Trading via Image Classification

Shashwat Mishra, Sacha Apelbaum, Rashi Mohta, Samy Kobbite

October 2024

Contents

1 Introduction

The intersection of quantitative finance and computer vision presents a novel frontier for exploring trading strategies. This study implements and extends the work of Cohen et al. (2020) in their paper "Trading via Image Classification", which proposes a paradigm shift in approaching financial time series analysis. The central question driving this research is: *Can we construct a system that identifies and replicates human trading behavior through visual pattern recognition?*

Traditionally, algorithmic trading strategies have relied on processing time-series data as numerical lists, seeking to detect patterns such as trends, cycles, and correlations. Upon identifying such patterns, analysts typically construct algorithms to leverage these insights for predictive modeling. However, this approach may not fully capture the intuitive visual analysis performed by experienced human traders.

The focus of this implementation is on representing financial time-series data as images and examining the efficacy of using these visual representations alone for identifying trading opportunities typical of technical analysis.

2 Motivation

- Bridging Disparate Domains: This research aims to bridge the gap between the fields of quantitative finance and computer vision. By leveraging the advanced CNN models in the context of financial data analysis, we explore potential synergies that may lead to novel insights and methodologies.
- Mimicking Human Decision-Making: In practice, there are financial domains in which investment decisions are made using visual representations alone (e.g., swap trade) Therefore, it's reasonable to examine the usefulness of visual representations as input to the model.
- Real-World Applicability: While a significant portion of artificial intelligence research remains conceptual or limited to synthetic datasets, this study aims to verify if the concept of visual time-series classification is effective and applicable to real-world financial data.

By implementing this innovative approach to financial time series analysis, the paper explores whether computer vision techniques can effectively capture and exploit the visual patterns that human traders intuitively recognize.

3 Implementation Details

Data Acquisition and Signal Generation

We utilized daily Open, High, Low, and Close (OHLC) data for all S&P 500 constituent companies from 2010 to 2018, sourced from Yahoo Finance. Following the paper's methodology, we implemented three technical indicators. These indicators were chosen for their widespread use in technical analysis and their varying complexity, allowing for a comprehensive evaluation of the image classification approach.

- Bollinger Bands (BB)
  - 20-day simple moving average with upper and lower bands at two standard deviations
  - Buy signal: price crosses above the lower band
- Moving Average Convergence Divergence (MACD)

- MACD line: difference between 12-day and 26-day exponential moving averages
- Signal line: 9-day EMA of the MACD line
- Buy signal: MACD line crosses above the signal line

- Relative Strength Index (RSI)
  - 14-day RSI calculation
  - Buy signal: RSI crosses above 30 (oversold threshold)

We applied these indicators to each stock's time series, generating binary labels (1 for buy, 0 otherwise) for each trading day.

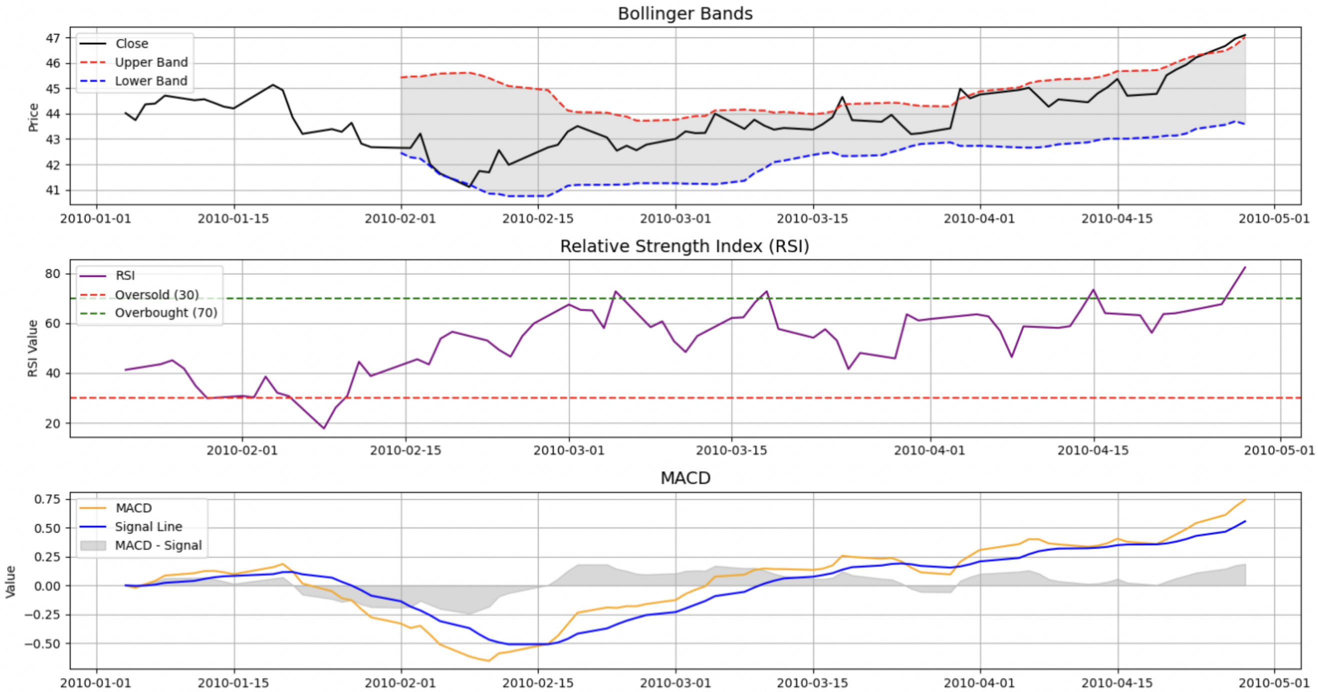

Figure 1: Visualisation of the 3 trading signals

Image Generation

We generated images by enveloping a window of stock activity (the red rectangles) before and including the buy-signal day activity. We also created negatively-labeled images from this time-series by enveloping activity, in the same way, for days with no buy signal.

We used a discrete form of the continuous data by accounting only for the start, max, min, and end values per stock per day. The color of each box reveals whether the open price finalized higher or lower than the close price for the same day.

Images were generated for each stock using specific window sizes - BB: 20-day, MACD: 26-day, RSI: 27-day.

All images were resized to 30x30 pixels, as the paper found this resolution to be optimal for classification performance.

Figure 2: Continuous Time Series Data to Candlesticks (from paper)

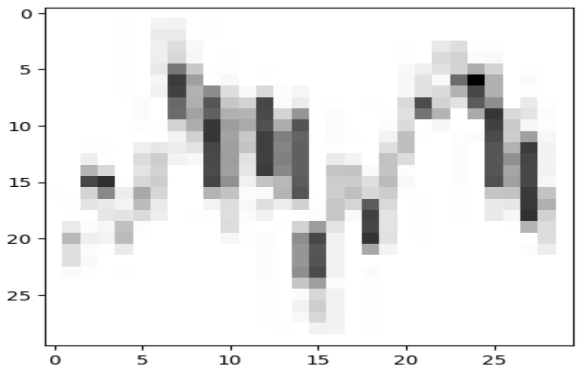

Figure 3: 30 x 30 pixel resolution in our code

Dataset Creation

For each indicator:

- Randomly selected 10 buy signals and 10 non-buy signals per stock
- Created corresponding images for each selected signal
- Compiled a balanced dataset of 5,000 samples per class (10,000 total images)

Model Training and Evaluation

We implemented and trained multiple classifiers as specified in the paper:

- Traditional algorithms: Logistic Regression, Gaussian Naive Bayes, Linear Discriminant Analysis, Quadratic Discriminant Analysis, Gaussian Process, K-Nearest Neighbors, Linear SVM, RBF SVM
- Ensemble methods: Decision Trees, Random Forest, Extra Randomized Forest, Ada Boost, Bagging, Gradient Boosting
- Neural Networks: Deep Neural Network (32x32x32 structure), Convolutional Neural Network (3 layers of 32 3x3 filters with ReLU activations and Max Pooling)

Extending beyond the paper's original scope, we conducted out-of-sample validation and testing. We train our models using the first 5 years data, use the next 3 years' data for validation and tested the models on the last 2 years. This additional step allowed us to evaluate the real-world applicability of the image classification approach to financial time series analysis.

4 Results

Our implementation achieved classification accuracies of approximately:

- 95% for BB and RSI signals
- 80% for MACD signals

These results are consistent with the original paper's findings. The lower accuracy for MACD signals is likely due to the increased complexity of this indicator, which involves multiple time scales and smoothing operations.

A comprehensive table detailing the accuracy and precision metrics for all indicators across different classifiers can be found in the Appendix. Notably, our analysis revealed that the performance differences between various classifiers were relatively small, suggesting that the image-based approach is robust across different machine learning algorithms.

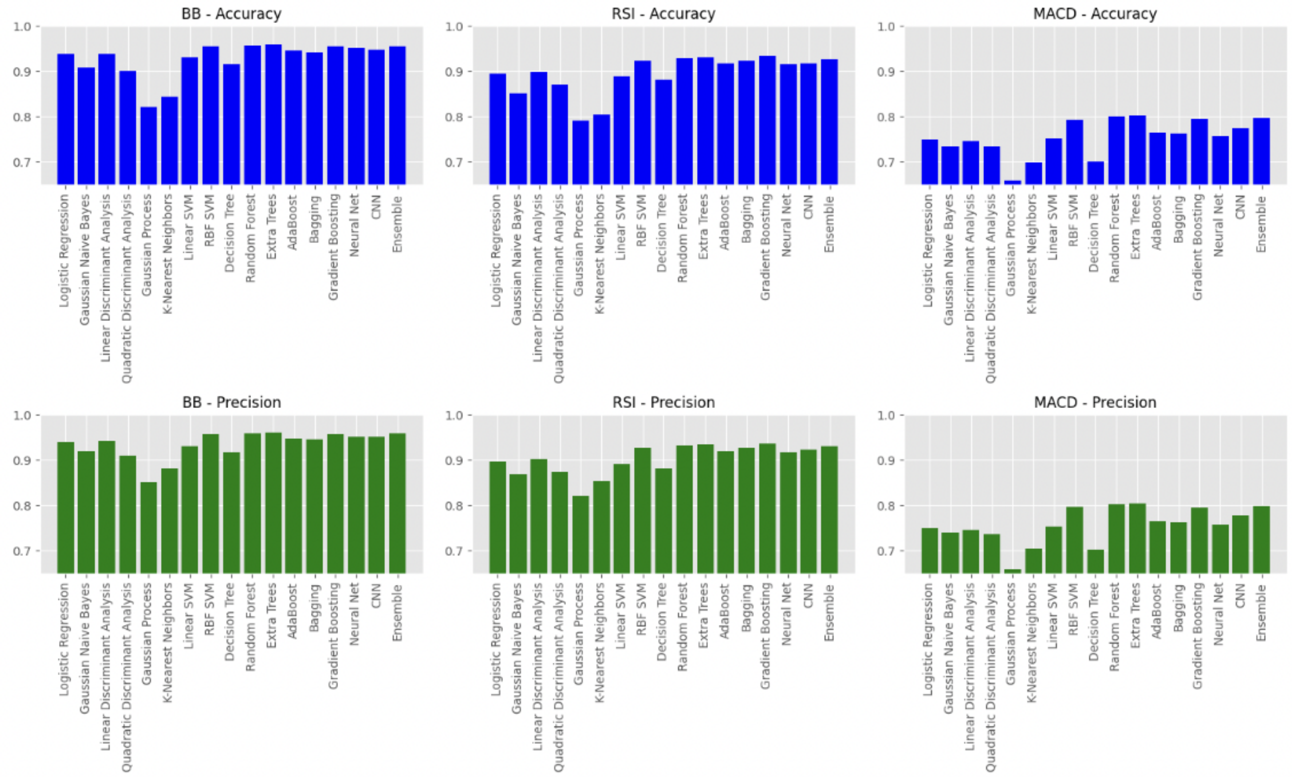

Figure 4: Classifier Comparison by Accuracy and Precision for BB, RSI and MACD Signals

5 Critical Evaluation of the paper

Strengths of the Approach

- Novel Approach:

  The paper introduces an innovative method of transforming time series analysis into image classification, effectively bridging the fields of quantitative finance and computer vision. This interdisciplinary approach opens new avenues for financial data analysis and potentially offers insights that might be overlooked in traditional numerical methods.

- Practical Inspiration

  The method draws inspiration from real-world trading practices, where traders often rely on visual chart analysis. By mimicking this human approach, the image-based method may offer more intuitive interpretations of trading signals compared to purely numerical approaches. This alignment with human cognition could potentially lead to more interpretable and trustworthy models.

- Comprehensive Testing

  The study demonstrates thoroughness in its evaluation, examining 16 different classifiers, various visual representations of financial data, and three distinct trading indicators (BB, RSI, MACD) for labeling. This comprehensive approach provides a robust foundation for assessing the viability of the image-based classification method.

Limitations and Areas for Improvement

- Lack of Theoretical Foundation

  While the approach is innovative, the paper does not provide a strong theoretical explanation for why image-based classification should be superior to traditional numerical approaches for algorithmic trading. The authors do not convincingly demonstrate why a computer would process visual representations more effectively than raw numerical data, which somewhat undermines the core premise of the approach.

- Computational Inefficiency

  The process of creating images from numerical data and then applying deep learning techniques to these images is inherently more time-consuming and computationally expensive than direct numerical analysis. This inefficiency is not adequately addressed or justified in the paper, raising questions about the method's practicality in real-world, high-frequency trading scenarios.

- Oversimplification of Trading Strategies

  The use of basic indicators like BB, RSI, and MACD as labeling strategies may not reflect the complexity of real-world trading decisions. Furthermore, the paper lacks a thorough comparison of the image-based approach's performance to state-of-the-art numerical time series analysis methods. This omission makes it difficult to assess the true value of the proposed method in the context of existing financial analysis techniques.

- Lack of Portfolio Performance Evaluation

  The paper does not extend the analysis to actual portfolio creation and performance evaluation, which limits its practical applicability. As it stands, the paper demonstrates a proof-of-concept rather than providing a novel trading strategy or alpha generation method. This leaves open questions about the real-world efficacy of the approach.

- Underutilization of CNN Potential

  The paper does not fully exploit the potential of Convolutional Neural Networks (CNNs) to extract complex or non-trivial signals from the image data. A more compelling justification for the image-based approach would have been to demonstrate that CNNs can identify intricate visual patterns in price charts that are difficult or impossible to define algebraically, potentially uncovering novel predictive signals. This limitation leaves the potential advantages of treating financial time series as images largely unexplored.

6 Our Proposed Strategy

Building upon the foundation laid by Cohen et al. and addressing some of the limitations identified in our critical evaluation, we developed our own trading strategy. Our approach aims to bridge the gap between the proof-of-concept demonstrated in the original paper and a more practical, portfolio-based implementation.

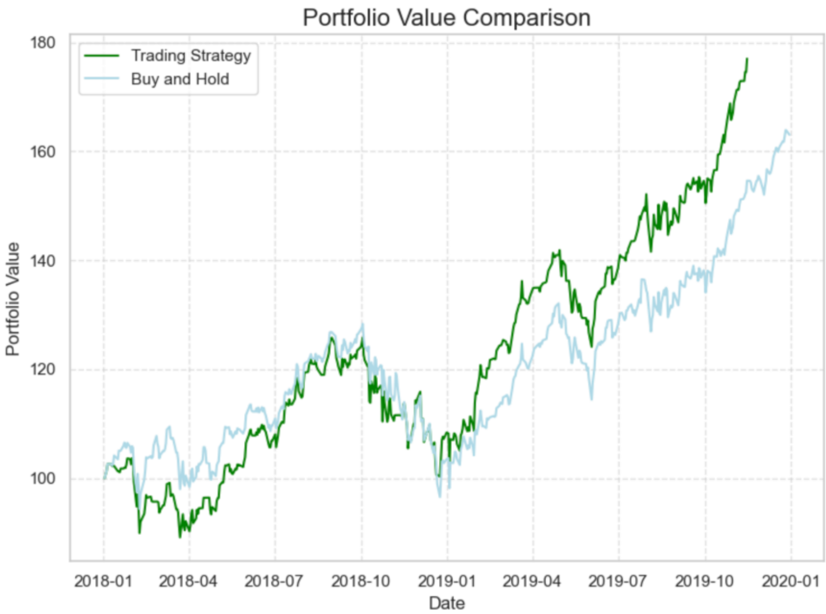

Figure 5: Cumulative returns of our trading strategy and a buy and hold strategy

Overview

Our strategy utilizes the image-based classification method but extends it to create and manage a simulated portfolio. We found that while the ensemble method (majority voting of all classifiers) performed well, it was not necessarily superior to simpler approaches. In fact, we consistently achieved comparable accuracy and precision using either the Extra Trees method or Gradient Boosting alone. Our algorithm addresses several limitations of the original paper:

- Portfolio Creation: We extend the image-based classification to actual portfolio management, allowing for a more realistic evaluation of the strategy's performance.
- Risk Management: By incorporating a risk-free asset and implementing a cooldown period after sell signals, we add basic risk management features.
- Signal Aggregation: Instead of relying solely on individual indicators, we aggregate buy signals over a 5-day window, potentially capturing more robust trading opportunities.
- Computational Efficiency: By using either Extra Trees or Gradient Boosting instead of an ensemble of all classifiers, we improve the computational efficiency without significantly sacrificing accuracy.

You can find the pseudocode of our trading strategy in the appendix.

Results

Our image-based trading strategy, inspired by Cohen et al., outperformed a simple buy-and-hold approach:

- Higher Returns: Achieved an annualized return of 39.26%, compared to 31.22% for buy-and-hold.
- Better Risk-Adjusted Performance: Sharpe ratio of 1.60 vs. 1.27 for buy-and-hold.
- Lower Drawdown: Maximum drawdown of -20.28%, compared to -24.77% for buy-and-hold.

Metric	Our Strategy	Buy and Hold
Annualized Return	39.26%	31.22%
Annualized Volatility	23.24%	22.96%
Sharpe Ratio	1.60	1.27
Max Drawdown	-20.28%	-24.77%
Sortino Ratio	1.98	1.66
Calamar Ratio	1.94	1.26

Table 1: Performance comparison of trading strategies

- Consistency: Higher Sortino ratio (1.98 vs. 1.66) and Calmar ratio (1.94 vs. 1.26), reflecting more stable returns.
- Effective Volatility Management: Similar volatility (23.24%) to buy-and-hold (22.96%).

The cumulative returns graph visually confirms these findings, showing that our trading strategy consistently outperformed the buy-and-hold approach over the testing period, with the outperformance becoming more pronounced towards the end of the period.

7 Final Remarks

While the concept of trading via image classification presents an interesting avenue for research, we are not yet fully convinced that it offers sufficient advantages to justify the additional complexity and computational resources required. The method's inability to consistently extract novel features that are not identifiable through traditional means suggests that simpler, more interpretable models might achieve similar results with greater efficiency.

Our attempt to implement an unsupervised learning approach, hoping to extract features not easily definable through algebraic means, did not yield satisfactory results. This leads us to question whether the model is effectively learning from the candlestick images alone, or if it's primarily capturing patterns that could be more efficiently identified through traditional time series analysis.

8 Appendix

Model	MACD		RSI		BB	
	Accuracy	**Precision**	**Accuracy**	**Precision**	**Accuracy**	**Precision**
Logistic Regression	0.7497	0.7499	0.8952	0.8960	0.9390	0.9399
Gaussian Naive Bayes	0.7354	0.7404	0.8508	0.8692	0.9077	0.9195
LDA	0.7462	0.7466	0.8980	0.9032	0.9393	0.9418
QDA	0.7349	0.7358	0.8717	0.8741	0.9024	0.9103
Gaussian Process	0.6597	0.6599	0.7908	0.8206	0.8224	0.8520
K-Nearest Neighbors	0.6990	0.7050	0.8044	0.8529	0.8450	0.8815
Linear SVM	0.7525	0.7529	0.8895	0.8907	0.9306	0.9317
RBF SVM	0.7936	0.7963	0.9228	0.9270	0.9560	0.9579
Decision Tree	0.7016	0.7016	0.8829	0.8841	0.9135	0.9151
Random Forest	0.8007	0.8017	0.9333	0.9366	0.9574	0.9591
Extra Trees	0.8031	0.8043	0.9306	0.9346	0.9583	0.9601
AdaBoost	0.7649	0.7649	0.9186	0.9197	0.9462	0.9471
Bagging	0.7716	0.7722	0.9218	0.9256	0.9458	0.9484
Gradient Boosting	0.7941	0.7949	0.9344	0.9369	0.9559	0.9575
Neural Net	0.7598	0.7598	0.9213	0.9221	0.9498	0.9504
CNN	0.6813	0.7202	0.8947	0.9078	0.9431	0.9438
Ensemble	0.7957	0.7975	0.9267	0.9318	0.9570	0.9596

Table 2: Performance of different models on MACD, RSI, and BB signals

Algorithm 1 Pseudocode for our strategy

```
procedure TradingStrategy(bb_signals, rsi_signals, macd_signals, prices, capital, risk_free_rate)
    Initialize: positions ← 0, portfolio_weights ← 0, days_since_sell ← −∞
    for each day do
        tradable_actions ← [], buy_signals_count ← 0
        for each action do
            if Cooldown period active then
                positions[action] ← 0
            else
                Generate Buy Signal: Count BB, RSI, MACD signals (last 5 days)
                if No Sell Signal then Append action to tradable_actions
                elseSet positions[action] ← 0, update cooldown
                end if

                if Buy signals detected then
                    Allocate capital proportionally to signal strength
                    risk_free_weight ← 1 − ∑ portfolio weights
                else
                    All capital assigned to risk-free
                end if
                Update Portfolio: Compute daily return, update portfolio_value

                return portfolio_weights, portfolio_value

```