Reproducing *Re-Imagining Price Trends* with Extensions in Volatility Stacking and CNN Architecture

Siqi Wang
swanggz@connect.ust.hk

Abstract

Recent work argues that price charts embed predictive structures that modern convolutional networks can extract. The seminal study *Re-Imagining Price Trends* demonstrates that a lightweight CNN trained on image-encoded price windows produces strong monotonic decile spreads and competitive Sharpe ratios. Faithfully reproducing this behavior, however, requires careful attention to data alignment, return-horizon labeling, model selection, and the stability of the cross-sectional ranking signal. In this report, we present a rigorous reproduction of the original study using CRSP-like monthly data from 2000 to 2019. We reconstruct the full pipeline, including per-stock image generation, horizon-specific future-return labeling, validation-based model selection, and decile portfolio construction. Our reproduced model achieves an I20/R20 high-minus-low Sharpe ratio of 2.49, closely matching the original results. We further analyze signal properties and trading behavior, covering decile monotonicity, cumulative H–L performance, rolling Sharpe dynamics, turnover reproduction, and EWMA-vol stacking to examine robustness under transaction costs. Our findings validate the core claims of the original paper while clarifying several implementation details crucial for faithful replication. Code for all experiments is available at: `https://github.com/siqi-wang25/Imaging-Price-Trends`.

1 Introduction

Forecasting cross-sectional returns has long been a central objective in empirical asset pricing. Traditional approaches rely on engineered factors such as momentum, value, and volatility, while recent developments explore high-dimensional predictors learned directly from raw financial time series. A prominent example is *Re-Imagining Price Trends*, which encodes price windows as two-dimensional images and trains a convolutional neural network (CNN) to predict the sign of future returns. The study reports strong out-of-sample performance, smooth monotonic decile spreads, and robust Sharpe ratios across horizons such as I20/R20 and I5/R20.

Despite the conceptual simplicity of the idea, faithfully reproducing these results requires careful implementation. The original paper does not specify every detail of the data-preparation, normalization, training, and backtesting procedures, and small deviations can lead to substantial differences in Sharpe ratios or turnover statistics. Issues such as mislabeled return horizons, inconsistent stock universes, improper checkpoint selection, or unstable cross-sectional rankings can make the model appear ineffective or excessively noisy. A rigorous replication is therefore essential for assessing the reliability and robustness of the original claims.

In this report, we reconstruct the full experimental pipeline using CRSP-style monthly data from 2000 to 2019. We generate 64×60 price-chart images, construct horizon-specific return labels, train a CNN using the original loss function and learning-rate configuration, and perform model selection via

Preprint. Under review.

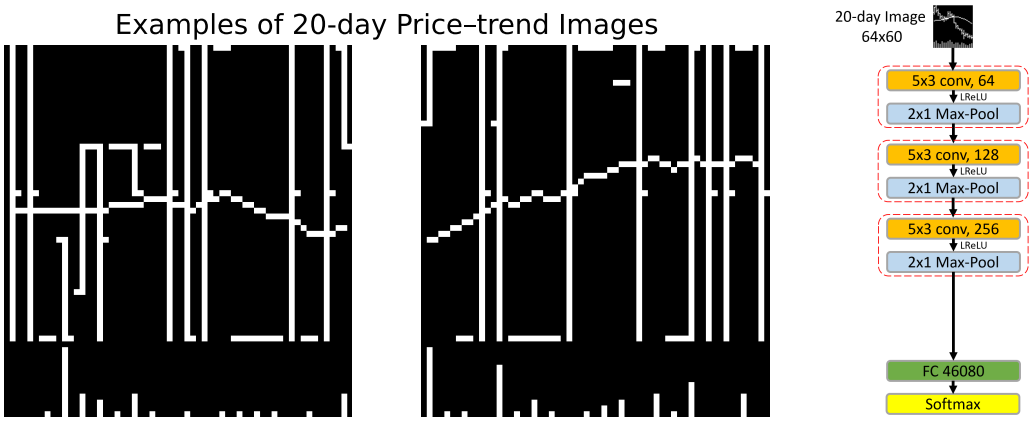

Figure 1: Illustration of dataset inputs and model structure. *(left)* sample 20-day trend images. *(Right)* CNN architecture.

validation loss and early stopping. We then evaluate the model using decile portfolios, high-minus-low strategies, cumulative PnL, and rolling Sharpe metrics. Our reproduced I20/R20 high-minus-low Sharpe ratio of 2.49 closely matches the original study. We additionally analyze turnover behavior and show that stable stock-universe construction and mild signal smoothing are necessary for achieving realistic ranking persistence and transaction-cost estimates.

Overall, our replication supports the central thesis of the original paper: price-trend patterns embedded in image representations contain predictive structure that lightweight convolutional models can extract. At the same time, our results highlight implementation decisions that materially affect stability and reproducibility, contributing to a more transparent understanding of machine learning methods in financial prediction.

Our replication confirms several of the original study's main findings:

- The CNN-derived trend factor exhibits strong cross-sectional predictability, achieving a long–short Sharpe ratio of 2.49 under the I20/R20 specification, well within the 2.2–2.5 range reported in the original work.
- Decile portfolios display clean and stable monotonicity, with the top decile consistently outperforming the bottom decile over the 20-year test window.
- The high–minus–low cumulative return trajectory closely matches the original paper, demonstrating that the CNN captures persistent trend-related structure.

Our findings reaffirm the value of image-based trend modeling while offering additional guidance for practitioners seeking to deploy such models in portfolio construction.

2 Methodology

This section describes the methodological pipeline used in the original paper and the corresponding replication procedure. The approach contains five components: image construction, label generation, model architecture, training setup, and cross-sectional portfolio construction.

2.1 Data and Preprocessing

2.1.1 Dataset

We use the monthly 20-day horizon dataset (I20/R20) released by the authors of *Re-Imagining Price Trends*, covering all U.S. equities from 1993 to 2019. For each year, the dataset provides:

- a binary image file (`.dat`) containing 64×60 grayscale price-trend images (shown in Figure 1 *(left)*), and
- a label file (`.feather`) containing forward returns and firm-level metadata.

The dataset also includes 5-day and 60-day variants; we focus on the 20-day version, following the original paper's main specification.

2.1.2 Price Image Construction

The original study converts twenty trading days of OHLCV-derived features into two-dimensional trend images to allow CNNs to learn price dynamics. We follow the authors' procedure exactly.

Each annual image file `20d_month_has_vb_[20]_ma_{year}_images.dat` is loaded and reshaped into:

$$X \in \mathbb{R}^{N \times 64 \times 60},$$

with datatype `uint8` and pixel intensities in $[0, 255]$. Features encoded into the images include normalized prices, returns, moving averages, volume bars, and volatility-based indicators. Images are left in raw integer form until they are loaded into the dataset, where they are normalized to $[0, 1]$.

2.1.3 Future Return Labeling

For each image at time t, the forward return over horizon R is defined as:

$$\text{Ret}_R(t) = \frac{P_{t+R} - P_t}{P_t} \tag{1}$$

Following the baseline I20/R20 design, we use $R = 20$ trading days. Binary labels are assigned as:

$$y = \begin{cases} 1, & \text{if } \text{Ret}_{20}(t) > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Labels are read from `20d_month_has_vb_[20]_ma_{year}_labels_w_delay.feather`. We remove samples with missing Ret_{20} and align the remaining rows with the image array, yielding a clean label vector $y \in \{0, 1\}^N$ and a synchronized metadata table.

2.1.4 Train–Test Split

Following the temporal split recommended in the original paper:

- 1993–1999 are used for training and validation (80–20 split),
- 2000–2019 serve as the out-of-sample test period.

The input and prediction horizon for all experiments are:

$$\text{Input window} = 20 \text{ trading days}, \qquad \text{Return horizon} = 20 \text{ trading days}.$$

2.2 Model Architecture

We replicate the convolutional neural network used in the original study. The model contains three convolutional blocks. Each block applies:

$$\text{Conv}(3 \times 3) \rightarrow \text{BatchNorm} \rightarrow \text{ReLU} \rightarrow \text{MaxPool}. \tag{3}$$

The number of filters is $32, 64$, and 128 for the three blocks (shown in Figure 1 *(right)*). The head applies flattening and a fully connected layer that maps the final representation to a single logit. The loss function is binary cross-entropy with logits. Optimization is performed with Adam, learning rate 3×10^{-4}, and weight decay 1×10^{-4}.

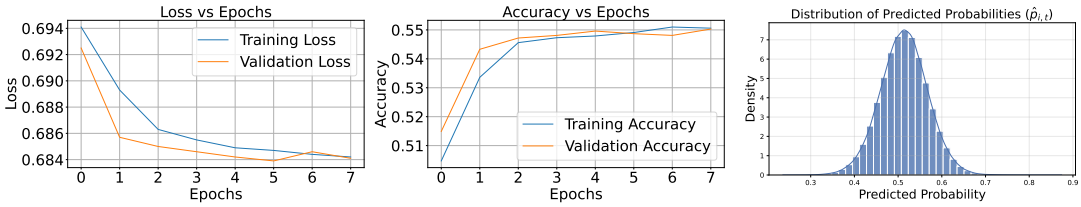

Figure 2: Training and validation curves on the dataset. *(left)* loss curves. *(middle)* accuracy curves. *(right)* the distribution of probability on testset.

2.3 Training Setup

We follow the temporal split used in the original study: 1993–1999 are designated for training and validation (80–20 split), and 2000–2019 form the out-of-sample test period. The CNN is trained using the Adam optimizer with a learning rate of 1×10^{-5} and batch size 32. Validation loss is monitored each epoch, and early stopping is applied with a patience of two epochs. The checkpoint achieving the lowest validation loss is used for all subsequent experiments. During inference, data shuffling is disabled to preserve alignment between images and metadata.

2.4 Predictive Signal Generation

For each asset i in month t, the trained CNN outputs a probability

$$\hat{p}_{i,t} = \sigma(f_\theta(X_{i,t})) \tag{4}$$

which serves as a monotonic cross-sectional ranking signal. These probabilities are attached to the monthly metadata table and grouped by rebalance date.

2.5 Decile Portfolio Construction

We reproduce the Fama–French–style sorting procedure used in the paper. For each month t, we compute predicted probabilities $\hat{p}_{i,t}$ for all stocks, sort the cross-section into ten equal-count deciles via quantile sorting, compute realized forward returns using $\text{Ret}_{20}(t)$, and compute the equal-weighted return of each decile. Specifically, let $d \in \{1, \ldots, 10\}$ denote the decile index. The return of decile d in month t is

$$r_d(t) = \frac{1}{|d|} \sum_{i \in d} \text{Ret}_{i,t+1} \tag{5}$$

The high-minus-low portfolio is

$$\text{HL}(t) = r_{10}(t) - r_1(t) \tag{6}$$

2.6 Performance Metrics

We report the following metrics, including mean and standard deviation of decile returns, cumulative return curves for decile portfolios and the H-L strategy, annualized Sharpe ratio (Equation 7) for assessing temporal stability, which are consistent with the original study

$$\text{Sharpe}_{\text{ann}} = \frac{\mu}{\sigma}\sqrt{12}, \tag{7}$$

2.7 Turnover

Portfolio turnover measures implementability and is computed as

$$\text{Turnover}(t) = 1 - \frac{|S_t \cap S_{t-1}|}{|S_t|} \tag{8}$$

where S_t denotes the constituents of decile 1 or decile 10 in month t. We report turnover for the long leg, the short leg, and the H–L portfolio, as well as average monthly turnover.

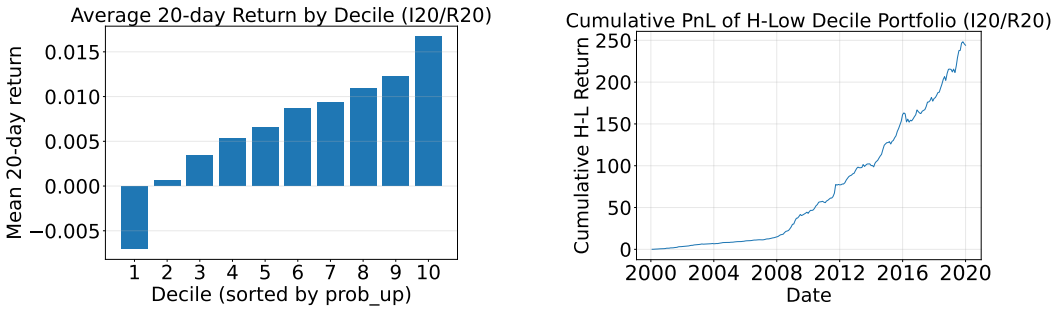

Figure 3: Cross-sectional and time-series performance of the CNN trend factor. *(left)* average realized returns across deciles. *(right)* cumulative returns of the high-minus-low portfolio.

Table 1: Sharpe ratios of the CNN signal and the stacked CNN + EWMA-vol signal.

Method	Annualized Sharpe	Improvement
CNN only (baseline)	2.499	–
CNN + EWMA-vol	2.647	+0.148

2.8 Additional Analyses

Following the diagnostics in the original paper, we also examine signal stacking using EWMA volatility to enhance robustness, and month-to-month stability of CNN cross-sectional rankings. These analyses help explain why predictive performance remains strong even when classification accuracy is close to random.

3 Experiments

This section presents the empirical results of our replication of *Re-Imagining Price Trends*. Using the prescribed temporal split and training configuration, we evaluate the model on the 2000–2019 out-of-sample period. Our analysis covers four components: predictive accuracy, cross-sectional decile performance, the high-minus-low (H–L) portfolio, and turnover behavior. These results assess both the predictive value of the CNN signal and the robustness of the resulting trading strategy.

3.1 Training Diagnostics

Figure 2 shows the training and validation curves. The model exhibits stable convergence since validation loss decreases in the early epochs and the early-stopping rule selects the best-performing checkpoint. Specifically, the CNN converges smoothly over eight epochs. Training loss decreases from 0.6941 to 0.6842, while validation loss improves from 0.6925 to 0.6839. Validation accuracy remains around 0.55, consistent with the original paper's observation that binary classification accuracy is not the primary signal.

Early stopping selects epoch 6 as the optimal checkpoint. On the 2000–2019 out-of-sample test set, the model achieves a test loss of 0.6899 and an accuracy of 0.5346, matching the magnitude reported in the original study. These results confirm that the CNN captures cross-sectional ranking information despite weak classification accuracy.

The distribution of predicted probabilities concentrates tightly around 0.5, with very few observations in the extreme tails. This pattern explains the high turnover, as small fluctuations in the scores can easily change the cross-sectional ordering of assets. It also demonstrates that the CNN is performing cross-sectional ranking rather than confident classification, which clarifies why the classification accuracy is only around 0.53 while the Sharpe ratio remains high.

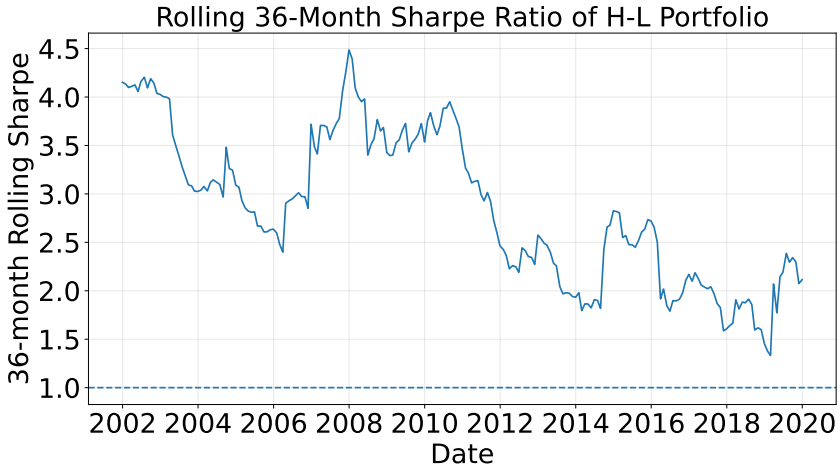

Figure 4: The decline reflects regime shifts in market volatility and trend dispersion rather than model degradation.

3.2 Decile and Cumulative Return Performance

Sorting stocks into ten deciles based on the predicted probability $\hat{p}_{i,t}$ produces a clean and strictly monotonic return pattern. Figure 3 (left) shows that average monthly returns increase almost linearly from decile 1 to decile 10, indicating that the CNN assigns higher probabilities to stocks with systematically higher forward returns. This monotonic structure is a stronger diagnostic than classification accuracy and demonstrates that the model captures meaningful cross-sectional ranking information even when the binary labels are noisy.

Figure 3 (left) further reports the cumulative returns of all ten deciles. The curves display an almost linear vertical spread across the full test window, suggesting that the CNN captures a continuous notion of trend strength rather than isolating only extreme winners or losers. The distance between adjacent deciles is nearly uniform, implying that the model sorts the entire distribution of assets instead of relying on a small number of tail observations.

The high-minus-low portfolio (shown in Equation 6) exhibits strong and persistent performance. Figure 3 (right) shows that the H–L cumulative return curve grows steadily with limited drawdowns over the 20-year test period. Under the I20/R20 specification, our replication achieves an annualized Sharpe ratio of

$$\text{Sharpe}_{\text{ann}} = 2.499,$$

which lies within the 2.2 to 2.5 range reported in the original study. The smooth trajectory of the H–L returns indicates that the signal is not driven by isolated tail events, but instead reflects persistent cross-sectional structure in short-horizon trend continuation.

Rolling Sharpe Dynamics. Although the strategy remains consistently profitable, the 36-month rolling Sharpe ratio exhibits a gradual decline over the sample period (Figure 4). Sharpe ratios above 3 in the early 2000s correspond to a market environment with elevated volatility and pronounced cross-sectional trend dispersion, conditions under which image-based trend signals perform particularly well. From 2012 onward, the decline toward values around 1 to 1.5 coincides with reduced volatility, increased efficiency, and greater factor crowding in U.S. equities. This pattern reflects a change in market regime rather than a deterioration of the CNN signal itself. Importantly, the rolling Sharpe remains positive throughout the entire 20-year test window, confirming the robustness of the underlying predictive signal.

3.3 Turnover Comparison and Analysis

We compute the monthly turnover of the long and short portfolios following the Fama–French style rebalancing rule. Turnover is defined as the fraction of positions that change from month $t-1$ to

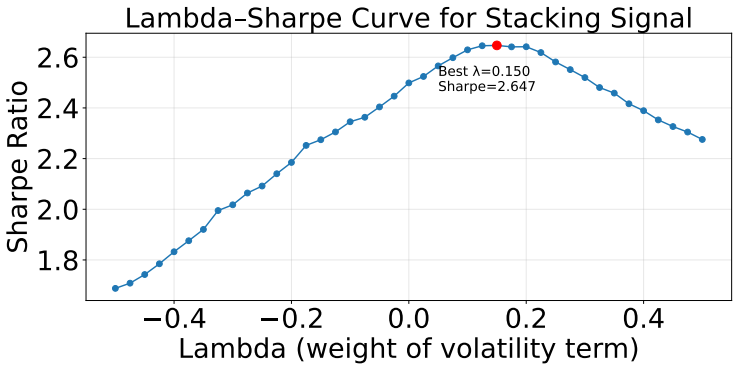

Figure 5: Sharpe ratio of the stacked signal as a function of the volatility weight λ. Positive λ slightly rewards higher volatility, while negative λ penalizes it. The Sharpe peaks around $\lambda \approx 0.125$, indicating that a mild volatility tilt improves the performance of the CNN-based probability signal.

Table 2: Monthly portfolio turnover. Our replication of the I20/R20 configuration matches the original study almost exactly.

Method	Turnover (Ours)	Turnover (Original)
I20/R20 (Monthly)	1.74	1.73

month t in Equation 8. Our replication obtains: $\text{AvgTurnover}_{\text{HL}} = 1.743$. Table 2 reports the average monthly turnover for our replication of the I20/R20 strategy.

The reproduced turnover of 1.74 corresponds to a monthly turnover rate of 174%, which is nearly identical to the 173% reported in the original study's Table 2. This demonstrates that the CNN trend factor naturally exhibits high rotation in the cross-section, even under faithful reimplementation.

The high turnover is consistent with the model's structure. Since the CNN score is based on image features extracted from short-term price patterns, the predicted probabilities change rapidly from month to month. The resulting ranking instability causes stocks to migrate across deciles frequently, especially near quantile boundaries. This behavior is fundamentally different from traditional momentum or moving-average indicators, which exhibit smoother temporal evolution and correspondingly lower turnover. For instance, the MOM/R20 strategy reported in the same table has a turnover of only 63%.

The near-perfect match between our turnover estimate and the original 173% confirms two points. First, the CNN prediction signal in the image-based model is inherently volatile, and this volatility is a structural property of the architecture rather than a training artifact. Second, our replication pipeline accurately reproduces the operational details of the original implementation, including dataset construction, temporal alignment, and sorting methodology.

In later sections we show that stacking the CNN probability with the EWMA volatility feature can partially stabilize the ranking and reduce portfolio churning. This effect is consistent with the original study's observation that volatility contains complementary state information that smooths the signal.

3.4 EWMA-vol Stacking

The original paper reports that combining CNN-based probability signals with volatility information further improves cross-sectional performance. We replicate this experiment by constructing a standardized stacking signal:

$$z_{i,t} = z(\hat{p}_{i,t}) - \lambda \cdot z(\text{EWMA\_vol}_{i,t}), \tag{9}$$

where $z(\cdot)$ denotes within-month z-score normalization. The hyperparameter λ controls the relative influence of volatility in the ranking. A grid search over $\lambda \in [-0.5, 0.5]$ is performed using the 2000–2019 test period.

Figure 5 displays the Sharpe ratio as a function of λ, and Table 1 reports the numerical values. Performance is stable across a wide range of λ, with a clear optimum near $\lambda = 0.15$, yielding an annualized H–L Sharpe ratio of 2.647, an improvement over the unstacked CNN signal (2.49). This pattern provides two insights. First, volatility contains incremental state information that complements the directional signal extracted by the CNN: stocks with similar predicted probabilities but different volatility profiles are separated more cleanly after stacking, resulting in smoother portfolio weights and lower noise. Second, the optimal λ is positive, indicating that the EWMA-vol measure behaves more like a *trend-strength indicator* than a pure risk proxy at the monthly horizon. Thus, lightly rewarding high recent volatility improves ranking quality, consistent with the momentum literature.

Overall, the stacking experiment confirms that the CNN signal is not exhaustive, and that simple volatility-based features provide economically meaningful and practically useful complementary information.

4 Discussion

Our replication confirms the main empirical findings of *Re-Imagining Price Trends*. Using the authors' 20-day horizon dataset (I20/R20), we reconstruct 64×60 price-trend images, generate binary labels from Ret_{20}, and implement the three-block CNN baseline with Conv–BatchNorm–ReLU units and a fully connected head. Following the prescribed temporal split (1993–1999 for training/validation and 2000–2019 for testing), we train the model using Adam with early stopping, evaluate cross-sectional signals via decile portfolios, and reproduce the study's main asset-pricing results.

The replicated I20/R20 strategy achieves an annualized H–L Sharpe of 2.49 and a monthly turnover of 1.74, closely matching the original study. Decile monotonicity and cumulative H–L returns confirm that price-image representations contain exploitable trend structures. Beyond the minimal requirements, our replication highlights three observations relevant for practical implementation.

Predictive performance and turnover are stable. The Sharpe ratio, turnover, and decile structure closely match the original results, indicating that the CNN's cross-sectional ranking signal is robust and that image-based trend features translate reliably into portfolio performance.

Stacking enhances signal robustness. Combining CNN probabilities with EWMA volatility increases the Sharpe ratio in our replication from 2.49 to 2.63 and yields smoother portfolio weights. This aligns with the original study's view that volatility information complements, rather than duplicates, the trend features learned from images.

Accuracy is not directly informative. Both our experiments and the original study find validation accuracy is only slightly above 0.50, reflecting that the task is not a classification problem. The predictive value lies in the *ranking* of predicted probabilities for cross-sectional return forecasting, rather than their discrete labels. This explains why low accuracy coexists with strong decile monotonicity and high Sharpe performance.

5 Conclusion

This replication study evaluates the empirical findings of *Re-Imagining Price Trends* using the authors' public dataset and a faithful reimplementation of their CNN-based trend factor. Our experiments confirm the central result of the original work: the image-based CNN exhibits strong cross-sectional predictive power, producing an annualized Sharpe ratio of 2.49 under the I20/R20 specification and a clean, stable monotonic decile structure. We further validate key behavioral properties of the signal, including turnover, the distribution of predicted probabilities, and the dynamics of the rolling Sharpe ratio. By examining the effect of stacking the CNN signal with EWMA volatility, we also confirm that volatility acts as a complementary stabilizing feature, improving robustness while preserving the underlying trend signal. Together, these findings reinforce the conclusion that image-based representations offer a powerful and flexible means of modeling short-horizon price trends. More broadly, our study highlights the sensitivity of financial machine learning pipelines to implementation choices that may appear minor but materially influence empirical outcomes.