Major Project Report on

Encoding Candlesticks as Images for Patterns Classification Using Convolutional Neural Networks

Submitted in partial fulfillment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

by

Sohanraj R (191IT149)

under the guidance of

Dr. Biju R Mohan

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE - 575025

February, 2023

DECLARATION

I/We hereby *declare* that the Major Project-I Work Report entitled ***"Encoding Candlesticks as Images for Patterns Classification Using Convolutional Neural Networks"*** , which is being submitted to the **National Institute of Technology Karnataka, Surathkal**, for the award of the Degree of Bachelor of Technology in Information Technology, is a *bonafide report of the work carried out by me/us.* The material contained in this Major Project Report has not been submitted to any University or Institution for the award of any degree.

Name of the Student (Registration Number) with Signature

(1) Sohanraj R(191IT149)

Department of Information Technology
Place : NITK, Surathkal
Date : 10/04/2023

CERTIFICATE

This is to *certify* that the Major Project Work Report entitled ***Encoding Candlesticks as Images for Patterns Classification Using Convolutional Neural Networks*** submitted by

Name of the Student (Registration Number)

(1) Sohanraj R (191IT149)

as the record of the work carried out by them/, is *accepted as the B.Tech. Major Project work report submission* in partial fulfillment of the requirement for the award of degree of Bachelor of Technology in Information Technology in the Department of Information Technology, NITK Surathkal.

(Dr. Biju R Mohan)
Assistant professor
Department of Information Technology
NITK Surathkal

ABSTRACT

Candlestick charts display the high, low, open, and closing prices for a certain time series. Candlestick patterns develop as a result of repetitive, repeated human activities and reactions that are modelled and documented in the candle creation. A total of 103 can- In Thomas Bulkowski's Encyclopedia of Candlestick Charts, candlestick patterns are shown. These are used by investors. Use patterns to determine when to come in and go out. The categorization of candlestick patterns, however, makes it simpler to distinguish them visually. The "GASF-CNN" approach of extended convolutional neural networks (CNN) is presented in this paper to automatically recognise candlestick patterns. Using the Gramian Angular Field, the time series are encoded as various types of images. (GAF). Then, we use the CNN for GAF image to discover eight essential categories of candlestick patterns. . The simulation and testing findings show that our technique is capable of identifying eight different types of candlestick patterns.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Overview

The forecasting of financial markets, such as predicting swings or volatility estimates for futures indexes, is one of the most crucial fields of research in commercial finance and information engineering. Because market prices are based on and subject to the predicted psychological effect of the whole market, it is possible to complete predictive models of financial demand by specific pre-processing and coding, or more complex model designs.

Numerous tools are already available to help people predict changes in stock prices and futures indices, but they are almost exclusively used in academic research. These tools include neural networks, fuzzy time series analysis, gene algorithms, classification trees, statistical regression models, support vector machines, etc. Rarely do traditional machine learning forecasting algorithms combine financial knowledge.Because the ordinary individual wants to make money from the transaction, the predictions made by the aforementioned model are insufficient for application in real-world operations. Given that investment predictions and model projections can occasionally have big gaps, investors are more likely to find a reliable entry and exit point than to simply predict prices. . The timing of their entry and exit (how much profit room they have) is all that matters to these anonymous investors, according to various research [1, 2, 3, 4, 5, 6]. To put it another way, it is better to directly include the core knowledge of transactions to create a reliable application model than to pursue irrational low-risk, high-accuracy profit models using machine learning or deep learning architecture.

1.2 Motivation

One of the most important skills for assessing market circumstances is the ability to recognise candlestick patterns . Traders frequently make decisions based on complex information that is presented to them. News, candlestick patterns, technical indicators, and other information are included in the content. Identification of candlestick patterns is one of them that is crucial for helping with certain trades. By identifying candlestick patterns, traders may determine the market's current asset price as well as whether or not the present buying or selling pressure will continue or shift. After then, this data is combined with data from other sources to assist traders in forecasting the future. Changes in price trend and a few more well-known morphologies, such as the Morning Star and the Evening Star, are examples of price reversal signals. Candlestick patterns, on the other hand, are gradually being detected through studies of trading skill rather than utilising numerical analysis. Due to this comprehension, traders are required to create judgements about how pictures seem visually.

The Convolutional Neural Networks (CNN) model is the best option for image recognition. In order to extract the best visual features, the CNN may train the appropriate weights and update its convolution kernel via backward propagation. By evaluating the association between traits and images, the model is then utilised to make judgements. In addition, a neural network suitable for photo recognition is built using a two-dimensional convolution. However, the financial time series data is essentially represented as a one-dimensional array. As a result, we must determine how to convert the time series data into an appropriate data matrix.

CHAPTER 2

LITERATURE REVIEW

2.1 Background and Related Works

(1)Shewhart charts are used to monitor the process mean (and/or variance) estimate and prediction. (e.g. Chen and Elsayed, 2000, Wu, 2006). When examining assignable causes and putting the required corrective measures in place as potential feedback controls, this estimate might be a useful tool. Using state-space models, autoregressive integrated moving average (ARIMA) models, exponentially weighted moving average (EWMA) control charts, cumulative sum (CUSUM) control charts, and EWMA control charts, you can describe and monitor processes in this category that are typically controlled by linear stochastic systems. These control charts are the best options to use when identifying minute process shifts requires great performance.
(2)The study by Boobalan (2014) is based on in-depth technical analysis of a small number of carefully selected companies, which helps in understanding share price behavior, the signals they transmit, and the crucial market price turning points. This article's objective is to do technical analysis on the stocks of the selected companies in order to support investment decisions made on the Indian Market based on forecasts from five companies: WIPRO, SBIN, GAIL, ONGC, and ITC. Technical analysis is crucial for predicting short- and medium-term market trends and assisting investors in selecting the appropriate trading approach. The various stock price trends of these businesses give an insight of the potential future direction of these businesses.
(3), Jun-Hao Chen, Yun-Cheng Tsai, and Wang3, Chun-Chieh They advise using the "GASF-CNN" approach, which extends Convolutional Neural Networks, for automatic candlestick pattern detection. (CNN).We use the Gramian Angular Field to encode the time series as different types of graphics. (GAF). After that, we use CNN with GAF encoding to find eight key types of candlestick patterns. The simulation and testing results demonstrate that our system has a 90 percent accuracy rate for automatically identifying the eight various types of candlestick patterns.Using Candlestick Charting and an Ensemble Machine Learning Technique with a Novelty Feature Engineering Scheme, Stock Trend Prediction. To forecast stock prices based

on pattern occurrence, the authors offer an ensemble learning approach for stock prediction.They have 8 trigram patterns and 13 key line patterns.

(5)Marc Velay and Fabrice Daniel's research (June 2018) compared the abilities of CNN and LSTM to recognise common chart patterns in historical stock data. It shows two common patterns, the method used to create the training set, the architecture of the neural network, and accuracy outcomes.

(6)Yaohu Lin, Shancun Liu, Haijun Yang, Harris Wu, and Bingbing Jiang conducted this investigation. PRML, a brand-new candlestick pattern detection model applying machine learning techniques, is introduced to improve stock trading decisions. The pattern recognition schedule starts by applying four well-known machine learning algorithms and eleven different feature types to every possible combination of daily patterns. Multiple time frames between one and ten days are used to calculate the impact of the predictions at different points in time. An investment strategy is developed based on the identified candlestick patterns and the best time window. They forecast all stocks on the Chinese market between January 1, 2000, and October 30, 2020 using PRML. There are also the data from January 1, 2015, to October 31, 2015.

(7)Artificial Intelligence System for Expert Advisor Trading System Prediction (2022) M. L. Bin Yasruddin, T. W. Keong, Z. Bin Husin, and M. A. H. Bin Ismail The paper discusses a technical analysis for predicting foreign exchange prices using Support Vector Machines and Japanese Candlestick Patterns. (SVM). The experiment was done on the Euro-Dollar (EURUSD) price, and the results showed that the Japanese candlestick pattern and SVM algorithm work well together to predict the EURUSD price.

2.2 Outcome of Literature Review

Evening star, morning star, and other candlestick patterns are important in making trade decisions because they can provide valuable information about potential trend reversals, confirm market sentiment, serve as entry and exit signals, and contribute to a trader's risk management strategy. However, it's important to note that no trading signal or pattern is 100 percent accurate, and it's always prudent to consider other factors, such as fundamental analysis, market conditions, and risk management, when making trading decisions. Various research as seen below has gone through to analyze these candlestick chart patterns. Different researches has shown the candlesticks patterns can be useful in predicting the nature of stock market movement.So we plan to build a machine learning model to recognize some of the patterns.

Table 2.2.1: Literature review

Case	Title	Inference
1.Chen and Elsayed, 2000, Wu, 2006	Process mean (and/or variance) estimation and prediction, monitored by Shewhart charts	Understand the way the stock market works and what the basic financial terms mean and affect the market
2.Boobalan (2014)	Technical analysis of a few carefully chosen firms	To understand how traders actually come up on decisions on real life scenarios.
3.Yun-Cheng Tsai,1 Jun-Hao Chen,2 Chun-Chieh Wang3, For automatic candle- stick pattern recognition	automatic candlestick pattern recognition by encoding into images	Explains conversion of time series data to images using GAF and analysing it with a CNN model.

4.Y. Lin, S. Liu, H. Yang and H. Wu,	Stock Trend Prediction Using Candlestick Charting and Ensemble Machine Learning Technique With a Novelty Feature Engineering Scheme	Gain knowledge of the no. of patterns present in candle stick chart which includes 13-key line patterns and 8 trigram patterns.
5.Marc Velay and Fabrice Daniel (June 2018)	Stock Chart Pattern recognition with Deep Learning	It dhows the effectiveness of CNN and LSTM models to predict certain patterns, as well as some data cleaning techniques and model architecture.

6.Yaohu Lin,Shancun Liu,Haijun Yang ,Harris Wu,Bingbing Jiang(August 6,2021)	Improving stock trading decisions based on pattern recognition using machine learning technology	Shows how different models(11 different Machine learning models) perform in predicting one day patterns in candle-stick charts. They also perform a comparative study to decide which model perform better. The PRML model proposed by them gives them good returns on investments on most type of trades such as 2-day ahead, 3-day ahead etc.

7.Prediction of Expert Advisor Trading System Using An Artificial Intelligence System (2022)	M. A. H. Bin Ismail, Z. Bin Husin, T. W. Keong and M. L. Bin Yasruddin	The paper discusses a technical analysis for predicting foreign exchange prices using Support Vector Machines and Japanese Candlestick Patterns. (SVM). The experiment was done using the Euro-Dollar (EURUSD) price, and the results showed that the Japanese candlestick pattern and SVM algorithm work well together to provide information for EURUSD price prediction

8. Modeling for Stock Trends: A Study of Two-Stage Pattern Strategy (2022)	Y. He, H. Zhou, S. Kimm and J. Xue	The paper offered a technique for identifying stock price trends and used a model to firmly back them. To calculate the prediction power and learn how the pattern may be exploited to increase return, two derived methodologies and two trading rules have been applied.This study comes to the conclusion that two consecutive white bars often suggest an upward trend whereas two consecutive black bars typically represent an adverse trend. .

9 Do Candlestick Patterns Work in Cryptocurrency Trading? (2021)	K. -H. Ho, T. -T. Chan, H. Pan and C. Li	The usefulness of candlestick patterns in bitcoin trading is examined in this research. Historical daily opening, high, low, and closing prices for the top 23 cryptocurrencies by market capitalization are included in the data set.68 popular candlestick patterns were examined using statistical analysis, and it was discovered that they are not very useful for trading cryptocurrencies. .

2.3 Problem Statement

To design a Deep learning based model to classify key line patterns in candlestick stock market charts.

2.4 Objectives of the Project

(1) To implement Gramian angular field method to convert time series data features to images.

(2) To create a dataset of patterns by recognizing them mathematically through time series data of highly fluctuating stocks.

(3) Build a image pattern recognition model that recognizes patterns.

(4) Do a performance analysis of the above model as per our requirements.

CHAPTER 3

PROPOSED METHODOLOGY

3.1 Simulation data generation and dataset creation

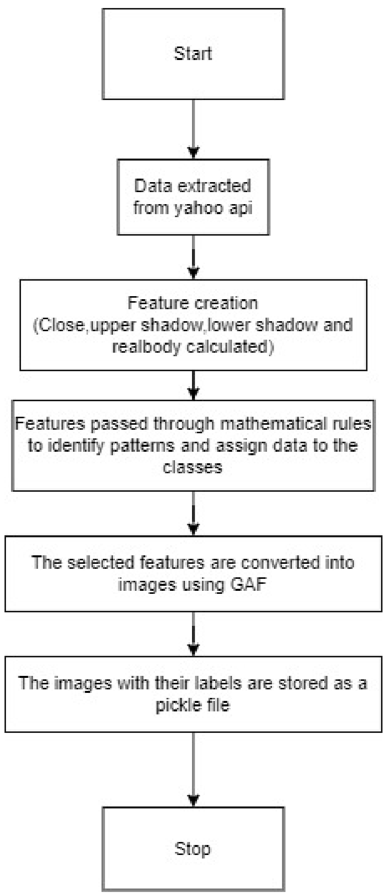

Figure 3.1.1: Flow diagram for dataset creation

1. Taking data from nsc stocks through python using the nsc api.
2. Generating simulation data using GBM(Geometric brownian motion) using the above nsc data as training set to widen the range of our data.

3. Calculating features such as close,lower shadow,upper shadow and realbody from the candlestick features.

4. Passing the features through mathematical functions to assign them into one of the 8 classes.

5. Converting the numerical features to images using GAF.

6. Assigning the classes to the images.

3.1.1 Dataset creation

To identify trends, we will use data from highly volatile equities like TCS, L T Infotech, etc.We selected the eight most common candlestick patterns as our training aim based on "The Major Candlesticks Signals," one of the major textbooks on candlestick patterns. As our eight candlestick patterns, we choose the Morning Star, Bullish Engulfing, Hammer, Shooting Star, Evening Star, Bearish Engulfing, Hanging Man, and Inverted Hammer. All of them are reversal patterns that show when the price is likely to change. The first four patterns indicate that the price changes from a downtrend to an uptrend, while the last four patterns demonstrate the opposite. . We'll give a thorough example of each situation in the section below. A price resurgence following a decline can be identified using the Morning Star pattern.The explanation of this design is broken down into three stages. A fall must first be demonstrated since it shows that there is no market confidence at all.Second, from the gloomy surroundings will finally emerge a big black bar.After a calm day, the third bar will transform into a big white bar. Investors believe that this is how the market's confidence will shift.Price trends that shift from upward to downward are recognised using the Evening Star pattern. The description of this design is broken down into three sections as well. First, it is necessary to confirm an upsurge, which shows that the market is generally quite optimistic. Second, the end of great days will be indicated by a broad white bar. After a calm day, the third bar will transform into a substantial black bar. In this sense, investors expect a shift in market confidence.The decline and upswing components are discernible by regression. Regardless of how steep or how low the slope is, the pattern will be confirmed. The other patterning rules also vary significantly between the simulation and the real data, which brings us to our last conclusion. The rules for the simulation data are identical to those in the book. But there aren't enough examples since the constraints on using real data are so rigorous. We will thus considerably narrow the criteria in order to get sufficient data. As a result, we will significantly reduce the criteria in order to collect enough information. For instance, in order for the Bullish Engulfing pattern to occur, the next bar's open price must be lower than the preceding bar's close price.

.

3.1.2 Generating simulation data using GBM

In a continuous-time stochastic process called a geometric Brownian motion (GBM), also known as an exponential Brownian motion, the logarithm of the randomly fluctuating variable follows a Brownian motion (also known as a Wiener process), with drift. It is used in mathematical finance primarily to mimic stock prices in the Black-Scholes model and is a prominent example of stochastic processes satisfying an SDE. The following are some justifications for using GBM to model stock prices:

- According to what we would predict in practice, the projected returns of GBM are independent to the process's value (the stock price).
- Only positive numbers, like actual stock prices, are presumptions in a GBM process.
- Similar to real stock values, a GBM process exhibits "roughness" in its routes.
- GBM procedures make calculations very simple.

Using The GBM model we added like double the initial stock data that was taken, therefore increasing the range of data we have and also taking into account all the variations that are possible.

3.1.3 Calculating new distinct features

The base paper has given an approach of calculating other distinct features using the candlestick features open, close,high and low. The other distinct features are as follows:

- Close
- lower shadow
- upper shadow
- Real body

The base paper states that these features when used for training gave better results in comparison to the basic candlestick features. These features are calculated as follows.

```
if slope:
        # process slpoe
        data['diff'] = data['close'] - data['open']
        data = data.query('diff != 0').reset_index(drop=True)
        data['direction'] = np.sign(data['diff'])
        data['ushadow_width'] = 0
        data['lshadow_width'] = 0

        for idx in trange(len(data)):
            if data.loc[idx, 'direction'] == 1:
                data.loc[idx, 'ushadow_width'] = data.loc[idx, 'high'] -
                data.loc[idx, 'lshadow_width'] = data.loc[idx, 'open'] -
            else:
                data.loc[idx, 'ushadow_width'] = data.loc[idx, 'high'] -
                data.loc[idx, 'lshadow_width'] = data.loc[idx, 'close']
```

3.2 Types of Patterrns and their Classification methods

3.2.1 Morning star

The visual pattern known as a morning star is composed of a tall black candlestick, a smaller black or white candlestick with a short body and long wicks, and a third tall white candlestick. The middle candle of the morning star represents a time of market turmoil when bulls begin to outnumber bears. The third candle may suggest a new upsurge, validating the reversal. The pattern known as the evening star, which contrasts with the morning star, indicates the change from an upswing to a decline.

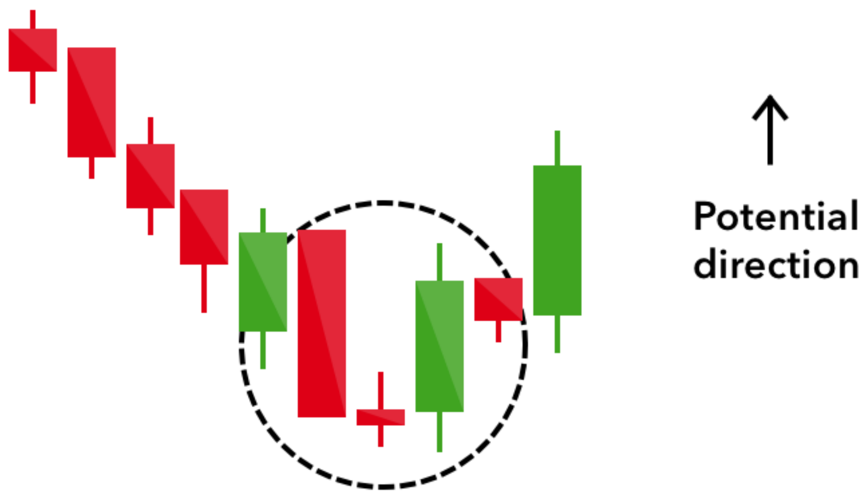

Figure 3.2.1: Morning Star

```
def detect_morning_star(data, q=None, multi=False, short_per=35, lon
'''Detect morning star pattern
Args:
    short_per (int): percentile for determination.
    long_per (int): percentile for determination.

Returns:
    dataframe.
'''
print('[ Info ] : detecting morning star')
temp = data[(data['previous_trend'] == -1)
& (data['direction'] == -1)].index
```

```
data['morning'] = 0
try:
    for idx in tqdm(temp):
        cond1 = (data.loc[idx, 'body_per'] >= long_per)
        cond2 = (data.loc[idx+1, 'body_per'] <= short_per)
        cond3 = (data.loc[idx+2, 'direction'] == 1)
        cond4 = (data.loc[idx+1, 'close'] +
        data.loc[idx+1, 'open'])/2 <= data.loc[idx, 'close']
        cond5 = data.loc[idx+2, 'close'] >= ((data.loc[idx, 'open']
        + data.loc[idx, 'close'])/2)
        cond7 = (data.loc[idx+2, 'open'] >= (data.loc[idx+1, 'open']
        + data.loc[idx+1, 'close'])/2)
        if cond1 & cond2 & cond3 & cond4 & cond5 & cond7:
            data.loc[idx+2, 'morning'] = 1
except:
    pass

if multi:
    q.put({'morning': np.array(data['morning'])})
else:
    return data
```

3.2.2 Evening star

On stock price charts, technical analysts use the evening star pattern to identify when a trend is about to reverse itself. The bearish candlestick pattern consists of three candles: a red candle, a small-bodied candle, and a massive white candlestick. The pinnacle of a market upsurge and the emergence of evening star patterns are two signs that the uptrend is about to stop. The evening star pattern is the opposite of the morning star pattern, which is considered to be a bullish sign.

Figure 3.2.2: Evening star

```
def detect_evening_star(data, q=None, multi=False, short_per=35, lon
'''Detect evening star pattern
Args:
    short_per (int): percentile for determination.
    long_per (int): percentile for determination.

Returns:
    dataframe.
'''
print('[ Info ] : detecting evening star')
temp = data[(data['previous_trend'] == 1) & (data['direction'] == 1)
data['evening'] = 0
try:
    for idx in tqdm(temp):
        cond1 = (data.loc[idx, 'body_per'] >= long_per)
        cond2 = (data.loc[idx+1, 'body_per'] <= short_per)
        cond3 = (data.loc[idx+2, 'direction'] == -1)
        cond4 = (data.loc[idx+1, 'close'] +
```

```
            data.loc[idx+1, 'open'])/2 >= data.loc[idx, 'close']
            cond5 = data.loc[idx+2, 'close'] <=
            ((data.loc[idx, 'open'] + data.loc[idx, 'close'])/2)
            cond7 = (data.loc[idx+2, 'open'] <=
            (data.loc[idx+1, 'open'] + data.loc[idx+1, 'close'])/2)
            if cond1 & cond2 & cond3 & cond4 & cond5 & cond7:
                data.loc[idx+2, 'evening'] = 1
    except:
        pass

    if multi:
        q.put({'evening': np.array(data['evening'])})
    else:
        return data
```

3.2.3 Shooting star

After a gain, a shooting star appears, signalling that the price may soon begin to decline. The price attempted to climb sharply during the day, but the sellers grabbed control and forced the price back down towards the open, making the pattern negative. Normally, after a shooting star, traders watch to observe what the subsequent candle (period) performs. They could sell or go short if the price falls throughout the next term. If the price increases following a shooting star, it's possible that the formation was a false signal or that the candle is indicating a potential resistance zone within the candle's price range.

```
    def detect_shooting_star(data, q=None,
    multi=False, short_per=35, long_per=65):
    '''Detect shooting star pattern
    Args:
        short_per (int): percentile for determination.
        long_per (int): percentile for determination.
```

Figure 3.2.3: Shooting star

```
Returns:
    dataframe.
'''
print('[ Info ] : detecting shooting star')
data['shooting_star'] = 0
temp = data[(data['previous_trend'] == 1)
& (data['direction'] == 1)].index
try:
    for idx in tqdm(temp):
        cond1 = (data.loc[idx, 'body_per'] >= long_per)
        cond2 = (data.loc[idx, 'direction'] == 1)
        cond3 = (data.loc[idx+1, 'ushadow_width'] > 2 * abs(data.loc
        cond4 = (min(data.loc[idx+1, 'open'],
        data.loc[idx+1, 'close']) >
        ((data.loc[idx, 'close'] + data.loc[idx, 'open']) / 2))
        cond5 = (data.loc[idx+1,
        'lower_per'] <= short_per - 10)  # 25
```

```
            cond6 = (data.loc[idx+1, 'upper_per'] >= long_per)
            if cond1 & cond2 & cond3 & cond4 & cond5 & cond6:
                data.loc[idx+1, 'shooting_star'] = 1
    except:
        pass

    if multi:
        q.put({'shooting_star': np.array(data['shooting_star'])})
    else:
        return data
```

3.2.4 Hanging man

A possible upward reversal is indicated by the hanging man, a type of candlestick pattern that refers to the shape and appearance of the candle. Candlesticks demonstrate how market sentiment impacts price changes by displaying the high, low, opening, and closing prices for a security over a specific time period. The hanging man pattern appears when an asset has been in an upswing and the candle has a small true body and a long lower shadow. .

```
    def detect_hanging_man(data, q=None, multi=False, short_per=35, long
    '''Detect hanging man pattern
    Args:
        short_per (int): percentile for determination.
        long_per (int): percentile for determination.

    Returns:
        dataframe.
    '''
    print('[ Info ] : detecting hanging man')
    data['hanging_man'] = 0
    temp = data[(data['previous_trend'] == 1) & (data['direction'] == 1)
```

Figure 3.2.4: Hanging man

```
try:
    for idx in tqdm(temp):
        cond1 = (data.loc[idx, 'lshadow_width'] > 2 * abs(data.loc[i
        cond2 = (data.loc[idx, 'body_per'] <= short_per)
        cond3 = (data.loc[idx, 'upper_per'] <= (short_per - 10))
        cond4 = (data.loc[idx, 'lower_per'] >= long_per)
        if cond1 & cond2 & cond3 & cond4:
            data.loc[idx, 'hanging_man'] = 1
except:
    pass

if multi:
    q.put({'hanging_man': np.array(data['hanging_man'])})
else:
    return data
```

3.2.5 Bullish engulfing

A bullish engulfing pattern is formed by a white candlestick with an opening that is lower than the previous day's close and a closing that is higher than the opening. when a little black candlestick, signifying a bearish trend, is followed the following day by a huge white candlestick, signifying a bullish trend, the latter of which has a body that completely covers or engulfs the former. The next day, a giant white candlestick, whose body entirely engulfs or overflows the body of the little black candlestick, is followed by a small black candlestick. A bullish engulfing candlestick pattern is what this pattern is called. It is more likely that bullish engulfing formations that are preceded by four or more black candlesticks represent reversals. Investors should also take a close look at the candlesticks that came before the two that make up the bullish engulfing pattern.

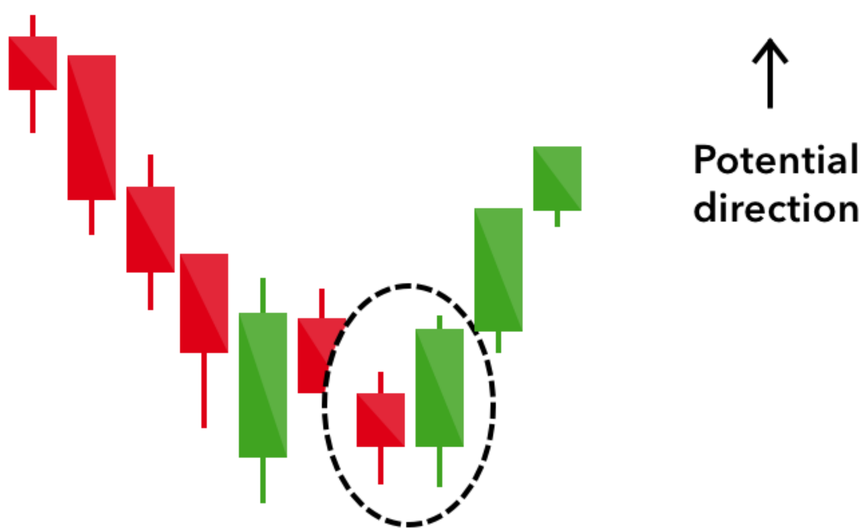

Figure 3.2.5: Bullish engulfing

```
def detect_bullish_engulfing(data, q=None, multi=False, short_per=35
'''Detect bullish engulfing pattern
Args:
```

```
        short_per (int): percentile for determination.
        long_per (int): percentile for determination.

    Returns:
        dataframe.
    '''
    print('[ Info ] : detecting bullish engulfing')
    data['bullish_engulfing'] = 0
    temp = data[(data['previous_trend'] == -1) & (data['direction'] ==
    try:
        for idx in tqdm(temp):
            cond1 = (data.loc[idx, 'direction'] == -1)
            cond2 = (data.loc[idx, 'body_per'] >= long_per)
            cond3 = (data.loc[idx+1, 'direction'] == 1)
            cond4 = (data.loc[idx+1, 'close'] > data.loc[idx, 'open'])
            cond5 = (data.loc[idx+1, 'open'] < data.loc[idx, 'close'])
            if cond1 & cond2 & cond3 & cond4 & cond5:
                data.loc[idx+1, 'bullish_engulfing'] = 1
    except:
        pass

    if multi:
        q.put({'bullish_engulfing': np.array(data['bullish_engulfing'])}
    else:
        return data
```

3.2.6 Bearish engulfing

A bearish engulfing pattern can appear anywhere, but after a price increase its significance increases. This may indicate a pullback in the upward direction or an upswing within a deeper decline. In respect to the price bars surrounding them, the proper size for both candles is important. Two really small bars could make an enveloping pattern, but it would be far less significant than if both candles were enormous. The "true body" of the candlesticks, or the difference between the open and closing prices, is what matters. The real body of the down candle must encompass the up candle. The pattern is substantially less prominent in volatile markets.

Figure 3.2.6: Bearish engulfing

```
def detect_bearish_engulfing(data, q=None, multi=False, short_per=35
'''Detect bearish engulfing pattern
Args:
    short_per (int): percentile for determination.
    long_per (int): percentile for determination.

Returns:
    dataframe.
'''
print('[ Info ] : detecting bearish engulfing')
data['bearish_engulfing'] = 0
temp = data[(data['previous_trend'] == 1)
& (data['direction'] == 1)].index
```

```
try:
    for idx in tqdm(temp):
        cond1 = (data.loc[idx, 'direction'] == 1)
        cond2 = (data.loc[idx, 'body_per'] >= long_per)
        cond3 = (data.loc[idx+1, 'direction'] == -1)
        cond4 = (data.loc[idx+1, 'close'] < data.loc[idx, 'open'])
        cond5 = (data.loc[idx+1, 'open'] > data.loc[idx, 'close'])
        if cond1 & cond2 & cond3 & cond4 & cond5:
            data.loc[idx+1, 'bearish_engulfing'] = 1
except:
    pass

if multi:
    q.put({'bearish_engulfing': np.array(data['bearish_engulfing'])})
else:
    return data
```

3.2.7 Hammer

Hammer candlesticks typically show up after a price decline. Their shadow at the bottom is long and their actual body is little. The hammer candlestick is created when buyers enter the market when prices are dropping. When the market closes, buyers have absorbed the selling pressure and have nearly restored the opening price of the market. The close may be above or below the beginning price, but it should be near to the open for the real body of the candlestick to remain modest. The bottom shadow should be at least twice as tall as the actual body. Hammer candlesticks point to a potential price reversal to the upside. The price must start moving higher after the hammer; this is referred to as confirmation.

```
def detect_hammer(data, q=None, multi=False, short_per=35, long_per=
'''Detect hammer pattern
```

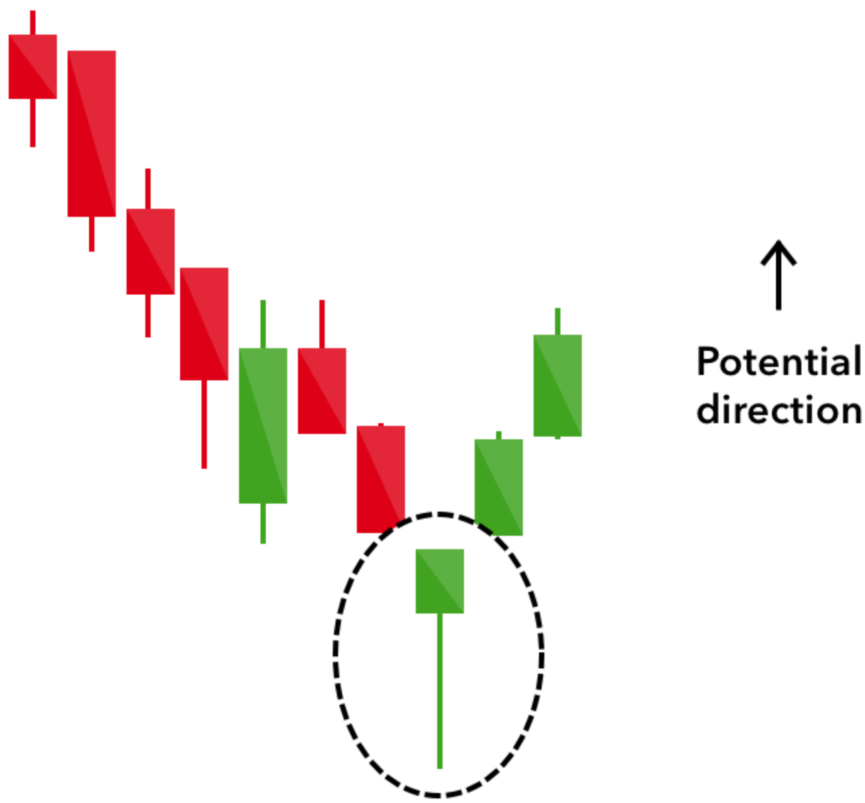

Figure 3.2.7: Hammer

```
Args:
    short_per (int): percentile for determination.
    long_per (int): percentile for determination.

Returns:
    dataframe.
'''
print('[ Info ] : detecting hammer')
data['hammer'] = 0
temp = data[(data['previous_trend'] == -1)
& (data['direction'] == -1)].index
try:
    for idx in tqdm(temp):
```

```
            cond1 = (data.loc[idx, 'lshadow_width'] > 2 * abs(data.loc[i
            cond2 = (data.loc[idx, 'body_per'] <= short_per)
            cond3 = (data.loc[idx, 'upper_per']
            <= (short_per - 15))
            cond4 = (data.loc[idx, 'lower_per'] >= long_per)
            if cond1 & cond2 & cond3 & cond4:
                data.loc[idx, 'hammer'] = 1
    except:
        pass

    if multi:
        q.put({'hammer': np.array(data['hammer'])})
    else:
        return data
```

3.2.8 Inverted Hammer

Another bullish pattern is the inverted hammer. The only difference is the length of the top wick in comparison to the shortness of the bottom wick. It means that after a strong buying demand, a weak selling pressure followed, failing to drop the market price. According to the inverse hammer, buyers will soon dominate the market.

```
    def detect_inverted_hammer(data, q=None, multi=False, short_per=35,
    '''Detect inverted hammer pattern
    Args:
        short_per (int): percentile for determination.
        long_per (int): percentile for determination.

    Returns:
        dataframe.
    '''
```

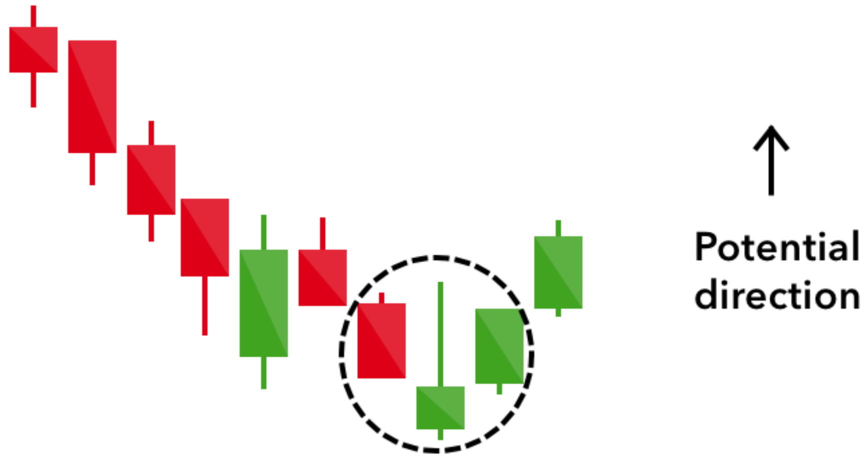

Figure 3.2.8: Inverted hammer

```
print('[ Info ] : detecting inverted hammer')
data['inverted_hammer'] = 0
temp = data[(data['previous_trend'] == -1)
& (data['direction'] == -1)].index
try:
    for idx in tqdm(temp):
        cond1 = (data.loc[idx, 'direction'] == -1)
        cond2 = (data.loc[idx, 'body_per'] >= long_per)
        cond3 = (data.loc[idx+1, 'ushadow_width'] > 2 * abs(data.loc
        cond4 = (max(data.loc[idx+1, 'open'], data.loc[idx+1, 'close
        cond5 = (data.loc[idx+1, 'lower_per'] <= short_per)
        cond6 = (data.loc[idx+1, 'upper_per'] >= long_per)
        if cond1 & cond2 & cond3 & cond4 & cond5 & cond6:
            data.loc[idx+1, 'inverted_hammer'] = 1
except:
    pass

if multi:
    q.put({'inverted_hammer': np.array(data['inverted_hammer'])})
else:
```

```
        return data
```

3.2.9 Bullish harami

The candlestick chart indication known as a bullish harami suggests that a downward trend may be coming to an end. Some investors can see a bullish harami as a cue to start long bets on particular assets.Candlestick charts get their name from its rectangular design with lines extending from the top and bottom, which resembles a candle with wicks. They are employed in the analysis of stock performance. A candlestick chart is used to show the opening, closing, high, and low prices of a stock on a certain day. To spot bear trend reversals, candlestick charts with a bullish harami indicator are utilised. .

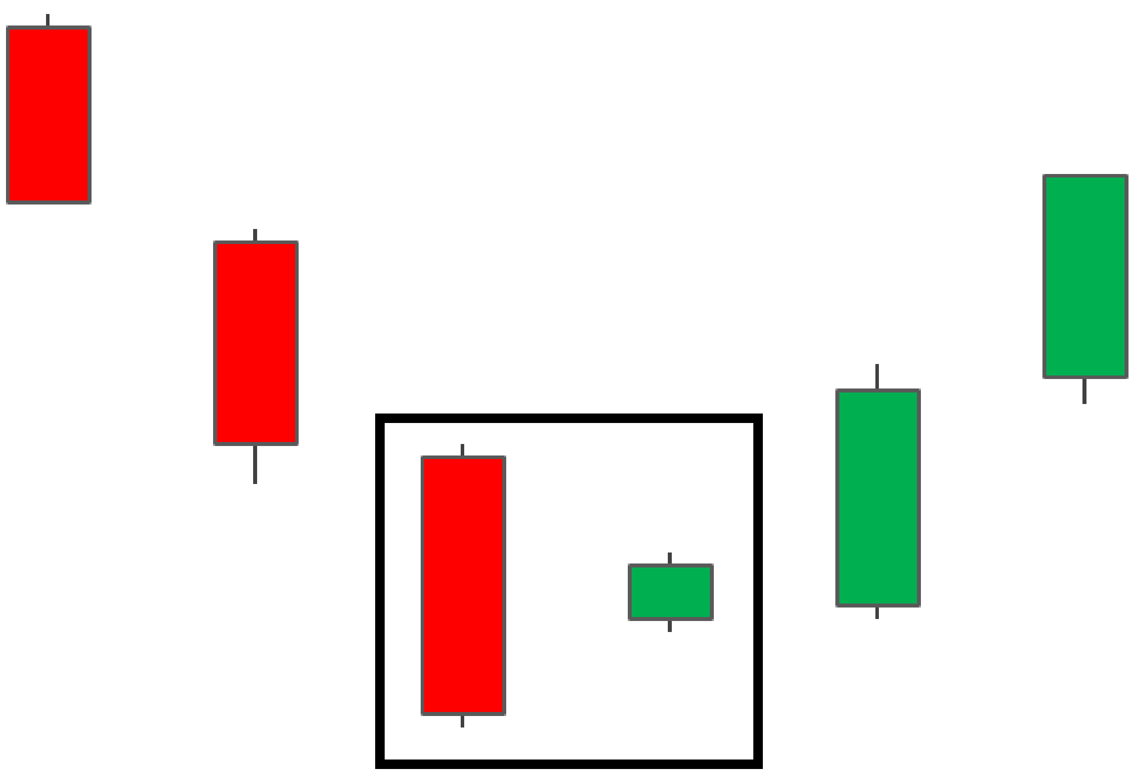

Figure 3.2.9: Bullish harami

```
def detect_bullish_harami(data, q=None, multi=False, short_per=35, l
    '''Detect inverted bullish harami pattern
    Args:
        short_per (int): percentile for determination.
        long_per (int): percentile for determination.

    Returns:
        dataframe.
    '''
    print('[ Info ] : detecting bullish harami')
    data['bullish_harami'] = 0
    temp = data[(data['previous_trend'] == -1) & (data['direction'] == -
    try:
        for idx in tqdm(temp):
            cond1 = (data.loc[idx, 'direction'] == -1)
            cond2 = (data.loc[idx, 'body_per'] >= long_per)
            cond3 = (data.loc[idx+1, 'direction'] == 1)
            cond4 = (data.loc[idx+1, 'close'] >= ((data.loc[idx, 'open']
            cond5 = (data.loc[idx+1, 'close'] < data.loc[idx, 'open'])
            cond6 = (data.loc[idx+1, 'open'] > data.loc[idx, 'close'])
            cond7 = (data.loc[idx+1, 'open'] <= ((data.loc[idx, 'open']
            cond8 = (data.loc[idx+1, 'body_per'] >= long_per)
            if cond1 & cond2 & cond3 & cond4 & cond5 & cond6 & cond7 & c
                data.loc[idx+1, 'bullish_harami'] = 1
    except:
        pass

    if multi:
        q.put({'bullish_harami': np.array(data['bullish_harami'])})
    else:
        return data
```

3.2.10 Bearish harami

The size of the second candle establishes the pattern's strength; the smaller it is, the more probable a reversal is to take place. The opposite of a bearish harami is a bullish harami, which is preceded by a downtrend and suggests that prices may reverse to the upside. Traders typically combine a bearish harami with other technical indicators to increase its strength as a trading tip. For example, a trader may use the 200-day moving average to determine that the market is in a long-term drop and start a short position when a bearish harami appears during a retracement.

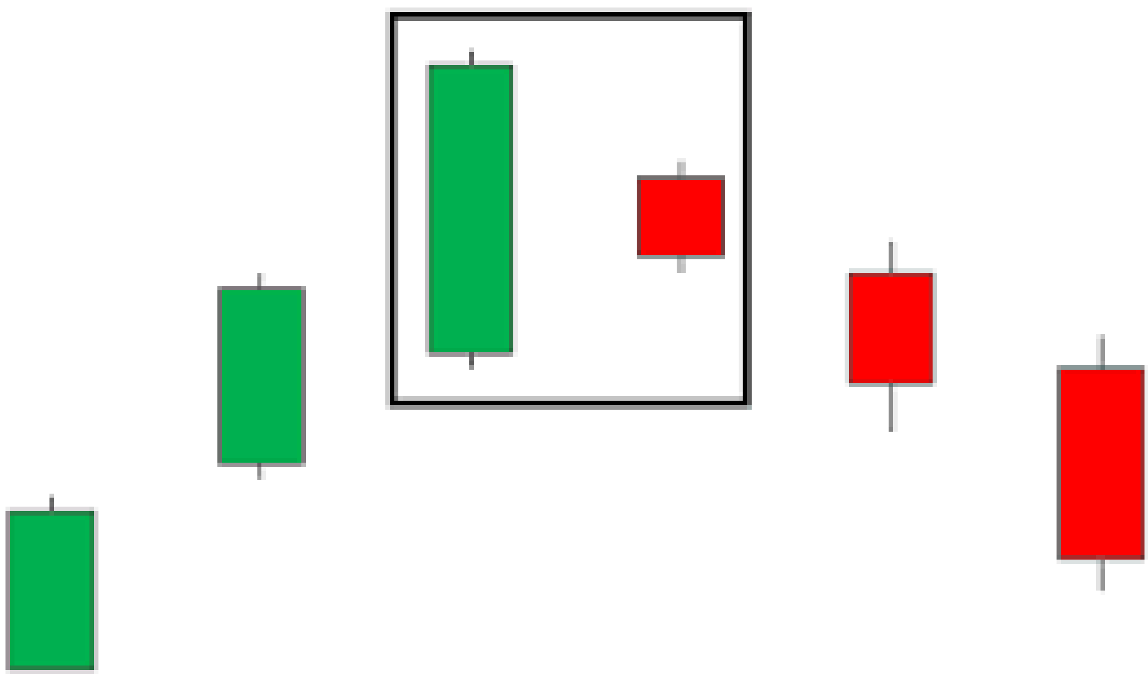

Figure 3.2.10: Bearish harami

```
def detect_bearish_harami(data, q=None, multi=False, short_per=35, l
'''Detect inverted bearish harami pattern
Args:
    short_per (int): percentile for determination.
```

```
        long_per (int): percentile for determination.

    Returns:
        dataframe.
    '''
    print('[ Info ] : detecting bearish harami')
    data['bearish_harami'] = 0
    temp = data[(data['previous_trend'] == 1) & (data['direction'] == 1)
    try:
        for idx in tqdm(temp):
            cond1 = (data.loc[idx, 'direction'] == 1)
            cond2 = (data.loc[idx, 'body_per'] >= long_per)
            cond3 = (data.loc[idx+1, 'direction'] == -1)
            cond4 = (data.loc[idx+1, 'close'] <= ((data.loc[idx, 'open']
            cond5 = (data.loc[idx+1, 'close'] > data.loc[idx, 'open'])
            cond6 = (data.loc[idx+1, 'open'] < data.loc[idx, 'close'])
            cond7 = (data.loc[idx+1, 'open'] >= ((data.loc[idx, 'open']
            cond8 = (data.loc[idx+1, 'body_per'] >= long_per)
            if cond1 & cond2 & cond3 & cond4 & cond5 & cond6 & cond7 & c
                data.loc[idx+1, 'bearish_harami'] = 1
    except:
        pass

    if multi:
        q.put({'bearish_harami': np.array(data['bearish_harami'])})
    else:
        return data
```

3.2.11 Feature selection

A closing or open price alone cannot be used to analyse a candlestick pattern. It is necessary to know the close, high, open, and low prices. Utilizing the close, high, open, and low prices is therefore intuitively sensible. In addition, we may employ the fair qualities of the actual body, lower shadow, and higher shadow. Prices at the close, high, open, and low as well as the upper shadow, close, lower shadow, and real body are used to determine the features of the Hanging Man and Bearish Engulfing patterns in Figures 4.2.1 and 4.2.2 below. The fundamental cause is that these four GASF matrices are significantly similar to one another given how little variance there is between the close, high, open, and low prices all at once. If there is an excessive amount of repetitive input, convolutional models will struggle to learn the key attributes. The near, higher, lower, and authentic body attributes will be used to process the data. It is clear from this change how fundamentally dissimilar the four characteristics are.This strategy is more natural and mimics traders' observation from a different perspective.

As a result, we will use simulation data to organise our testing, which will cover (1) close, high, open, low, and (2) upper shadow, close, lower shadow, true body features. The more effective one will then be used with actual data.

3.2.12 Converting numerical features to Images

Markov Transition Field

A diagram known as a Markov transition field serves as the representation for the field of transition probabilities in a discretized time series. A number of methods may be used to bin time series. The MTF comprises temporal information and accurately depicts the multispan transition probabilities of the time series on the basis of the Markov transfer matrix. A state transition for a single time stamp as well as state transitions over several time bins owing to changes in the MTF's component components may be defined by such an extension.

$$x' = \frac{x - x_{mean}}{x_{max} - x_{min}}, \text{ x} \in (x_1, x_2, x_3 ... x_n), \text{ x} \in (x_1, x_2, x_3 ... x_n), \text{ x} \in (x_1, x_2, x_3 ... x_n), \text{ x} \in (x_1, x_2, x_3 ... x_n)$$

$$w_{i,j} = \sum \forall x \in qi, y \in q_j, x + 1 = y1 / \sum j = 1^Q wi, j \tag{2}$$

Gramian angular field and Image classification model

Every pair of values in a time series has some type of temporal link, which is represented by a time series representation known as a Gramian angular field. GADF and GASF are the two methods.Take into consideration a time series that has been rescaled and has the form (x 1, x 2,...x n). After transforming the angular viewpoint into the polar coordinate system, we can simply use it. , i.e. $\theta_i = arccos(x_i)$,by calculating the trigonometrical sum/difference between each point to establish the chronological relationship between distinct time periods: The i-j element of the n*n GAF matrix is defined as the sine or cosine of the sum of the i-th and j-th angles. As seen in the following graphic, a time series in cartesian coordinates has been converted into a GAF image utilising an intermediate polar coordinate system step. Following this, we intend to create a CNN-based picture classification model. We're going to use the resnet Neogenic CNN model that was suggested in 3. The investigation is still ongoing.

3.2.13 Why GAF transformation is needed?

The Gramian Angular Field (GAF) has the following advantages:

1. The GAF provides a way to preserve temporal dependency since time increases as the position moves from top-left to bottom-right.

2. The GAF contains temporal correlations because the Gramian Angular represents the relative correlation by superposition and difference of directions for the time interval.

3. The primary diagonal of the Gramian Angular Field matrix is the particular case.

4. The diagonal of the Gramian Angular Field matrix contains the original value and angular information.

5. From the main diagonal, we can reconstruct the time series from the high-level features learned by the deep neural network.

3.3 Model architecture and training

3.3.1 GASF-CNN model

In order to simulate a large quantity of data inside the parameters of the simulation data, we will use the GBM model, one of the financial classic models. For training and testing, there will be 2000 and 500 data points, respectively. In order to increase the model's robustness, we employed three times as much simulation data for the other categories with the label 0. after data generation, We will calculate the open, high, low, and closing prices at each time point using 20 data points per bar.Then, using 10 as our window size, the data will be normalised with a variety of characteristics, such as (1) close, high, open, low, or (2) upper shadow, close, lower shadow, and actual body, to [0, 1]. . Using the cosine function of the angles, a GASF matrix may be created for each set of normalised data. Finally, we will train these matrices using our simple modified convolutional model.Once the data has been divided into (1) , high,close, low, open or (2) upper shadow,close, lower shadow, and actual body, the four GASF feature matrices will pass through four channels to our convolutional model. This simple model consists of just two convolutional layers and one fully connected layer. .

3.3.2 GASF-model architecture

The GASF model is a convolutional neural network model. The model consists of total of 6 blocks that includes two dimensional convolution layers, pooling layers and fully connected dense layers. The architecture of the model is shown in Figure 3.5.1. The convolution dense layers is made up of 16 kernels each with sigmoid function as the activation function. The base paper performed a comparison of the result using the pooling layer and without using them. The results showed that the model gave better accuracy by not using pooling layers. So in this paper I have ignored the pooling layers while building the model. The fully connected dense layer consists of 128 kernels and softmax function is used as activation function to classify the values to multiple classes.

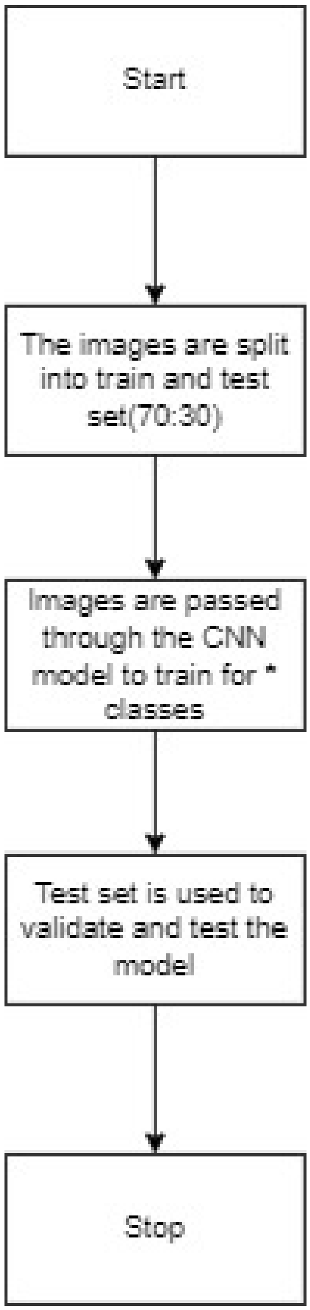

Figure 3.3.1: Flow chart for training and testing model

Layer (type)	Output Shape	Par
conv2d_6 (Conv2D)	(None, 10, 10, 16)	272
max_pooling2d_4 (MaxPooling2D)	(None, 5, 5, 16)	0
dropout_8 (Dropout)	(None, 5, 5, 16)	0
conv2d_7 (Conv2D)	(None, 5, 5, 16)	1040
max_pooling2d_5 (MaxPooling2D)	(None, 2, 2, 16)	0
dropout_9 (Dropout)	(None, 2, 2, 16)	0
conv2d_8 (Conv2D)	(None, 2, 2, 16)	1040
dropout_10 (Dropout)	(None, 2, 2, 16)	0
flatten_2 (Flatten)	(None, 64)	0
dense_4 (Dense)	(None, 128)	8320
dropout_11 (Dropout)	(None, 128)	0
dense_5 (Dense)	(None, 9)	1161
activation_2 (Activation)	(None, 9)	0

Figure 3.3.2: GASF-model-1 architecture for pattern recognition

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 10, 10, 16)	272
activation (Activation)	(None, 10, 10, 16)	0
conv2d_1 (Conv2D)	(None, 10, 10, 16)	1040
activation_1 (Activation)	(None, 10, 10, 16)	0
flatten (Flatten)	(None, 1600)	0
dense (Dense)	(None, 128)	204928
dense_1 (Dense)	(None, 9)	1161
activation_2 (Activation)	(None, 9)	0

Figure 3.3.3: GASF-model-2 architecture for pattern recognition

CHAPTER 4

RESULTS AND ANALYSIS

.

4.1 Dataset creation results

1. Created eight functions to recognize the 8-trigram patterns (evening star,morning star,
   shooting star, hanging man, bullish engulfing, bearish engulfing, hammer, inverted hammer bullish harami,bearish harami).

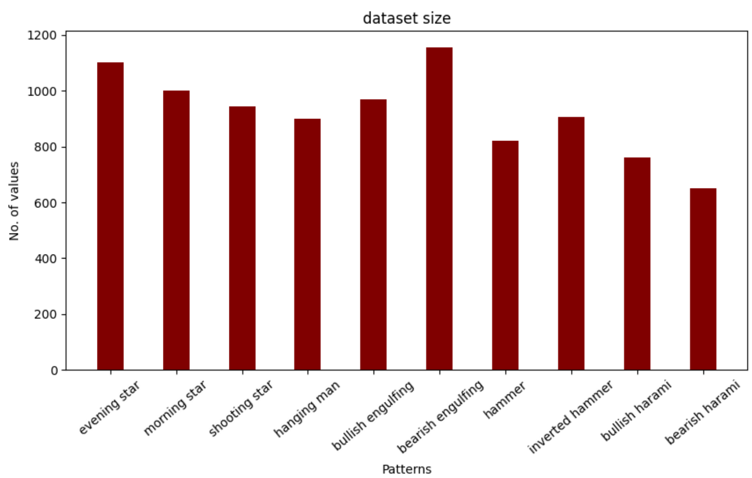

Figure 4.1.1: Distribution of all detected patterns that are present in the dataset created.

2. Used multiple high varying stocks such as TATA steel, L T infotech, CL etc to create a dataset containing data of minimum 1000 time series blocks for each pattern.
3. The batch size taken was varied accordingly to get the best results.

4.2 GAF transformation

The time series data created using GBM and other stocks was passed on to the GAF function for transforming it into images.

There are two types of conversions:

1. Conversion into images using open,high,low and close features.

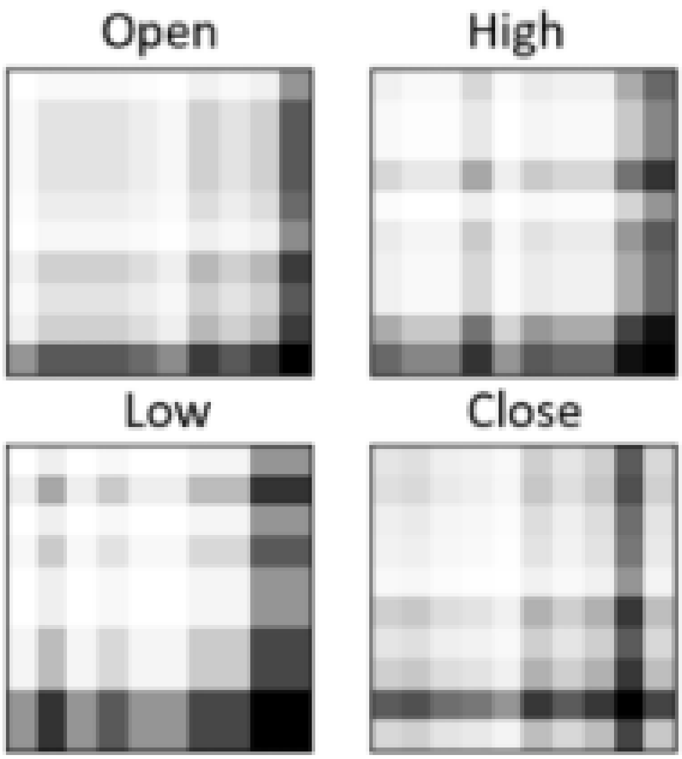

Figure 4.2.1: GAF transformation images using open,close,high and low features.

2. Conversion into images using close, upper, lower and realbody.

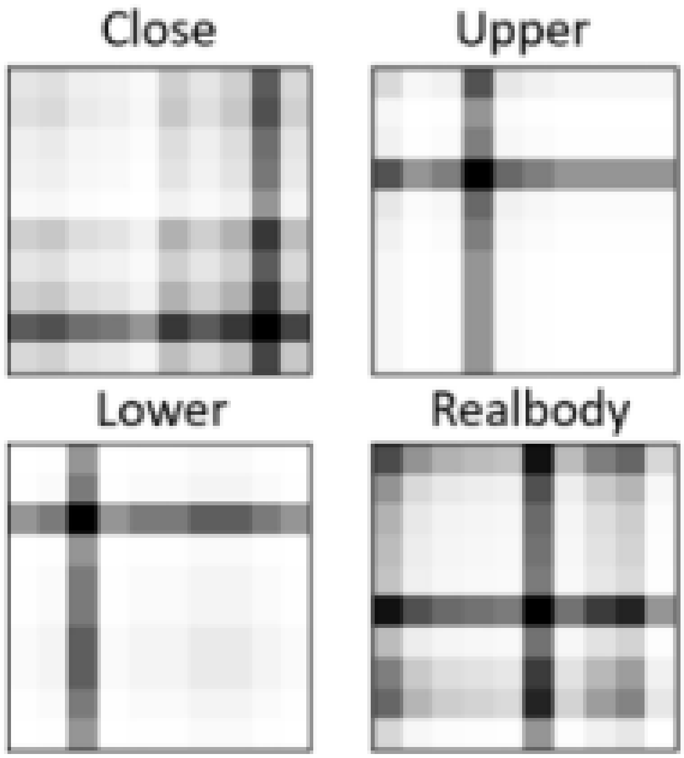

Figure 4.2.2: GAF transformation images using close,upper,lower and RealBody features.

4.3 Model results and analysis

We use meticulously updated TATA and L T infotech data from 2022 in our actual data system. Since there is less data in the actual data framework than there is in the simulated data, the labelling criteria have been loosened. In the first nine months of 2022, we used a training set of 1000 data points from each class. The most recent three months, with 350 data points for each class, will then comprise the testing set. As a consequence, we used three time data in the training set for class 0, which is the noisy data for the other classes. This enhances the model's robustness and capability to recognise patterns clearly.

We have trained the data on two types of model architectures one stated by the base paper [3] and another by rooming the Maxpooling and the dropout layers. The maxpooling and drop out layers are used to reduce the dimensionality of the data.

During this process important data might be ommited as all data is important in a time series to detect the patterns. As we can see there is an increase of around 1-2 percent in accuracy for our current dataset. This proves that removing the pooling layers is more effective for our application.

The classes specified in the below figures are as follows:

1. Evening star.
2. Morning star
3. Shooting star
4. Hanging man
5. Bullish engulf
6. Bearish engulf
7. Hammer
8. Inverted Hammer

0 being non of the above.

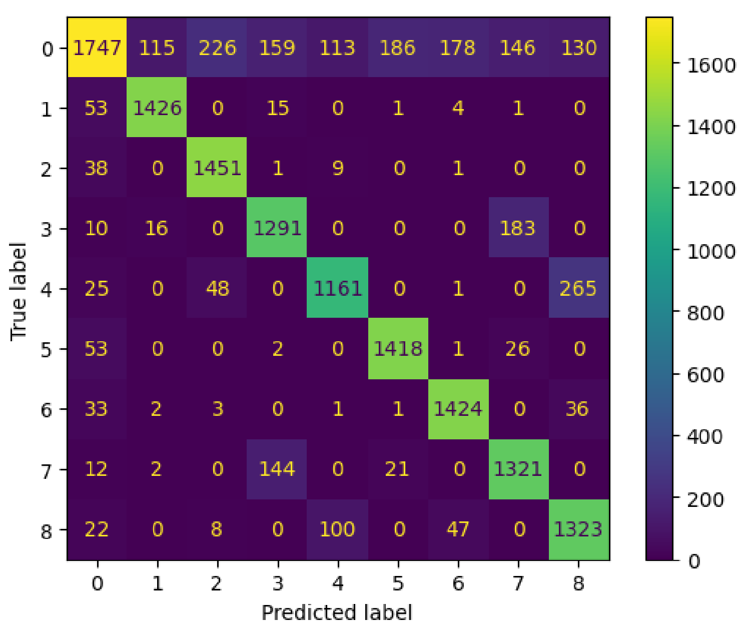

Figure 4.3.1: Confusion matrix results for the model-1 for the training set

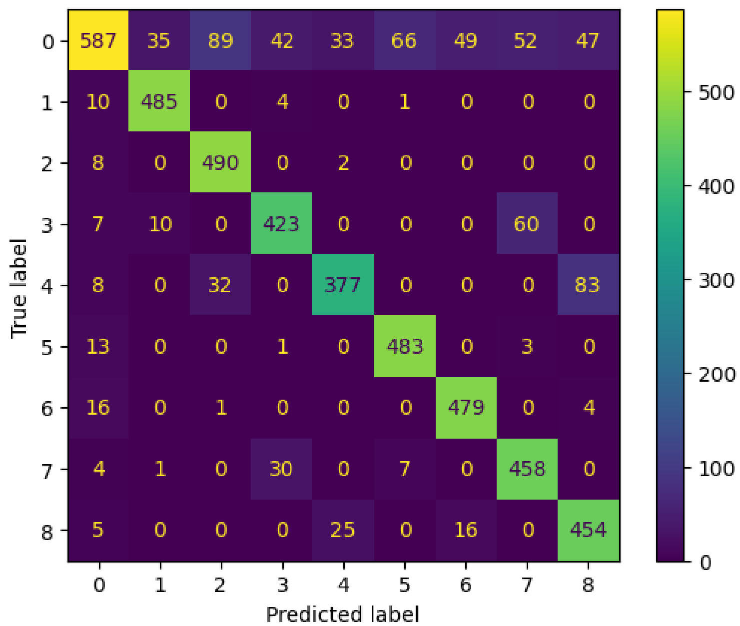

Figure 4.3.2: Confusion matrix results for the model-1 for the test set

	precision	recall	f1-score	support
0.0	0.877	0.582	0.700	3000
1.0	0.914	0.951	0.932	1500
2.0	0.836	0.967	0.897	1500
3.0	0.801	0.861	0.830	1500
4.0	0.839	0.774	0.805	1500
5.0	0.872	0.945	0.907	1500
6.0	0.860	0.949	0.902	1500
7.0	0.788	0.881	0.832	1500
8.0	0.754	0.882	0.813	1500
accuracy			0.837	15000
macro avg	0.838	0.866	0.846	15000
weighted avg	0.842	0.837	0.832	15000

Figure 4.3.3: Evaluation metrics of model-1 for training set

	precision	recall	f1-score	support
0.0	0.892	0.587	0.708	1000
1.0	0.913	0.970	0.941	500
2.0	0.801	0.980	0.881	500
3.0	0.846	0.846	0.846	500
4.0	0.863	0.754	0.805	500
5.0	0.867	0.966	0.914	500
6.0	0.881	0.958	0.918	500
7.0	0.799	0.916	0.854	500
8.0	0.772	0.908	0.835	500
accuracy			0.847	5000
macro avg	0.848	0.876	0.856	5000
weighted avg	0.853	0.847	0.841	5000

Figure 4.3.4: Evaluation metrics of model-1 for test set

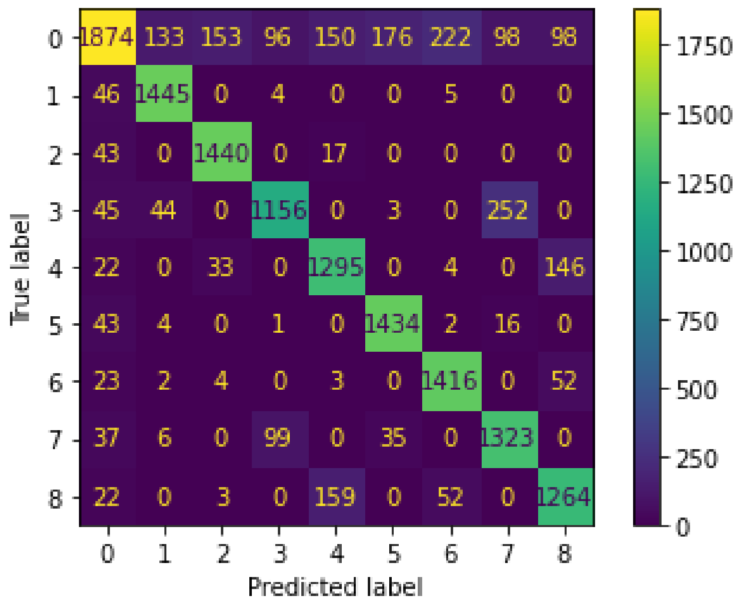

Figure 4.3.5: Confusion matrix results for the model-2 for the training set

	precision	recall	f1-score	support
0.0	0.870	0.625	0.727	3000
1.0	0.884	0.963	0.922	1500
2.0	0.882	0.960	0.919	1500
3.0	0.853	0.771	0.810	1500
4.0	0.797	0.863	0.829	1500
5.0	0.870	0.956	0.911	1500
6.0	0.832	0.944	0.885	1500
7.0	0.783	0.882	0.830	1500
8.0	0.810	0.843	0.826	1500
accuracy			0.843	15000
macro avg	0.842	0.867	0.851	15000
weighted avg	0.845	0.843	0.839	15000

Figure 4.3.6: Evaluation metrics of model-2 for training set

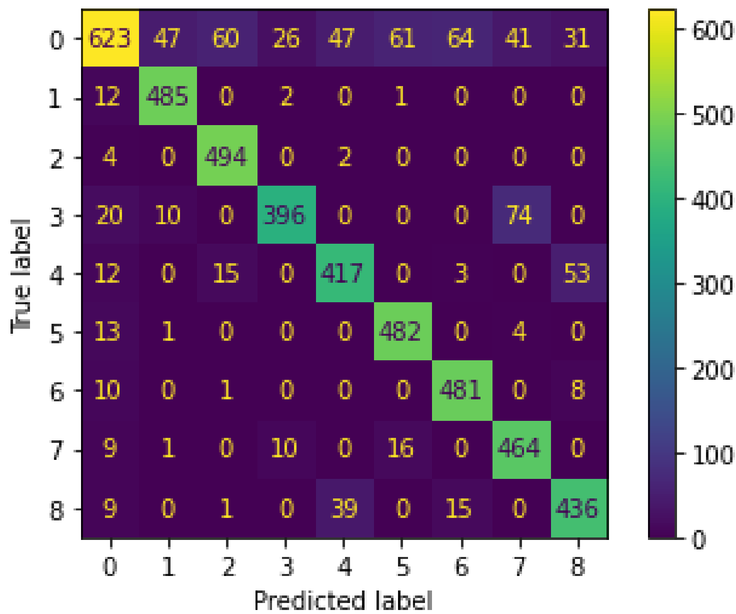

Figure 4.3.7: Confusion matrix results for the model-2 for the test set

	precision	recall	f1-score	support
0.0	0.875	0.623	0.728	1000
1.0	0.892	0.970	0.929	500
2.0	0.865	0.988	0.923	500
3.0	0.912	0.792	0.848	500
4.0	0.826	0.834	0.830	500
5.0	0.861	0.964	0.909	500
6.0	0.854	0.962	0.905	500
7.0	0.796	0.928	0.857	500
8.0	0.826	0.872	0.848	500
accuracy			0.856	5000
macro avg	0.856	0.881	0.864	5000
weighted avg	0.858	0.856	0.850	5000

Figure 4.3.8: Evaluation metrics of model-2 for test set

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

In this research, we created a pattern identification model based on convolutional neural networks that categorises 8 distinct candlestick patterns. Using simulation data generated by the GBM (Granular Brownion Motion) model, which aids in modelling stock data for research and sampling reasons, we develop our own dataset. Using data from TATA and L T Infotech, a test set is produced. The dataset's allocated candlestick data are transformed into 10x10 2D pictures before being given into the model for training. The GASF-CNN model is the one we suggest here, and it provided us with an average accuracy of about 84 percent for the train set and 85 percent for the test set, which is a respectable accuracy for application.

Future work we plan on creating a new model called MTF-CNN , that is using MTF instead of GAF for the purpose of image conversion and check the results. Also, plan to add more classes into the multi class classification.

REFERENCES

1. Y. Lin, S. Liu, H. Yang and H. Wu, "Stock Trend Prediction Using Candlestick Charting and Ensemble Machine Learning Techniques With a Novelty Feature Engineering Scheme," in IEEE Access, vol. 9, pp. 101433-101446, 2021, doi: 10.1109/ACCESS.2021.3096825.

2. Tsai, Yun-Cheng Chen, Jun-Hao Wang, Chun-Chieh. (2019). Encoding Candlesticks as Images for Patterns Classification Using Convolutional Neural Networks.

3. Kaya, Celal Yılmaz, Alperen Uzun, Gizem Kilimci, Zeynep. (2020). Stock Pattern Classification from Charts using Deep Learning Algorithms. Academic Perspective Procedia. 3. 445-454. 10.33793/acperpro.03.01.89.

4. X. Guo, X. Liang and X. Li, "A Stock Pattern Recognition Algorithm Based on Neural Networks," Third International Conference on Natural Computation (ICNC 2007), 2007, pp. 518-522, doi: 10.1109/ICNC.2007.145.

5. C. Wu, "Empowering Financial Technical Analysis using Computer Vision Techniques," 2021 2nd International Conference on Computer Science and Management Technology (ICCSMT), 2021, pp. 179-184, doi: 10.1109/ICCSMT54525.2021.00043.

6. K. -H. Ho, T. -T. Chan, H. Pan and C. Li, "Do Candlestick Patterns Work in Cryptocurrency Trading?," 2021 IEEE International Conference on Big Data (Big Data), 2021, pp. 4566-4569, doi: 10.1109/BigData52589.2021.9671826.

7. Lin Y, Liu S, Yang H, Wu H, Jiang B (2021) Improving stock trading decisions based on pattern recognition using machine learning technology. PLoS ONE 16(8): e0255558. https://doi.org/10.1371/journal.pone.0255558;Available at:https://journals.plos.org/plosone/article?

8. K. Jearanaitanakij and B. Passaya, "Predicting Short Trend of Stocks by Using Convolutional Neural Network and Candlestick Patterns," 2019 4th International Conference on Information Technology (InCIT), 2019, pp. 159-162, doi: 10.1109/INCIT.2019.8912115.Available at:https://ieeexplore.ieee.org/document/8912115 17

9. Kusuma, Rosdyana Ho Thi, Trang Kao, Wei-Chun Ou, Yu Yen Hua, Kai-Lung. (2019). Using Deep Learning Neural Networks and Candlestick Chart Representation to Predict Stock Market.Available at:https://arxiv.org/abs/1903.12258

10. Hung C-C, Chen Y-J (2021) DPP: Deep predictor for price movement from candlestick charts. PLoS ONE 16(6): e0252404. https://doi.org/10.1371/journal.pone.0252404

5.1 Bio Data

Student-Name:Sohanraj R

Roll no.:191IT149

Email id:sohanrajgowda@gmail.com

Address:No.23, 8th cross, Shardhamba nagar ,Jalahalli village, Bangalore 560013