



An Integrated Technical and
Fundamental Stock Analysis Model for
the Saudi Stock Market (Tadawul)

Prepared By

Name: Malek M. Atarji

ID No.: 252000474

Submitted in fulfillment of the requirements for the degree Master of
Science in Artificial Intelligence

At

College of Informatics

Midocean University

UAE

Supervisor Name: Dr. Ashraf El-Sayed

College of Informatics, Midocean University, Comoros

Email: 252000474@st.midocean.edu.km

Acknowledgements

I would like to express my sincere gratitude to Midocean University for providing the academic environment and institutional support necessary for the successful completion of this research.

I am especially grateful to my supervisor, Dr. Ashraf El-Sayed, for his continuous guidance, professionalism, and invaluable academic insight throughout all stages of this study. His constructive feedback and critical evaluations significantly enhanced the academic rigor and intellectual depth of this work.

I also acknowledge the contributions of previous scholars and researchers whose theoretical and empirical studies provided the foundation upon which this research is built.

Finally, I extend my appreciation to my colleagues and fellow researchers for their valuable discussions, encouragement, and support throughout the research process.

Abstract

Currently available literature on approaches to predicting trading signals mainly utilizes either technical analysis of market data or analyses of the fundamental values of the companies listed on the market. Each of these approaches has its advantages and disadvantages for applying to the prediction of trading signals.

The dataset utilized in this project consists of five years of trading and financial data for 20 companies listed on the Tadawul All Share Index. The raw data contained 24,966 trades but were cleaned and preprocessed to contain 19,742 trades. The class imbalance for the data was moderate, with around 32.00% of trades being successful trades, and around 25.78% of trades being successful within the out-of-sample testing data. A successful trade is defined as one in which the share price of the company increased 5% (take-profit) before dropping 2% (stop-loss) in a 5–10-day time period.

A supervised machine learning approach was used to learn the characteristics of trades that were successful. Two models were utilized for prediction: a bidirectional Long Short-Term Memory (BiLSTM) neural network with 60-day sequences of data, and an XGBoost model that used technical and fundamental financial ratio features. The two models were combined with a weighted Hybrid.

The model was tested with a backtesting algorithm that factored in the costs of trading at 0.15% per trade, and tested with a range of thresholds for the probability that a trade would be successful between 0.50 and 0.75. Each of the individual models was compared to the Hybrid of the two models. This improvement is primarily driven by the model's ability to filter low-confidence trades rather than The Hybrid model had the best results for all probabilities tested, outperforming the individual models in risk-adjusted trading performance. The best performance was achieved at the 0.75 threshold.

The outcome of this study indicates that utilizing technical and fundamental analyses within an artificial intelligence framework can enhance trading signals. Furthermore, the models are entirely based upon publicly available market data for the Saudi Stock Market and comply with the regulations of the Capital Market Authority (CMA) of Saudi Arabia.

The outcomes of this study contribute to the existing literature on AI-based trading systems in that it presents a framework with economic performance as its main

goal. Additionally, the use of this algorithm within the Saudi Stock Market indicates its potential use within other markets around the world as well.

Executive Summary

Presented is a Hybrid framework for short-term trading signals within the Saudi Stock Market. The framework utilizes AI models based on technical and fundamental analysis of the Tadawul markets. Results from five years of market data indicate that the hybrid AI model outperformed standalone models in terms of both predicted accuracy and financial performance metrics. Furthermore, the framework is reproducible, uses publicly available market data, and complies with Saudi Capital Market Authority regulations.

List of Abbreviations

AI – Artificial Intelligence

AUC – Area Under the Curve

CMA – Capital Market Authority (Saudi Arabia)

CVaR – Conditional Value at Risk

D/E – Debt-to-Equity Ratio

DL – Deep Learning

EMA – Exponential Moving Average

EMH – Efficient Market Hypothesis

F1 – F1 Score

GARCH – Generalized Autoregressive Conditional Heteroskedasticity

LSTM – Long Short-Term Memory

MA – Moving Average

MACD – Moving Average Convergence Divergence

MDD – Maximum Drawdown

ML – Machine Learning

NI – Net Income

OHLCV – Open, High, Low, Close, Volume

OLS – Ordinary Least Squares

P/E – Price-to-Earnings Ratio

PM – Profit Margin

PR – Precision-Recall

RF – Random Forest Model

ROE – Return on Equity

ROC – Receiver Operating Characteristic

RSI – Relative Strength Index

SE – Stockholders' Equity

SL – Stop-Loss

SVM – Support Vector Machine

TASI – Tadawul All Share Index

TP – Take-Profit

TR – Total Revenue

VaR – Value at Risk

XGBoost – Extreme Gradient Boosting

Table of Contents

Contents

Acknowledgements.....	2
Abstract.....	3
Executive Summary	4
List of Abbreviations	5
Table of Contents	7
Chapter 1: Introduction	10
1.1 Background and Problem Definition	10
1.2 Research Methodological Positioning	10
1.3 Research Objectives.....	11
1.4 Research Hypotheses	11
1.5 Scope of the Study	12
1.6 Research Contributions.....	12
1.6.1 Model Development Contribution.....	12
1.6.2 Trading Framework Contribution	12
1.6.3 Interpretability Contribution.....	13
1.6.4 Generalizability Contribution.....	13
1.6.5 Literature Contribution	13
1.7 Thesis Organization	13
1.8 Research Clarification	14
Chapter 2: Background and Related Work	15
2.1 Introduction.....	15
2.2 Stock Market Forecasting	15
2.3 Fundamental Analysis	15
2.4 Technical Analysis	16
2.5 Machine Learning in Finance	16
2.6 combined into a weighted hybrid model	16
2.7 Trading-Oriented Evaluation Metrics	17
2.8 Evolution of Financial Prediction Models.....	17
2.9 Machine Learning and Deep Learning Studies.....	18

2.10 Critical Evaluation of Prior Studies	18
2.11 Hybrid Models in Literature	18
2.12 Studies in Emerging and Saudi Markets	18
2.13 Research Gap	18
2.14 Comparative Analysis with Existing Studies	21
2.15 Positioning of the Present Study	21
2.16 Chapter Summary	21
Chapter 3: Proposed Approach (Methodology)	22
3.1 Overview of the Hybrid Framework	22
3.2 Dataset Description	22
3.3 Data Preparation	23
3.4 Feature Engineering	24
3.4.1 Technical Features	24
3.4.2 Fundamental Features	25
3.5 Trading Constraints Justification	25
3.6 Model Architecture	25
3.6.1 XGBoost Model	25
3.6.2 Bidirectional LSTM Model	26
3.7 Hybrid Integration	26
3.8 Mathematical Formulation	27
3.9 Backtesting Framework	27
3.10 Sharpe Ratio	28
3.11 Performance Metrics	28
3.12 Reproducibility	28
3.13 Ethical and Regulatory Compliance	29
3.14 Chapter Summary	29
Chapter 4: Experimental Results and Discussion	30
4.1 Experimental Setup	30
4.2 Baseline Model Performance	30
4.2.1 Interpretation of Classification Performance	30
4.3 Hybrid Model Performance	31
4.3.1 Risk-Adjusted Trading Performance	31

4.4 Key Insight.....	32
4.5 Threshold Sensitivity Analysis	32
4.6 Risk-Oriented Perspective.....	33
4.7 Impact of Transaction Costs	34
4.8 Equity Curve and Trading Dynamics	34
4.9 Distribution of Trade Returns	35
4.10 Mechanism Behind Hybrid Model Superiority	35
4.11 Economic Interpretation	36
4.12 Limitations.....	36
4.13 Practical Deployment.....	37
4.14 Chapter Summary	37
Chapter 5: Conclusion and Future Work	38
5.1 Conclusions.....	38
5.2 Theoretical Contribution.....	38
5.3 Practical Implications	39
5.4 Implementation Constraints	39
5.5 Generalizability of the Framework	39
5.6 Future Research Directions.....	40
5.7 Final Remarks	40
References.....	41

Chapter 1: Introduction

1.1 Background and Problem Definition

The Saudi Stock Market, also known as Tadawul, is one of the largest stock markets in the Middle East and North Africa (MENA) region.

The Saudi Stock Market has become increasingly liberalized in recent years with various reforms introduced as part of the Vision 2030 initiative. These reforms have increased the number and size of the companies that trade on the market and increased the complexity of trading activity within the market.

Despite the benefits of such reforms, the increased complexity of the market has increased the volatility of the market. Furthermore, the volatility has introduced nonlinearity in the price movements of the stocks that trade on the market.

Investors in the stock market typically employ two main methods to make trading decisions within the market: technical analysis or fundamental analysis of the stocks that they trade. Technical analysis is beneficial for generating short-term trading signals. However, it does not provide information about the financial strength of the companies that trade on the market. Fundamental analysis of stocks provides an understanding of the financial strength of those companies. However, it fails to provide trading signals for short-term trading activities.

Furthermore, the bulk of machine learning literature is based on the use of statistical models to maximize the accuracy of trading signals. However, the stock markets that are regulated in the Kingdom of Saudi Arabia require that trading systems are easily interpretable and comply with specific regulations. Hence, there is a need for a prediction model that is based on regulation compliance and can generate short-term trading signals for the Saudi Stock Market.

1.2 Research Methodological Positioning

This study will use experimental research methods to test the performance of various models. Furthermore, the study will employ machine learning methods to develop these models.

Rather than using the stock market's price movements to determine trading signals, this research will use the information in the stocks' financial reports. Furthermore,

the problem of trading signal generation will be represented as a classification problem, where the classification of trades will be based on whether a predefined target return is achieved before a stop-loss is triggered for a stock during a trading period.

Performance of the models will be evaluated using a set of financial metrics that represent the economic performance of the trading strategy rather than statistical accuracy measures. Furthermore, a method will be used that ensures that the model is tested on data that was available to traders at the time of trading (chronological out-of-sample testing) to ensure that it can be used for making trading decisions.

1.3 Research Objectives

The objective of this research is to develop a trading signal model for the Saudi Stock Market that is based on artificial intelligence.

Furthermore, this model will have the following sub-objectives:

To develop a model that can combine technical and fundamental indicators of company strength.

To formulate a target that can be used to train the model based on economic trading logic.

To test and evaluate the performance of the model using financial metrics.

To ensure that it complies with regulations of the Saudi Stock Market by using publicly available data.

To develop a framework for the model that can be replicated for other trading applications.

1.4 Research Hypotheses

Hypothesis 1: The artificial intelligence model that is developed for this research will exhibit risk-adjusted returns that are superior to those of standalone technical or fundamental models.

Hypothesis 2: The artificial intelligence model will experience less drawdown in its returns than the standalone models.

Hypothesis 3: The signals that are produced by the model will be based on public and structured data, and these signals will be sufficient to develop AI trading systems that can effectively trade in the Saudi Stock Market.

Hypothesis 4: The developed model will exhibit stability in its performances under validation on different market conditions.

1.5 Scope of the Study

The main focus of this study will be on 20 of the most liquid companies of the Tadawul stock exchange between 2021 and 2025.

Furthermore, the trading signals that will be tested will be for short-term trading strategies within a specific time window. The strategy will be limited to a target return of +5% of the trading position of the investor and a stop-loss of -2% of the trading position of the investor within a time window of between 5 and 10 trading days.

Furthermore, these trading signals will be developed as a framework for decision-making that can be applied to the Saudi Stock Market. However, they are not financial advice for investors.

Furthermore, while the models will exclude trading costs such as transaction costs and slippage, these limitations will be discussed in Chapter 5.

1.6 Research Contributions

1.6.1 Model Development Contribution

The model that will be developed will be a hybrid model that includes two main machine learning methods. The Long Short-Term Memory (LSTM) model will be used to analyze temporal data of the stock prices. The XGBoost machine learning model will be used to analyze the financial data of the companies whose stocks trade in the Saudi Stock Market.

1.6.2 Trading Framework Contribution

Furthermore, rather than the common approach of using an AI model to predict stock price movements, the trading signals will be based on a predefined target profit and stop loss for each position.

1.6.3 Interpretability Contribution

The developed model will also provide interpretations of each of the trading signals in terms of the importance of each variable in the model's determination of the trading signal. Hence, it will provide interpretable trading signals for traders.

1.6.4 Generalizability Contribution

Furthermore, the model is generalizable to different firms in the Saudi Stock Market whose stocks trade on the Tadawul exchange.

1.6.5 Literature Contribution

Finally, another contribution of this study is the desire to contribute to the literature on stock market trading by filling the existing gaps in the literature on Saudi Stock Market trading and using an AI model that adheres to the regulations of the stock exchange.

1.7 Thesis Organization

The remainder of this thesis is organized as follows.

Chapter 2 presents the background and related work in the area of artificial intelligence and trading signals in the stock market.

Chapter 3 presents the Hybrid methodology for this research.

Chapter 4 presents the results of the experiments that were performed on the models.

Chapter 5 draws conclusions from the presented results.

1.8 Research Clarification

The research that will be performed will be based on the structured and quantitative data that is publicly available for the Saudi Stock Market.

Hence, there will be no use of unstructured data in the study. All variables will be based on publicly available financial data and trading price data.

Model: "sequential"

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 20, 64)	20,224
dropout (Dropout)	(None, 20, 64)	0
lstm_1 (LSTM)	(None, 32)	12,416
dropout_1 (Dropout)	(None, 32)	0
dense (Dense)	(None, 1)	33

Figure 1.1: Research Gap – Technical vs Fundamental Fragmentation

Chapter 2: Background and Related Work

2.1 Introduction

This chapter presents the theoretical background and related work in the area of stock market prediction using AI. It reviews the theories of fundamental and technical analysis as well as the various models of machine learning. Furthermore, it also presents relevant studies conducted in various countries, including Saudi Arabia, to identify the existing gaps in the literature.

2.2 Stock Market Forecasting

Stock market forecasting is a process of determining the future price movement of stocks based on available information from past movements. Stock markets are characterized by non-linear and non-stationary time series data due to the variety of factors influencing stock prices. Moreover, the challenges of stock market forecasting are more pronounced in emerging markets like the Saudi Stock Market.

2.3 Fundamental Analysis

Fundamental analysis evaluates the intrinsic value of a company by analyzing its financial data, such as financial ratios. Commonly used financial ratios include:

- Price-to-earnings (P/E) ratio
- Return on equity (ROE)
- Debt-to-equity (D/E) ratio

This approach has been proven to be successful in determining the long-term value of stocks. However, it is not effective in providing traders with information regarding the short-term movements of stock prices.

2.4 Technical Analysis

Contrary to fundamental analysis, technical analysis determines the future price of stocks by studying past price and volume movements.

Various technical indicators are used, including the relative strength index (RSI), moving averages, MACD indicators, and Bollinger bands.

Technical analysis is useful for short-term movements but does not account for the financial strength of the company.

2.5 Machine Learning in Finance

Machine learning models can identify non-linear relationships in large data sets.

Various models have been applied, including Random Forest, support vector machines (SVM), and gradient boosting models.

Deep learning models such as long short-term memory (LSTM) networks have shown potential for stock market analysis. However, the majority of models rely solely on price data.

2.6 combined into a weighted hybrid model

Hybrid models use various data sources and models to improve accuracy.

The approach here uses a weighted Hybrid of two models:

an LSTM for technical analysis and an XGBoost model for fundamental analysis.

The final stock price is determined by the weighted combination of the two models.

Technical analysis covers short-term movements, while fundamental analysis covers long-term value.

By combining these approaches, the model aims to improve trading decision quality.

Comparison of Stock Market Prediction Approaches			
Technical Analysis vs Fundamental Analysis vs Hybrid Approach			
Aspect	Technical Analysis	Fundamental Analysis	Hybrid Model (Proposed)
Data Type	Price & Volume (High-frequency)	Financial Statements (Low-frequency)	Both Price Data & Financials
Time Horizon	Short-term (Days/Weeks)	Long-term (Months/Years)	Short to Medium-term
Key Indicators	RSI, MACD, MA, Bollinger Bands	P/E, ROE, D/E, Cash Flow	Combined Technical & Fundamental
Strengths	Market timing, Entry/Exit signals	Company valuation, Intrinsic value	Timing + Valuation
Weaknesses	Ignores company health, False signals	Ignores market sentiment, Lagging data	Complex integration, Model overfitting

Source: Academic Research 2024

Figure 2.1: Comparison of Technical, Fundamental, and Hybrid Approaches

2.7 Trading-Oriented Evaluation Metrics

Most literature on stock market prediction using ML focuses on accuracy metrics. This study uses trading performance metrics, including:

- Sharpe Ratio
- Maximum Drawdown
- Win Rate
- Equity Curve Stability

The target variable is defined economically: a trade is successful if +5% is achieved before -2% within 5–10 trading days.

2.8 Evolution of Financial Prediction Models

Early models were linear econometric approaches such as ARIMA and regression, supported by the Efficient Market Hypothesis (Fama, 1970). Empirical evidence showed nonlinear behavior in financial markets, leading to the adoption of machine learning techniques.

2.9 Machine Learning and Deep Learning Studies

Breiman (2001) introduced powerful Hybrid methods, and Chen & Guestrin (2016) presented gradient boosting. Fischer & Krauss (2018) showed that LSTM models outperform traditional approaches in finance. However, most studies rely only on technical indicators. Transformer-based architectures, such as the Informer and Time Series Transformer, have demonstrated strong performance in modeling long-range dependencies in financial time series (Zhou et al., 2021).

2.10 Critical Evaluation of Prior Studies

Despite progress, several limitations remain. First, many studies emphasize predictive accuracy over financial performance. Second, many rely exclusively on technical indicators, ignoring firm-level financial information. Third, few studies incorporate explicit trading rules, such as take-profit and stop-loss constraints.

2.11 Hybrid Models in Literature

Hybrid approaches combine multiple data sources and models. However, most existing studies focus on accuracy rather than trading performance, lack explicit trading logic, and ignore risk-adjusted evaluation.

2.12 Studies in Emerging and Saudi Markets

Research in emerging markets, including Saudi Arabia, shows increasing adoption of AI models. However, existing studies often use only technical indicators, do not integrate financial ratios, and lack risk-based evaluation.

2.13 Research Gap

A multidimensional gap exists:

- lack of integration between technical and fundamental analysis
- focus on statistical accuracy rather than trading performance
- absence of explicit trading rules
- limited consideration of regulatory compliance.

This study addresses these limitations with a hybrid, risk-aware, and regulation-compliant framework.

Figure 2.2 illustrates the research gap between technical and fundamental analysis, and the Hybrid model addresses this limitation.

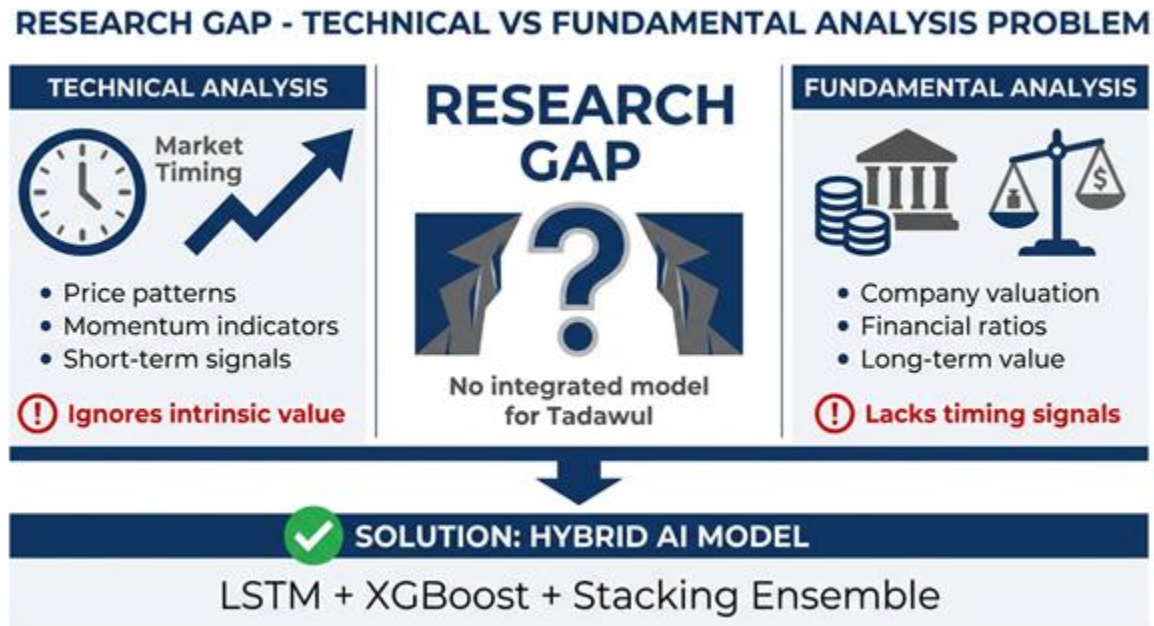


Figure 2.2: Research Gap and Hybrid Model

Author	Market	Data Type	Model Used	Key Contribution	Limitation
Nti et al. (2020)	Global	Technical & Fundamental	ML Review	Comprehensive classification of prediction methods	Limited focus on integrated models
Daradkeh (2022)	UAE	Sentiment + Financial	CNN-BiLSTM	Multi-feature fusion improves prediction accuracy	Relies on sentiment data
Al-Nefaie & Aldhyani (2022)	Saudi	Price Data	LSTM, MLP	Strong predictive performance in Tadawul	Focus on price prediction only
Alsulmi & Al-Shahrani (2022)	Saudi	Market Data	ML Framework	Automated trading decision system	Limited integration of financial ratios
Ansari et al. (2024)	Global	Technical + Fundamental	RL (A2C, DDPG)	Adaptive trading strategy using RL	Complex and unstable training
Daoria et al. (2024)	Saudi	Sentiment + Price	LSTM, SVM	Combines sentiment and ML for prediction	Heavy reliance on social media noise
Khurayshi & Alharbi (2025)	Saudi	Price Data	ML Models	Comparison of ML algorithms in Saudi market	Weak generalization across datasets
Alsinglawi et al. (2025)	Saudi	Financial Data	AI Models	FinTech-based prediction approach	Limited evaluation on trading performance
Saberironaghi et al. (2025)	Global	Mixed	ML/DL Review	Comprehensive review of ML/DL models	Lacks economic evaluation metrics
Alhazmi et al. (2025)	Saudi	Financial Reports	AI Analysis	Impact of AI on financial reporting quality	Not focused on trading prediction

Table 2.1 – Enhanced Literature Review Comparison

2.14 Comparative Analysis with Existing Studies

The existing literature on stock market prediction using AI primarily utilizes technical indicators and ML models to improve accuracy. However, most do not incorporate financial fundamentals or trading rules. The main contribution of this research is a framework that integrates financial data, trading rules, and risk metrics for evaluating the models.

2.15 Positioning of the Present Study

This study identifies gaps in the literature and presents a framework addressing them. Many studies rely on technical indicators and focus on accuracy rather than live-market performance. They also lack trading rules for the stocks being predicted. The framework herein combines multiple data types, includes trading rules, and uses risk metrics to evaluate models, aiming to advance AI models used in stock market trading.

2.16 Chapter Summary

This chapter reviewed theories of technical and fundamental analysis and various ML models, including studies in Saudi Arabia, to identify gaps. The need for the Hybrid framework is established. Chapter 3 presents the Hybrid methodology.

Chapter 3: Proposed Approach (Methodology)

3.1 Overview of the Hybrid Framework

This chapter describes the architecture and implementation of the Hybrid AI framework for trading signal generation for the Saudi stock market (Tadawul). The framework combines technical and fundamental data sources and employs two complementary machine learning models.

In contrast to existing models, the Hybrid approach aims to predict trades rather than prices by incorporating trading constraints. The overall framework consists of data acquisition, preprocessing, feature engineering, model training, model ensembling, and backtesting against realistic trading conditions.

3.2 Dataset Description

The dataset comprises five years (2021-2025) of trading and financial data for 20 companies in the Tadawul All Share Index.

The raw dataset had 24,966 observations, which were reduced to 19,742 observations through cleaning.

The dataset consists of companies from various sectors to ensure diversity:

Sector	Number of Companies
Banking	5
Petrochemicals	5
Telecommunications	5
Energy	5

Each observation contains a combination of technical indicators derived from price data and fundamental financial ratios derived from financial statements.

The target variable is defined in the context of trading outcomes: A trade is classified as positive (1) if a +5% return (take-profit) occurs before a -2% stop-loss within a holding period of 5 to 10 trading days.

The dataset has a moderate degree of class imbalance:

- **32.00% positive trades in the cleaned dataset**
- **25.78% positive trades in the out-of-sample test set**

Data were collected from publicly available sources such as Yahoo Finance and validated against official market data where applicable.

3.3 Data Preparation

3.3.1 Data Cleaning

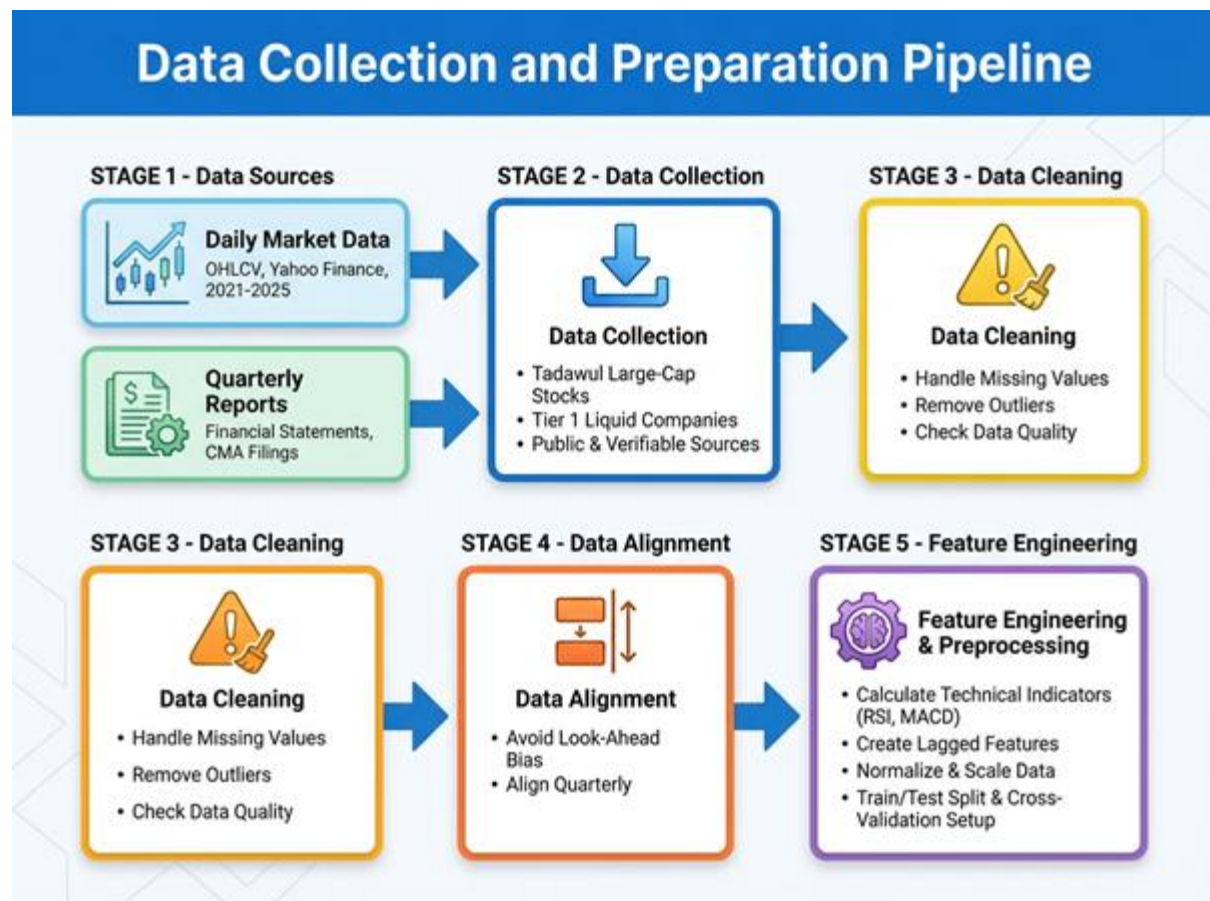


Figure 3.2: Data Collection and Preparation Pipeline

The raw data contain missing values for both price and financial variables. Stocks with missing critical price observations were removed. Remaining missing values

in financial features were imputed using median values on a per-company basis. All variables were typed to numerical values.

3.3.2 Leakage Prevention

A strict chronological framework is applied to eliminate look-ahead bias.

Trading signals are generated using information available up to the market close of day t , but trades are executed at the opening price of day $t+1$.

Intraday high and low prices are used to determine take-profit and stop-loss prices only during day $t+1$ and all trading days following day t .

By using only information that is available to traders during the market trading days, the back-testing of trading systems ensures that the results are realistic and valid.

3.3.3 Standardization

Standardization techniques were applied to the features using StandardScaler without data leakage by fitting it only to training data.

This transformation ensures numerical stability for the model:

$$Z = (X - \mu) / \sigma$$

where Z is the standardized value, X is the feature value, μ is the mean, and σ is the standard deviation.

3.4 Feature Engineering

Feature engineering combines technical and fundamental features.

3.4.1 Technical Features

The following technical features were engineered:

- Returns
- Moving averages (simple and exponential)
- Distance to moving averages
- RSI (Relative Strength Index)
- MACD and its signal line
- Different volatility measures

3.4.2 Fundamental Features

The following fundamental features were engineered:

- Return on Equity (ROE)
- Debt-to-Equity ratio
- Profit Margin
- Earnings Per Share (EPS)

These features are calculated with stable division techniques to avoid numerical issues.

3.5 Trading Constraints Justification

The following trading parameters were used:

- Take-profit (TP): +5%
- Stop-loss (SL): -2%

These parameters allow a risk-reward ratio of 2.5, a standard ratio for short-term trading strategies.

The holding period for trades is kept between 5 and 10 trading days to limit exposure.

3.6 Model Architecture

3.6.1 XGBoost Model

The XGBoost model was trained on the entire tabulated feature space after preprocessing and feature integration.

Training parameters include:

- $n_estimators = 400$
- $max_depth = 6$
- $learning_rate = 0.05$

Class imbalance was addressed using *scale_pos_weight* calculated on the training set.

3.6.2 Bidirectional LSTM Model

The following sequential model was implemented as a Bidirectional LSTM (BiLSTM) using 60-day rolling windows as input sequences.

The architecture contains:

- Bidirectional LSTM layer (64 units; returns sequences)
- Dropout layer (0.5 rate)
- Bidirectional LSTM layer (32 units)
- Dropout layer (0.5 rate)
- Dense output layer with sigmoid activation

The model is trained with binary cross-entropy loss using the Adam optimizer with early stopping based on validation AUC.

3.7 Hybrid Integration

The final prediction is created using a weighted Hybrid based on fixed weights of:

$$0.6 \text{ (XGBoost)} + 0.4 \text{ (LSTM)}$$

These weights are applied consistently across out-of-sample predictions.

3.8 Mathematical Formulation

The Hybrid model is defined as:

$$P_{\text{final}} = w_1 P_{\text{LSTM}} + w_2 P_{\text{XGB}}$$

where w_1 and w_2 are weights determined by:

$$w_1 + w_2 = 1$$

3.9 Backtesting Framework

The trading system makes trades based on crossing a defined probability threshold.

Returns are calculated using:

$$\text{Equity}_t = \prod (1 + r_i)$$

where r_i is individual trade returns.

The backtesting framework contains the following procedures:

- Entry price is set at the closing price after the signal generation point in time.
- Exit occurs when take-profit (TP) or stop-loss conditions are met or after 10 trading days of holding.
- Transaction costs are modeled at 0.15% per trade.

Returns for the total portfolio are calculated as follows:

$$\text{Portfolio_Return} = \sum \text{trade_returns}[t]$$

Entry into trades is executed at the next day's opening price to ensure that all decisions are based solely on information available at the time of signal generation.

3.10 Sharpe Ratio

The Sharpe Ratio is calculated using:

$$\text{Sharpe} = (R_p - R_f) / \sigma_p$$

where R_p is portfolio return, R_f is the risk-free rate, and σ_p is the portfolio return's standard deviation.

3.11 Performance Metrics

The performance of model predictions and financial results is evaluated using the following metrics:

Statistical Metrics:

- Precision
- Recall
- F1-score
- Area Under Curve (AUC)

Financial Metrics:

- Sharpe Ratio
 - Maximum Drawdown
 - Win Rate Percentage (%)
 - Total Return Percentage (%)
-

3.12 Reproducibility

Rigorously applying the following techniques ensures reproducibility of results:

- Fixed random number seeds
 - Chronological split of datasets without shuffling
 - Persisting trained models
 - Applying consistent preprocessing to all models under identical condition
-

3.13 Ethical and Regulatory Compliance

All data used in these studies are publicly available. No unstructured or sentiment-based datasets were utilized outside publicly available indicators or social media aggregates.

The modeling framework complies with any applicable regulations from the Saudi Capital Market Authority (CMA).

3.14 Chapter Summary

This chapter introduced and described the design and implementation of the Hybrid artificial intelligence framework for trading signal generation. The framework integrates technical analysis, fundamental analysis, and modeling techniques into a unified pipeline/architecture while adhering to realistic trading constraints modeled in a usable format for traders/users.

Experimental results comparing this approach to existing techniques are provided in the next chapter.

Chapter 4: Experimental Results and Discussion

4.1 Experimental Setup

The experiments were performed using the Saudi Stock Market (Tadawul).

A chronological split of the available data was used with an 80-20 split between training and out-of-sample testing data. Furthermore, no shuffling was applied to the data prior to training the models.

The resulting dataset contained 19,742 observations after cleansing the initial data source that contained 24,966 records. More specifically, the training and testing data contained 15,790 and 3,952 observations, respectively.

The trading costs for all models were set to 0.15% per trade. Additionally, the accuracy of the model's predictions was tested at a range of probability thresholds between 0.50 and 0.75.

4.2 Baseline Model Performance

Two separate models were employed as a baseline for the performance of the Hybrid Model:

- LSTM utilizing technical market features
- XGBoost utilizing financial and technical market features

Beyond evaluating the accuracy of these models in classifying price movement, the performance of the stocks was also evaluated.

At thresholds of 0.50, the standalone models exhibited higher trade counts than the hybrid model; however, they also exhibited inferior risk-adjusted performance. Furthermore, XGBoost exhibited unstable trading performance.

4.2.1 Interpretation of Classification Performance

Although the classification performance of the model is moderate, as indicated by its AUC, such results do not contradict the strong trading performance that the model achieved.

In trading applications, the goal is not to optimize prediction accuracy, but to optimize the trading decisions that are made based on the model's predictions.

By utilizing a threshold to filter which predictions are acted upon by the trading platform, the model optimizes for this latter goal.

Thus, even with moderate classification accuracy, the model achieved strong economic performance.

4.3 Hybrid Model Performance

The hybrid model includes the predictions of both the XGBoost and Bidirectional LSTM models that are both trained with the same market data; the two predictions are weighted to create a final trading signal.

The best performance of the hybrid model was exhibited at a threshold of 0.75.

4.3.1 Risk-Adjusted Trading Performance

Model	Trades	Win Rate	Total Return	Sharpe	Max Drawdown
XGBoost	54	29.63%	-11.03%	-1.02	-26.24%
LSTM	61	36.07%	14.87%	1.16	-13.03%
Hybrid	21	42.86%	15.87%	3.50	-4.55%

It is important to note that the hybrid model determines a relatively small number of trades to be executed at the optimal threshold (21 trades).

Only when both the LSTM model and the XGBoost model indicate strong and consistent signals is the trade to be executed by the model.

Although a general concern of fewer trades is the potential for less statistical stability in the results, the model was specifically constructed to aim for a smaller number of trades to be executed with high confidence as opposed to a large number of trades with potentially lower quality trading signals. Thus, this design choice for the model reflects the fact that fewer trades with high confidence will result in improved trading performance compared to the alternative strategy of executing a large number of trades with lower quality signals.

The superior performance of the proposed model is not derived from the improved accuracy of its trade classifications but from the superior quality of the trades it executes.

By applying a higher threshold to the probability of a trade, the model reduces the number of trades it executes but increases the quality of those trades, improving the risk-adjusted performance of its trading strategy.

Unlike the standalone models, which trade a higher number of less-confident trades, the hybrid model only executes trades with strong agreement between the technical and fundamental trading signals.

4.4 Key Insight

The improvement in the results of the hybrid trading model was not due to an increase in the accuracy of the trades it made, but rather to making fewer trades—trades of higher quality.

By applying a higher threshold to the probability of a trade being made, the trading model makes fewer trades, but increases the quality of those trades.

The model avoids making trades in which it does not have high confidence, and only makes trades where the technical and fundamental indicators agree.

As a result, the model has a higher Sharpe ratio and lower drawdowns than either of the standalone trading models.

4.5 Threshold Sensitivity Analysis

The effect of the probability thresholds on the performance of the models is depicted

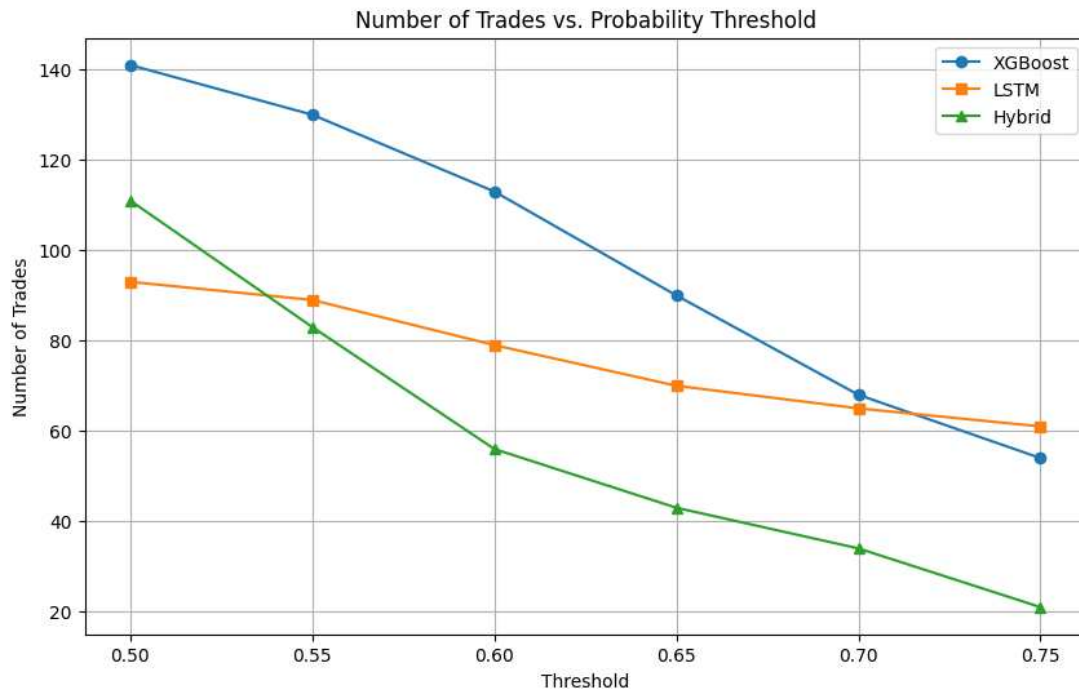


Figure 4.7: Number of Trades vs Probability Threshold

As the threshold for the probability of a positive return is increased, the algorithm becomes more conservative in its approach to trading.

Although the number of trades is reduced, the average profitability of the trade's increases. This indicates that the model is beneficial at only trading those stocks that are more likely to return a profit. The reduction in trade frequency at higher thresholds demonstrates the model's ability to filter out low-confidence opportunities, resulting in a more robust and stable trading strategy.

4.6 Risk-Oriented Perspective

One of the main benefits of the Hybrid model is that it can control the downside risk for the investor.

At the tested thresholds, the model has a maximum drawdown of -4.55%, which is the strongest risk-adjusted performance among the models evaluated.

4.7 Impact of Transaction Costs

The cost of 0.15% per trade was incorporated into the calculations of the model performances.

As such, the performance statistics for each model are reflective of the costs that would be experienced during real trading of those stocks.

Nevertheless, the results for the Hybrid model still indicate enhanced profitability and risk-adjusted performance relative to the standalone models.

4.8 Equity Curve and Trading Dynamics

The equity curve for the Hybrid model exhibits a more stable upward trend in comparison to the other models.

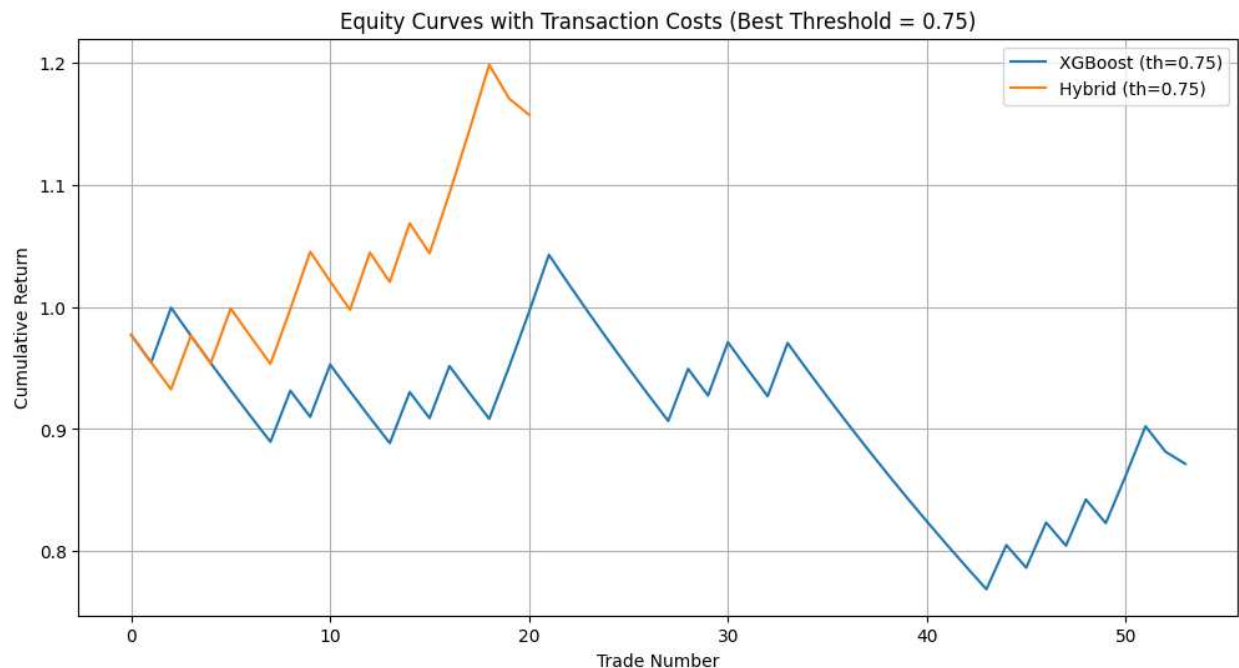


Figure 4.6: Cost-Adjusted Equity Curves for XGBoost and Hybrid Models (Threshold = 0.75)

Furthermore, the volatility of returns for the Hybrid model is significantly reduced in comparison to those of the other two models. The smooth upward trend

indicates consistency in returns and effective risk control, which is critical in real-world trading applications

4.9 Distribution of Trade Returns

Figure 4.8 presents the distribution of trade returns for the hybrid model at the optimal threshold.

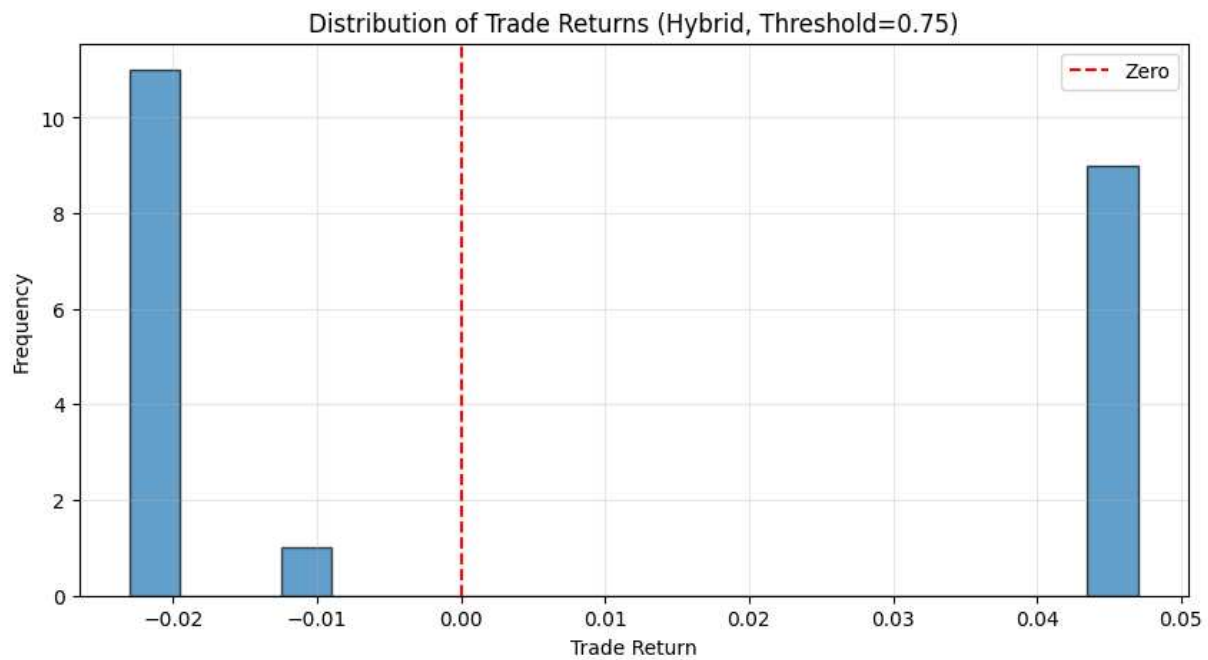


Figure 4.8: Distribution of Trade Returns for Hybrid Model (Threshold = 0.75)

The distribution reflects the rule-based structure of the strategy:

- Losses are concentrated near the stop-loss region
- Gains cluster near the take-profit level

This confirms that the trading logic is effectively enforced.

4.10 Mechanism Behind Hybrid Model Superiority

The superior performance of the hybrid model is due to the fact that each of its components possesses different strengths within the trading process.

Specifically, the Bidirectional LSTM model is found to have strengths in recognizing short-term price movement, while the XGBoost model is found to be strong in recognizing financial and technical market relationships.

Thus, the model's ability to filter trading opportunities based on the strength of both signals ensures that only trades with high confidence are executed.

The effectiveness of the hybrid model can also be explained by the models themselves.

The LSTM model can identify short-term patterns in the prices of the cryptocurrencies, while the XGBoost model indicates the relationship between the features of the cryptocurrencies. By using both of these models, the hybrid model can validate signals from the LSTM model with the XGBoost model to reject false-positive trades.

4.11 Economic Interpretation

By simultaneously utilizing technical and fundamental trading information within the decision-making process, the Hybrid model avoids entering into trades that have strong signals for only one of the model's components; such decisions by the model lead to improved consistency in the model's trades and a reduction in the downside risk of the strategy.

4.12 Limitations

The framework and results of this study are limited in the following aspects

- The model was tested on only the stocks that trade on the Saudi Stock Market (Tadawul).
 - The historical data used by the model was contained to a five-year span of market data.
 - The model has not been tested for performance under changing market conditions.
-

4.13 Practical Deployment

In the real world, additional considerations would have to be made for the deployment of the Hybrid model:

- The model would have to be tested for its performance in markets of differing levels of liquidity.
- The model would have to account for slippage in real markets.
- The model would have to account for latency of market orders.
- In addition, the model would have to be continually retrained to account for changes in the market.

4.14 Chapter Summary

The results of this study indicate that the Hybrid model outperformed the two existing models in terms of risk-adjusted trading performance.

The main finding of this study is that enhancing the trading decisions of the model—rather than the accuracy of its price prediction—is what provides the model with the greatest value to investors. Thus, this aspect is crucial in the development and implementation of trading algorithms based upon AI systems.

Chapter 5: Conclusion and Future Work

5.1 Conclusions

The main results of this study indicate that the Hybrid model is effective in generating trading signals for the Saudi Stock Market (Tadawul). Furthermore, the results reveal that the Hybrid model is able to significantly outperform both of the standalone models in terms of risk-adjusted trading performance.

Specifically, at a threshold of 0.75, the Hybrid model indicates trading performance results of 3.50 for its Sharpe ratio, a win rate of 42.86%, and a maximum drawdown of -4.55%, while trading fewer trades than the other two models. These enhanced risk-adjusted performance results are due to the enhanced selectivity of the trades that are executed by the model relative to those of the other two models.

These findings confirm that focusing upon trading decisions and the accuracy of risk-adjusted performance is more valuable than enhancing the accuracy with which the model can predict the movement of stock prices. Furthermore, by incorporating both technical and fundamental trading information into the model, weak trading signals are automatically filtered by the model prior to being entered into the market.

Although tested under realistic conditions that include transaction costs of 0.15% per trade, these results are still significantly enhanced relative to the standalone models.

5.2 Theoretical Contribution

The theoretical literature on trading signals and financial models could benefit from the findings of this study. More specifically, the integration of technical and fundamental analyses into a single model framework is a novel contribution of the study.

By utilizing both valuation-based indicators as well as technical indicators of the strength of each stock, the model ensures that both analyses contribute to each trading decision. Consequently, this leads to the enhancement of the trading system and model relative to those that utilize only one type of analysis.

5.3 Practical Implications

The findings of this study have a variety of implications for the trading system.

First, it indicates that models can be created that utilize only publicly available information about each trade to make predictions of the direction of the stocks' prices.

Second, the fact that the model considers the cost of trading helps to indicate that the results of the trading system are realistic relative to the actual trading that would occur in the stock market.

Third, the findings of the study help to suggest that traders should focus upon risk-adjusted performance rather than accuracy of price movement predictions.

Furthermore, because the model is compatible with the regulations of the Saudi Capital Market Authority (CMA), it may be directly applied to the Saudi Stock Market (Tadawul) for trading purposes.

5.4 Implementation Constraints

The findings of the study are positive, but there are some challenges for the implementation of the Hybrid model.

For instance, issues like liquidity and slippage were not considered in the creation of the model. Additionally, while the impact of slippage was incorporated into the backtesting process, other considerations regarding implementation would have to be made prior to its implementation into the market.

5.5 Generalizability of the Framework

While the framework was applied to the Saudi Stock Market (Tadawul), it is not market-specific.

It can be applied to any market that exposes stocks to the same types of financial and price data.

5.6 Future Research Directions

There are a variety of different ways in which the research that was performed can be extended.

Some potential research directions that could be undertaken in the future may include:

- testing the model on different markets around the world
- incorporating additional economic indicators into the model
- applying reinforcement learning to determine trading strategies
- enhancing the model to account for slippage

finding dynamic thresholds for the models to increase their performance.

These are only a few of the ways in which the study can be extended.

5.7 Final Remarks

Overall, this study indicates the importance of focusing upon trading decisions rather than the accuracy of the AI system's predictions of the prices of the stocks.

By integrating technical and fundamental analyses of the companies and their stocks, and by applying thresholds to filter trades with high probabilities of success, the model indicates that it can significantly outperform existing models for trading decisions.

Thus, the implications of this study suggest that the value of AI algorithms in the stock market is not in their ability to accurately predict the movements of stock prices, but in their potential to automate trading decisions for investors.

References

- Abu-Mostafa, Y. S., Magdon-Ismael, M., & Lin, H. T. (2012). Learning from data. AMLBook.
- Adebisi, A. A., Adewumi, A. O., & Ayo, C. K. (2014). Stock price prediction using the ARIMA model. In Proceedings of the UKSim-AMSS 16th International Conference on Computer Modelling and Simulation (pp. 106–112). IEEE.
- Ahmed, N. K., Atiya, A. F., Gayar, N. E., & El-Shishiny, H. (2010). An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews*, 29(5–6), 594–621. <https://doi.org/10.1080/07474938.2010.481556>
- Bao, W., Yue, J., & Rao, Y. (2017). A deep learning framework for financial time series using stacked autoencoders and LSTM. *PLOS ONE*, 12(7), e0180944. <https://doi.org/10.1371/journal.pone.0180944>
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
- Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). <https://doi.org/10.1145/2939672.2939785>
- Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, 25(2), 383–417. <https://doi.org/10.2307/2325486>
- Fischer, T., & Krauss, C. (2018). Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*, 270(2), 654–669. <https://doi.org/10.1016/j.ejor.2017.11.054>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, 259(2), 689–702. <https://doi.org/10.1016/j.ejor.2016.10.031>

Tsai, C. F., & Wang, S. P. (2009). Stock price forecasting by hybrid machine learning techniques. *Expert Systems with Applications*, 36(4), 7551–7559.
<https://doi.org/10.1016/j.eswa.2008.09.029>

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12), 11106–11115.

Yahoo Finance. (2024). Historical market data from <https://finance.yahoo.com>

Tadawul (Saudi Exchange). (2025). Historical market data. Retrieved from <https://www.saudiexchange.sa>

Saudi Capital Market Authority (CMA). (2024). Market regulations and guidelines. Retrieved from <https://www.cma.org.sa>



Facial Emotion Recognition for Visually Impaired People

Prepared By

Name: [[Chadi Omar Elminawi](#)]

IDNo.: [[252001554](#)]

Submitted in fulfillment of the requirements for the degree Master of
Science in Artificial Intelligence

At

College of Informatics

Midocean University

UAE

Supervisor Name: Dr. Ashraf El-Sayed

College of Informatics, Midocean University, Comoros

Email: 252001554@st.midocean.edu.km

Table of Contents

Abstract.....	3
Chapter 1 Introduction.....	4
Section 1.1 Problem Definition	4
Section 1.2 Thesis Objectives.....	4
Section 1.3 Thesis Contribution	5
Chapter 2 Background and Related Work.....	9
Section 2.1 User Needs	9
Section 2.2 Requirement Specification	9
Section 2.3 Overview of The System	12
Section 2.4 Concept Design.....	12
Section 2.5 Machine/Deep Learning	12
Chapter 3 Proposed Approach.....	14
Section 3.1 Camera	16
Section 3.2 Preliminary Work.....	17
Section 3.3 System Design Overview	23
Chapter 4 Experimental Results and Discussion	26
Section 4.1 Prototype Implementation.....	26
Section 4.2 Prototype Verification	29
Chapter 5 Conclusion	32
References	33
Appendix A The Project Plan	35
Section A.1 Work Breakdown Structure	35
Section A.2 Gantt Chart	37
Section A.3 Cost Estimation.....	37

Abstract

This project focuses on developing a deep learning-based facial emotion recognition system designed to assist visually impaired individuals in social interactions. By accurately identifying and interpreting facial expressions, the system aims to bridge the communication gap faced by the visually impaired community, often hindered by an inability to perceive emotional and social cues visually. The system will operate on an embedded platform, ensuring portability and accessibility. It will analyse facial expressions in real time, leveraging advanced deep learning algorithms tailored for high accuracy and low latency. The identified emotions will be conveyed to the user through audible reports, enabling an intuitive understanding of the social and emotional context during interactions. This innovative approach not only enhances the social engagement of visually impaired individuals but also fosters inclusivity and empathy within the broader community. By providing a technologically advanced solution, this project seeks to empower visually impaired individuals with greater independence in social settings and contribute significantly to improving their quality of life.

Chapter 1

Introduction

Section 1.1 Problem Definition

Visually impaired individuals encounter significant challenges in their social interactions, primarily due to their inability to perceive non-verbal cues, especially facial expressions. This limitation critically impedes their ability to fully comprehend the emotions and intentions of others, which can lead to misunderstandings and an increased risk of social isolation. The World Health Organization reports that approximately 285 million people worldwide are visually impaired, with 39 million classified as blind and 246 million having low vision. Moreover, around 2.2 billion people globally live with some form of vision impairment or blindness, underscoring the widespread prevalence and profound impact of visual impairment. This vast number of affected individuals highlights the urgency of addressing this issue [1][2].

The difficulty in interpreting non-verbal cues has a profound effect on the social well-being and mental health of visually impaired individuals. Research indicates that this challenge contributes to increased feelings of loneliness and depression among those who are visually impaired [3]. Consequently, it is imperative to recognize the significance of this issue to improve the quality of life for those with visual impairments and promote social inclusion and mental well-being throughout society. Addressing this need is crucial to mitigate the adverse effects on the mental health and social interactions of visually impaired individuals, emphasizing the importance of societal efforts to foster a more inclusive environment for all.

Section 1.2 Thesis Objectives

The facial emotion recognition system will employ computer vision algorithms to analyze facial expressions. Utilizing a device's camera as input, the system will capture facial features, extract relevant emotional cues, and classify expressions into distinct emotions. The output will consist of audible feedback to convey recognized emotions to the user, maintaining a non-intrusive and user-friendly interaction. The system's adaptability to varying environmental conditions and compatibility with standard devices aim to provide an inclusive and seamless user experience for individuals with visual impairment. Detailed design and implementation specifications will be developed in subsequent project phases.

Section 1.3 Thesis Contribution

Facial Emotion Recognition (FER) technology analyzes facial expressions from images and videos to infer emotional states. It's pivotal in non-verbal communication, offering insights into human emotions and being crucial for social interaction and communication. However, challenges such as variability in facial expressions, contextual and cultural differences, and technological limitations like lighting and occlusions affect accuracy. FER has vast applications in healthcare, detecting disorders, education, gauging student engagement, security, monitoring public spaces, and customer service to personalize experiences. This technology, while promising, faces ethical and privacy concerns, necessitating careful consideration in deployment. Yet traditional methods exist that can be utilized in FER efforts.

The application of various techniques in FER, including neural networks and traditional machine learning methods. Specifically, it mentions the use of Histogram of Oriented Gradients (HOG) and Local Binary Patterns (LBP) as traditional feature descriptors combined with data classifiers like Support Vector Machine (SVM), k-nearest Neighbors (KNN), or Random Forest for emotion detection and recognition. These methods were compared to Artificial Neural Networks (ANNs), highlighting the shift towards using ANNs due to their superior performance in detecting and recognizing emotions in a subject-independent manner. The document suggests that while traditional methods have their merits, the integration of ANNs, especially deep learning techniques, has opened new possibilities for practical applications across various fields such as healthcare, security, business, education, and manufacturing, offering improved accuracy and efficiency in emotion recognition tasks. Emotion Detection using Image Processing in Python outlines a methodology for detecting human emotions from facial expressions using Python, OpenCV, and NumPy. It details creating a dataset, processing images to extract emotions, and classifying emotions using Fisher Face classifiers. The study involves converting images to grayscale, cropping, and organizing them into directories based on emotion types. It emphasizes the importance of using consistent image size and properties for accurate emotion recognition, suggesting an 80/20 split for training and classification to enhance accuracy and efficiency [4][5].

In [6], the authors outline a facial expression recognition project employing Convolutional Neural Networks (CCN) specifically to identify human emotions from facial images. It covers the essential stages of FER systems, including preprocessing, face detection, feature extraction, and expression classification. However, the document primarily focuses on using CNN to recognize seven key human emotions and does not explicitly detail the use of Local Binary Patterns, Histogram of Oriented Gradients, facial landmarks, or local feature descriptors as requested. These techniques are integral to traditional FER methods and contribute to the feature extraction phase by analyzing texture, shape, and changes in facial attributes, which are critical for accurate emotion classification.

In [7], the authors discuss an innovative Facial Expression Recognition method that enhances FER performance through facial action units-based image retrieval. This technique leverages Multi-layer Perceptrons (MLPs) to assess the reliability of FER results. Unreliable outcomes trigger a search for similar facial expressions using facial action units, aiding in the reclassification process. It integrates traditional feature extraction methods like LBP and HOG, alongside advanced neural network models, to refine emotion classification. This approach signifies a melding of classical feature analysis and modern deep learning, aiming to improve accuracy in emotion detection.

In [8], the authors present a system for automatic facial expression recognition, utilizing image processing techniques and neural networks. It focuses on extracting features from the contours of eyebrows, eyes, and mouth using a scalable rectangle, aiming to improve manual point selection and complex face models. The approach reduces the number of features to quicken recognition time while maintaining accuracy. This method involves preprocessing to adjust lighting, detecting facial features, normalizing these features to account for scale differences, and classifying expressions using SimNet, a combination of Fuzzy Logic and ANNs.

In [9], the authors provide a comprehensive overview of recent advancements in FER using deep learning techniques. It details the challenges in the field, such as variations in facial expressions and environmental conditions, and discusses the effectiveness of various deep learning architectures like CNN and Recurrent Neural Networks (RNNs) in overcoming these challenges. The paper highlights the importance of large and diverse datasets for training models and reviews several databases used in FER research. A comparative analysis of different studies offers insights into improving FER systems' accuracy and efficiency. "FER Using Deep Learning" [10], explores FER using deep learning techniques. It includes a comprehensive examination of CNN and SVM for FER, detailing the methodology, dataset preparation (FER2013 and CK+ datasets), and implementation of these models. The thesis aims to improve the accuracy and efficiency of FER systems, addressing challenges such as occlusion, contrast variations, class imbalance, and intra-class variations.

In [11], the authors present a study focusing on improving facial expression recognition using CNN. The research utilizes the FER2013 dataset, comprising samples of seven universal facial expressions, to train the model. The results demonstrate that the CNN-based technique enhances facial expression recognition accuracy without necessitating the encoding of multiple CNN layers, which typically results in a computationally intensive model. This approach offers a solution to high computational costs associated with facial expression recognition, achieving near state-of-the-art accuracy while ensuring efficiency and prediction capabilities in the health sector.

It focuses on the critical role of facial expression recognition in conveying human emotions. It provides a comprehensive review of automated FER systems, emphasizing the challenges and progress in the field. The paper highlights the efficiency of deep learning architecture, particularly CNN, as a novel solution for FER issues. It investigates recent and highly cited works, detailing

various deep learning methods applied to facial expression recognition, the databases used, and the success of these techniques in improving recognition accuracy [12].

In [13], the authors explore the application of deep learning and transfer learning to FER, focusing on evaluating model performance and the impact of transfer learning on traditional sequential models. The thesis uses the FER-2013 database, containing 35,887 images across seven emotion categories, and employs a k-fold split method for training and testing. It compares the performance of a sequential CNN model, a pre-trained ResNet50 model, and a pre-trained VGG16 model, using metrics like F1 score and confusion matrices. The study demonstrates the effectiveness of transfer learning in reducing computational time and achieving comparable accuracy, making it a recommended approach for FER tasks.

In [14], the author delves into a sophisticated system leveraging deep learning to empower visually impaired people by enabling them to perceive facial expressions through auditory feedback. This technology harnesses the power of CNN, a class of deep neural networks highly effective in analyzing visual imagery, to identify a range of facial expressions. By training these networks with extensive datasets, the system can accurately classify different emotions conveyed through facial expressions. Once identified, these emotions are translated into audio signals, providing real-time, intuitive feedback to the user. Otherwise, each not only enhances social interactions for the visually impaired by giving them access to non-verbal cues that are otherwise inaccessible but also contributes to their emotional understanding and connectivity in interpersonal communications. Also presents a deep learning-based facial expression recognition system for assisting visually impaired people, implemented on a wearable device. It utilizes MobileNetV2 for its efficiency and speed, which makes it suitable for real-time applications on edge devices like Raspberry Pi. The system combines datasets including CK+, ANN, and JAFFE for training, achieving notable accuracy. Experiments in indoor and outdoor settings with visually impaired participants showed success rates of 80% and 78%, respectively, demonstrating the system's effectiveness in real-world conditions for enhancing social interactions for the visually impaired through auditory feedback of facial expressions.

In [15], the authors outline a comprehensive study on developing a lightweight FER system tailored for visually impaired individuals. This system leverages a novel approach using partial transfer learning with a custom-trained Convolutional Neural Network, aiming to improve accuracy in emotion recognition while ensuring the system remains efficient for real-time applications on edge devices. The document highlights the challenges faced by visually impaired people in social interactions due to the inability to perceive visual cues and proposes this technology as a solution. With an emphasis on portability and low computational requirements, the system demonstrated promising results, achieving high recognition accuracy, particularly for emotions like happiness, sadness, and surprise, making it a significant advancement in assistive technologies for enhancing the quality of life and social integration of visually impaired individuals. The designed system showcased an impressive achievement in terms of accuracy and efficiency. The system achieved a notable accuracy rate of 82.1% on the enhanced Facial Expression Recognition 2013 dataset

such as EFER2013 and CK+, an improvement over the previous model. This high level of performance is particularly significant considering the system's design for real-time application on edge devices, which are characterized by their limited processing power. The system's architecture, optimized for efficiency, comprises only 1.49 million parameters, demonstrating its suitability for low-resource environments and its potential to significantly aid visually impaired individuals in social interactions by accurately interpreting facial emotions.

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

Figure 1.1 Samples from the EFER2013 dataset with the actual and predicted labels. [15]

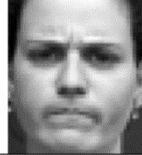
Actual Label	Anger	Sad	Anger	Fear	Happy	Disgust	Surprise
Face Image							
Predicted Label	Sad	Sad	Anger	Sad	Happy	Disgust	Surprise

Figure 1.2 Samples from the CK+ dataset with the actual and predicted labels. [15]

The studies on FER highlight the pivotal role of deep learning techniques in advancing the field by addressing traditional challenges such as variability in facial expressions, contextual and cultural differences, and technological limitations. Deep learning methods, particularly CNN, have demonstrated superior performance in detecting and recognizing emotions compared to traditional methods like Histograms of Oriented Gradients and Local Binary Patterns. These studies reveal that while traditional methods have their merits, the integration of deep learning techniques has significantly improved the accuracy and efficiency of emotion recognition systems. Applications span across various fields including healthcare, security, education, and customer service, showcasing the practical benefits of deep learning. Furthermore, innovations such as using facial action units for reclassification and employing deep learning on edge devices for real-time applications, particularly for assisting visually impaired individuals, underscore the transformative potential of deep learning in FER. The comprehensive examination of deep learning architectures and their comparative analysis against traditional methods affirm the efficacy and adaptability of deep learning, making it the cornerstone technique for advancing FER technology.

Chapter 2 Background and Related Work

Section 2.1 User Needs

User needs in projects refer to the specific requirements, preferences, and expectations of the end users that must be met by the project deliverables or outcomes.

- The system should be based on a general off-the-shelf embedded platform.
- The system should be capable of recognizing and classifying facial expressions with high accuracy.
- The system should be easy to operate, where the user captures a picture of a person's face to analyze their facial expression.
- The system should return the recognized facial emotion audibly through a speaker.
- The system should be flexible enough to recognize a wide range of facial expressions accurately.
- The system should be reliable in identifying emotions across different individuals and lighting conditions.
- The system should be cost-effective, considering both hardware and software components.
- The system should have a high scanning speed to analyze facial expressions quickly.
- The system should be designed to minimize power consumption.

Section 2.2 Requirement Specification

- The system should use an off-the-shelf embedded platform, due to its superior performance with deep learning models compared to other options.
- The system should accurately recognize and classify facial expressions with at least 90% accuracy.
- The system's training data should include a diverse range of facial expressions to ensure high accuracy in recognition.
- The system should perform a quick scan of facial images within 3 seconds or less.
- The system shall classify and output recognized emotion within 3 seconds.
- The system should include (ON/OFF) to optimize energy consumption.
- The system should audibly output recognition results within 5 seconds.
- The project budget is limited to 1000 SAR and must not exceed this limit.

Table 2.1 matching Marketing requirements to Engineering requirements.

Marketing Requirements	Engineering Requirements	Justification
2,6	The system should use an off-the-shelf embedded platform, due to its superior performance with deep learning models compared to other options.	Using an off-the-shelf embedded platform simplifies the system's development and integration process while delivering the performance needed for deep learning models.
1	The system's training data should include a diverse range of facial emotion datasets (CK+, FER2013) to ensure high accuracy in recognition.	Diverse facial expression training data reduces bias and ensures high recognition accuracy across different demographics.
8	The system should accurately recognize and classify facial expressions with at least 90% accuracy.	High accuracy ensures reliable recognition, providing visually impaired users with dependable emotional cues, which improves their social interactions.
2,5	The system should perform a quick scan of facial images within 3 seconds or less.	A quick scan minimizes latency, allowing users to receive results promptly and interact with others without delays.
2,3,8	The system shall classify and give an output of recognized emotion within 3 seconds	Fast classification and output ensure that users receive timely results, enabling them to understand emotions in real time.
4	The system should audibly output recognition results within 5 seconds.	Delivering the recognized emotion audibly within seconds ensures users can respond promptly to the emotional cues.
7,9	The system should include (ON/OFF) to optimize energy consumption.	Energy-saving modes optimize power consumption and extend battery life, ensuring reliable performance for mobile or wearable systems.
7	The project budget is limited to 1000 SAR and must not exceed this limit.	Limiting the budget ensures cost-effectiveness and accessibility, while still providing high-quality performance.

Marketing Requirements:

1. The system should be based on a general off-the-shelf embedded platform.
2. The system should be capable of recognizing and classifying Facial emotions with high accuracy.
3. The system should be easy to operate where the user takes a picture of a facial emotion.
4. The system should return the recognized facial emotion audibly.
5. The system should be flexible enough to recognize a wide range of facial emotions.
6. The system should be reliable.
7. The system should be cost-effective.
8. The system should have a high scanning speed.
9. The system should not consume too much power.

Section 2.3 Overview of The System

Our FER design should be based on general off-the-shelf embedded platforms with a suitable programming language. It should be capable of recognizing facial expressions with high accuracy and return the recognized emotions audibly, ensuring that visually impaired individuals receive prompt, clear, and accessible feedback.

Section 2.4 Concept Design

Table 3.1 Concept overview.

Machine Learning	Hardware Platform	Camera	Speaker
Deep Learning	Raspberry Pi	Raspberry Pi camera V1	Speaker
Image Processing	Arduino	Raspberry Pi camera V2	

Section 2.5 Machine/Deep learning:

Machine/Deep Learning is a branch of Artificial Intelligence that enables systems to learn and improve from experience automatically, without the need for explicit programming.

Table 3.2 Technique comparison.

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

Our project on FER for visually impaired people can be approached using two methods: image processing and deep learning. Image processing techniques, such as Histograms of Oriented Gradients and Local Binary Patterns, involve extracting features manually and using classifiers like Support Vector Machines or k-nearest Neighbors for emotion detection. While these methods are computationally less intensive and easier to implement, they often struggle with variations in lighting, facial expressions, and occlusions, leading to lower accuracy. In contrast, deep learning, particularly CNN, automatically learns features from data, making it more robust to such variations and significantly improving recognition accuracy. Deep learning models can handle large, diverse datasets and capture complex patterns in facial expressions, which is crucial for accurate emotion detection. Given the need for high accuracy and real-time processing in assistive technologies for visually impaired individuals, deep learning offers superior performance and adaptability, making it the preferred choice for our project.

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

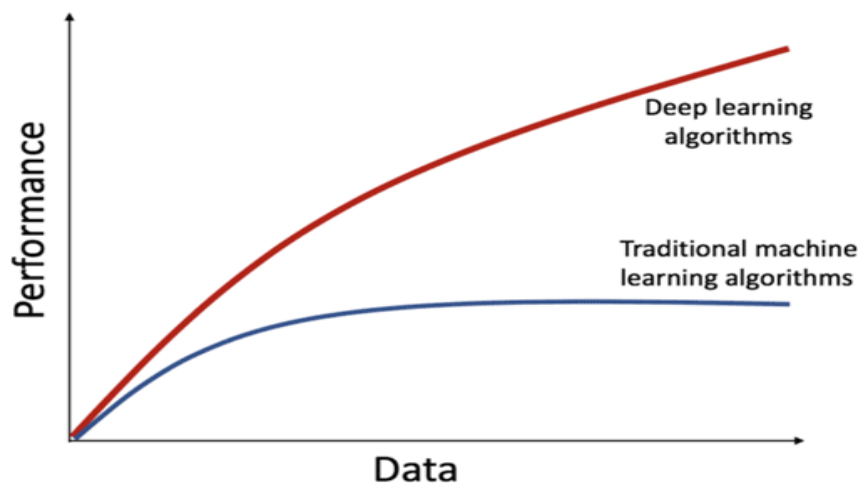


Figure 3.1 Deep Learning: Comparative Study

Chapter 3 Proposed Approach

A hardware platform consists of a group of compatible hardware components that can run software applications. Programs must be designed for a specific platform, which includes a standardized type of processor and associated hardware components, as each platform has its machine language.

Table 3.3 Hardware comparison.

Method	Strengths	Weaknesses
Raspberry Pi	- Low power consumption - Fast memory - great with Python - High Speed - Have his own Linux OS	- High cost - Non-upgradable memory. - Long setup .
Arduino	- Lower cost. - No need for setup. - Lower power.	- Single-tasking. - Lower speed - Doesn't have an OS. - It doesn't deal with Python or deep learning.

Raspberry Pi and Arduino are both highly popular hardware platforms used by computer engineers, whether they are beginners or professionals. Although Raspberry Pi is mainly intended for learning computer programming, Arduino is focused on rapid programming and circuit prototyping. In this discussion, we will explore the similarities and differences between these two platforms and explain the reasons behind choosing Raspberry Pi over Arduino.

We chose **Raspberry Pi** for its multitasking processor and support for Python programming language, which was essential for developing the Deep Learning module for the Arabic handwritten character/digit recognition dataset.

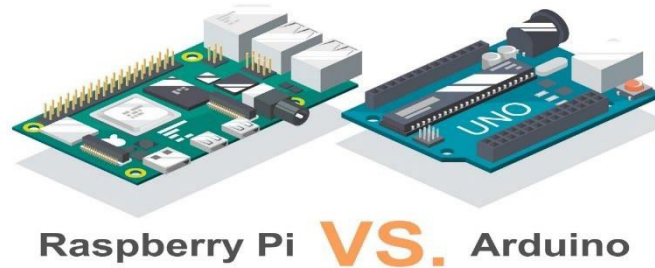


Figure 3.2 Comparison of hardware.

Table 3.4 Hardware comparison.

Raspberry Pi		Arduino
- A mini-computer with its own Raspberry OS capable of handling multiple programs at a time.	Definition	- It is a microcontroller which is just a normal part of the computer. It can be used to run a single program code again and again.
- It has 4 USB ports for establishing a connection with different devices.	Number of USB ports	- It has just 1 USB port to connect to the computer.
- Python, Scratch, Ruby, C, and C++ are the primary programming languages for developing applications on the Raspberry Pi.	Languages Supported	- C or C++ programming languages can be used to program Arduino.
- 32 bits	CPU Architecture	- 8 bits
- The Raspberry Pi processor speed- 700MHz- 1.5GHz	Processor speed.	- The average speed of most of the Arduinos is 16 MHz

Section 3.1 Camera

The camera is a crucial hardware component that collects the required data for the project application.

Table 3.5 Camera (Strengths & Weaknesses).

Camera type	Strengths	Weaknesses
Raspberry Pi Camera V1	- 5 Megapixels. - Easy to install. -Lower cost than the V2.	- Lower Resolution. - requires configuration
Raspberry Pi Camera V2	- 8 Megapixels. - Easy to install.	-Thermal issues -Limited frame rate

For our project on FER for visually impaired people, we chose the Raspberry Pi Camera Module v2 over the v1 due to its superior features. The v2 camera has an 8-megapixel sensor, offering higher-resolution images compared to the 5-megapixel sensor in the v1. This higher resolution is crucial for accurately detecting facial expressions. Additionally, the v2 camera provides better image quality and color fidelity, enhancing the performance of our deep-learning models. Despite some minor limitations, the v2 camera's improvements in resolution and clarity make it the better choice for our project's needs.

So, we chose: **Raspberry Pi Camera V2**

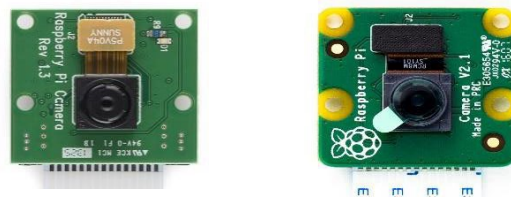


Figure 3.3 Camera V1 and Camera V2 respectively.

Section 3.2 Preliminary Work

Datasets are an essential part of any computer vision system. A benchmark dataset allows researchers to compare different machine learning approaches and techniques quickly and fairly. In this project, we utilize two of the most used datasets, namely **FER2013** and **CK+ (Extended Cohn-Kanade Dataset)**.

FER2013Dataset:

FER2013 is a publicly available dataset created for facial expression recognition tasks. It was introduced during the ICML 2013 Challenges in Representation Learning. The dataset consists of 35,887 grayscale images of 48x48 pixels, categorized into seven different facial expressions: anger, disgust, fear, happiness, sadness, surprise, and neutral. These images were collected from various sources, ensuring diversity in age, gender, and ethnicity, making it a valuable benchmark for training and evaluating facial emotion recognition models.



Figure 3.4 A sample from FER2013

CK+ Dataset (Extended Cohn-Kanade Dataset):

CK+ is a widely used dataset for facial expression recognition, containing sequences of images that capture the progression of facial expressions. It includes 593 sequences from 123 participants, with a focus on seven universally recognized expressions: anger, contempt, disgust, fear, happiness, sadness, and surprise. The images are annotated with both Action Units (AUs) and emotion labels, making the dataset ideal for training and evaluating facial expression recognition models. The data were collected under controlled laboratory conditions to ensure consistency and high-quality annotations.



Figure 3.5 A sample from CK+ dataset

Why We Chose Convolutional Neural Networks (CNNs)

For our preliminary work, we utilized **CNNs**, a cornerstone technique in deep learning, particularly effective for image-related tasks. CNNs are designed to automatically and adaptively learn spatial hierarchies of features from images, making them highly suited for tasks like facial emotion recognition.

In this project, we focused on developing a **facial emotion recognition system for visually impaired people**, we trained our CNN model specifically for this task. This allowed us to tailor the architecture and parameters of the model to better suit the FER2013 dataset and the unique requirements of facial emotion recognition.

This approach was chosen over transfer learning to ensure the model was optimized specifically for the dataset and task at hand, without relying on pre-existing features from unrelated datasets. The trained CNN model was further integrated into our system, providing analysis of facial expressions and delivering auditory feedback to visually impaired users.

After deciding to use Convolutional Neural Networks (CNNs) for their ability to learn spatial hierarchies of features and their effectiveness in emotion recognition tasks, the next step was to select the most suitable CNN architecture for our project. To ensure optimal performance, we compared two widely recognized models, VGGNet and DenseNet121. This comparison allowed us to evaluate their strengths and weaknesses, particularly in terms of accuracy, efficiency, and suitability for real-time applications in facial emotion recognition. By analyzing their performance, we aimed to identify the model that best aligns with the requirements of our system.

The table compares the performance of VGGnet, ResNet152 on the FER2013 dataset using various evaluation metrics including precision, recall, F1-score, and accuracy.

Table 3.6 Comparisons of models on FER2013

Model	Dataset	Precision	Recall	F1-Score	Accuracy
VGGnet	FER2013	0.683	0.651	0.633	0.701
Densenet121		0.579	0.576	0.575	0.576

True positives (TP): refers to cases where the system correctly recognizes an emotion,

True negatives (TN): refers to cases where the system correctly identifies no face or no emotion.

False positives (FP): refers to cases where the system incorrectly identifies a no emotion as an emotion.

False negatives (FN): refers to cases where the system fails to recognize an emotion.

Table 3.7 tp,tn,fn and fp of densenet121

Class	Emotions	TP	TN	FP	FN
0	Angry	403	4494	438	407
1	Fear	330	4486	443	483
2	Happy	1192	3919	360	271
3	Sad	467	4162	657	456
4	Surprise	436	4947	155	204
5	Neutral	448	4390	349	555

		Confusion Matrix					
True Label	Angry	403	80	54	150	23	87
	Fear	112	330	56	178	65	66
	Happy	61	55	1192	68	30	52
	Sad	115	130	80	467	13	114
	Surprise	19	94	40	22	436	28
	Neutral	108	75	126	222	22	448
		Angry	Fear	Happy	Sad	Surprise	Neutral
		Predicted Label					

Figure 3.6 Confusion matrix of densenet121

DenseNet121 demonstrates strong capabilities in facial emotion recognition, excelling in identifying certain emotions like "Happy," which achieves a high true positive rate of 1,192. Its dense connectivity allows for efficient feature reuse and improved gradient flow, which enhances training speed and overall performance. Additionally, the model's parameter efficiency reduces the computational load compared to other deep architectures, making it suitable for tasks with limited memory. However, it struggles to distinguish between similar emotions, such as "Sad" and "Fear," as reflected by higher false positive rates and confusion in the matrix. The model's computational demands, due to its depth and dense connections, can hinder real-time applications, particularly on devices with limited processing power. Furthermore, its performance heavily depends on the quality and balance of the training dataset, with biases leading to lower accuracy for certain classes like "Fear" (TP = 330). Finally, the high complexity of DenseNet121 increases the risk of overfitting on small or imbalanced datasets, emphasizing the need for careful dataset preparation and model optimization to maximize its potential.

The Final Model (VGGnet):

Following a thorough evaluation and comparison, it was evident that VGGNet outperformed ResNet152 in terms of both accuracy and performance. This model exhibited superior results and proved to be the most suitable choice for our needs. By leveraging the power of Kaggle cloud for training and carefully selecting the VGGNet model, we successfully obtained a highly accurate and efficient solution for our requirements.

Table 3.8 Comparison of tp,tn,fn and fp on VGGnet

Class	Emotions	TP	TN	FP	FN
0	Angry	407	4740	253	342
1	Fear	323	4681	234	504
2	Happy	1251	4048	221	222
3	Sad	510	4378	432	422
4	Surprise	479	4995	85	183
5	Neutral	792	4024	706	220

The confusion matrix for its performance in this project shows its strong ability to accurately detect emotions like "Happy" and "Surprise," with high True Positive rates, demonstrating its reliability in identifying distinct facial expressions. While the model achieves impressive results overall, there is room to refine its recognition of more subtle emotions to further enhance its performance.

true label	Angry	407	46	52	99	8	131
	Fear	117	323	36	163	54	133
	Happy	22	23	1251	36	16	125
	Sad	51	74	36	510	4	256
	Surprise	18	56	35	16	479	58
	Neutral	23	27	57	111	2	792
		predicted label					
		Angry	Fear	Happy	Sad	Surprise	Neutral

Figure 3.7 Confusion matrix of VGGnet

Moreover, VGGNet's structured approach enables it to consistently process and classify complex visual data, maintaining reliability across most emotion categories. While the outcomes suggest slight limitations in recognizing subtler or overlapping emotions, the model's overall performance underscores its robustness and suitability as a foundation for building assistive technologies, especially those aimed at improving social interaction for visually impaired individuals.

Section 3.3 System Design Overview

Design Level 0:

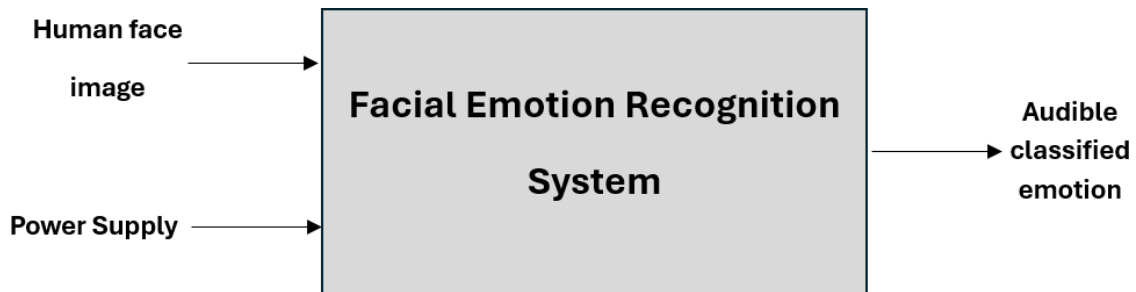


Figure 3.8 System design level 0.

Module	FER System
Input	Human face image: 8Mpixels Power Supply: 220 volts AC rms, 60Hz.
Output	Audio output signal
Functionality	Capturing an image of a person's face using a camera and detecting the face, the system extracts facial features and classifies the emotion using a trained deep learning model. Finally, it outputs the detected emotion through an audio message.

Table 1.9 Explaining the functionality of the system.

Design Level 1:

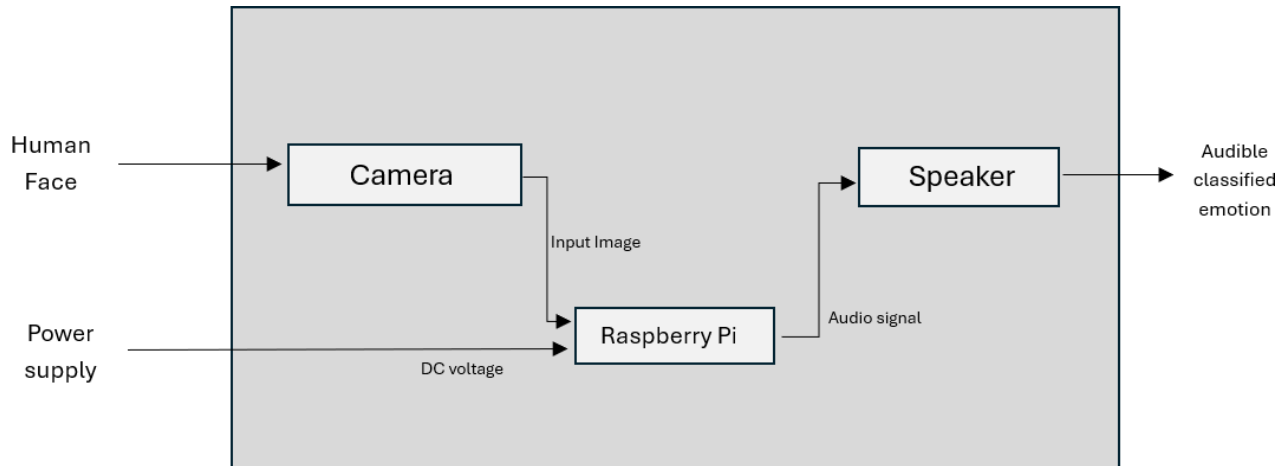


Figure 3.9 Detailed system design.

Module	Raspberry Pi Camera V2
Input	Human Face
Output	Image of the human face
Functionality	The camera captures an image of a human face and then outputs it to Raspberry Pi.

Table 3.10 Functionality of Camera.

Module	Raspberry Pi 4 model b
Input	Image of the human face DC voltage
Output	Audio signal
Functionality	The Raspberry Pi processes the input image then performs tasks like extracting facial features, identifying emotions expressed on the human face and emotion classification then outputs the classified emotion as an audio signal.

Table 3.11 Functionality of Raspberry Pi.

Module	Speaker
Input	Audio signal
Output	Audible classified emotions
Functionality	The speaker receives this audio signal and produces an audible output, allowing the user to hear the classified emotion

Table 3.12 Functionality of speaker.

Chapter 4 Experimental Results and Discussion

The implementation phase is where all the components of the project come together, and we provide the necessary details to clarify our project idea. This section aims to provide a clear understanding of the project's feasibility and applicability to the intended objectives. We will also discuss any challenges encountered during the implementation process, providing a comprehensive overview of our journey.



Figure 4.1 Component of the system

Section 4.1 Prototype Implementation

Our implementation process began by installing the Raspbian operating system on a MicroSD card using the Raspberry Pi Imager software. However, compatibility issues arose with certain versions of Raspbian, as not all supported the Tensorflow library. To address this, we identified and utilized a specific 64-bit version of the Raspberry Pi's operating system that was compatible with Tensorflow.



Figure 4.2 MicroSD card and Raspberry Pi Imager

Following the successful installation of the operating system, we set up the required environment and dependencies, including Tensorflow, PiCamera2 and OpenCV libraries. Tensorflow served as the backbone for building and training our CNN, which replaced the initial plan to use Fastai and transfer learning. PiCamera2 facilitated interaction with the Raspberry Pi camera module. These tools collectively enhanced the system's functionality, enabling the implementation of a robust deep learning application.

After that, we utilized Thonny, a Python IDE, to write code for testing the model using images stored on the desktop. We initially tested the model, which was built based on a VGGNet-inspired architecture, with the FER2013 dataset and specifically took a test image of a happy face. At this stage, we had not yet connected the camera and speaker to the Raspberry Pi. By executing the code, the output was printed in the Thonny shell, allowing us to observe the model's predictions and verify its performance. This step was crucial in ensuring that the model, designed to leverage the strengths of VGGNet's structured architecture, was functioning correctly and accurately identifying emotions based on the provided test image.

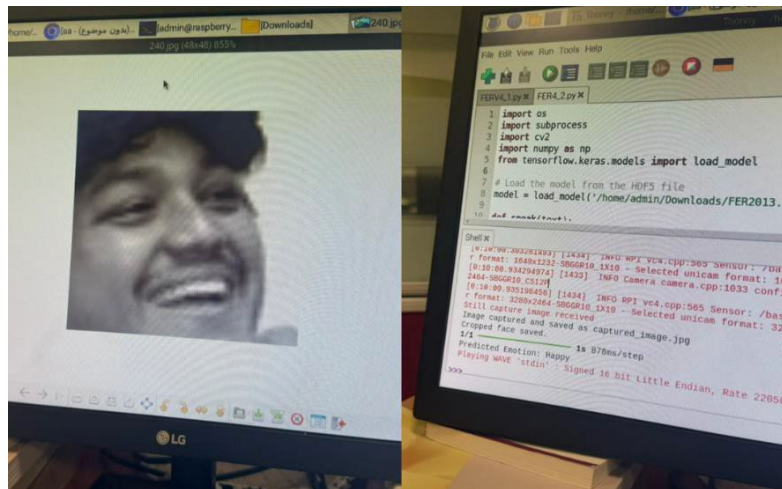


Figure 4.3 A sample from dataset and the predicted output

After assembling all the basic components of the project and setting up the system, we tested the camera on real-time images and observed the crucial importance of having a stable setup for capturing images. Ensuring that the images were properly aligned and centered was essential for accurate emotion recognition. Initially, we used a setup created with a 3D-printed Raspberry Pi 4 case that securely held all the components except the audio output as in figure 4.4. This setup provided a stable base for the camera and other essential components, preventing unwanted movement during image capture. The design of the case allowed for precise positioning and easy integration of the Raspberry Pi, camera module, and other hardware. This ensured a reliable and consistent system for testing and fine-tuning the emotion recognition model to meet our project's requirements.

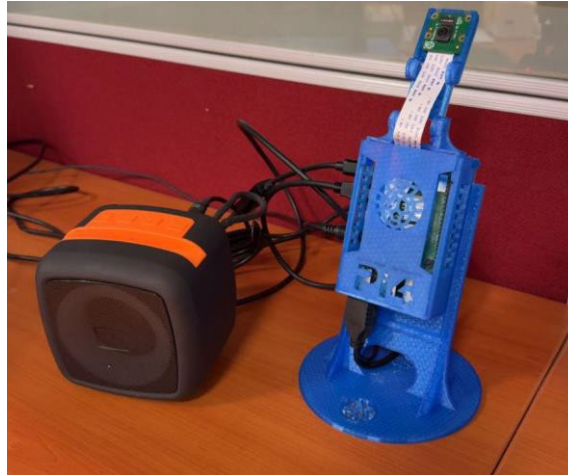


Figure 4.4 The final prototype

The user interface, shown in Figure 4.5, was developed to be simple and easy to use, ensuring visually impaired individuals can interact with the system without difficulty. It was created using Python and Tkinter, which provides a basic and reliable framework for building user-friendly interfaces. The interface includes essential features such as capturing images, analyzing facial expressions, and providing spoken feedback. With clear buttons and sound prompts, it guides users through each step. This design focuses on making the system straightforward while still being fast and effective, helping users easily understand and respond to emotions in real time.

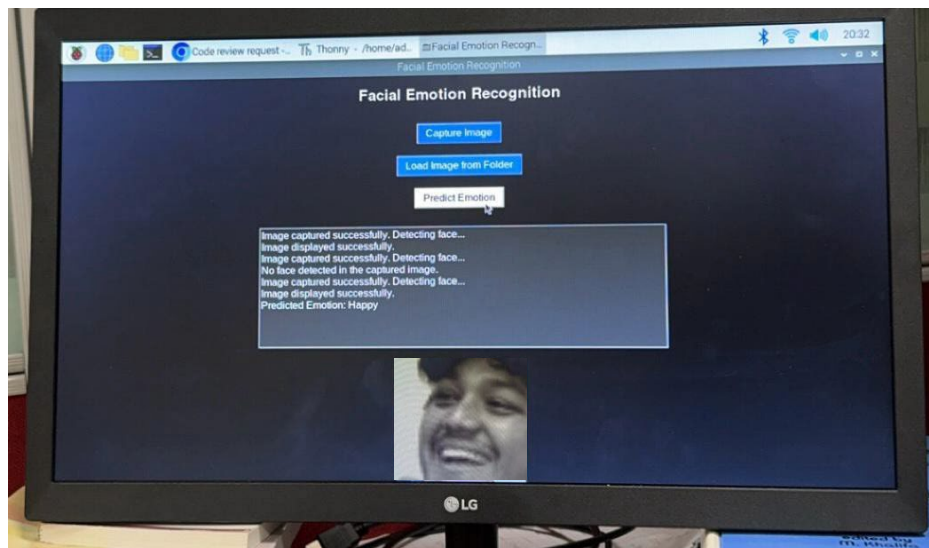


Figure 4.5 Interface of the system

Section 4.2 Prototype Verification

This section showcases the prototype testing results, which were conducted to verify the design requirements. The testing process included acceptance tests and integration tests.

Table 4.1 Acceptance Test

Test Case Name:		Facial emotion recognition System – Camera module test	Test ID #:		FERS-01	
Description:		capture the image of emotion and return the exact value of the emotion using a speaker.	Type:		■ white box	
Setup:		Connecting camera V2, and integrating space key and with speaker to the raspberry pi4 with good lighting and clear image				
Step	Action	Expected Results	Pass	Fail	N/A	Comments
1	Connect a speaker or headphone	Sound is clear	✓			
2	Capture the image	When capturing there should be a sound of camera click	✓			
3	Listening to value note	Listening to the value of the note captured by the camera.	✓			
Overall test result:			✓			

Table 4.2 Integration Test

Test Case Name:		Facial Emotion Recognition system – Camera with deep learning models	Test ID #:			FERS-02
Description:		The system's main requirement is for a Deep Learning model to recognize the emotions of an individual from the camera image.	Type:			■ black box
Setup:		Select an accurate Deep Learning model for recognizing Emotions from camera images under optimal conditions				
Step	Action	Expected Results	Pass	Fail	N/A	Comments
1	Load the deep learning model using python code	The VGGNet model should load effortlessly using Python and TensorFlow libraries on the Raspberry Pi system.	✓			Try again if its not recognized
2	Take a picture of happy	The model should accurately detect each emotion, particularly the emotion “happy” with high precision.	✓			Try again if its not recognized
3	Take a picture of sad	The model should accurately detect each emotion, particularly the emotion “sad” with high precision.	✓			Try again if its not recognized

4	Take a picture of angry	The model should accurately detect each emotion, particularly the emotion "angry" with high precision.	✓			Try again if its not recognized
5	Take a picture of neutral	The model should accurately detect each emotion, particularly the emotion "neutral" with high precision.	✓			Try again if its not recognized
6	Take a picture of fear	The model should accurately detect each emotion, particularly the emotion "fear" with high precision.	✓			Try again if its not recognized
7	Take a picture of surprised	The model should accurately detect each emotion, particularly the emotion "surprised" with high precision.	✓			Try again if its not recognized
Overall test result:			✓			

Chapter 5 Conclusion and Future Work

In conclusion, this project successfully developed a facial emotion recognition system using deep learning techniques, specifically designed to assist visually impaired individuals in understanding emotions around them. The system demonstrates the value of integrating technology to enhance quality of life by providing accurate and fast emotion recognition, converting facial expressions into easy-to-understand audio outputs. The results highlight the model's efficiency in classifying primary emotions, emphasizing its role as a supportive tool for improving social interaction for this group. This project marks a significant step towards fostering inclusivity through technological innovation.

References

- [1] W. S. Liste, "Blindness and vision impairment," World Health Organization, 10 August 2023. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment>. [Accessed 4 February 2024].
- [2] Alarcos Cieza, Stuart Keel, Ivo Kocur, "World report on vision," World Health Organization, Switzerland, 2019.
- [3] Hawton A, Green C, Dickens AP, Richards SH, Taylor RS, Edwards R, Greaves CJ, Campbell JL, "The impact of social isolation on the health status and health-related quality of life of older people," National Library of Medicine, 20 February 2011. [Online]. Available: <https://link.springer.com/article/10.1007/s11136-010-9717-2#citeas>. [Accessed 4 February 2024].
- [4] Andrada-Livia Cîrneanu, Dan Popescu and Dragos Iordache, "New Trends in Emotion Recognition Using Image Analysis by Neural Networks, A Systematic Review," Sensor, 23 August 2023. [Online]. Available: <https://doi.org/10.3390/s23167092>.
- [5] Manas Sikri, Archit Gupta, Raghav Puri, ShivendraGoel, Nitish Pathak and Mohit Tiwari, "Emotion Detection using Image Processing in Python," in *Computing for Sustainable Global Development*, New Delhi, 2018.
- [6] ANGANA MITRA , SOUVIK CHOUDHURY and SUSMITA MOITRA, "IMAGE PROCESSING FACIAL EXPRESSION RECOGNITION," RCC Institute of Information Technology, West Bengal, 2018.
- [7] S.-W. Jung, "Facial Action Units-Based Image Retrieval for Facial Expression Recognition," IEEE, South Korea, 2019.
- [8] Hsi-Chieh Lee, Chia-Ying Wu and Tzu-Miao Lin, "Facial Expression Recognition Using Image Processing Techniques and Neural Networks," National Quemoy University, Kinmen, 2013.
- [9] Wafa Mellouka and Wahida Handouzia, "FER using deep learning: review and insights," 12 August 2020. [Online]. Available: <http://www.sciencedirect.com/>.
- [10] D. D. Patil, "FER USING DEEP LEARNING," California State University, Northridge, 2023.
- [11] Edeh Michael Onyema, Piyush Kumar Shukla, Surjeet Dalal, Mayuri Neeraj Mathur, Mohammed Zakariah and Basant Tiwari, "Enhancement of Patient Facial Recognition through Deep Learning Algorithm: ConvNet," *Journal of Healthcare Engineering*, p. 8, 2021.

- [12] Meriem Sari, Abdelouahab Moussaoui and Abdenour Hadid, "Automated Facial Expression Recognition Using Deep Learning Techniques: An Overview," *International Journal of Informatics and Applied Mathematics*, p. 15, 2019.
- [13] E. B. Kirkvik, "FER using Deep Learning," Oslo Metropolitan University, Oslo, 2022.
- [14] Hendra Kusuma, Muhammad Attamimi and Hasby Fahrudin, "Deep learning based facial expressions recognition system for assisting visually impaired persons," *Bulletin of Electrical Engineering and Informatics*, vol. 9, p. 12, 2020.
- [15] DINA SHEHADA, AYAD TURKY, WASIQ KHAN, BILAL KHAN and ABIR HUSSAIN, "A Lightweight FER System Using Partial Transfer Learning for Visually Impaired People," *IEEE Access*, vol. 11, p. 9, 2023.

Appendix A The Project Plan

Section A.1 Work Breakdown Structure

Table A.1 WBS

ID	Activity	Description	Deliverables / Checkpoints	Duration (days)	People	Resources	Predecessors
	FER			265			
	Research	Searching over the internet and research papers	Have some knowledge about the project	20	Bassam Mshari Hatem	IEEE Research Gate	
	Gather Information	Collect related information from different sources	Sort out the gathered information	20	Bassam Mshari Hatem	IEEE Research Gate	
	Update With Supervisor			3	Bassam Mshari Hatem		
	Report 1			47			
1	Problem Statement	Define the issue and how to solve it	Have a need statement, objective statement, and user needs	7	Bassam Mshari Hatem		1.1,1.2,1.3
	Update With Supervisor			1	Bassam Mshari Hatem		
2	Research Survey	Gathering data through participant responses.		10	Bassam Mshari Hatem		
3	Full Requirement Specification	Translate user needs to engineering requirements	Extract the engineering requirements	10	Bassam Mshari Hatem		3.1,3.2

4	Design	Choose the suitable components and define how to implement it	A final chosen design	20	Bassam Mshari Hatem	IEEE Research Gate Raspberry pi	4.1,4.2
	Update With Supervisor			1	Bassam Mshari Hatem		
	Buy Components	Identify parts, Place order	have the components needed to the system	3	Bassam Mshari Hatem	Amazon	
	Hardware integration	Combine all hardware components	All hardware components are combined	7	Bassam Mshari Hatem		
	Software And Coding	Writing codes needed for the system	Meet all the requirement and translate it to code	70	Bassam Mshari Hatem		
	Testing	Testing codes	Check each function and verify	14	Bassam Mshari Hatem		
	Report 2	Writing report 2		4	Bassam Mshari Hatem		
	Implementation and verification	Implement and verify the whole system	Testing each module on the field	7	Bassam Mshari Hatem		
	Summary and conclusion	Writing summary and conclusion		1	Bassam Mshari Hatem		
	Testing project	Test the whole system	Have the system work on the field	14	Bassam Mshari Hatem		

Section A.2 Gantt Chart

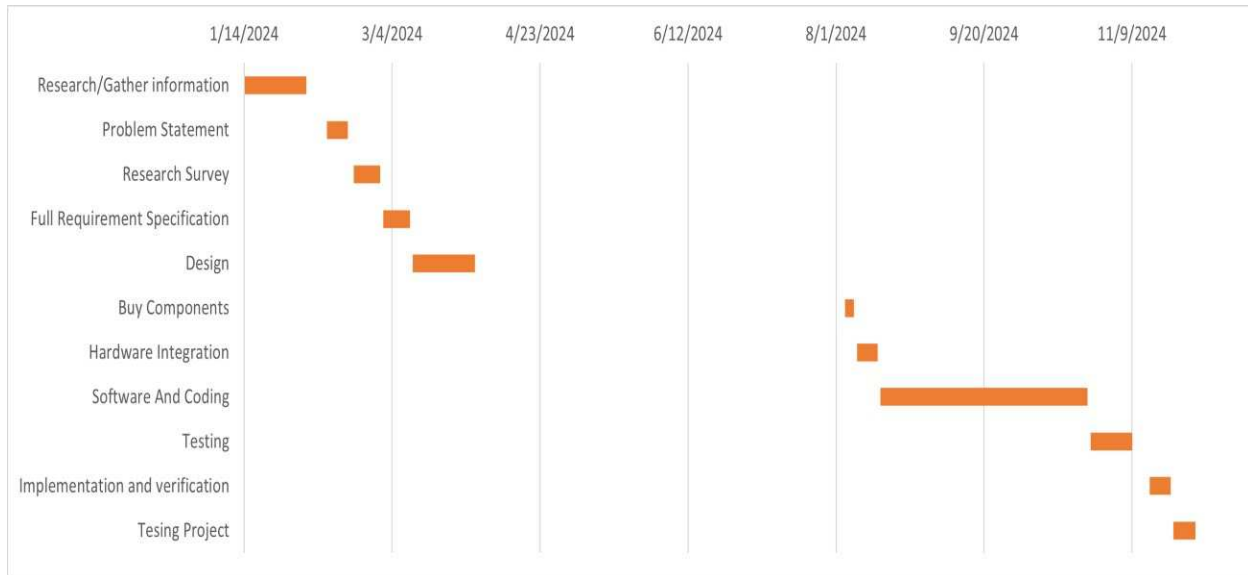


Figure A.1 Gantt Chart

Section A.3 Cost Estimation

Table A.2 Cost Estimation

Equipment	Cost
Raspberry Pi 4 + components	550 SR
Camera	115 SR
Speaker	60 SR
Total Price	725 SR