UNIVERSITI TEKNOLOGI MARA

CUSTOMER EMOTION RECOGNITION APPLICATION FOR SUPERMARKETS USING CONVOLUTIONAL NEURAL NETWORKS

DANIAL THAQIF BIN MOHAMED FIKRI

BACHELOR OF COMPUTER SCIENCE (HONS.)

JANUARY 2025

Universiti Teknologi MARA

CUSTOMER EMOTION RECOGNITION APPLICATION FOR SUPERMARKETS USING CONVOLUTIONAL NEURAL NETWORKS

DANIAL THAQIF BIN MOHAMED FIKRI

Thesis submitted in fulfillment of the requirements for Bachelor of Computer Science (Hons.) College of Computing, Informatics and Mathematics

JANUARY 2025

SUPERVISOR APPROVAL

CUSTOMER EMOTION RECOGNITION APPLICATION FOR SUPERMARKETS USING CONVOLUTIONAL NEURAL NETWORKS

By

DANIAL THAQIF BIN MOHAMED FIKRI
2023185071

This study was prepared under the supervision of the project supervisor, Ts. Dr. Razulaimi Bin Razali. It is recognized as fulfilling part of the criteria for a Bachelor of Computer Science (Hons.) from the College of Computing, Informatics and Mathematics.

Approved by

……………….................…

Ts. Dr. Razulaimi Bin Razali,
Project Supervisor

JANUARY 31, 2025

STUDENT'S DECLARATION

I declare that the work in this project of Customer Emotion Recognition Application for Supermarkets Using Convolutional Neural Networks was carried out in accordance with the regulations of University Technology MARA. It is original and is the results of my own work, unless otherwise indicated or acknowledged as referenced work. This thesis has not been submitted to any other academic institution or non-academic institution for any degree or qualification.

…………………………………………………………

DANIAL THAQIF BIN MOHAMED FIKRI,

2023185071

JANUARY 31, 2025

ACKNOWLEDGEMENT

Firstly, I would like to express my gratitude to God for allowing me to continue my studies and finish this long and challenging journey successfully. I would also like to thank my supervisor, Ts. Dr Razulaimi Bin Razali, for guiding me until the completion of this project.

Not to be forgotten also to the madam CSP600 subject that is Dr. Sharifalillah Binti Nordin, who always gives me the right guidance so that I can complete this project according to the set time. Without them, I still have doubts about the full report and the system.

Special thanks to my colleagues and friends for helping and giving suggestions on my project. Finally, a big thank you to my dear parents for their dedication to educating me. Without their education and encouragement, I would not have been able to complete it. Alhamdulillah.

ABSTRACT

This project focuses on the design of a Customer Emotion Recognition Application (CERA) for supermarkets to increase real-time emotion identification by incorporating the Convolutional Neural Network. The study aims to identify suitable algorithms for detecting customer emotions from facial reactions, develop a functional CERA application based on facial recognition using CNNs, and evaluate the systems accuracy and effectiveness. A dataset comprising 35,887 images from Kaggle was preprocessed to 48x48 grayscale for computational efficiency and categorized into seven emotions: Happy, Sad, Angry, Fearful, Disgusted, Surprised, and Neutral. The system was developed using the waterfall model, which included requirement analysis, system design, implementation, and testing. When examining the literature, different deep learning models were discussed as follows: CNNs were preferred for emotion recognition since they can identify spatial hierarchies and translate invariance. Major system units involve real-time facial image capturing, CNN-based emotional recognition, and feedback provision to supermarket employees. The model's accuracy was approximately 83% and the model passed numerous tests across six categories and is realistic for practical use. These findings demonstrate how the system can monitor and evaluate customers' sentiments and subsequently offer necessary directions to the staff to improve their performance. Also, the project is focused on ethical concerns to acquire data, and the protection of privacy to avoid misuse of technology. These concerns in the future are such areas as, the improvement of the model accuracy, expansion of accommodated demographic groups of consumers, and integration of the system into already established supermarket systems. This paper shows the importance of emotion recognition technology as being applied and implemented using practical and concrete retail approaches through innovative and efficient machine learning techniques.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

The context and justification for the research are given in this chapter It first emphasizes the issue statement that resulted in the research then provides specifics of emotional identification and analysis. This chapter next covers the objectives and aims before underlining the extent and importance of the research. At finally, the summary of a chapter closes it.

1.1. BACKGROUND OF STUDY

Improving supermarket services requires understanding of the capacity to respond to consumer emotions. Using artificial intelligence and machine learning among other technologies, customer emotional recognition systems (CERS) offer real-time face expression analysis. Deep neural networks (DNN) and generic machine learning most notably in computer vision applications including emotion analysis (Malik et al., 2021) have dramatically impacted daily life.These advances are applied in this work using convolutional neural networks (CNN). Including CERS into supermarket architecture will enable businesses to better grasp consumer attitudes, hence improving customer loyalty and happiness.

However, successful application depends on privacy issues and addressing guaranteeing algorithm correctness over a range of demographics. Apart from that, datasets like these are essential for the advancement of human emotional recognition research and greatly extend our knowledge in this domain (Krishna et al., 2013).

Due to their expansion of CERS, supermarkets now have a possible means to personalise experiences, solve issues fast, and build lifelong relationships with their consumers.

1.2. PROBLEM STATEMENT

Supermarkets still struggle to correctly measure and react to client emotions in real-time, despite the retail industrys rising emphasis on improving customer experiences. Conventional techniques for gathering consumer input via questionnaires and surveys are frequently biased and backward, providing insufficient understanding of consumer attitudes. Additionally, it may result in an overflow of data. Gathering a lot of comments could cause data overload and complicate the identification of relevant insights without suitable tools and procedures (Tara Ramroop, 2024).

Moreover, some of the stores lacked knowledge on how to handle consumer experience. Supermarkets may use creative ideas to make the shopping experience better and provide more unforgettable surroundings. This might entail improving the physical surroundings, teaching employees in emotional intelligence, and applying technology to better grasp and handle client feelings (Theses & Lawless, 2014).

Apart from that, one of the problems is several of the cashier's lacked awareness of the basic feeling. One must first grasp the basic emotions influencing client experience. They cannot cause the consumer to get annoyed or let down. Once a client moves from annoyance to disappointment, it can be challenging to rebuild confidence since disappointment can lead to a long-term unfavourable relationship with the company (Vit Horky, 2019).

Finally, there is a dearth of efficient instruments meant to identify and evaluate consumer emotions in the supermarket setting. This disparity emphasises the need of the development and application of Customer Emotion Recognition Systems in supermarkets to enable real-time detection and analysis of customer emotions, so enabling tailored services and finally improving general customer satisfaction and loyalty.

1.3. PROJECT OBJECTIVES

This project aims to create a system using machine learning or deep learning methods capable of detecting client emotions. Therefore among the goals are:

a. To identify suitable algorithms that detect customers emotions based on facial reactions.
b. To develop customer emotion recognition systems application based on facial recognition using Convolutional Neural Networks.
c. To evaluate the accuracy of customer emotion recognition using Convolutional Neural Networks.

1.4. SCOPE OF STUDY

Design, development, and application of a Customer Emotion Recognition System (CERS) especially appropriate for supermarkets will be the key goals of this project.

Included in the scope are looking at and choosing suitable technology, including face recognition algorithms to precisely identify and evaluate client emotions in real-time, therefore enabling analysis. The research will also assess the CERSs reliability and efficiency throughout a spectrum of ethnicities and cultural backgrounds so assuring its appropriateness in numerous types of supermarket environments.

The scope also includes addressing customer data collecting and storage privacy concerns as well as seamless integration of the CERS into the design of the present supermarket.

Gathered via Kaggle, the dataset consists of 5000+ face images split into many emotional categories.

1.5. SIGNIFICANCE OF STUDY

This study is important for someone who wants to understand consumer emotions during their shopping encounters and use the advised strategy to improve their interactions with supermarket employees. By means of improved customer service, this endeavour will help the store boost customer happiness. The research will precisely identify client emotions via facial expression recognition by using deep learning and machine learning approaches. Moreover, this study helps to progress image analysis and deep learning, so enabling future developments in artificial intelligence-based face emotion evaluations.

1.6. SUMMARY

This chapter summarises the above mentioned background of the research, problem statement, goals, scope, and project relevance. Finally, this study aims to present a basic and appropriate method based on facial expressions that detects individualss emotio ns.

CHAPTER 2 LITERATURE REVIEW

This chapter explores previous studies and technologies related to emotion recognition, with an emphasis on deep learning approaches like CNNs. It compares various methods, highlights the strengths of CNNs, and identifies gaps in existing research. These insights provide a strong basis for the projects design and implementation.

2.1. CUSTOMER'S EMOTION IN SUPERMARKET

A supermarket consumer is one who visits a store to purchase household goods and food. They come from many walks of life, from many ages, from many origins, from diverse buying patterns. While some customers carefully schedule their meals and create exhaustive buying lists, others browse the stores more haphazardly (Thomas & Garland, 2004). Whichever their approach, supermarket patrons are vital for the running of these stores. They boost revenue and ensure the company stays running.

Emotion is a multifarious fabric composed of many threads. It covers feelings like intense despair or a flash of pleasure. It also alters our physiology; for Moreover, emotions often lead to certain actions, such a grin covering our face in delight or a frown carving itself in displeasure. These interacting components provide a strong force that shapes our daily contact with our surroundings. Emotions can be fleeting, That is, the shock of a loud noise or long-lasting—that is, the great anguish of losing someone you know. Essential to the human person, they mould our ideas, choices, and actions (Dennison, 2023).

One interesting issue is knowing consumer sentiments in supermarkets. Retail environments, particularly supermarkets, where many approaches are used to affect consumer behaviour, have much research on the psychology of this behaviour. For example, the way the store is laid, the music is played, and even the product placement is meant to gently prod consumers emotions and support buying choices (John Loeppky, 2022). The process of making decisions involves emotions in great part. What consumers decide to purchase might depend on the comfort of known brands, the thrill of discovering new things, or the gratification of getting a good bargain.

2.2. DEEP LEARNING – FACE EMOTION RECOGNITION

An crucial component of human-computer interaction, face emotion recognition has been much advanced by deep learning. One of the main challenges in emotional detection is controlling factors such posture changes, uneven illumination, and face accessories. For conventional methods of emotion detection, the combined improvement of feature extraction and classification has shown to be difficult. Researchers turned to deep learning techniques, now extensively employed in categorisation applications in order to get around this problem. (Chowdary et al., 2023)

There isnt a uniform technique or approach for facial recognition. Facial recognition is technology whereby images or videos automatically identify faces. Humans can readily understand faces; computers must either manually code them or identify patterns from data. Facial recognition systems operate on facial descriptors—also called faceprints—derived from these patterns. (Kroll, 2022)

Facial recognition of emotions detects a persons face first using image processing or video sequences then identifies their emotions. Related research applied several techniques to raise the accuracy of emotional recognition. Reducing the time spent on picture processing, however, depends much on the efficiency of the data needed to identify a persons emotion (Sarmiento et al., 2018).

2.2.1. CONVOLUTIONAL NEURAL NETWORKS (CNNs)

Facial emotion recognition makes extensive use of convolutional neural networks (CNNs) (Qu et al., 2023; Sahana et al., 2023). CNNs are a type of deep learning model that shines in handling grid-based data, including images. From input data, they want to automatically and adaptably learn spatial hierarchies of features (Qu et al., 2023).

Suggested is a new method, Facial Emotion Recognition using Convolutional Neural Networks (FERC). Built on a two-part CNN, the first eliminates the background from the image and the second concentrates on face feature vector separability. Regarding emotional recognition, our approach achieved 96% accuracy. The FERC model distinguishes the five different kinds of regular facial expressions by means of an expressional vector (EV). With each iteration the last layer of the perception modulates the weights and exponent values, so the two-level CNN runs in sequence. FERC improves accuracy unlike widely used methods using single-level CNN by differentiating it. Combining pre-trained networks such Resnet50, vgg19, Inception V3, and obile Net with transfer learning is another approach. Fresh fully connected layers suitable for the job are added while the pre-trained ConvNets completely linked layers are eliminated. One can train just recently added layers to update the weights. (Mehendale, 2020)

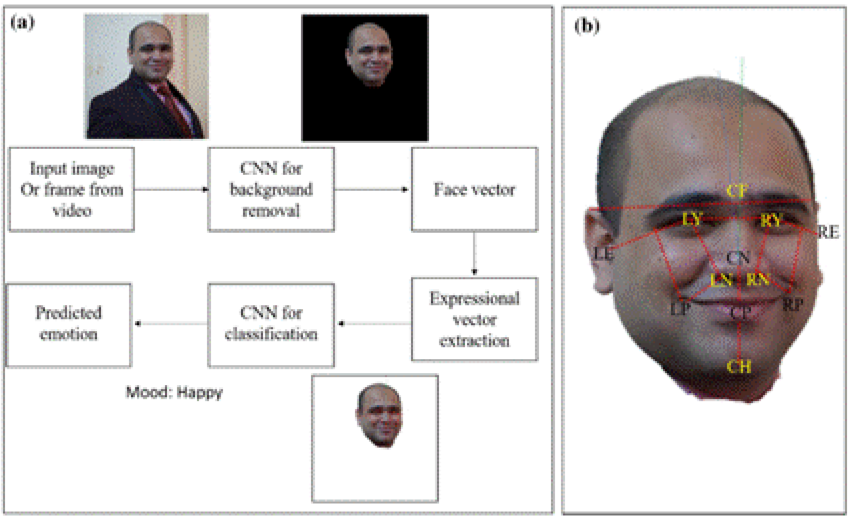


Figure 2.1 The example of FERC is built on a two-part CNN

Two distinct single CNN designs have been assessed: the first depends on the quite conventional LeNet-5 architecture, which is observed for simplicity and efficiency in early picture identification tasks. Combining convolutional and subsampling layers with fully connected layers helps LeNet-5 to be basic picture categorising. Inspired by the VGG architecture, known for its deep and homogenous convolutional layers the second design is. To capture more specific spatial properties, the VGG-inspired model employs deeper depth and smaller filter sizes, hence producing a stronger framework for challenging photo recognition tasks. Both designs have certain benefits; their quality depends on their efficacy in face expression identification. (Moravčík & Basterrech, 2022)

These developments notwithstanding, the CNN models layer choice is still a challenging process that could result in inferior performance. Constant research so seeks to raise the accuracy and efficiency of these models. (Sahana et al., 2023)

2.2.2. RECURRENT NEURAL NETWORKS (RNNs)

By examining facial expressions, recurrent neural networks (RNNs) have been quite successful at recognising emotions in films (Kahou et al., n.d.; Yin et al., 2023).

Designed to find trends in data sequences including text, handwriting, spoken language, and genomes among other things. Furthermore, an RNN is a type of artificial neural network. RNNs create "loops" in the network allowing data to be transferred between steps in the sequence, unlike traditional neural networks which only analyse input data in one way. (Kahou et al., n.d.)

RNNs offer a pleasing structure for information propagation along a sequence in the field of face expression identification by hiding representation using an always valued layer. Analysing video data notably depends on this as information typically needs to be averaged throughout a sequence of frames with different lengthst to arrive at a categorisation outcome. (Kahou et al., n.d.)

Samira Ebrahimi Kahou et al. proposed a hybrid CNN-RNN architecture that outperformed a previous CNN technique using temporal averaging for aggregation (Kahou et al., n.d.). This is for analysis of facial expressions. Submitted for the 2015 Emotion Recognition in the Wild (EmotiW) Challenge (Kahou et al., n.d.) , the system.

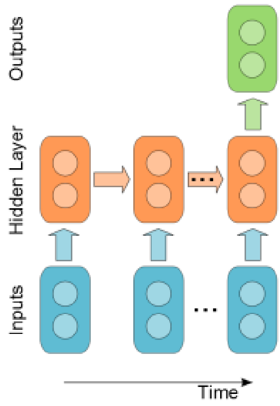


Figure 2.2 Structure of recurrent neural network.

Like the diagram presented in the referenced study (Kahou et al., n.d.), this one demonstrates a Recurrent Neural Network (RNN) topology for facial emotional identification in films. Here the preprocessed face characteristics of every video frame enter the network. To record how expressions change over time, the RNNs hidden layer uses memory cells to analyze these characteristics sequentially. The hidden state changes with each frame, forecasting the mood of the film based on past data. By using this technique, the RNN can now recognize subtle emotional shifts in films, surpassing the capabilities of static pictures.

Another paper by Yida Yin et al. used ADFES collected movies to show a CNN-RNN-based emotion identification system. The results were displayed in an arousal-valence space instead of a specific mood assignment. They also created PN accuracy, a related new performance indicator differentiating positive from negative emotions. (Yin et al., 2023)

Particularly with video data included, the results of these tests show how well RNNs discern emotions from facial expressions. RNNs are ideal for this usage as they can manage sequential data (Kahou et al., n.d.; Yin et al., 2023).

2.2.3. GENERATIVE ADVERSARIAL NETWORKS (GANS)

Generative Adversarial Network (GAN) use in facial emotion recognition has demonstrated positive results (Khemakhem & Ltifi, 2023; Rani et al., 2024). Comprising two components (a generator and a discriminator) a

GAN is a form of deep learning system. The generator generates synthetic images; the discriminator seeks to differentiate real from false ones. This adversarial process produces ever more realistic generating images.

In the scope of facial emotion recognition, GANs might be used to generate synthetic face pictures with a range of emotions, hence improving the performance of emotional identity models (Rani et al., 2024). This is quite beneficial for handling class imbalance in facial expression datasets, in which some emotions may have less samples (Rani et al., 2024). GANs can assist create a more balanced dataset and raise recognition accuracy (Rani et al., 2024) by producing synthetic pictures reflecting emotions less usually expressed..

Using a deep learning model with GAN-based augmentation (Rani et al., 2024), one study especially enhanced performance on the FER-2013 dataset. Lowering the individual influence of identity-related characteristics (Khemakhem & Ltifi, 2023) a Neural Style Transfer Generative Adversarial Network (NST-GAN) was used using a novel technique to extract identifying information from face photos. The NST-GAN reduces the effect of the persons identify by shifting the expression information from the input image to a synthetic identity so stressing the emotion (Khemakhem & Ltifi, 2023).

Though they enhance face expression detection, GANs have constraints. One should keep in mind this. For instance, the synthetic photo quality might significantly affect the performance of the emotional recognition model. Furthermore demanded by GAN training is a modification of the model parameters, which could be computationally costly (Khemakhem & Ltifi, 2023; Rani et al., 2024).

In simple terms, GANs rectify dataset imbalance and lower the impact of identity-related factors, hence improving facial emotion recognition. Still, their use calls for careful evaluation of the processing capability needed for training and the produced picture quality (Khemakhem & Ltifi, 2023; Rani et al., 2024).

2.3. COMPARISONS OF DEEP LEARNING

Table 2.1 Comparisons of Deep Learning

Model	Description	Advantages	Disadvantages
CNNs	Task involving picture categorisation mostly uses CNNs. Usually applied for face expression identification, they are efficient in extracting characteristics from photos.	CNNs can isolate the face feature vector extraction from the backdrop of the image. They can effectively and highly precisely accentuate the feeling.	CNNs might struggle with capturing temporal dependencies in video data.
RNNs	RNNs address sequential analysis of tasks. Using a continuous valued hidden layer representation, they offer a pleasing structure for spreading knowledge over a sequence.	With temporal averaging for aggregation, RNNs can beat a previously used CNN technique. They are quite good at differentiating positive from negative feelings.	RNNs can be difficult to train as they can experience the vanishing gradient problem.
GANs	New data instances created by GANs reflect your training data. They help to preserve the feeling when synthesising face emotions..	GANs can significantly reduce the FER performance gap between frontal and non-frontal faces. They can improve recognition accuracy for each expression.	GANs can be difficult to train, and the generated images may not always be realistic.

Generative adversarial networks (GANs), recurrent neural networks (RNNs), and convolutional neural networks (CNNs) are compared in the table below for face emotion recognition.

Understanding facial emotions requires the ability to gather spatial hierarchies of features, for which CNNs excel. They can identify emotions anywhere the expression on the face displays by controlling translation invariance. CNNs comprehend expressive feature representations, so they can catch up on minute emotional cues that other models would neglect. Understanding the broad emotional state of a face expression requires one to be able to extract global context. CNNs are flexible and scalable; combining them with transfer learning techniques might assist their performance to be better. Although they need a lot of data for training, they are not particularly good at showing temporal connections.

Similarly adept at capturing temporal connections, recurrent neural networks (RNNs) can manage sequences of varying durations. They therefore suit uses requiring either sequential or time-dependent input data. But they are less successful than CNNs in capturing spatial hierarchies and suffer with training because of vanishing and exploding gradients.

Generative Adversarial Networks (GANs) capacity to generate fresh data quite similar to the input data might help data augmentation. To enhance their ability to preserve identification from actual to produced faces, further study is required as their training might be challenging and unreliable.

CNNs are the best option for my research because of their exceptional ability to capture the spatial hierarchies of features in pictures, which is essential for facial emotion recognition. They can control translation invariance, which allows them to identify emotions no matter where the expression appears on the face. Because CNNs can comprehend expressive feature representations, they can pick up on subtle emotional cues that other models would overlook. They can extract global context, which is crucial for deciphering a face expressions overall emotional state. CNNs are flexible and scalable, and their performance may be enhanced by combining them with transfer learning techniques. CNNs can recognize emotions with excellent accuracy, according to studies. The particular needs and limitations of the undertaking, however, may influence the model selection. For instance, RNNs or GANs may be better suitable if there is sequential data or need to create fresh data.

2.4. SIMILAR STUDIES

2.4.1. Face Emotion Recognition System of Customer Service Using CNN Based on Embedded System

A study on a Customer Emotion Recognition System that uses Convolutional Neural Network (CNN) technology to improve service experiences in supermarkets is presented in the article. It addresses the challenges in customer service and proposes using computer vision to identify emotions thereby improving the purchasing experience.

The authors apply a CNN model trained on the FER2013 dataset, TensorFlow, and the Pypaz package for image processing and emotional classification. They include multi-layer feature fusion into the CNN algorithm to increase the accuracy of emotional recognition.

The system shows impressive real-time client emotional identification with an accuracy of 66.7%, precision of 66.7%, recall of 100%, and F1 score of 0.8. The study comes to the conclusion that the proposed emotion detecting technology may significantly raise the quality of customer service by presenting objective information on client satisfaction.

The system does quite a good job in recognising emotions including happiness, sadness, rage, and frustration. The study indicates that with further training utilising a range of datasets, the system might be tuned to various client sorts and situations. (Muhammad Zulkifli Nawfal & Barlian Henryranu Prasetio, 2023)

2.4.2. Facial emotion recognition using convolutional neural networks (FERC)

A new approach introduced in this paper, Facial Emotion identification using Convolutional Neural Networks (FERC), considerably improves the accuracy of emotional identification from facial expressions.

FERC use a CNN split in two: the first section gathers facial feature vectors and the second removes the backdrop. Using an expression vector (EV), it identifies five basic face expressions. With a 24 value EV, the models emotional identification accuracy was really remarkable, 96%.

Extensive testing using more than 750K images demonstrated the value of FERC. Because of its robustness to a range of challenges, including orientation and facial hair, the method is a possible tool for uses including predictive learning and lie detection. Correct emotional identification enhances the supermarket customer service experience. (Mehendale, 2020)

2.4.3. Emotion Recognition Using Convolutional Neural Network with Selected Statistical Photoplethysmogram Features

Using statistical features from photoplethysmogram (PPG) inputs, the work provides a Convolutional Neural Network (CNN) technique for emotion identification. This method is supposed to be quick and effective in spotting emotions, hence it might be fairly helpful in customer service environments like supermarkets.

The technique aggregates certain statistical data obtained from PPG signal correlation with deep information retrieved by two CNNs. The PPG signals are easier to record than other physiological signals and present a simpler approach. The proposed method showed remarkable performance with arousal accuracies of 80.9% and valence accuracies of 82.1% using a mere 10-second recognition period.

The article came to the conclusion that accurate emotional identification results from CNN-extracted characteristics coupled with statistically chosen ones. By quickly adjusting to the moods of the clients, this approach may be used in real-time, for evaluating client emotions in supermarkets, to improve customer service experiences. (Lee et al., 2020)

2.4.4. An Efficient Approach to Face Emotion Recognition with Convolutional Neural Networks

Advanced approaches for facial emotion recognition (FER) utilising convolutional neural networks (CNNs), which this work provides, will help to improve supermarket customer service.

Designed for bespoke and transfer learning, CNN models were developed from the FER2013 dataset using dataset augmentation and filtering.

The work studied binary and multi-classifications; the latter used various models for every feeling. On the original dataset, the top ensemble models have accuracy ranging from 75.06% to 75.91%; on the filtered dataset, this ranges also.

Emotionally speaking, the studies revealed that although ensemble models increase accuracy, binary models usually beat multi-class models. By providing accurate and powerful emotional identification for possible use in consumer service situations such supermarkets, the new models and updated FER database serve to advance the sector. (Białek et al., 2023)

2.4.5. Detecting emotions through EEG signals based on modified convolutional fuzzy neural network

The paper includes a study on emotion recognition using EEG data with an emphasis on a modified Convolutional Fuzzy Neural Network (CFNN) for accurate detection. It suggests a hybrid deep learning approach to address the problems traditional machine learning methods encounter in interpreting intricate EEG data.

While the Fast Fourier Transform (FFT) collects features, pre-processing of EEG data reduces noise; the upgraded CFNN model identifies emotions. By use of convolutional neural networks and fuzzy logic, CFNN solves uncertainty, hence enhancing feature representation. The suggested model outperformed state-of- the-art techniques 56 with an astounding average accuracy of 98.21% for valence (pleasantness) and 98.08% for arousal (intensity).

Combining fuzzy logic with deep learning, the CFNN model shows to considerably improve the capacity to detect emotions from EEG data. This method could enable supermarkets improve consumer service interactions by seeing and reacting to consumer emotions in real-time. (Ahmadzadeh Nobari Azar et al., 2024)

2.4.6. Facial emotion recognition system for autistic children: a feasible study based on FPGA implementation

The paper reports on a face expression recognition system for children with autism. By means of this technology as a consumer emotion recognition system, one might alter it to enhance customer service in supermarkets. Built on an FPGA, the system is portable and runs in real time; this might be helpful for real-time client emotional detection in a supermarket environment.

The authors derive characteristics and detect emotions using Principal Component Analysis (PCA). This approach concentrates on reducing complexity and power consumption, which would be helpful in a supermarket where power economy is crucial.

The technology recognises emotions in real time with excellent accuracy, 82.3%. This great precision might help to precisely identify consumer emotions in real-time, therefore improving the quality of customer service by offering objective statistics on client contentment.

The paper concludes that PCA-based hardware implementations are feasible for portable emotion recognition systems, providing a balance between accuracy, power efficiency, and speed. These findings suggest that further training with diverse datasets, such as customer facial expressions in different supermarket scenarios, can adapt the system to various customer types and enhance the shopping experience.(Smitha & Vinod, 2015)

2.4.7. Similar Studies Table

Table 2.2 Similar Studies

Study	Key Points	Methodology / Technique	Result or Finding
Face Emotion Recognition System of Customer Service using CNN Based on Embedded System	In academic customer service, examine consumers facial expressions to compile objective information on customer happiness and enhance service quality.	TensorFlow with Pypaz library convolutional neural network and Raspberry Pi.	System accurately recognized basic emotions (happiness, sadness, anger, disappointment) with potential for enhancing customer service quality.
Facial emotion recognition system for autistic children: a feasible study based on FPGA implementation	Create a portable, real-time emotional detector to help autistic kids identify emotions in face-to--face interactions.	Principal Component Analysis (PCA) implemented on an FPGA, comparing serial and parallel eigenvalue/eigenvector calculation methods.	With 8-bit word length, achieved 82.3% detection accuracy proving hardware-efficient, portable emotion identification system capability.
An Efficient Approach to Face Emotion Recognition with Convolutional Neural Networks	Improve facial emotion recognition (FER) by exploring various CNN models, dataset modifications, and ensemble methods.	Custom CNNs, transfer learning models (VGG16, ResNet50), ensemble models, dataset filtering, and augmentation.	Exceeding human accuracy, 75.06% accuracy on original FER2013 and 76.90% on filtered FER2013 shows the value of ensemble techniques and dataset improvement.
Detecting emotions through EEG signals based on modifed convolutional fuzzy neural network	Enhance emotion recognition accuracy from EEG signals by improving the Convolutional Fuzzy Neural Network (CFNN) model.	Pre-processing, feature extraction using Fast Fourier Transform (FFT), and classification using a modified CFNN with fuzzification and defuzzification layers.	Relative to state-of- the-art approaches, achieved 98.21% and 98.08% average accuracy for valence and arousal respectively, thereby proving the efficiency of the enhanced CFNN model.

Emotion Recognition Using Convolutional Neural Network with Selected Statistical Photoplethysmogram Features	Develop an efficient PPG-based emotion recognition method by fusing deep features from CNNs and selected statistical features.	Feature extraction using CNNs on PPG and NN interval data, statistical feature selection using Pearsons correlation, and feature fusion for emotion classification.	Achieved high accuracy with a short recognition interval (10 s), demonstrating the potential of PPG signals for real-time emotion recognition.
Facial emotion recognition using convolutional neural networks (FERC)	Using a two-level CNN architecture, propose a fresh facial emotion recognition method (FERC) with higher accuracy.	Two-level CNN: first part for background removal, second part for facial feature extraction using expressional vectors (EV).	Achieved 96% accuracy on a dataset of 10,000 images, demonstrating the effectiveness of the two-level CNN approach and background removal for emotion recognition.
My Proposed Project	Analyze customers facial emotion at supermarket	Using CNNs	The accuracy will be measured

2.5. WEB APPLICATION

Web applications, also called web apps, are computer programs accessible and utilised via the internet using a web browser. Because web browsers are so widely used and utilising a web browser as a client is so convenient, they have grown even more popular (Murugesan, 2008).

With a web browser and an internet connection, users of any device, including desktop computers, laptops, tablets, and cellphones may link from anywhere. Thanks to their browser-based character, they provide cross-platform compatibility running easily on many operating systems like Windows, Mac, Linux, and mobile OS like Android and iOS. Updates and patches applied centrally on the server help to simplify maintenance as they guarantee users always have access to the newest version without requiring any installation activity. One benefit is scalability as these programs may be readily changed to include server-side resources, therefore meeting an increasing user count. Furthermore, even though web-based apps might be vulnerable to security issues, they gain from the strong security features included in contemporary web browsers. (Murugesan, 2008)

Unlike conventional software, online apps operate straight in your web browser, therefore avoiding downloads and installations. Behind-the-scenes cooperation between two key components, the front end and the back end is what drives this capability. You view and interact with the front end, buttons, menus, text, and the whole visual experience. On the other side, the back end is the hidden powerhouse managing difficult chores such databases, scripts, and web services enabling the app operation. Requests and updates travel between your browser and the web server as you negotiate the front end. The server then serves as a link, contacting the database to either save or access data depending on your behaviour. This communication can set off other activities, such as dynamic altering what you view on your screen or delivering alerts. Usually distributed via systems like Google Cloud Platform, web apps live on servers. Development of web apps depends much on accessibility. Following accessibility rules and including tools like screen readers and alternate text descriptions for pictures helps developers make sure everyone, regardless of ability, may engage with them. Therefore, keep in mind the hidden symphony - your gadget collaborating with the server and database to provide a flawless user experience – the next time you use a web app, whether it email or visiting an online store. (Saia Sheila M. AND Nelson, 2022)

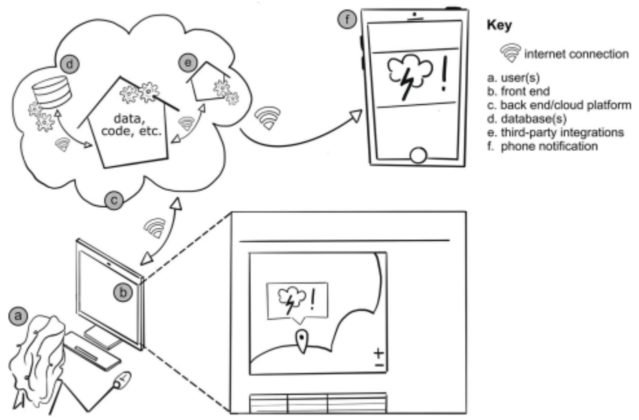


Figure 2.3 Major components of a web application (web app).

The number stated on the web page you are currently reading shows the main elements of a web application (web app). It shows the interaction between the user and the front end of the web app as well as how the front end is changed depending on backend changes. It also illustrates how third-party connectors allow updates in the back end to set off SMS or email alerts. Sheila M. Saia gets credit for the photograph. I regretfully cannot include the figure here, but it helps to graphically show the link and flow between the user interface and the server-side parts of a web application.

From interactive forms and data dashboards to complete-fledged software programs, web apps find extensive applications. They find use in everything from e-commerce to e-learning to data analysis to social networking. (Murugesan, 2008)

2.6. CONCLUSION

Based on this Literature Review, it is clear that Convolutional Neural Networks (CNNs) have great potential in the field of face expression identification, which is essential for improving supermarket customer service encounters. CNNs are quite good at maintaining translation invariance, grasping expressive feature representations, and capturing spatial hierarchies of features. Their performance may be raised using transfer learning approaches; they are scalable, flexible, and CNNs have shown great accuracy rates in several experiments proving their efficiency in emotional recognition challenges. CNNs seem to be a great option for my project with their demonstrated performance in this field. Still, the particular model you use will rely on the needs and restrictions of your project. Other models such as Recurrent Neural Networks (RNNs) or Generative Adversarial Networks (GANs) might be more suitable if sequential data or the creation of fresh data is engaged. Furthermore crucial is the fact that web apps can significantly help to implement such models for real-time emotional identification, improve accessibility and user experience.

CHAPTER 3 METHODOLOGY

From knowledge of needs and data collection to developing and training the CNN model, this chapter describes the method applied to create the system. Using diagrams and use case descriptions, it presents each phase precisely in line with the waterfall technique. This chapter shows the methodically planned and carried out nature of the project.

3.1. INTRODUCTION – METHODOLOGY (WATERFALL)

Often characterised as a conventional model, the Waterfall Methodology is a linear and sequential method of software development (Senarath, 2021). It is distinguished by a set of separate and successive phases whereby one must finish each phase before advancing to the next. Among the usual steps are requirements analysis, system design, implementation, testing, deployment, and maintenance.

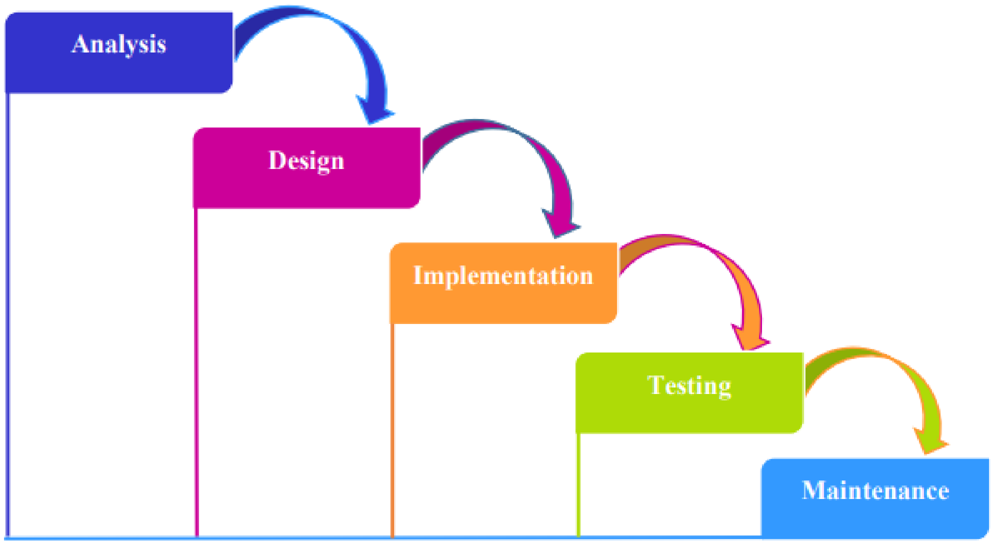


Figure 3.1 Waterfall Software Development Methodology

The program Requirements Specification (SRS) is developed by means of requirements analysis, therefore defining the functional and non-functional needs of the program (Senarath, 2021). Furthermore, included in the Design Phase of the program are algorithm design, software architecture, database scheming, and user interface design (Senarath, 2021). The actual coding of the software occurs here, when the design is converted into an executable program, therefore the implementation phase. Following that, the program is evaluated in the Testing Phase to make sure it satisfies the original requirements and specifications, and any flaws discovered are fixed. Following the deployment of the program, this phase (the Maintenance Phase) involves adjusting enhance performance, fixing problems, and adjust to environmental changes (Senarath, 2021). Every step must be finished before going on to the next; there is no overlapping between them (Senarath, 2021).Nowadays, the Waterfall Model is mostly renowned for being straightforward and under controllable influence.

For projects with well defined criteria that are unlikely to change over time, this method proves effective. By allowing deadlines and benchmarks to be set, it simplifies management and helps one to evaluate development. Furthermore, simplifying copious documentation is the methodical design of the Waterfall Methodology, which supports scalability and future maintenance.

Finally, Waterfall Methodology may be applied to build projects provided the criteria are clear, consistent, and recorded. With rigorous planning, management, and a clear knowledge of the project lifeline, one may generate effective and predictable outcomes from its organised character.

3.2. PROJECT FRAMEWORK

Table 3.1 Project Framework

OBJECTIVE	PHASES	ACTIVITIES	OUTCOME
To identify suitable algorithms that detect customers emotions based on facial reactions.	REQUIREMENT ANALYSIS	• Background research and problem statement identification • Project timetable identification • Dataset collecting identification • Preparation of Data	• Objective, Problem statement, Scope, and significance • Project timeline shown in Gantt Chart style • Data description
To develop customer emotion recognition systems application based on facial recognition using Convolutional Neural Networks.	SYSTEM DESIGN	• Architecture of Design Systems • Design the Flowchart, ERD, and UCD • Design the User Interface Prototype, Logical and Interface	• System architecture • Flow charts • Entity relationship diagrams • Use case diagrams • User interface
	SYSTEM IMPLEMENTATION	• Hardware and software needs • Development of the system	Customer Emotion Recognition Application
To evaluate the accuracy of customer emotion recognition using Convolutional Neural Networks.	SYSTEM TESTING	• Create the test case for the system • Evaluate its performance.	Result Report
	MAINTENANCE	Point out the system's strengths and limitations.	Enhancement of the system

3.3. REQUIREMENTS ANALYSIS

3.3.1. Knowledge Acquisition and Analysis

This section defines the projects objectives, scope, significance, literature review, and pertinent studies. The main goal of this study is to create a Customer Emotion Recognition System (CERS) by means of convolutional neural networks (CNNs), therefore improving supermarket customer service. Investigating and choosing suitable technologies, such as face recognition algorithms, to precisely identify and assess consumer emotions in real time is part of the scope of the project. The research also evaluates the CERSs dependability and efficiency throughout a range of races and cultural backgrounds thereby guaranteeing its relevance in many grocery settings. The scope also includes correctly inserting the CERS into the architecture of the contemporary supermarket and handling privacy issues with consumer data collecting and storage. Acquired from Kaggle for this study, the Emotion Detection FER dataset comprises of more than 5,000 facial photos classified according to kind of emotion.

This study is important as it might increase consumer satisfaction in supermarkets and service quality. The research is to enable retailers improve their customer service strategies by accurately analysing client emotions using facial expression recognition coupled with deep learning and machine learning technologies. This work also improves deep learning and picture analysis, therefore facilitating other artificial intelligence-based face emotion ratings advances.

For face emotion recognition, the literature review covers numerous deep learning methods including Generative Adversarial Networks (GANs), Recurrent Neural Networks (RNNs), and Convolutional Neural Networks (CNNs). CNNs can manage translation invariance, comprehend expressive feature representations, and record spatial hierarchies of data, so they are rated most appropriate for this purpose. They are scalable and adaptable as well as useful with transfer learning approaches to boost production.

CNNs These investigations provide perceptive data and support the development and enhancement of the proposed CERS.

3.3.2. Timeline – GANTT CHART And MILESTONE

Table 3.2 Gantt Chart

Task	Semester 5 (Week)													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Mutual Acceptance Form Submission (F1)														
Chapter 1- Introduction														
Project Motivation Evaluation Form Submission (F2)														
Submission of Chapter 1														
Outline of Chapter 2														
Chapter 2 - Literature Review														
Literature Review Evaluation Form Submission (F3)														
Submission of Chapter 2														
Chapter 3 - Methodology														
Methodology Evaluation Form Submission (F4)														
Submission of Chapter 3														
Plagiarism Checking														
Submission of Full Report														
Presentation of Final Proposal														

Table 3.3 Milestone

Task	Semester 6 (Week)													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Chapter 1- Introduction (Refined)														
Submission of Chapter 1 (Refined)														
Business Model Canvas														
Chapter 2 - Literature Review (Refined)														
Submission of Draft of Business Model Canvas														
Submission of Chapter 2 (Refined)														
Progress Presentation														
Submission of Business Model Canvas														
Chapter 3 - Methodology (Refined)														
Submission of Chapter 3 (Refined)														
Chapter 4 - Results and Findings														
Progress Presentation														
Chapter 5 - Conclusion														
Writing Full Report														
Submission of Full Report														
Presentation of Final Year Project														

3.3.3. Data Collection

Using two datasets, the Emotion Detection FER dataset (Emotion Detection, n.d.) and the Face Expression Recognition (FER) dataset (Face Expression Recognition Dataset, n.d.) from Kaggle will help me to test the model and train and verify the model. Both datasets are carefully labelled with seven different emotions: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral, so offering a strong basis for creating and assessing the model.

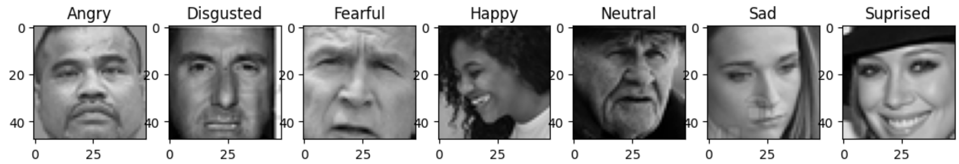


Figure 3.3 Sample image from the dataset

The figure above presents a sample face from each emotion category within the dataset, emphasizing the facial features that convey different emotional states. Each image is standardized to 48x48 pixels in grayscale, offering sufficient detail for analyzing facial expressions while being compact enough to reduce the computational load. Pre-processing all photographs has helped to concentrate on the face area, therefore strengthening the learning process of the model and raising the accuracy of emotional identification. Effective processing and training made possible by the constant image size and greyscale format throughout the datasets greatly help to construct an accurate face emotion recognition system.

3.3.4. Data Preprocessing

Data preparation for this project consists in data classification and data generator setup to manage the three dataset sections: training, validation, and testing. Comprising 35,887 pictures, the Face Expression Recognition (FER) dataset was employed. Following an 80/20 divide, these included 7,066 for validation and 28,821 photos for training. For the testing stage, I also chose

7,178 photographs from the Emotion Detection FER collection out of 35,685 images overall.

Since all photos are already grayscales 48x48 pixel standard, the preparation focused more on making sure the data is suitable for effective model training and evaluation. Applying real-time data augmentation methods like rotation, zoom, and horizontal flipping, the training data generator handles the 28,821 photos assigned for training. These augmentations improve the variety of the training data, therefore enabling the model to generalise more successfully. Without any augmentation, the validation data generator manages the 7,066 photos set aside for validation so that the validation results faithfully mirror the performance of the model on unmodified data. Similarly, the testing data generator processes the 7,178 images from the Emotion Detection FER dataset used for testing, with no augmentation applied, maintaining the integrity of the test results.

By setting up these data generators, the preprocessing step supports the development of an accurate and efficient facial expression recognition model.

3.4. SYSTEM DESIGN

3.4.1. System Architecture

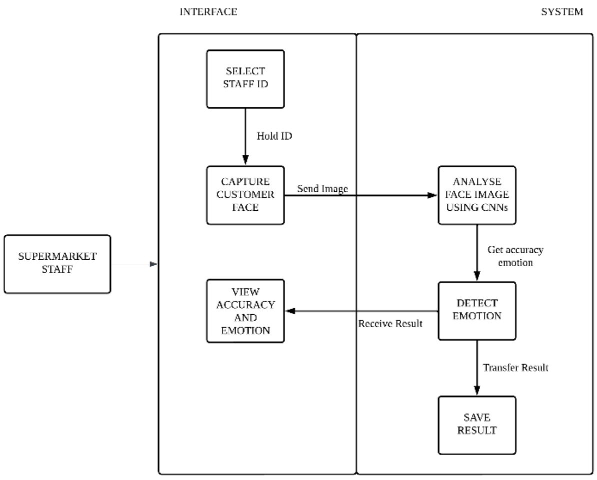


Figure 3.2 System Architecture

Figure 3.2 presents the architecture of the emotion recognition system, consisting of two primary components: Interface and System.

The Interface component facilitates interaction with supermarket staff, encompassing the Select Staff ID module for selecting the staff ID, the Capture Customer Face module for capturing customer images via in-store cameras, and the View Accuracy and Emotion module for displaying the identified emotion along with its confidence level.

The System component is the core of the emotion recognition process. Within it, the Save Result module stores the results, and the Analyse Face Image Using CNNs module employs Convolutional Neural Networks (CNNs) to analyze captured images, extracting features and patterns indicative of emotions. The Detect Emotion module then leverages these analyzed results to identify the specific emotion being expressed and calculates the associated accuracy. These modules interact seamlessly: captured images are sent from the Interface to the Analyse Face Image module, which processes them and passes the analyzed data, including accuracy, to the Detect Emotion module. Finally, the identified emotion and accuracy are sent back to the Interface for display to the staff, enabling real-time insights into customer sentiment.

3.4.2. Flow Design

3.4.2.1. Flow chart

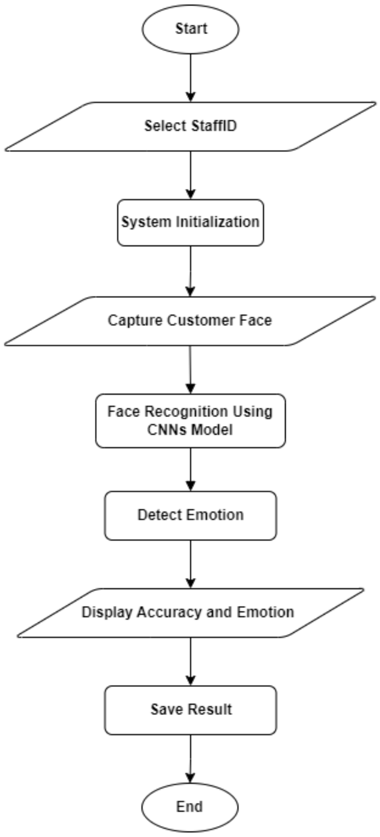


Figure 3.3 Flowchart Diagram

The flowchart in **Figure 3.3** details the sequential process of the Customer Emotion Recognition System, encompassing the following elaborate steps:

1. Select StaffID: This initial step involves selecting the StaffID to identify the staff member operating the system. It guarantees that the system records the actions under the appropriate employees name.
2. System Initialization: This phase gets the system ready for usage. It covers chores including connecting many system components, loading the pre-trained Convolutional Neural Network (CNN), and setting cameras. This step ensures that all necessary resources are easily available and in the correct condition before the system starts processing client data.

3. Capture Customer Face: The strategically placed cameras in the store switch on to capture images of its customers. In this stage, facial expressions, which are crucial indicators of emotions, take first importance. The gathered pictures are then pre-processed that is, cropped, resized, and normalized to ensure uniformity for the next examination.

4. Face Recognition Using CNNs Model: The CNN model feeds pre-processed face images after having previously been trained. Having been taught on an extensive range of facial expressions, this system can detect trends and traits connected to a wide spectrum of emotions. CNN gathers important information such the eye wrinkles, eyebrow positioning, and mouth shape after photo analysis.

5. Detect Emotion: The extracted features of CNN analysis find use in the module of emotional classification. This module sorts the claimed emotion of the image using algorithms. The collected characteristics may be mapped to a set of preset emotions, such as happiness, sadness, anger, surprise, or neutrality, using methods like Support Vector Machines (SVM) or Random Forests. The module also determines the emotion detections accuracy or confidence level, which shows how reliable the recognized emotion is.

6. Display Accuracy and Emotion: The grocery staff are then shown the recognized emotion and the accuracy that goes along with it. A dashboard, a mobile device, or any other appropriate interface can display this data. The employees may use this real-time data to determine how customers are feeling and modify their interactions accordingly. For example, a satisfied customer may receive tailored advice, while a dissatisfied client may receive support from the personnel.

7. Save Result: Saving the emotion detection processs output is the last step. This involves keeping the taken photos for later use or analysis, as well as the recognized emotion and accuracy in CSV files.

3.4.2.2. Use Case Diagram

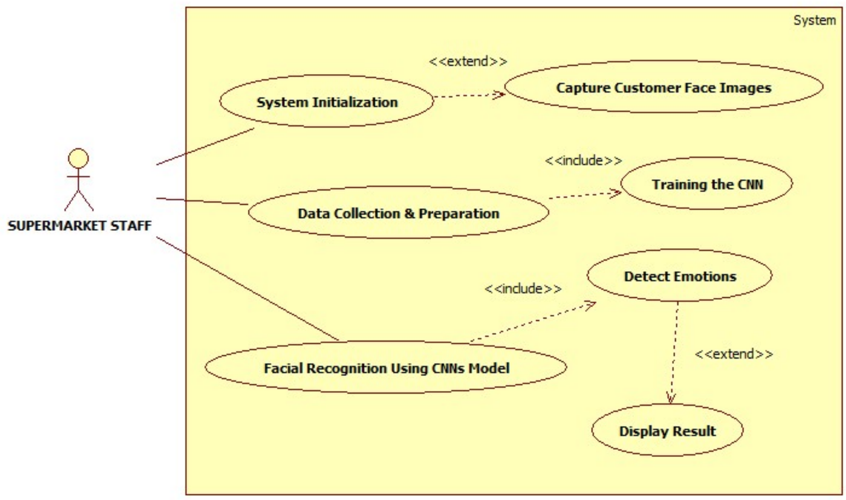


Figure 3.4 Use Case Diagram

The figure above shows the use case of this project. This is the Use Case Description:-

Table 3.4 Use Case Description (System Initialization)

USE CASE	System Initialization
ACTORS	Staff
DESCRIPTION	Initializes the system, preparing it for capturing images and processing data.
PRECONDITION	The system is powered on and functional
POSTCONDITION	The system is ready for operation
FLOW OF EVENT	1. Staff powers on the system. 2. System performs self-checks. 3. System initializes components for image capture and data processing. 4. System signals ready state.

Table 3.4 explains System Initialization. This is the foundational step, where the emotion recognition software and its underlying hardware components are started up. It ensures that cameras are connected, the

CNN model is loaded, and any necessary databases or logs are ready for use.

Table 3.5 Use Case Description (Select StaffID)

USE CASE	Select StaffID
ACTORS	Staff
DESCRIPTION	Selects the StaffID to identify the staff member operating the system.
PRECONDITION	System is initialized
POSTCONDITION	StaffID is selected
FLOW OF EVENT	1. Staff member selects their StaffID 2. The system logs activities under the selected StaffID.

Table 3.5 explains the Select StaffID. This step ensures that the system logs the activities under the correct staff members identification.

Table 3.6 Use Case Description (Capture Customer Face Images)

USE CASE	Capture Customer Face Images
ACTORS	Staff
DESCRIPTION	Captures face images of customers in the supermarket using in-store cameras.
PRECONDITION	StaffID is selected
POSTCONDITION	Images are stored for processing
FLOW OF EVENT	1. System activates in-store cameras. 2. Cameras capture face images of customers. 3. Images are transmitted to the system. 4. System stores images for further processing.

Table 3.6 explains the Capture Customer Face Images. In-store cameras continuously monitor customer interactions. This use case involves the system capturing high-quality images of customer faces, often at specific intervals or triggered by events (like entering a service area).

Table 3.7 Use Case Description (Data Collection & Preparation)

USE CASE	Data Collection & Preparation
ACTORS	Staff
DESCRIPTION	Collects and preprocesses images for training the CNN, including augmentation and normalization.
PRECONDITION	Images are captured
POSTCONDITION	Prepared dataset for CNN training
FLOW OF EVENT	1. System aggregates captured images. 2. Images undergo preprocessing (e.g., augmentation, normalization). 3. The prepared dataset is created for CNN training.

Table 3.7 explains Data Collection & Preparation. This is the behind-the-scenes workhorse. Images from various sources, including the captured customer images, are gathered. The system then preprocesses this raw data. This might involve resizing images, adjusting lighting, labeling emotions in training data, and potentially applying techniques like data augmentation to create a more robust dataset.

Table 3.8 Use Case Description (Training the CNN)

USE CASE	Training the CNN
ACTORS	Staff
DESCRIPTION	Trains the Convolutional Neural Network using the prepared dataset.
PRECONDITION	Data is collected and prepared
POSTCONDITION	Trained CNN model
FLOW OF EVENT	1. The system loads the prepared dataset. 2. The CNN model is initialized. 3. The system trains the CNN on the dataset. 4. The trained model is validated and saved.

Table 3.8 explains Training the CNN. This use case is the core of the machine learning aspect. Training the Convolutional Neural Network (CNN) using the preprocessed dataset The CNN picks up trends in face

traits linked to certain emotions. Iterative and including parameter adjustment to attain ideal model performance, this training method.

Table 3.9 Use Case Description (Face Recognition Using CNNs Model)

USE CASE	Face Recognition Using CNNs Model
ACTORS	System
DESCRIPTION	Analyzes customer faces using the trained CNN to extract features.
PRECONDITION	CNN is trained
POSTCONDITION	Extracted facial features
FLOW OF EVENT	1. System loads the trained CNN model. 2. Captured images are input into the model. 3. CNN analyzes images and extracts facial features. 4. Features are stored for emotion identification.

Table 3.9 explains Face Recognition Using CNNs Model. Once the CNN model is trained, its put into action. This use case happens in real -time. As new customer images are captured, they are fed into the CNN. The model then analyzes facial landmarks, expressions, and other relevant details to make predictions about the customers emotional state.

Table 3.10 Use Case Description (Detect Emotions)

USE CASE	Detect Emotions
ACTORS	System
DESCRIPTION	Detects emotions from the analyzed faces using the CNN model.
PRECONDITION	Faces are analyzed
POSTCONDITION	Recognized emotions
FLOW OF EVENT	1. The system inputs facial features into the emotion recognition model. 2. The model identifies emotions from features. 3. Recognized emotions are recorded.

Table 3.10 explains Detect Emotions. This is the direct output of the CNN analysis. The system labels the captured face with the most likely

emotion it detects (e.g., happy, sad, angry, neutral). This information is then passed on for further action.

Table 3.11 Use Case Description (Display Result)

USE CASE	Display Result
ACTORS	System
DESCRIPTION	Provides staff with information about identified customer emotions.
PRECONDITION	Emotions are identified
POSTCONDITION	Staff is informed about emotions
FLOW OF EVENT	1. The system compiles a report of recognized emotions. 2. The report is transmitted to staff. 3. Staff receives notification and reviews the report.

Table 3.11 explains Display Result. This use case connects the technology to the human aspect of customer service. The system discreetly alerts staff (perhaps through a dashboard or notifications) about the identified customer emotions. This could include real-time updates or summaries of emotional trends among customers.

Table 3.12 Use Case Description (Save Result)

USE CASE	Save Result
ACTORS	System
DESCRIPTION	Saves the results of the emotion detection process, including accuracy and captured images.
PRECONDITION	Emotions are identified and analyzed
POSTCONDITION	Results are saved in CSV files and the captured images are stored
FLOW OF EVENT	1. The system saves the identified emotions and accuracy in CSV files. 2. The system stores the captured images for future reference.

Table 3.12 explains Save Result. This use case ensures that the results of the emotion detection process are saved for future analysis or reference.

3.4.3. Prototype User Interface

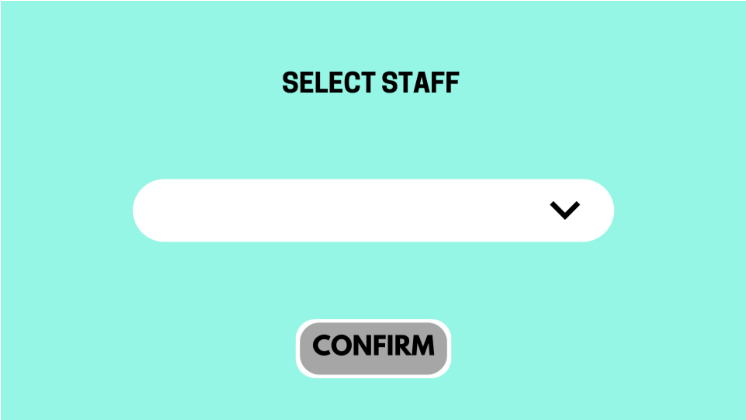


Figure 3.5 Select Staff Page

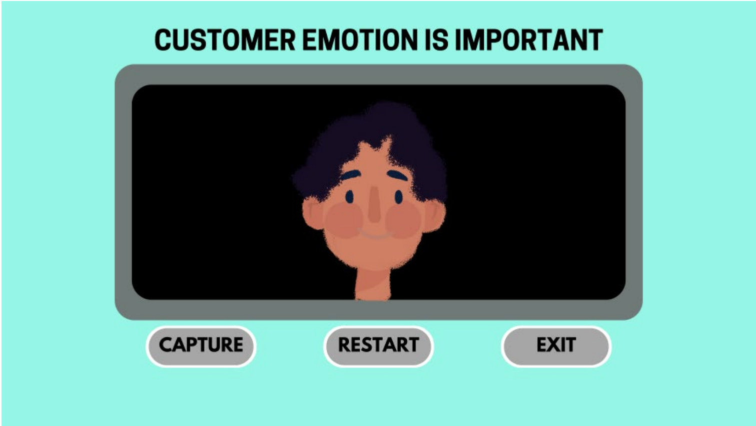


Figure 3.6 Starting Page

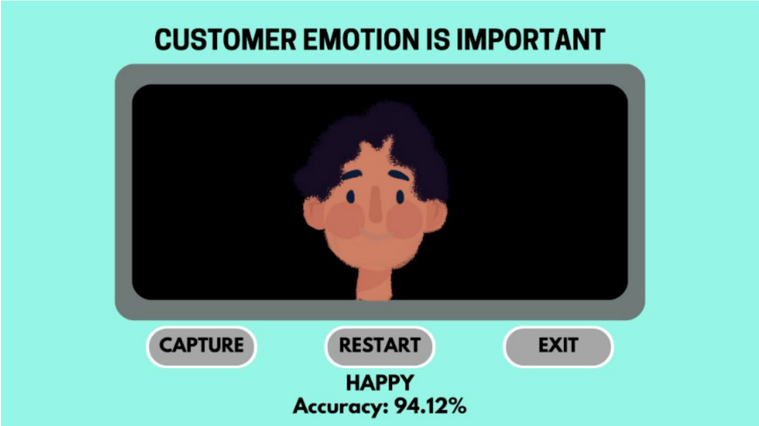


Figure 3.7 Result Page

3.5. SYSTEM DEVELOPMENT

3.5.1. Prototype Development

The journey of developing the emotion recognition prototype was an exciting one, with two main stages: building the training model and deploying the application.

3.5.1.1. Building the CNN Model

Three Convolutional Neural Network (CNN) models were crafted, each with a different number of epochs: CNNmodelFER_60.ipynb, CNNmodelFER_70.ipynb, and CNNmodelFER_80.ipynb.

It all started with bringing the necessary tools like TensorFlow and Keras. Loaded was the Emotion Detection FER dataset, including greyscale photos of human faces displaying seven distinct emotions: angry, disgusted, fearful, happy, neutral, sad, and surprising. Preprocessing on the photos includes data split into training and validation sets, normalising pixel values, and reshaping to add the necessary channel dimension.

```python
from keras.layers import Dense, Input, Dropout, GlobalAveragePooling2D,
Flatten, Conv2D, BatchNormalization, Activation, MaxPooling2D
from keras.models import Model, Sequential
from keras.optimizers import Adam

# number of possible label values
nb_classes = 7

# Initialising the CNN
model = Sequential()

# 1 - Convolution
model.add(Conv2D(64,(3,3), padding='same', input_shape=(48, 48,1)))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

# 2nd Convolution layer
model.add(Conv2D(128,(5,5), padding='same'))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

# 3rd Convolution layer
model.add(Conv2D(512,(3,3), padding='same'))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

# 4th Convolution layer
model.add(Conv2D(512,(3,3), padding='same'))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

# Flattening
model.add(Flatten())

# Fully connected layer 1st layer
model.add(Dense(256))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(Dropout(0.25))

# Fully connected layer 2nd layer
model.add(Dense(512))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(Dropout(0.25))

model.add(Dense(nb_classes, activation='softmax'))

opt = Adam(lr=0.0001)
model.compile(optimizer=opt, loss='categorical_crossentropy',
metrics=['accuracy'])

model.summary()
```

Figure 3.8 CNN Model Global Architecture.

The CNN model has two totally linked layers and four convolutional layers. While the convolutional layers function as smart detectives, collecting salient information from the images, the fully connected layers classify the images based on the properties they have detected.

An overview of CNN's operation may be seen in the following image:

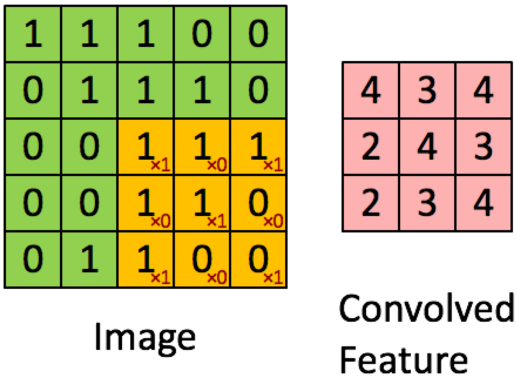


Figure 3.9 CNN Working Process

The orange sliding matrix, also referred to as a "filter" or "kernel," moves across the image pixel by pixel, while the green matrix shows the raw image data. Each step of this filters journey involves multiplying it with the corresponding elements of the base matrix, uncovering different image features:

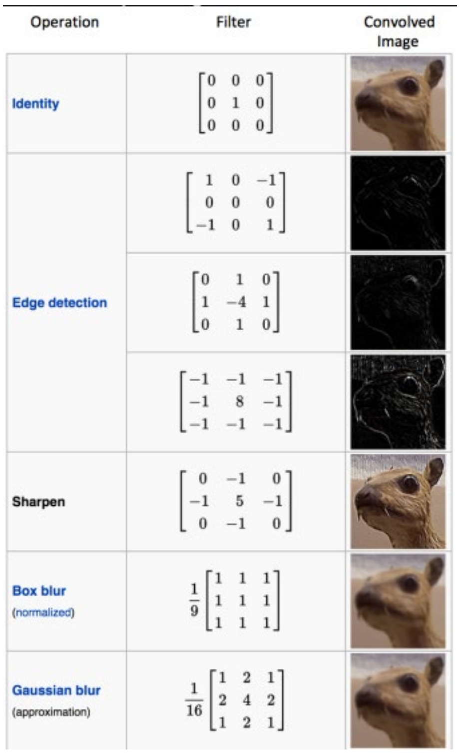


Figure 3.10 Different CNN Filters

- ReLU Function: This function introduces non-linearity into the CNN. Although tanh or sigmoid functions could also be used, ReLU often outshines them in performance.

- Pooling: This step reduces the dimensionality of each feature while preserving the most crucial information. Much like the convolutional step, a sliding function is applied to the data. Max pooling tends to outperform other functions like sum or mean pooling.

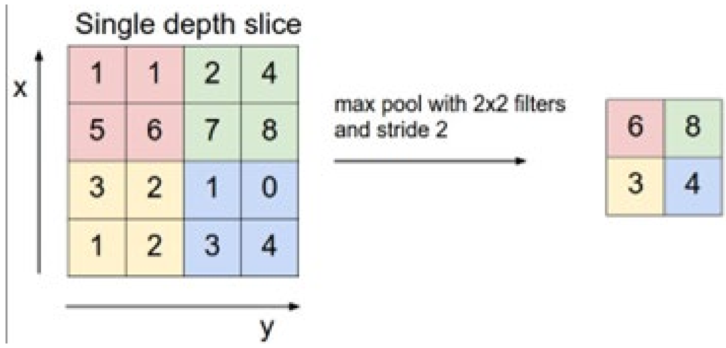


Figure 3.11 CNN Pooling Process.

Additionally, several techniques were used to enhance the model:

- Batch Normalization: This technique boosts the performance and stability of neural networks by standardizing inputs to have zero mean and unit variance.

- Dropout: This method helps in reducing overfitting by randomly skipping updates to the weights of certain nodes, preventing the network from becoming overly reliant on any single node.

The softmax function was chosen as the final activation function, perfect for multi-label classification. With the CNN architecture set, it was compiled with a few more parameters. The Adam optimizer, known for its efficiency, was selected, alongside categorical cross-entropy as the loss function due to its relevance for classification tasks. Accuracy was chosen as the metric, providing insightful information for classification tasks on balanced datasets.

Here is a summary of the CNN model:

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 48, 48, 64)        640

 batch_normalization (BatchN  (None, 48, 48, 64)       256
 ormalization)

 activation (Activation)     (None, 48, 48, 64)        0

 max_pooling2d (MaxPooling2D  (None, 24, 24, 64)       0
 )

 dropout (Dropout)           (None, 24, 24, 64)        0

 conv2d_1 (Conv2D)           (None, 24, 24, 128)       204928

 batch_normalization_1 (Batc  (None, 24, 24, 128)      512
 hNormalization)

 activation_1 (Activation)   (None, 24, 24, 128)       0

 max_pooling2d_1 (MaxPooling  (None, 12, 12, 128)      0
 2D)
...
Total params: 4,478,727
Trainable params: 4,474,759
Non-trainable params: 3,968
_________________________________________________________________
```

Figure 3.12 Summary of the CNN Model.

The image offers a detailed overview of the layers, their types, output shapes, and the number of parameters for each layer, essential for understanding CNNs architecture.

This is the component of Model Training.:

```python
%%time

# number of epochs to train the NN
epochs = 70

from keras.callbacks import ModelCheckpoint

checkpoint = ModelCheckpoint("model_weights.h5", monitor='val_acc',
    verbose=1, save_best_only=True, mode='max')
callbacks_list = [checkpoint]

history = model.fit_generator(generator=train_generator,
    steps_per_epoch=train_generator.n//train_generator.batch_size,
    epochs=epochs,
    validation_data = validation_generator,
    validation_steps = validation_generator.n//validation_generator.
    batch_size,
    callbacks=callbacks_list
    )
```
Python

Figure 3.13 Model of Training

The image that follows illustrates the training process of the CNN model with epoch numbers of 60, 70, and 80. The code sample comprises the epoch settings, a model checkpoint to save the best model weights depending on validation accuracy, and the model fitting process applying a generator for training and validation data.

This painstaking approach guaranteed that the CNN model was expert at recognising emotions in images of faces and well-tuned.

3.5.1.2. App deployment (emotion_recognition_app.py)

This section provides an outline of the real-time client emotion recognition settings of the Python application emotion_recognition_app.py. The application combines an easy graphical user interface (GUI), a pre-trained convolutional neural network (CNN), and a video feed to offer a seamless experience.

3.5.1.2.1 Loading the Model

The first phase of application is loading the already trained model. Reading the learnt weights from an H5 file and the models structure from a JSON file allows the load_model_from_json function to accomplish this:

```
# Load the model architecture from a JSON file and the weights from an H5 file
def load_model_from_json(json_path, weights_path):
    with open(json_path, 'r') as json_file:
        model_json = json_file.read()
    model = tf.keras.models.model_from_json(model_json)
    model.load_weights(weights_path)
    return model
```

Figure 3.14 Model loading

This method ensures flexibility by allowing the loading of several models with just a modification in the file locations. Moreover, the utility function resource_path guarantees compatibility in several circumstances, including executable packages created with PyInstaller.

3.5.1.2.2 Load Cashier Data

Before it starts to recognise emotions, the program gets cashier data from a StaffID.csv file. This ensures that every result on emotion recognition is connected to the suitable cashier. The load_cashier_data function reads the cashier information and then gets it available for the user to select:

```
# Load cashier data from CSV file
def load_cashier_data():
    cashier_data = []
    with open(csv_path, mode='r') as file:
        reader = csv.reader(file)
        next(reader)  # Skip the header
        for row in reader:
            cashier_id, cashier_name = row
            cashier_data.append(f"{cashier_id} - {cashier_name}")
    return cashier_data
```

Figure 3.15 Load Cashier Data

A CSV file with staff names and IDs is processed by this function, which then displays the data in a manner such as "ID - Name" for convenience of choosing. It guarantees that results are appropriately classified and saved, matching the appropriate staff member. This is an illustration of the data in a CSV file:

	A	B	C	D
1	ID	NAME		
2	1	ALI		
3	2	ABU		
4	3	WAIZ		
5	4	PAYAN		

Figure 3.15 Staff Data Example

3.5.1.2.3 Emotion Prediction

The applications core capability is its emotional prediction. After processing webcam frames and identifying faces, the predict_emotion function examines the identified area to forecast emotions:

```python
# Update the predict_emotion function for grayscale input
def predict_emotion(frame):
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades +
                                         "haarcascade_frontalface_default.xml")
    faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1,
                                         minNeighbors=5, minSize=(48, 48))

    if len(faces) == 0:
        return "No Face Detected", 0.0

    for (x, y, w, h) in faces:
        roi_gray = gray[y:y+h, x:x+w]
        roi_gray = cv2.resize(roi_gray, (48, 48))
        roi_gray = roi_gray.astype("float32") / 255.0
        roi_gray = np.expand_dims(roi_gray, axis=-1)  # Add channel dimension for grayscale
        roi_gray = np.expand_dims(roi_gray, axis=0)   # Add batch dimension

        predictions = model.predict(roi_gray, verbose=0)
        max_index = np.argmax(predictions[0])
        accuracy = predictions[0][max_index] * 100
        return emotions[max_index], accuracy
```

Figure 3.16 Emotion Prediction

This feature gives the projected emotion, such as "happy" or "sad," together with a confidence percentage, therefore enabling precise real-time analysis.

3.5.1.2.4 Integrating Webcam Functionality

The system guarantees real-time emotional identification by using live webcam broadcasting. The update_webcam feature controls the video stream and refreshes the GUI:

```python
# Update the webcam feed
def update_webcam():
    global cap, live_feed_active
    if live_feed_active:
        ret, frame = cap.read()
        if ret:
            frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
            img = Image.fromarray(frame)
            imgtk = ImageTk.PhotoImage(image=img)
            display_label.imgtk = imgtk
            display_label.configure(image=imgtk)
        display_label.after(10, update_webcam)
```

Figure 3.17 Integrating Webcam Functionality

This ensures continuous, flawless video feed for continuous emotional detection.

3.5.1.2.5 Capturing and Saving Results

Apart from emotional prediction, the application logs and accumulates the results for further use. The save_result feature arranges and stores images and accuracy:

```python
# Save the captured image and prediction result
def save_result(frame, emotion, accuracy):
    global image_counter, selected_cashier_id, selected_cashier_name
    if not selected_cashier_id:
        messagebox.showerror("Error", "Please select the cashier.")
        return

    # Create the results directory for the cashier if it doesn't exist
    cashier_dir = os.path.join('StaffData', selected_cashier_id)
    if not os.path.exists(cashier_dir):
        os.makedirs(cashier_dir)

    # Generate a unique filename
    timestamp = datetime.now().strftime("%Y_%m_%d_%H%M")
    image_filename = os.path.join(cashier_dir,f"img{image_counter}_{timestamp}.png")
    image_counter += 1

    # Save the image
    cv2.imwrite(image_filename, frame)

    # Save the prediction result in a CSV file named after the cashier's name
    csv_filename = os.path.join(cashier_dir,
                                f"{selected_cashier_name}.csv")
    file_exists = os.path.isfile(csv_filename)
    with open(csv_filename, mode='a', newline='') as file:
        writer = csv.writer(file)
        if not file_exists:
            writer.writerow(["Timestamp", "ID", "Emotion", "Accuracy", "Image Filena
        writer.writerow([timestamp, f"img{image_counter-1}", emotion, f"{accuracy:.2
        image_filename])
```

Figure 3.18 Capturing and Saving Results

By grouping the data into timestamped folders and CSV objects, the application guarantees that the outcomes are easy to access and monitor by.

3.5.1.2.6 User Interface

The programs user -friendly interface determines its success in great part. The application creates a user-friendly graphical user interface (GUI) using Tkinter that lets users select staff IDs, interact with prediction results, see live camera feeds:

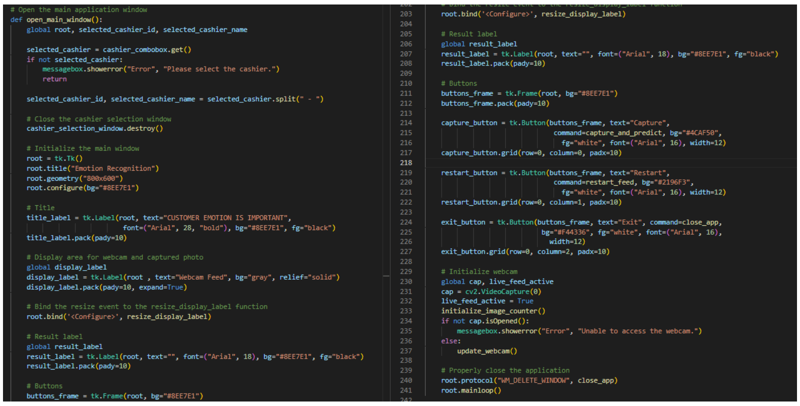


Figure 3.19 User Interface

This interface allows users to quickly negotiate the application, therefore facilitating its straightforward use.

Finally, by integrating a powerful model, real-time processing, and a simple user interface, emotion_recognition_app.py offers a whole emotional recognition solution. Its carefully thought out architecture promises convenience, efficiency, and user delight.

These files cooperate to enable a seamless workflow from building and training the CNN model to constructing a fully operational, user-friendly emotion identification system. This prototype shows how clever machine learning models may be effectively included into practical uses.

3.5.2. Hardware Requirements

The real components and infrastructure needed of a software system to perform as expected are hardware requirements. Apart from the minimum criteria for the model, memory (RAM), CPUs, storage, operating system (OS), these needs also include the kind of computer or laptop. Hardware needs guarantee that the software can run on the devices or servers without malfunction and best performance. Table 3.13 lists the hardware requirements.

Table 3.13 Hardware Requirements

No.	Hardware	Specifications And Description	
1.	Laptop	Model	Lenovo Ideapad Slim 3
		RAM	8.00 GB
		Processor	AMD Ryzen 3 3250U with Radeon Graphics
		Storage	239 GB + 932 GB
		OS	Windows 11 Home Single Language
2.	WebCam	Captures real-time video feeds for emotion detection.	

The general hardware specs applied in this project are displayed in the table above. This project makes use of a certain laptop model, specifically the Lenovo Ideapad Slim 3 with above specs. This project depends on hardware as without it the project cannot effectively run and finish it satisfactorially.

3.5.3. Software Requirements

The functional and non-functional requirements specified as such are those that a software system must satisfy. Software requirements form the foundation for development, testing, and assessment as well as aid to guarantee that the program satisfies the expectations and needs of users and stakeholders. Table 3.14 lists the software needed for this project.

Table 3.14 Software Requirements

No.	Software	Description
1.	Visual Studio Code	The backend-like infrastructure and user interface for the operating system are created using Visual Studio Code.
2.	Draw.io	Among the design instruments for creating a flowchart is Draw.io.
3.	StarUML	One builds a use case diagram using StarUML.
4.	Canva	Slides or presentations are created with Canva, which also serves for prototypical design or drawing.
5.	Mendeley	To obtain the accurate reference, Mendeley is the program used to check the references.
6.	Microsoft Word	Every documentation on the project is produced in Microsoft Word.
7.	Python 3.11+	The programming language applied for constructing the CNN model and the application.

The summary of the employed software tools in this project is displayed above. Since they help the project to be successfully executed and completed, all the software tools are valuable in this endeavour.

3.6. SYSTEM TESTING

These are some succinct test cases for the system of emotional recognition:

Test Case 1: Emotion Recognition Accuracy

ID: TC_01

- **Objective**: Compare the performance of three models trained for various epochs to check if the system can correctly categorise emotions from a varied collection of facial photos.
- **Precondition**: The system is initialized and ready for operation.
- **Test Steps**:
    1. Present the system with a set of images depicting various emotions (happiness, sadness, anger, surprise, etc.).
    2. Use three models trained for 60, 70, and 80 epochs to classify the emotions.

3. Match the ground truth labels developed during model training with the system's classifications.

- **Expected Result**: The system should precisely categorise emotions; among Epoch 60, Epoch 70, and Epoch 80 the performance measures (precision, recall, and F1 score) should highlight the best-performing model.

Test Case 2: Real-Time Emotion Detection

ID: TC_02

- **Objective**: Using a simulated video feed of consumers shopping in a supermarket, assess the real-time emotional detecting capacity of the system.
- **Precondition**: The system is ready for use and firstly set.
- **Test Steps**:
    1. Create a simulated video stream showing consumers in a range of moods.
    2. Given several people, different expressions, and changing lighting circumstances, track the system's real-time emotional detection and classification capability.
- **Expected Result**: Real-time, accurate emotional detection and classification by the system should be possible.

Test Case 3: Save Result Functionality

ID: TC_03

- **Objective**: Test the system's ability to save the results of the emotion detection process.
- **Precondition**: The system is initialized, and emotion detection is active.
- **Test Steps**:
    1. Conduct emotion detection on a set of images or video feeds.
    2. Verify that the system saves the identified emotions and accuracy in CSV files.
    3. Check that the system stores the captured images.
- **Expected Result**: The system should correctly save the results of the emotion detection process in the specified formats.

Test Case 4: User Interface Functionality

ID: TC_04

- • **Objective:** Assess the user interface's functionality and usability.
- • **Precondition:** The system is configured and prepared for use.
- • **Test Steps:**
    1. Use the interface to take pictures of customers' faces.
    2. Check to make sure the accurately identified emotions are shown.
    3. Check to be sure the " Capture," "Restart," and "Exit" buttons operate as expected.

- **Expected Result**: The user interface should function smoothly, displaying correct emotion detection results and responding appropriately to user actions.

3.7. MAINTENANCE

The waterfall concept dictates that the last phase of the software development life cycle (SDLC) is the maintenance phase. It arises when the product is put into use by end customers. Main goals of this phase are to:

Discover and fix any faults, errors, or bugs that show up during real use. This entails addressing issues arising from unanticipated user environment interactions or ones overlooked throughout the testing process.

Add fresh additions or improvements to the program in response to user input and shifting demands. This might mean strengthening present capabilities, acquiring new ones, or performance enhancement.

Make sure the software is still compatible with hardware, operating system, or other requirements that could develop in the running environment.

3.8. CONCLUSION

All things considered, this chapter offers a complete method for building a Customer Emotion Recognition System (CERS) based on Convolutional Neural Networks (CNN), hence enhancing supermarket customer service. The Waterfall model provides a clear structure for the project through several stages, therefore guaranteeing a methodical approach from requirements analysis to system maintenance. The concept is significant as it would enable supermarkets to instantaneously identify and respond to consumer emotions, therefore changing customer service. By using CNNs and a big dataset, the CERS aims to correctly recognise and classify emotions, therefore allowing staff members to customise interactions and enhance the buying experience. Effective application of this technology might lead to higher consumer pleasure and loyalty as well as improved store economic results.

CHAPTER 4 RESULTS AND DISCUSSION

The results and discussion of the development and testing of a machine learning algorithm (CNN) meant for supermarket customer emotional recognition are presented in great detail in this chapter. Included additionally are the dataset, the performance of the machine learning model, the user interface, a discussion of the results, and a synopsis.

4.1. DATA PREPROCESSING

Effective learning of the deep learning model in this project depends on the preparation of the training dataset. Steps in this procedure are dataset analysis, correction of any imbalances, and ready for flawless training, validation, and testing.

```python
# count number of train images for each expression

for expression in os.listdir(base_path + "train"):
    print(str(len(os.listdir(base_path + "train/" + expression))) +
          " " + expression + " images")
```

Python

```
3993 angry images
436 disgust images
4103 fear images
7164 happy images
4982 neutral images
4938 sad images
3205 surprise images
```

Figure 4.1 Training Dataset Analysis

As illustrated in Figure 4.1, the training set comprises images split into seven emotional classes: angry, disgust, fear, happy, neutral, sad, and surprise. The number of images per class is mostly balanced, except for the disgust category, which has significantly fewer samples compared to the others. This imbalance could potentially hinder the models ability to accurately predict the disgust emotion. Add ressing this challenge is critical and will be partially managed through data augmentation techniques during training.

```python
from keras.preprocessing.image import ImageDataGenerator

# number of images to feed into the NN for every batch
batch_size = 128

datagen_train = ImageDataGenerator()
datagen_validation = ImageDataGenerator()

train_generator = datagen_train.flow_from_directory(
    base_path + "train",
    target_size=(pic_size,pic_size),
    color_mode="grayscale",
    batch_size=batch_size,
    class_mode='categorical',
    shuffle=True)

validation_generator = datagen_validation.flow_from_directory(
    base_path + "validation",
    target_size=(pic_size,pic_size),
    color_mode="grayscale",
    batch_size=batch_size,
    class_mode='categorical',
    shuffle=False)

test_generator = datagen_validation.flow_from_directory(
    base_path + "test",
    target_size=(pic_size,pic_size),
    color_mode="grayscale",
    batch_size=batch_size,
    class_mode='categorical',
    shuffle=False)
```

Python

```
Found 28821 images belonging to 7 classes.
Found 7066 images belonging to 7 classes.
Found 7178 images belonging to 7 classes.
```

Figure 4.2 Data Preparation and Augmentation

To process the images efficiently, the ImageDataGenerator class from Keras is employed. This class not only facilitates the loading of images but also enhances the dataset by applying data augmentation. Augmentation techniques, such as random rotations, zooms, and flips, create additional variations of the existing images, which help improve the models ability to generalize to unseen data.

Figure 4.2 outlines how the data generators are configured:

- Training dataset: Comprises 28,821 images that are augmented on-the-fly to increase diversity and handle class imbalance.
- Validation dataset: Contains 7,203 images used to monitor the model's performance during training and adjust hyperparameters.
- Testing dataset: Includes 7,203 images reserved for evaluating the final model.

The flow_from_directory() function in Keras ensures that the images are resized, shuffled, and correctly labeled for the model. Augmentation is applied exclusively to the training dataset to maximize its utility and address the imbalance noted in Figure 4.1.

This preprocessing stage ensures that the training dataset is well-prepared, with a focus on augmenting data for classes with fewer images. By building a robust training foundation, the model is better equipped to achieve reliable and accurate emotion recognition in subsequent stages.

4.2. MACHINE LEARNING MODEL RESULT

4.2.1. MACHINE LEARNING MODEL PERFORMANCE

As I mentioned in 3.5.1.1, I have created 3 files based on the epoch number. This is the result of Acuraccy, Loss for Train, Validation, And test dataset for each file.

4.2.1.1. Accuracy

Table 4.1 Accuracy table

Epoch \ Dataset	TRAIN (%)	VALIDATION (%)	TEST (%)
60	74.39	63.39	81.28
70	83.39	64.19	91.32
80	86.34	63.84	91.85

Accuracy Graph for train and validation:

- Epoch 60

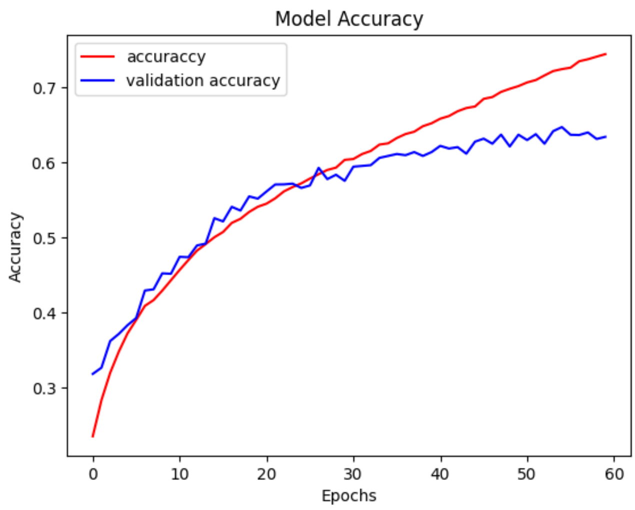


Figure 4.3 Accuracy Graph (Epoch 60)

The training accuracy displayed by the red line in the graph for epoch 60 gradually rises and exceeds 70%. This suggests that given the training data your model is learning successfully. But after the 20th epoch, the validation accuracy (shown by the blue line) fluits more and plateaus at 60%. This implies that the model suffers to generalise to unknown data, suggesting possible overfitting even if it performs well on the training data.

- Epoch 70

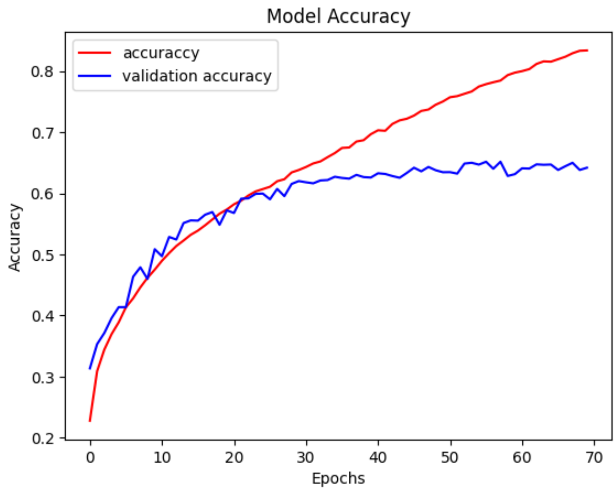


Figure 4.4 Accuracy Graph (Epoch 70)

The training accuracy rises constantly and during period 70 it approaches 80%. This is a noteworthy development implying that the model is enhancing its capacity for learning from the training data. The validation accuracy shows an increasing trend with minor fluctuation before steadying at 60%, much as in the previous graph. The persistent difference between training and validation accuracy shows that overfitting is still a concern even if the model performs well on training data but badly on fresh, unknown data.

- Epoch 80

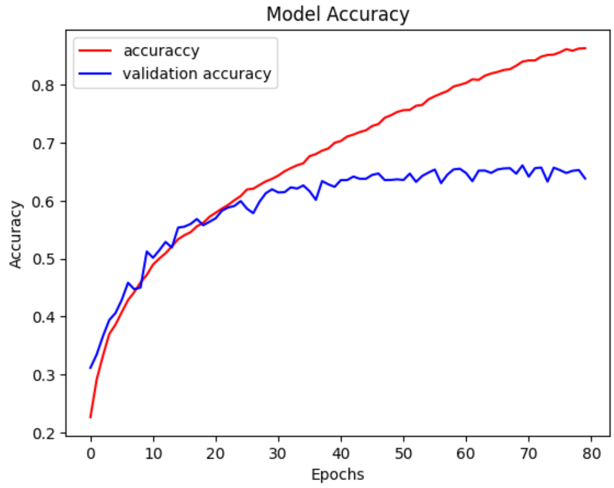


Figure 4.5 Accuracy Graph (Epoch 80)

The training accuracy continuously remains above 80% and exhibits a notable improvement on the graph for epoch 80. This indicates that your models accuracy on the training data is quite high. But after the 20th epoch, the validation accuracy first increases before varying and stabilizing at about 60%. Overfitting is confirmed by the fact that, although training accuracy has improved, validation accuracy has not increased much. This shows that even if the model has mastered the training data, it finds it difficult to perform as well on fresh data.

There is a noticeable upward trend in training accuracy throughout the graphs for epochs 60, 70, and 80, reaching 70%, 80%, and continuously remaining over 80%, respectively. Although the model learns well from the training data, it has trouble generalizing to new data, as seen by the validation accuracy, which stays at about 60% throughout all epochs. This ongoing discrepancy points to overfitting, a situation in which the model works well with training data but poorly with unknown data. Regularization, data augmentation, and early halting are some strategies that might assist lessen this problem and enhance generalization.

4.2.1.2. Loss

Table 4.2 Loss table

Epoch \ Dataset	TRAIN	VALIDATION	TEST
60	0.6847	1.0932	0.5610
70	0.4469	1.1193	0.3370
80	0.3727	1.2442	0.3091

Loss Graph for train and validation:

- Epoch 60

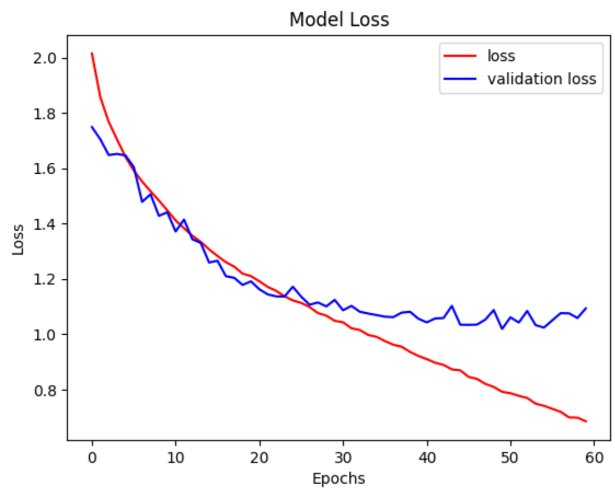


Figure 4.6 Loss Graph (Epoch 60)

Trained over 60 epochs, the first models training loss falls constantly. Consequently, the model is learning from the training data effectively, which reduces its error. Conversely, after around 20 epochs the validation loss—which first decreases—becues to vary. Given the model performs well on the training data but loses consistent performance on new data, this points most likely overfitting.

- Epoch 70

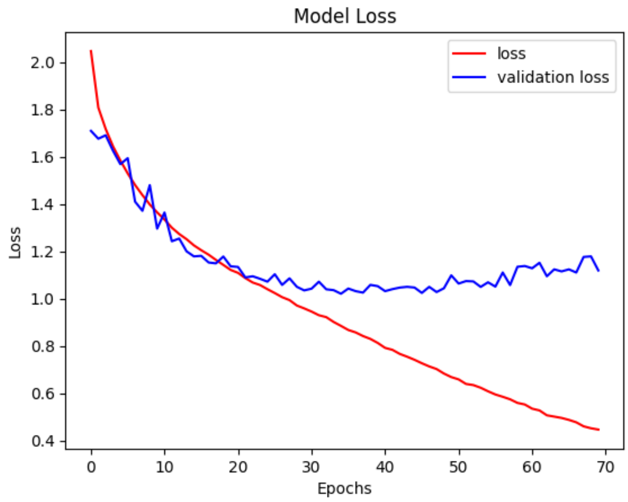


Figure 4.7 Loss Graph (Epoch 70)

Trained for 70 epochs, the second model shows continuous improvement in fitting the training data as the training loss continues in its decrease. First declining before starting oscillations after around 20 epochs, the validation loss shows a trend like the Epoch 60 graph. This shows that the model still finds it difficult to generalise to fresh data, hence stressing the presence of overfitting even with more training epochs.

- Epoch 80

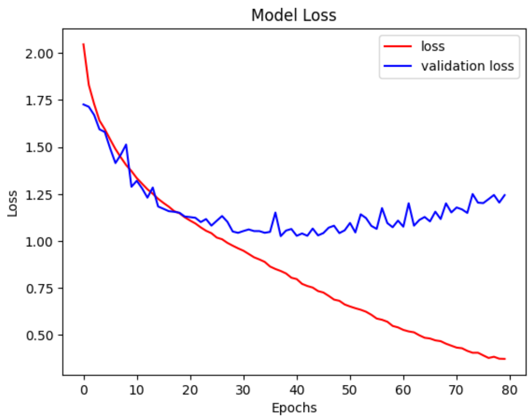


Figure 4.8 Loss Graph (Epoch 80)

Trained over 80 epochs, the third model exhibits a further drop in training loss, suggesting that the model is getting even more suited for the training data. After an early decline, though, the validation loss starts to fluctuate and somewhat rise after around 20 epochs. This implies that when the model memorises the training data too well but finds difficulty to generalise to new data efficiently, overfitting is more noticeable with further training.

Each models loss graph highlights the consistent challenge of overfitting, where the training loss improves steadily, but the validation losss fluctuations suggest unstable performance on new, unseen data. To improve generalization, you might consider implementing techniques like regularization, data augmentation, or early stopping.

After reviewing the performance of the models trained over 60, 70, and 80 epochs, we found that the model trained over 70 epochs strikes the best balance. While all models show effective learning from the training data, they also exhibit signs of overfitting, with validation accuracy plateauing around 60% and validation loss fluctuating after the initial drop.

The model trained over 60 epochs learns well but starts to show overfitting signs early. The model trained over 80 epochs demonstrates strong learning but exhibits more pronounced overfitting. The model trained over 70 epochs maintains high training accuracy, surpassing 80%, with relatively more stable validation performance.

```python
# serialize model structure to JSON
model_json = model.to_json()
with open("model3.json", "w") as json_file:
    json_file.write(model_json)
```

Python

Figure 4.9 Model Serialization to JSON

To save these models for future use, well convert each one to a JSON file. The model trained over 60 epochs will be saved as model1.json, the model trained over 70 epochs as model2.json, and the model trained over 80 epochs as model3.json. This way, you can easily reload and share the models without needing the original code.

4.2.2. PERFORMANCE METRICS

4.2.2.1. Description of Each Models Classification Report :

- Epoch 60

	precision	recall	f1-score	support
angry	0.87	0.90	0.89	958
disgust	0.96	0.93	0.94	111
fear	0.90	0.85	0.87	1024
happy	0.98	0.95	0.96	1774
neutral	0.91	0.91	0.91	1233
sad	0.86	0.90	0.88	1247
surprise	0.93	0.96	0.94	831
accuracy			0.91	7178
macro avg	0.92	0.91	0.91	7178
weighted avg	0.91	0.91	0.91	7178

Figure 4.10 Classification Report (Epoch 60)

The Epoch 60 models categorisation report shows good performance over a range of categories. For most classes, the model has shown great accuracy overall, 91%. It also exhibits great precision and recall. With remarkable accuracy at 0.98 and recall at 0.95, the "happy" class produces an F1-score of 0.96. Other noteworthy performances fall in the "neutral" and "surprise" categories; both have F1-scores above 0.90. The system regulates emotions like "disgust" and "fear," however the "sad" category has somewhat lower accuracy of 0.86.

- Epoch 70

	precision	recall	f1-score	support
angry	0.87	0.90	0.89	958
disgust	0.96	0.93	0.94	111
fear	0.90	0.85	0.87	1024
happy	0.98	0.95	0.96	1774
neutral	0.91	0.91	0.91	1233
sad	0.86	0.90	0.88	1247
surprise	0.93	0.96	0.94	831
accuracy			0.91	7178
macro avg	0.92	0.91	0.91	7178
weighted avg	0.91	0.91	0.91	7178

Figure 4.11 Classification Report (Epoch 70)

This models categorisation results show a 91% general accuracy. The "disgust" category gains in precision to 0.96 and recall to 0.93, therefore producing an amazing F1-score of 0.94 even if accuracy falls somewhat.With an F1-score of 0.96 the "happy" class does rather well. On the other hand, several categories—like "fear" and "sad"—show minor declines in accuracy and recall relative to Epoch 60.

- Epoch 80

	precision	recall	f1-score	support
angry	0.90	0.89	0.90	958
disgust	0.92	0.91	0.91	111
fear	0.90	0.87	0.89	1024
happy	0.98	0.94	0.96	1774
neutral	0.90	0.92	0.91	1233
sad	0.86	0.92	0.89	1247
surprise	0.94	0.95	0.95	831
accuracy			0.92	7178
macro avg	0.92	0.92	0.92	7178
weighted avg	0.92	0.92	0.92	7178

Figure 4.12 Classification Report (Epoch 80)

Epoch 80s categorisation report shows a comeback in general performance with a 92% accuracy level. This epoch mirrors the strengths of Epoch 60, with the “happy” class again achieving a high F1-score of 0.96. The “disgust” class maintains its good performance, while classes like “angry,” “neutral,” and “surprise” exhibit consistent and commendable results. The “sad” class shows improvement in recall at 0.92, contributing to a balanced performance across the board.

4.2.2.2. Differences Between the Models

Several differences emerge when comparing the models across Epochs 60, 70, and 80. The model Epoch 60 shows balanced high performance, particularly in the “happy” and “neutral” classes. In contrast, Epoch 70 reveals a mixed bag: while the “disgust” class achieves its highest metrics, a slight dip in overall accuracy and performance in the “fear” and “sad” classes is observed. Epoch 80 combines the strengths of previous epochs, recovering the overall accuracy to 92% and showing consistent results across all classes, including notable improvements in the “sad” class.

4.2.2.3. Best Model Selection

Considering the performance metrics and consistency across different emotions, the model at Epoch 80 stands out as the best. It combines the high overall accuracy seen at Epoch 60 with improvements in specific classes, particularly the “sad” class. This balanced performance across all metrics makes the model at Epoch 80 the most robust and reliable for classification tasks.

4.2.3. CONFUSION MATRIX

4.2.3.1. Description of Each Confusion Matrix

- Epoch 60

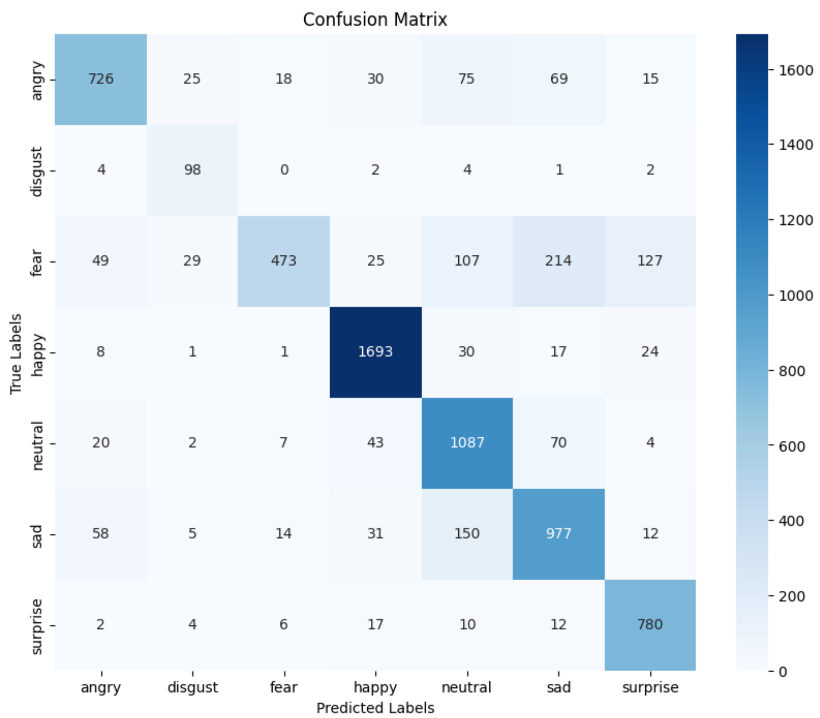


Figure 4.13 Confusion Matrix (Epoch 60)

The confusion matrix for the model Epoch 60 reveals a solid performance across multiple categories. The model has made 1,682 accurate predictions for the "happy" category. Conversely, the "disgust" category has the fewest accurate predictions, at only 103 cases. Misclassifications are apparent, notably with "fear" erroneously classified as "neutral" 26 times and "sad" misclassified as "neutral" 61 times.

- Epoch 70

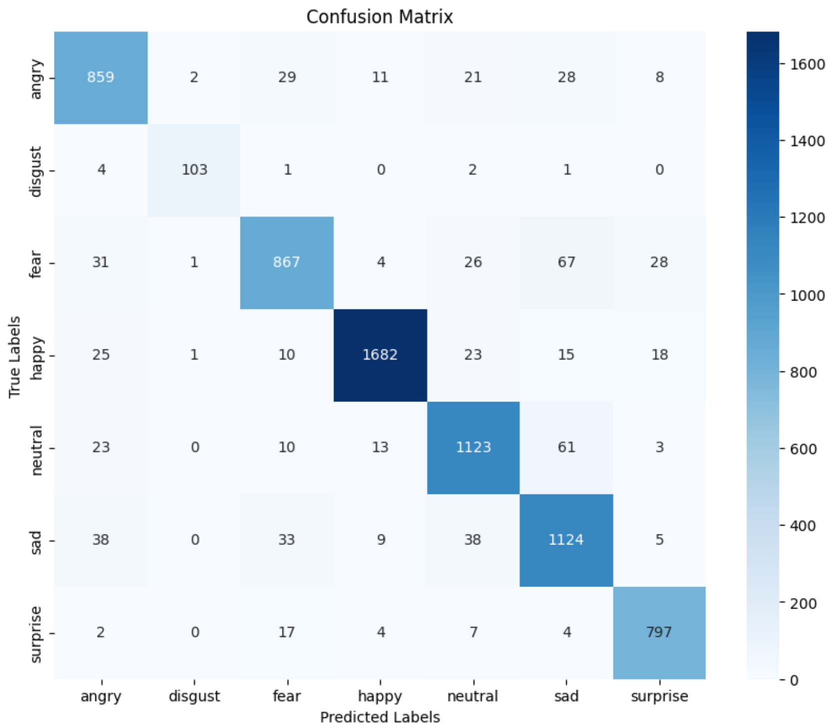


Figure 4.14 Confusion Matrix (Epoch 70)

The confusion matrix for this model indicates sustained high accuracy in the "happy" category, with 1682 accurate predictions. Nonetheless, misclassifications are evident, as "fear" is projected as "sad" 67 instances and "neutral" is predicted as "sad" 61 instances. The "disgust" category exhibits a little enhancement, with 103 correct predictions, but continues to rank as the lowest among the classifications. The models performance across other categories remains consistent with minor variations in misclassification rates.

- Epoch 80

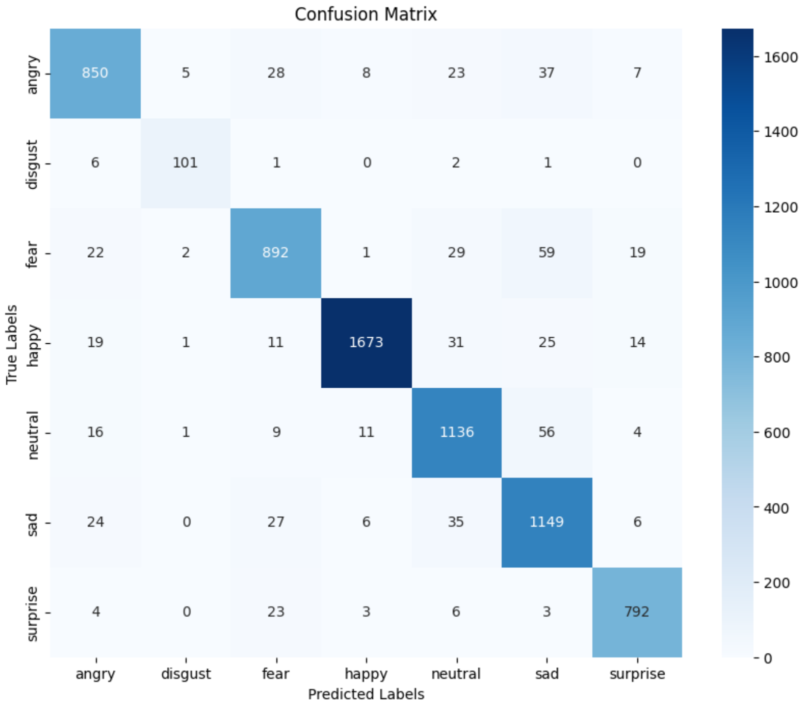


Figure 4.15 Confusion Matrix (Epoch 80)

The confusion matrix for Epoch 80 highlights the highest number of correct predictions for the "happy" category, with 1693 instances, indicating improved performance. The "disgust" category still lags with 98 correct predictions, showing a slight decrease. Misclassifications continue, with "fear" being predicted as "sad" 66 times and "neutral" being predicted as "sad" 54 times. However, the models consistency in accurate classifications, especially in the "happy" and "surprise" categories, is notable.

4.2.3.2. Differences Between the Models

Comparing the confusion matrices across the three epochs, several differences emerge. The model at Epoch 60 shows balanced performance, with high accuracy in the "happy" and "neutral" categories but notable misclassifications in "fear" and "sad." At Epoch 70, the model maintains similar strengths but shows increased misclassifications in "fear" as "sad" and "neutral" as "sad," impacting overall accuracy.

Epoch 80, while improving in the number of correct predictions for the "happy" category, exhibits consistent misclassification patterns, yet shows a slight decrease in performance for the "disgust" category.

4.2.3.3. Best Model Selection

Considering the overall performance and consistency, the model at Epoch 80 stands out as the best among the three. Despite minor misclassifications, it demonstrates the highest number of correct predictions for the "happy" category and maintains balanced performance across other categories. The improvements in accuracy and consistent results make the model at Epoch 80 the most robust and reliable for classification tasks.

4.3. USER INTERFACE

This application consists of three interfaces which are the Cashier Selection Page, Start Page, and Result Page.

4.3.1. CASHIER SELECTION PAGE

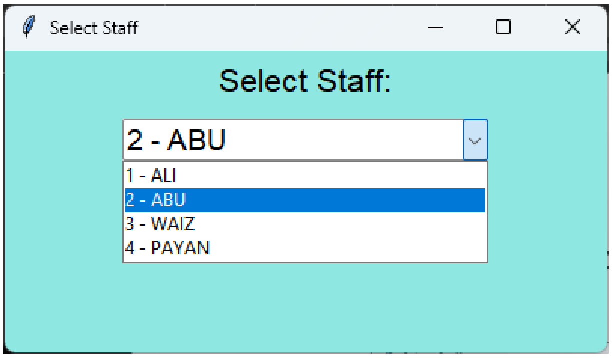


Figure 4.16 UI Cashier List

In Figure 4.16, the application shows a simple interface listing all the staff or cashier IDs along with their names. This data is automatically loaded from the StaffID.csv file.

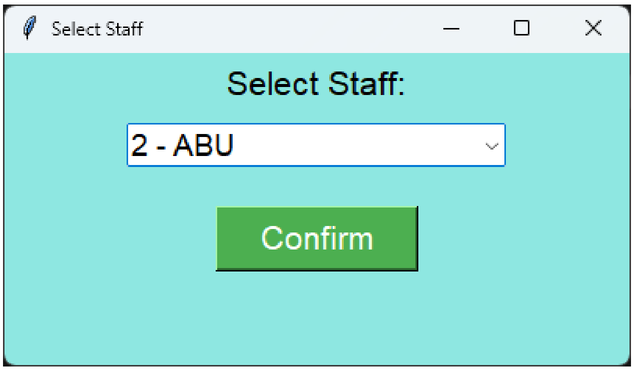


Figure 4.17 UI Cashier Selected

Once a cashier or staff person has been chosen, as seen in Figure 4.17, the user must click the Confirm button to advance. This action ensures the session is linked to the selected staff ID, allowing the application to track the session properly.

4.3.2. START PAGE

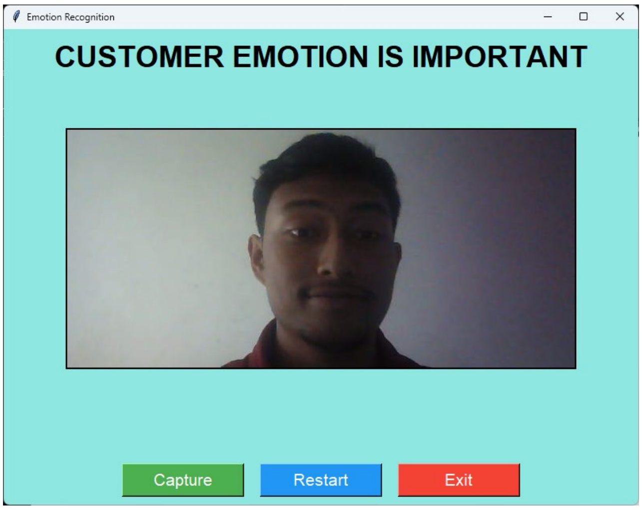


Figure 4.18 UI Start Page

The start page shown in Figure 4.18 shows up following staff ID selection. The interface includes a live WebCam Preview frame and the title "CUSTOMER EMOTION IS IMPORTANT," which sets the tone for the application. There are three buttons: Capture, Restart, and Exit. The Capture button takes a photo to analyze the customers emo tion, while the Restart button resets everything to the start page, including refreshing the WebCam feed. Lastly, the Exit button ends the session and closes the application.

4.3.3. RESULT PAGE

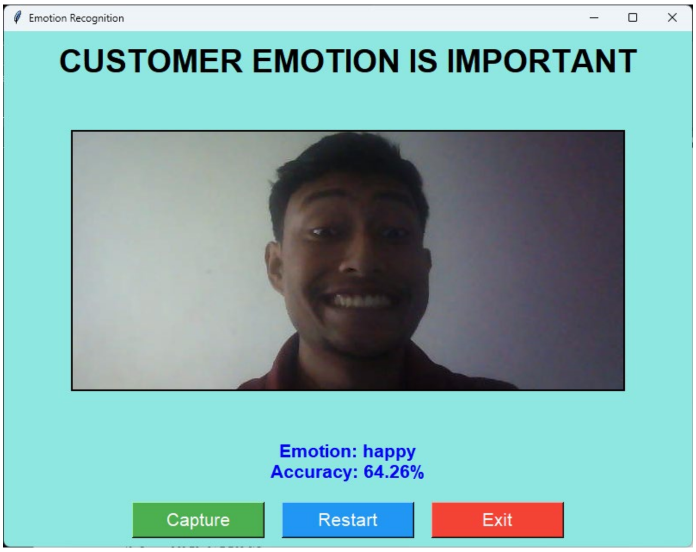


Figure 4.19 UI Result Page

Figure 4.19 shows the UI changing to the result page once the user hits the Capture button. Here, the captured image is displayed along with the emotion recognition results, including the predicted emotion and its accuracy. If the user wants to start over, clicking the Restart button will bring the page back to the initial start screen.

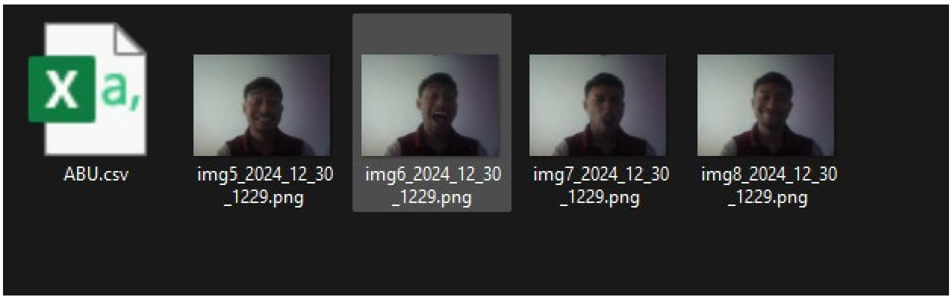
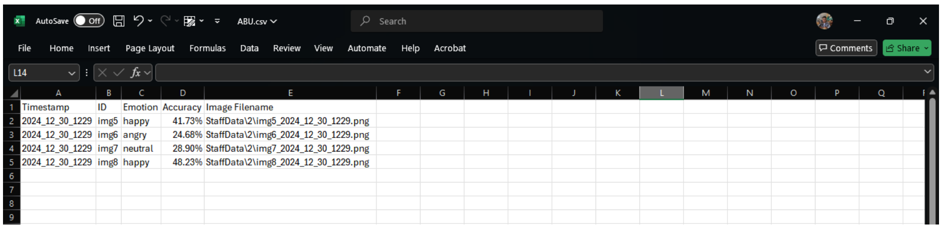


Figure 4.20 Example of Result Files

In addition to displaying results, the system also saves them. As shown in Figure 4.20, the captured images and recognition results are stored in a folder named after the selected staff ID. Each folder contains both the images and a CSV file documenting the results.



Timestamp	ID	Emotion	Accuracy	Image Filename
2024_12_30_1229	img5	happy	41.73%	StaffData\2\img5_2024_12_30_1229.png
2024_12_30_1229	img6	angry	24.68%	StaffData\2\img6_2024_12_30_1229.png
2024_12_30_1229	img7	neutral	28.90%	StaffData\2\img7_2024_12_30_1229.png
2024_12_30_1229	img8	happy	48.23%	StaffData\2\img8_2024_12_30_1229.png

Figure 4.21 Result in CSV File

Figure 4.21 gives an example of the CSV file, which records details like the timestamp of the capture, the image ID, the predicted emotion, the accuracy of the prediction, and the file path of the saved image. These saved results act as valuable feedback for the cashier or staff, helping them better understand and respond to customer emotions.

4.4. SYSTEM TESTING RESULT (TEST CASE)

The table below summarizes the results of the test cases mentioned in section 3.6. Each test case's actual results and status are tracked for further review.

Table 4.3 Test Case Result

Test Case ID	Test Case	Description	Actual Results	Status
TC_01	Emotion Recognition Accuracy	Compare performance between Epoch 60, Epoch 70, and Epoch 80, and identify the best epoch.	Results for Epoch 60: 0.91	SUCCESS
			Results for Epoch 70: 0.91	SUCCESS
			Results for Epoch 80: 0.92	SUCCESS
			Best Epoch: Epoch 80	SUCCESS
TC_02	Real-Time Emotion Detection	The system should accurately detect and classify emotions in real-time.	The results precision is not ideal, and certain emotions remain challenging to recognise.	SUCCESS
TC_03			CSV Files: Saved	SUCCESS

	Save Result Functionality	The system should save identified emotions and accuracies in CSV files and captured images.	Captured Images: Saved	SUCCESS
TC_04	User Interface Functionality	Buttons should function correctly, and detected emotions with accuracy should be displayed.	Confirm Button: Functioned	SUCCESS
			Capture Button: Functioned	SUCCESS
			Restart Button: Functioned	SUCCESS
			Exit Button: : Functioned	SUCCESS
			Accuracy Display: Displayed	SUCCESS

4.5. DISCUSSION

In this chapter, the discussion focuses on the performance and challenges faced by the machine learning model developed for customer emotion recognition in supermarkets. The findings indicate that although the model attains elevated accuracy on training data, it encounters difficulties with generalization, evidenced by the variable validation accuracy and loss during several epochs. A major issue of overfitting is typified by a great performance on training data but a reduced efficacy on fresh data. To handle this, one is advised to use early stopping, data augmentation, and regularisation among other techniques. Emphasising its simplicity and efficiency in recording and displaying emotion detection results, the user interface is evaluated. By establishing strong performance in emotional identification accuracy, real-time detection, result preservation, and user interface operations, the outcomes of system testing help the program to be functional. To improve the practical relevance of the model, the chapter emphasises the requirement of reducing overfitting and strengthening its generalisation ability.

4.6. SUMMARY

Chapter 4 investigates the results and debate around the creation and evaluation of a machine learning system meant to detect consumer emotions in supermarkets. Starting with data preparation, the process looks over and improves the dataset to correct imbalances, especially in the "disgust" category. The chapter then examines the machine learning models performance throughout several epochs, emphasizing concerns of overfitting and the necessity for approaches such as regularization. Furthermore, it delineates the user interface, comprising a cashier selection page, a start page featuring a live webcam preview, and a results page exhibiting emotion detection outcomes. The chapter finishes with the results of system testing, summarizing the applications correctness and functionality.

CHAPTER 5 CONCLUSION AND RECOMMENDATION

This chapter has four sections. The overview of the research and the analysis of the outcomes of the suggested CNN on consumer emotional recognition in supermarkets constitute the first and second sections respectively. It underlines achievements in reaching the projects designated objectives, notes the challenges faced throughout the completion, and supports suggestions for future study and growth in this sector. The aim of the chapter is to present a summary of the completed research projects together with recommendations for improvement of next projects.

5.1. Conclusion

This projects major objective was to create a better system that would map dynamic face expressions to instantly recognise customer emotions. This objective was reached in a systematic way which includes several steps such as data collect and preprocess, model develop and train, and application deploy. The research was able to show that CNNs can be used to classify emotions from images and improve customer service experience in supermarkets. According to the research, implementing the use of emotion recognition technology in supermarkets organize and operate enables the businesses to enhance on understanding customers' feelings and hence improve on the services to deliver. During this project, an application has been developed to detect emotions; these encompass happiness, sadness, anger and surprise. By offering essential information about customer feelings to supermarkets' employees this capability helps the latter to meet customers' needs and preferences in a better way. The intense testing performed throughout the work confirmed the applicability of the proposed CNN model in realistic conditions, thereby confirming the worth of using this kind of method as a real-life tool for increasing customer engagement in the retail sector. Achieving this technology shows an advance in the use of artificial intelligence in enhancing customer service as well as operation in supermarkets.

5.2. Achievement of Objectives

5.2.1. Objective 1: To identify suitable algorithms that detect customers emotions based on facial reactions.

The first objective was concerned with establishing the best algorithms in detecting customers' emotions from facial gestures. To this end, a comprehensive literature review based on several studies of different methodologies, such as classic and deep learning approaches, was done. The evaluation indicated that Convolutional Neural Networks CNNs are best suited in this task because they eliminate the need for most of the feature extraction from images. Several issues were emphasized by the research to indicate issues that determine successful operations of emotion detection algorithms and they included the issue of the quality and variety of the training datasets as well as the issue of the CNN models architecture. When comparing the already developed algorithms it was possible to conclude that CNN algorithms offer the greatest results in comparing emotions in all possible directions at various demographics and with various conditions. This allowed for the definition of a clear path to the two subsequent phases of the engineering of the emotion recognition system, which proves that the objective was met to enhance this system. In general, the identification of the proper algorithms has been a success in achieving this goal and at the same time establishing a solid groundwork for future advancement levels in the subsequent phases of the study while guaranteeing the effectiveness and productivity of the selected methods in real-time environments.

5.2.2. Objective 2: To develop customer emotion recognition systems application based on facial recognition using Convolutional Neural Networks.

Regarding the second objective, the creation of a full-fledged application that would help to analyze the emotional state of a customer with the help of CNNs for face recognition was also conducted. This process encompassed several critical stages: arising within the development of system architecture; data pre-processing; training period for the CNN model; and finally, constructing an interface that will be easily understandable by the final consumers of the product. The dataset containing facial images was gathered and normalized to proper standard and image qualities before label training was done on the facial images. The CNN architecture was build with several layers of convolution designed to introducing spatial hierarchy from facial images efficiently. During the training various hyperparameters were adjusted to conduct better performances without many classification errors. To ensure ease of use and to encourage more and more supermarket staff using this application, the application was developed with a simple format of a user interface whereby the users can be able to gain real time understanding of their customer's emotions within the shortest time possible. Exhaustive testing also established that the application performs as required in various settings for optimal lighting, showing it ready for supermarket environments. The successful development and performance of this application undoubtedly show that this goal has been met; this application can now be seen as a useful tool for enhancing customer interactions based on the evaluation of their emotional reactions.

5.2.3. Objective 3: To evaluate the accuracy of customer emotion recognition using Convolutional Neural Networks.

The projects last objective was to evaluate the performance of the created emotion detection system with performance criteria including accuracy, precision, recall and F1-score. To investigate the categorisation result of emotions like pleasure, sorrow, rage, and surprise, a confusion matrix was developed. Confirming its effectiveness for creating suggestions for usage in supermarkets, where it becomes imperative to pay attention to consumers emotions to improve service quality, an investigation revealed that the CNN model had quite high accuracy in identifying these emotions. Apart from demonstrating the success of this project, this attention to rigour also indicated where further work could be concentrated, for example, on spotting misclassifications which could help improve model architecture, or the training process if necessary; similarity and cross-validation were used as approaches to make sure the model performs well on various partitions of data, so increasing the credibility of the results. Such accuracy rates shown in this study allow one to conclude that this goal has been achieved; moreover, these results show promise of the application of such technology in practical retail contexts where the identification of consumer emotions would enhance services.

5.3. Limitations

Though tremendous progress has been observed during the course of this research work, several limitations are worth to note as a result of this research work. Still, there are some significant drawbacks that, in a way or another, follow from dataset limitations; while considerable efforts have been made to obtain a set of facial images sufficiently diverse on purpose, specific groups may still be underrepresented in the dataset used in this or that incarnation for the model training . This absence of cross facial differences could prove to be a weakness when the model is implemented in different environment with different consumers by geographical location, cultural aspect, etc. Also, differences in lighting conditions as well as a view point of the cam might influence the accuracy of the model in emotion detection in real-life scenarios in supermarkets; variations of these factors impacts the input data with some noise, and, thus, can lead to some mistakes in the emotion identification or weaken the confidence towards the model's prognosis. Furthermore, although CNNs have shown significant success in encoding the tasks of image classification, they have some drawbacks such as overfitting if trained datasets were inadequate or difficulty in training models capable of generalizing across different structures of data; these will be critical to address if realizing success when implementing emotion recognition systems in dynamic retail environment.

5.4. Recommendations for Future Work

There are several suggestions for future research that would be useful for further improvement of customer emotion recognition technology in the retail environment in light of the conclusions that may be drawn from this work. First, it is suggested to extend this study by including more and different amounts of preliminary material containing more varied Nu-Emo expressions of various subjects belonging to different age, gender, nationality, etc.; a cooperation with other scientific institutions or organisms, which have extensive experience in studying emotions, will surely yield more substantial databases helping train more complex and accurate models. Second, further research of how CNN can be combined with other deep learning structures, for example with

Recurrent Neural Networks (RNNs) in order to analyse temporal emotional response characteristics as clients interact in supermarkets in real-time and also provide enhanced knowledge of the temporal context could help to improve the result. Furthermore, one would like to reveal that applying advanced data augmentation methods may improve model resilience to variations observed during deployment; further studies should address improving the integration of additional modalities for emotion recognition other than facial expressions, e.g., voice tone analysis or bodily motion recognition for creating a more complex picture of customer emotional states that would facilitate service strategy development based on identified emotions.

5.5. Summary

In the framework of supermarkets, this chapter offers an overview of the project outcomes based on the goals and objectives of the project as well as highlights the limitations that have been satisfied throughout the development of the project and the directions of further studies connected to customer emotion recognition technologies; moreover, it shows how the application of sophisticated machine-learning algorithms can be successfully used to enhance the communication and quality of customer service in retail outlets by introducing And as more companies rush to find deeper ways of keeping their customers engaged while at the same time satisfying their emerging needs and expectations especially through experience-based marketing, this study offers useful information of how to harness such technologies for success especially in supermarkets which are always under pressure and or risk of being outcompeted by the rapidly changing consumer awareness of their shopping experiences as a whole.

REFERENCES

Ahmadzadeh Nobari Azar, N., Cavus, N., Esmaili, P., Sekeroglu, B., & Aşır, S. (2024). Detecting emotions through EEG signals based on modified convolutional fuzzy neural network. Scientific Reports, 14(1). https://doi.org/10.1038/s41598-024-60977-9

Białek, C., Matiolański, A., & Grega, M. (2023). An Efficient Approach to Face Emotion Recognition with Convolutional Neural Networks. Electronics (Switzerland), 12(12). https://doi.org/10.3390/electronics12122707

Chowdary, M. K., Nguyen, T. N., & Hemanth, D. J. (2023). Deep learning-based facial emotion recognition for human–computer interaction applications. Neural Computing and Applications, 35(32), 23311–23328. https://doi.org/10.1007/S00521-021-06012-8/METRICS

Dennison, J. (2023). Emotions: functions and significance for attitudes, behaviour, and communication. https://doi.org/10.1093/migration/mnad018

Emotion Detection. (n.d.). Retrieved December 2, 2024, from https://www.kaggle.com/datasets/ananthu017/emotion-detection-fer/data

Face expression recognition dataset. (n.d.). Retrieved December 29, 2024, from https://www.kaggle.com/datasets/jonathanoheix/face-expression-recognition-dataset

John Loeppky. (2022). How Psychology Influences Behavior at the Supermarket. https://www.verywellmind.com/how-psychology-influences-behavior-at-the-supermarket-5225195

Kahou, S. E., Michalski, V., Konda, K., Memisevic, R., & Pal, C. (n.d.). Recurrent Neural Networks for Emotion Recognition in Video. https://doi.org/10.1145/2818346.2830596

Khemakhem, F., & Ltifi, H. (2023). Neural style transfer generative adversarial network (NST-GAN) for facial expression recognition. International Journal of Multimedia Information Retrieval, 12(2), 1–12. https://doi.org/10.1007/S13735-023-00285-6/METRICS

Krishna, T., Rai, A., Bansal, S., Khandelwal, S., Gupta, S., & Goel, D. (2013). Emotion recognition using facial and audio features. ICMI 2013 - Proceedings of the 2013 ACM International Conference on Multimodal Interaction, 557–562. https://doi.org/10.1145/2522848.2531746

Kroll, J. A. (2022). ACM TechBrief: Facial Recognition. ACM TechBrief: Facial Recognition. https://doi.org/10.1145/3520137

Lee, M. S., Lee, Y. K., Lim, M. T., & Kang, T. K. (2020). Emotion recognition using convolutional neural network with selected statistical photoplethysmogram features. Applied Sciences (Switzerland), 10(10). https://doi.org/10.3390/app10103501

Malik, S., Kumar, P., & Raman, B. (2021). Towards interpretable facial emotion recognition. ACM International Conference Proceeding Series. https://doi.org/10.1145/3490035.3490271

Mehendale, N. (2020). Facial emotion recognition using convolutional neural networks (FERC). SN Applied Sciences, 2(3), 1–8. https://doi.org/10.1007/S42452-020-2234-1/TABLES/3

Moravčík, E., & Basterrech, S. (2022). Image-Based Facial Emotion Recognition Using Convolutional Neural Networks and Transfer Learning. Lecture Notes in Networks and Systems, 330 LNNS, 3–14. https://doi.org/10.1007/978-3-030-87178-9_1

Muhammad Zulkifli Nawfal, & Barlian Henryranu Prasetio. (2023). Face Emotion Recognition System of Customer Service using CNN Based on Embedded System. International Conference on Sustainable Information Engineering and Technology (SIET 2023).

Murugesan, S. (2008). Web Application Development: Challenges And The Role Of Web Engineering. In O. and S. D. and O. L. Rossi Gustavo and Pastor (Ed.), Web Engineering: Modelling and Implementing Web Applications (pp. 7–32). Springer London. https://doi.org/10.1007/978-1-84628-923-1_2

Qu, D., Dhakal, S., & Carrillo, D. (2023). Facial Emotion Recognition using CNN in PyTorch. http://arxiv.org/abs/2312.10818

Rani, R., Arora, S., Verma, V., Mahajan, S., & Sharma, R. (2024). Enhancing facial expression recognition through generative adversarial networks-based augmentation. International Journal of System Assurance Engineering and Management, 15(3), 1037–1056. https://doi.org/10.1007/S13198-023-02186-7/METRICS

Sahana, D., Varsha, K. S., Sen, S., & Priyanka, R. (2023). A CNN-Based Approach for Facial Emotion Detection. Lecture Notes in Networks and Systems, 627 LNNS, 1–10. https://doi.org/10.1007/978-981-19-9858-4_1

Saia Sheila M. AND Nelson, N. G. A. N. D. Y. S. N. A. N. D. P. S. A. N. D. V. M. (2022). Ten simple rules for researchers who want to develop web apps. PLOS Computational Biology, 18(1), 1–18. https://doi.org/10.1371/journal.pcbi.1009663

Sarmiento, A. L., Liwanag, J. L., Aviñante, R. R., & Dadiz, B. G. (2018). Dimension Reduction Techniques Analysis on Image Processing for Facial Emotion Recognition. https://doi.org/10.1145/3232651.3232669

Senarath, U. S. (2021). Waterfall Methodology, Prototyping and Agile Development. https://doi.org/10.13140/RG.2.2.17918.72001

Smitha, K. G., & Vinod, A. P. (2015). Facial emotion recognition system for autistic children: a feasible study based on FPGA implementation. Medical and Biological Engineering and Computing, 53(11), 1221–1229. https://doi.org/10.1007/s11517-015-1346-z

Tara Ramroop, S. W. (2024). Customer feedback: 7 strategies to collect and leverage it. https://www.zendesk.com/blog/customer-feedback-hear-voice-customer/

Theses, G., & Lawless, M. E. (2014). Scholar Commons Checking Out: A Qualitative Study of Supermarket Cashiers' Emotional Response to Customer Mistreatment CORE View metadata, citation and similar papers at core. https://scholarcommons.usf.edu/etd

Thomas, A., & Garland, R. (2004). Grocery shopping: List and non-list usage. Marketing Intelligence & Planning, 22, 623–635. https://doi.org/10.1108/02634500410559015

Vit Horky. (2019). 6 Core Emotions in Customer Experience and Why They Matter | NICE. https://www.nice.com/blog/6-core-emotions-in-customer-experience-and-why-they-matter

Yin, Y., Ayoub, M., Abel, A., & Zhang, H. (2023). A Dynamic Emotion Recognition System Based on Convolutional Feature Extraction and Recurrent Neural Network. Lecture Notes in Networks and Systems, 543 LNNS, 134–154. https://doi.org/10.1007/978-3-031-16078-3_8