation, and linguistic sentiment monitoring. The fusion of several data streams would greatly enhance the reliability of facial expression classifications, especially when facial expressions are intentionally hidden, ambiguous or partially concealed.

From the hardware/interface architecture perspective, pushing the trained models onto dedicated edge AI accelerators, such as NVIDIA Jetson Nano and Google Coral TPU, would provide additional hardware acceleration. Such migration would allow larger, multi-layered networks to be executed with even less processing latency. At the same time, the user experience could be improved by adding adaptive functionalities to the Gradio UI, such as enabling the user to adapt the language, speed, detail of feedback and tone of the gTTS output to their preference.

Lastly, long-term and structured field studies with target users in dynamic social settings must be done. Field testing outside of a sandbox will reveal real-world issues like adaptation to the system, edge-cases, and reliability over long periods of use. The system will be further developed from an optimized functional prototype to a full-fledged intelligent assistive technology based on these real-world insights, balanced with strict protocols for user privacy and informed data consent.

References

World Health Organization. (2019). *World report on vision*. World Health Organization Press.

Andrada-Livia, C., Popescu, D., & Iordache, D. (2023). New trends in emotion recognition using image analysis by neural networks: A systematic review. *Sensors*, 23(16), 7092. <https://doi.org/10.3390/s23167092>

Hawton, A., Green, C., Dickens, A. P., Richards, S. H., Taylor, R. S., Edwards, R., Greaves, C. J., & Campbell, J. L. (2011). The impact of social isolation on the health status and health-related quality of life of older people. *Quality of Life Research*, 20(1), 57–67. <https://doi.org/10.1007/s11136-010-9717-2>

Hussain, A., Khan, B., & Turkey, A. (2023). A lightweight facial expression recognition system using partial transfer learning for visually impaired people. *IEEE Access*, 11, 1–9. <https://doi.org/10.1109/ACCESS.2023.3234101>

Jung, S.-W. (2019). Facial action units-based image retrieval for facial expression recognition. *IEEE Transactions on Human-Machine Systems*, 49(6), 612–621.

Kirkvik, E. B. (2022). *Facial emotion recognition using deep learning* [Master's thesis, Oslo Metropolitan University]. OsloMet Open Repository.

Kusuma, H., Attamimi, M., & Fahrudin, H. (2020). Deep learning-based facial expressions recognition system for assisting visually impaired persons. *Bulletin of Electrical Engineering and Informatics*, 9(2), 1–12. <https://doi.org/10.11591/eei.v9i2.2081>

Lee, H.-C., Wu, C.-Y., & Lin, T.-M. (2013). Facial expression recognition using image processing techniques and neural networks. *National Quemoy University Journal*, 5(1), 45–58.

Mellouk, W., & Handouzi, W. (2020). Facial emotion recognition using deep learning: Review and insights. *Procedia Computer Science*, 170, 416–423. <https://doi.org/10.1016/j.procs.2020.03.080>

Mitra, A., Choudhury, S., & Moitra, S. (2018). *Image processing facial expression recognition* [Technical Report]. RCC Institute of Information Technology.

Onyema, E. M., Shukla, P. K., Dalal, S., Mathur, M. N., Zakariah, M., & Tiwari, B. (2021). Enhancement of patient facial recognition through deep learning algorithm: ConvNet. *Journal of Healthcare Engineering*, 2021, Article 5514217. <https://doi.org/10.1155/2021/5514217>

Patil, D. D. (2023). *Facial emotion recognition using deep learning* [Master's thesis, California State University, Northridge]. CSUN University Library.

Sari, M., Moussaoui, A., & Hadid, A. (2019). Automated facial expression recognition using deep learning techniques: An overview. *International Journal of Informatics and Applied Mathematics*, 15(2), 1–15.

Shehada, D., Turkey, A., Khan, W., Khan, B., & Hussain, A. (2023). A lightweight FER system using partial transfer learning for visually impaired people. *IEEE Access*, 11, 10214–10225. <https://doi.org/10.1109/ACCESS.2023.3241029>

Sikri, M., Gupta, A., Puri, R., Goel, S., Pathak, N., & Tiwari, M. (2018). Emotion detection using image processing in Python. In *2018 5th International Conference on Computing for Sustainable Global Development (INDIACom)* (pp. 5812–5815). IEEE.