CAW-KAN: CONTEXT-AWARE DUAL-GRID WAVELET KAN WITH LOW-RANK EDGE-FUNCTION MIXING FOR LONG-HORIZON FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Time series are rarely governed by a single simple relationship: once a variable is embedded into a latent representation, its future behavior depends on nonlinear interactions among the resulting latent coordinates. Kolmogorov-Arnold Networks (KANs) can model such interactions with learnable nonlinear edge functions, but giving every latent edge its own basis expansion requires a three-way coefficient tensor of size $d_{\text{in}} \cdot d_{\text{out}} \cdot N$, which erodes efficiency as latent width or dictionary size grows. We present CAW-KAN, a channel-independent forecaster that separates temporal context extraction from nonlinear mixing in value space. For each observed variable, reversible instance normalization and a value embedding produce a d_{model} -dimensional latent sequence, then a dense temporal convolution mixes across neighboring time steps and latent dimensions. A token-wise Wavelet-KAN evaluates a shared dictionary, built from two offset grids for complementary coverage of latent-value space, on the resulting scalar values; the grid centers and scales are currently fixed after initialization, with only the CP-factorized coefficients learned. To retain cross-feature edge functions without materializing the full coefficient tensor, CAW-KAN learns a low-rank CP factorization of these coefficients. Across 12 datasets and four forecast horizons, CAW-KAN attains the lowest error or ties for it in 59 of 96 MSE/MAE cells. Resource measurements show that CAW-KAN operates with significantly fewer parameters than other state-of-the-art models. Here, "context-aware" refers only to the local temporal representation fed into the Wavelet-KAN, not to a context gate or a temporal wavelet transform. The source code and implementation details are publicly available at https://github.com/tftpbe/CAW-KAN.

1 INTRODUCTION

Multivariate time-series forecasting supports decisions in domains ranging from energy and environmental monitoring to transportation and industrial operations. The long-horizon setting is particularly demanding because a predictor must extrapolate many future steps from a finite history while accommodating nonstationarity, local fluctuations, and dependencies that appear at different temporal scales. Recent work has approached this problem through linear decomposition, temporal convolution, patch-based attention, recurrent mixing, and multiscale architectures (Zeng et al., 2023; Nie et al., 2023; Wu et al., 2023; Wang et al., 2024; Liu et al., 2024; Hou & Yu, 2024). These model families differ substantially, yet they share a central design question: how should temporal evidence be converted into a compact representation on which expressive nonlinear prediction remains computationally practical?

KAN-based models replace fixed scalar edge weights with learnable univariate functions. This construction makes the nonlinear map itself adaptable rather than selecting one activation globally for an entire layer (Liu et al., 2025). Its use in forecasting has begun to show how edge functions can complement conventional temporal modules (Huang et al., 2025). At the same time, a direct basis expansion of every edge introduces a coefficient tensor indexed by output feature, input feature, and basis function. Its size grows as $d_{\text{out}} d_{\text{in}} N$, which can erase the efficiency sought in compact forecasters when either the latent width d or dictionary size N increases.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Two related tensions motivate our design. **First**, temporal context extraction and nonlinear latent-feature interaction are not interchangeable. A depthwise temporal operator can learn changes over time within each latent coordinate, but it does not mix information across coordinates. Conversely, applying an edge-function layer independently at each time step does not enlarge its temporal receptive field. A forecasting block therefore benefits from an explicit division of labor: local temporal and cross-feature mixing before token-wise nonlinear transformation. **Second**, a wavelet dictionary initialized on a single grid must trade broad value-space coverage against dense local resolution. This matters after normalization because most activations may concentrate near the origin while a non-negligible tail extends farther away. Treating wavelets as temporal filters would not resolve this allocation problem because the relevant argument is the scalar latent activation, not the time index.

We address these tensions with CAW-KAN, a context-aware dual-grid Wavelet-KAN for direct multi-horizon forecasting. The model processes each original variable independently with shared parameters, thereby avoiding unintended mixing between observed variables. Within a variable, however, its latent coordinates are deliberately coupled. Reversible instance normalization (RevIN) uses statistics from the observed window only (Kim et al., 2022); a short value embedding and optional observed-time descriptors initialize the latent sequence. Each encoder block first applies a dense, length-preserving temporal convolution and residual normalization. The resulting context representation is then passed to a Wavelet-KAN that acts pointwise in time. Thus, the convolution determines which neighboring observations inform a token, while the KAN determines how that token's latent coordinates interact nonlinearly.

The Wavelet-KAN shares N basis functions across latent edges. Their centers are initialized by two offset grids over the same bounded value range: a wide group emphasizes coverage, and a narrow group adds interior centers at a smaller initial scale. Both groups belong to one dictionary and one prediction path; they are not "trend" and "detail" branches. The centers and positive scales remain fixed after initialization, with only the CP-factorized coefficients being optimized. The selected range is diagnosed through the fraction of pre-KAN activations outside it. This construction relies on the initial bounded interval as a stable spatial anchor for the wavelet dictionary.

To preserve a genuine KAN edge-function map, each output latent feature remains a sum of learned univariate functions of all input latent features. CAW-KAN compresses only the coefficient tensor of this map. Specifically, it learns a rank-R canonical polyadic (CP) factorization over the output, input, and basis axes (Kolda & Bader, 2009). The factors are trained directly, and the full tensor is never materialized in the forward pass. Replacing the edge map with an elementwise wavelet activation would remove cross-feature edge functions, whereas CP factorization retains them subject to a controllable rank constraint. Here cross-feature refers to mixing latent coordinates within a single variable, not cross-channel interactions.

Contributions. Our contributions are:

- an implementation-aligned architecture combining channel-independent processing of observed variables with dense temporal and latent-feature mixing inside each variable, without a context gate, temporal wavelet frame, or future-covariate head;
- a dual-grid wavelet dictionary in latent-value space whose complementary initial center and scale allocations remain fixed after initialization (via Proposition 3.3), with non-linear feature mixing learned strictly through CP-factorized coefficients;
- a directly trained CP factorization that preserves cross-feature edge functions, accompanied by explicit approximation, expressivity, parameter, and operation analyses; and
- a twelve-dataset evaluation across four horizons, complemented by component ablations, dictionary and context sensitivity studies, and hardware measurements of parameters, inference time, and peak memory.

2 Related Work

Long-horizon forecasting. Long-horizon forecasters span several architectural families. Informer reduces the cost of attention for long input sequences (Zhou et al., 2021), PatchTST represents each univariate history through temporal patches (Nie et al., 2023), and iTransformer assigns attention

tokens to variables (Liu et al., 2024). Linear and MLP-based methods show that carefully chosen inductive biases can remain competitive without a large attention backbone: DLinear couples decomposition with simple linear maps (Zeng et al., 2023), whereas TimeMixer exchanges information across decomposed temporal scales (Wang et al., 2024). TimesNet maps temporal variation into structured two-dimensional representations (Wu et al., 2023), and RWKV-TS adapts recurrent mixing to time-series prediction (Hou & Yu, 2024). CAW-KAN instead shares one channel-independent encoder across observed variables and combines a local dense convolution with a token-wise edge-function map.

KANs and wavelet parameterizations. KANs parameterize network edges with learnable univariate functions and sum their contributions at each output (Liu et al., 2025). Wav-KAN instantiates such functions with translated and dilated wavelets (Bozorgasl & Chen, 2024), while TimeKAN combines KAN-based representation learning with frequency decomposition for forecasting (Huang et al., 2025). More recently, HaKAN (Hasan et al., 2026) leverages Hahn polynomials to parameterize KAN layers, modeling local and global temporal dynamics through hierarchical intra-patch and inter-patch structures without grid discretization. CAW-KAN takes a different route: its wavelet argument is a scalar hidden value, and the same dictionary is evaluated independently at every time index. Accordingly, the dual grid allocates basis functions along the latent-value axis; it neither decomposes the sequence into trend and detail components nor performs convolution with a temporal wavelet bank. Temporal context enters through the dense Conv1D that precedes the edge-function layer.

Low-rank tensor parameterization. CP decomposition represents a multiway array as a sum of rank-one outer products and provides a standard mechanism for trading tensor capacity against storage and computation (Kolda & Bader, 2009). For a Wavelet-KAN coefficient array in $\mathbb{R}^{d_{\text{out}} \times d_{\text{in}} \times N}$, rank R replaces $d_{\text{out}} d_{\text{in}} N$ coefficients with $R(d_{\text{out}} + d_{\text{in}} + N)$ factor parameters. CAW-KAN trains these factors as layer parameters rather than decomposing a pretrained dense tensor. The resulting model is therefore rank constrained from initialization onward; whether a particular rank improves forecasting remains an empirical question.

3 Methodology

3.1 Problem setting

Let $\boldsymbol{X} \in \mathbb{R}^{M \times L}$ contain L observed time steps for M variables, and let $\boldsymbol{Y} \in \mathbb{R}^{M \times T}$ contain the next T observations. CAW-KAN performs direct multi-horizon forecasting,

$$\widehat{\boldsymbol{Y}} = f_\theta(\boldsymbol{X}) \in \mathbb{R}^{M \times T}. \tag{1}$$

Optional deterministic timestamp descriptors are available only for the observed window. The encoder and forecasting head receive no measured future covariates and no future target-derived statistics. Hyperparameters are selected using validation data; the test split is not used to set the grid, normalization, or model configuration.

For $\boldsymbol{H} \in \mathbb{R}^{M \times L \times d}$, $\boldsymbol{H}_{m,t}$ denotes the d-dimensional latent vector for variable m at time t. The implementation is described channel independently at the level of the M observed variables: encoder parameters are shared, while the state of one observed variable is not mixed with another. Dense mixing occurs only among the d latent features associated with the same variable.

3.2 Architecture overview

For each variable m, CAW-KAN follows

$$\boldsymbol{x}^{(m)} \xrightarrow{\text{RevIN}} \boldsymbol{x}^{(m)}_{\text{norm}} \xrightarrow{\text{embedding}} \boldsymbol{H}^{(m)}_0 \xrightarrow{B \text{ blocks}} \boldsymbol{H}^{(m)}_B \xrightarrow{\text{head}} \widehat{\boldsymbol{y}}^{(m)}_{\text{norm}} \xrightarrow{\text{RevIN}^{-1}} \widehat{\boldsymbol{y}}^{(m)}. \tag{2}$$

Each block extracts local context with a dense temporal Conv1D and then applies a token-wise Wavelet-KAN. As emphasized in Figure 1, the wavelet dictionary operates on latent values; it is not a temporal filter bank.

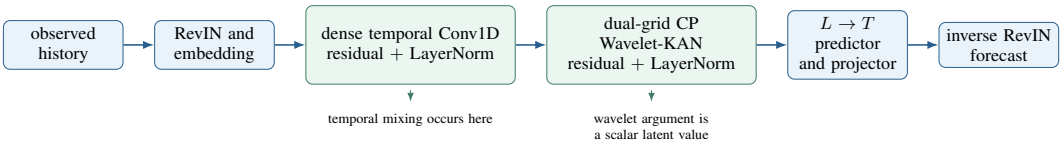

Figure 1: Code-aligned CAW-KAN architecture. One encoder is shared across observed variables. Within each variable, a dense temporal convolution mixes neighboring time steps and latent channels; the subsequent Wavelet-KAN acts token-wise in latent-value space.

3.3 Reversible normalization and embeddings

For variable m, RevIN computes statistics only from the observed window (Kim et al., 2022):

$$\mu_m = \frac{1}{L}\sum_{t=1}^{L} X_{m,t}, \qquad \sigma_m = \sqrt{\frac{1}{L}\sum_{t=1}^{L}(X_{m,t}-\mu_m)^2 + \varepsilon_{\mathrm{R}}}, \tag{3}$$

$$X_{\mathrm{norm},m,t} = \gamma_m \frac{X_{m,t}-\mu_m}{\sigma_m} + \beta_m. \tag{4}$$

The trainable affine parameters are initialized as $\gamma_m = 1$ and $\beta_m = 0$. Matching the working implementation description, inverse normalization uses $\delta_{\mathrm{R}} = \varepsilon_{\mathrm{R}}^2$:

$$\widehat{Y}_{m,t} = \sigma_m \frac{\widehat{Y}_{\mathrm{norm},m,t} - \beta_m}{\gamma_m + \delta_{\mathrm{R}}} + \mu_m. \tag{5}$$

This inverse is numerically approximate whenever $\delta_{\mathrm{R}} > 0$ and can amplify perturbations if $|\gamma_m + \delta_{\mathrm{R}}|$ becomes small. To guarantee numerical stability and strictly prevent division-by-zero during training, the implementation clamps this denominator to a minimum absolute magnitude (e.g., 10^{-5}) while preserving its original sign. The exact discrepancy is given in Section A.1.

A width-three temporal convolution embeds the normalized univariate sequence:

$$\boldsymbol{E}_{\mathrm{val}}^{(m)} = \mathrm{Conv1D}_3(\boldsymbol{X}_{\mathrm{norm}}^{(m)}) \in \mathbb{R}^{L\times d}, \tag{6}$$

where $\mathrm{Conv1D}_3$ denotes a 1-dimensional convolutional layer with a kernel size of 3. When observed timestamp descriptors $\boldsymbol{Z} \in \mathbb{R}^{L\times K}$ are enabled, a linear projection gives $\boldsymbol{E}_{\mathrm{time}} = \boldsymbol{Z}\boldsymbol{W}_{\mathrm{time}}$, with $\boldsymbol{W}_{\mathrm{time}} \in \mathbb{R}^{K\times d}$. Otherwise $\boldsymbol{E}_{\mathrm{time}} = \boldsymbol{0}$. The first hidden state is

$$\boldsymbol{H}_0^{(m)} = \mathrm{Dropout}\Big(\boldsymbol{E}_{\mathrm{val}}^{(m)} + \boldsymbol{E}_{\mathrm{time}}\Big) \in \mathbb{R}^{L\times d}. \tag{7}$$

Only the L observed timestamps are consumed.

3.4 Dense temporal context encoder

Let $k = 2q+1$ be the context kernel width. For block b, the standard dense Conv1D contains matrices $\boldsymbol{Q}_{b,\tau} \in \mathbb{R}^{d\times d}$ for $\tau \in \{-q,\dots,q\}$ and bias $\boldsymbol{c}_b \in \mathbb{R}^d$. With a length-preserving padding map p,

$$[\mathcal{A}_b(\boldsymbol{H})]_{t,:} = \boldsymbol{c}_b^\top + \sum_{\tau=-q}^{q}\Big(\boldsymbol{Q}_{b,\tau}\boldsymbol{H}_{p(t-\tau),:}^\top\Big)^\top. \tag{8}$$

The context representation is

$$\boldsymbol{C}_b^{(m)} = \mathrm{LN}_{b,1}\Big(\boldsymbol{H}_{b-1}^{(m)} + \mathcal{A}_b(\boldsymbol{H}_{b-1}^{(m)})\Big) \in \mathbb{R}^{L\times d}. \tag{9}$$

3.5 Dual-grid wavelet dictionary in latent-value space

The values in $\boldsymbol{C}_b^{(m)}$ are passed directly to a shared wavelet dictionary. Let $N_{\mathrm{w}} = \lceil N/2 \rceil$ and $N_{\mathrm{n}} = N - N_{\mathrm{w}}$ for $N \geq 5$. On the initialization interval $[-S_g, S_g]$, the wide centers are

$$\Delta_{\mathrm{w}} = \frac{2S_g}{N_{\mathrm{w}}-1}, \qquad b_{\mathrm{w},i} = -S_g + i\Delta_{\mathrm{w}}. \tag{10}$$

The narrow centers are offset into the interior:

$$\Delta_{\mathrm{n}} = \frac{2S_g - \Delta_{\mathrm{w}}}{N_{\mathrm{n}} - 1}, \tag{11}$$

$$b_{\mathrm{n},j} = -S_g + \frac{\Delta_{\mathrm{w}}}{2} + j\Delta_{\mathrm{n}}. \tag{12}$$

These equations define two complementary initial grids, not two temporal branches. Exact midpoint interleaving occurs for the odd-N construction $N = 2r + 1$; for even N, including the legacy setting $N = 8$, Equation 12 is an offset interior grid and must not be described as exact alternation.

For a target Mexican-Hat midpoint response $\rho \in (0, 1)$ and validated narrow-to-wide ratio $\rho_a \in (0, 1)$, the initial scales are

$$a_{\mathrm{w}}^{(0)} = c(\rho)\Delta_{\mathrm{w}}, \qquad a_{\mathrm{n}}^{(0)} = \rho_a c(\rho)\Delta_{\mathrm{w}}, \tag{13}$$

where $c(\rho)$ is given in Proposition 3.3. Supported basis families include

$$\psi_{\mathrm{MH}}(u) = (1 - u^2)e^{-u^2/2}, \quad \psi_{\mathrm{Morlet}}(u) = \cos(\omega_0 u)e^{-u^2/2}, \quad \psi_{\mathrm{DoG}}(u) = -ue^{-u^2/2}. \tag{14}$$

The closed-form scale in Equation 13 is specific to the positive lobe of the Mexican-Hat basis.

For diagnostics, the pre-KAN overflow rate at block b is

$$r_{\mathrm{out},b}(S_g) = \frac{1}{MLd} \sum_{m=1}^{M} \sum_{t=1}^{L} \sum_{i=1}^{d} \mathbf{1}\left\{ \left| C_{b,t,i}^{(m)} \right| > S_g \right\}. \tag{15}$$

Translations and scales remain fixed after initialization, so $[-S_g, S_g]$ dictates the permanent spatial layout of the dictionary. However, the input values are not strictly clipped, allowing the model's robustness to be diagnosed by the fraction of activations falling outside this interval.

The empirical evaluation of this overflow rate across multiple datasets and random seeds is detailed in Appendix B.6, demonstrating the stability and robust coverage of our dual-grid initialization.

3.6 CP-FACTORIZED EDGE-FUNCTION WAVELET-KAN

For a token $\boldsymbol{z} = (z_1, \ldots, z_d)^\top$, define

$$\Phi_{i,k}(\boldsymbol{z}) = \psi\left(\frac{z_i - b_k}{a_k} \right), \qquad i = 1, \ldots, d, \quad k = 1, \ldots, N. \tag{16}$$

A dense edge-function layer with d' outputs is

$$F_j(\boldsymbol{z}) = \sum_{i=1}^{d} \sum_{k=1}^{N} W_{j,i,k} \Phi_{i,k}(\boldsymbol{z}), \qquad j = 1, \ldots, d'. \tag{17}$$

This retains the KAN form $F_j = \sum_i \phi_{j,i}(z_i)$ (Liu et al., 2025). CAW-KAN parameterizes the coefficient tensor directly with CP rank R:

$$W_{j,i,k} = \sum_{r=1}^{R} A_{j,r} B_{i,r} D_{k,r}, \tag{18}$$

where $\boldsymbol{A} \in \mathbb{R}^{d' \times R}$, $\boldsymbol{B} \in \mathbb{R}^{d \times R}$, and $\boldsymbol{D} \in \mathbb{R}^{N \times R}$. The forward pass avoids materializing $\boldsymbol{W}$:

$$U_{i,r}(\boldsymbol{z}) = \sum_{k=1}^{N} D_{k,r} \Phi_{i,k}(\boldsymbol{z}), \tag{19}$$

$$v_r(\boldsymbol{z}) = \sum_{i=1}^{d} B_{i,r} U_{i,r}(\boldsymbol{z}), \tag{20}$$

$$F_j(\boldsymbol{z}) = \sum_{r=1}^{R} A_{j,r} v_r(\boldsymbol{z}). \tag{21}$$

The same map is applied independently at each time index. In each encoder block $d' = d$, and the block output is

$$\boldsymbol{H}_b^{(m)} = \mathrm{Dropout}\Big[\mathrm{LN}_{b,2}\Big(\boldsymbol{C}_b^{(m)} + \mathrm{FKAN}_b(\boldsymbol{C}_b^{(m)})\Big)\Big]. \tag{22}$$

To ensure stability during the initial forward pass and prevent activation variance explosion prior to tensor reconstruction, the CP factor matrices A, B, and D are systematically initialized using the Kaiming uniform distribution with $a = \sqrt{5}$ (He et al., 2015).

3.7 Forecasting head

A shared linear predictor maps the look-back axis to the forecast horizon while retaining the latent features:

$$P_{t,j}^{(m)} = \sum_{s=1}^{L} W_{h,t,s} H_{B,s,j}^{(m)} + b_{h,t}, \quad \boldsymbol{W}_h \in \mathbb{R}^{T \times L}. \tag{23}$$

A shared projector returns one normalized value,

$$\widehat{Y}_{\mathrm{norm},m,t} = \sum_{j=1}^{d} w_{o,j} P_{t,j}^{(m)} + b_o, \tag{24}$$

and Equation 5 restores the original scale. The confirmed working description contains neither a low-rank horizon head nor an additive future-covariate path.

3.8 Implementation-linked properties

The following results clarify three design choices. They do not imply forecasting superiority, statistical consistency, or distribution-free generalization. Proofs and auxiliary stability results are in Section A.

Lemma 3.1 (Distribution-free LayerNorm coverage)**.** *Let*

$$q_i = \frac{x_i - \bar{x}}{\sqrt{d^{-1}\sum_{j=1}^{d}(x_j - \bar{x})^2 + \varepsilon_{\mathrm{LN}}}}.$$

For any $S > 0$,

$$\frac{1}{d}\#\{i : |q_i| > S\} \le \min\left\{1, \frac{1}{S^2}\right\}. \tag{25}$$

If $c_i = \lambda_i q_i + \rho_i$, $|\lambda_i| \le \Lambda$, $|\rho_i| \le P$, $S > P$, and $\Lambda > 0$, the right-hand side becomes $\min\{1, \Lambda^2/(S-P)^2\}$.

Lemma 3.1 is a deterministic second-moment bound, not a Gaussian three-sigma statement. It motivates reporting the empirical overflow in Equation 15: even without affine parameters, the worst-case bound outside $[-3, 3]$ is $1/9$, not 0.3%.

Theorem 3.2 (CP rank–error–cost trade-off)**.** *Let full and rank-R coefficient tensors $\boldsymbol{W}$ and $\boldsymbol{W}_R$ use the same dictionary, and assume $|\psi(u)| \le B_\psi$. Then*

$$\|\boldsymbol{F}(\boldsymbol{z}) - \boldsymbol{F}_R(\boldsymbol{z})\|_2 \le B_\psi \sqrt{dN}\, \|\boldsymbol{W} - \boldsymbol{W}_R\|_F. \tag{26}$$

Every tensor in $\mathbb{R}^{d' \times d \times N}$ has an exact CP representation at some $R \le \min\{d'd, d'N, dN\}$. Excluding an optional output bias, the full and CP layers contain

$$P_{\mathrm{full}} = d'dN, \qquad P_{\mathrm{CP}} = R(d' + d + N). \tag{27}$$

After dN basis evaluations, their coefficient contractions require respectively $d'dN$ and $dNR + dR + d'R$ MACs per token.

Thus parameter saving requires $R < d'dN/(d' + d + N)$, whereas contraction saving requires the different threshold $R < d'dN/(dN + d + d')$. Since CAW-KAN trains the factors directly, Equation 26 compares two function classes; it does not state that optimization finds the best rank-R approximation.

Proposition 3.3 (Mexican-Hat midpoint overlap). *Let $\psi(z) = (1 - z^2)e^{-z^2/2}$. Place two copies at adjacent centers separated by $\Delta > 0$, with common scale $a = c\Delta$. For target response $\rho \in (0, 1)$ at their midpoint, there is a unique $c > 1/2$:*

$$c(\rho) = \frac{1}{2\sqrt{1 - 2W_0(\rho e^{1/2}/2)}}, \tag{28}$$

where W_0 is the principal Lambert-W branch. For $\rho = 1/2$, $c \simeq 0.7988$.

For equally weighted neighboring atoms, each contributes ρ at the midpoint and their sum is 2ρ. The midpoint is the boundary of the adjacent centers' Voronoi cells and the point farthest from the nearest center inside either half-cell. Controlling the response there therefore controls the least-covered location between the two centers. This initialization argument neither chooses the narrow ratio ρ_a nor transfers to Morlet or DoG.

4 Experiments

We study CAW-KAN from four complementary perspectives: long-horizon forecasting accuracy, the contribution of its architectural components, hyperparameter sensitivity, and practical efficiency. All numerical values in this section are taken from the reported experiments in the supplied manuscript; the full comparison tables are reproduced in Section B.2.

4.1 Experimental setup

Datasets. The evaluation contains 12 multivariate series drawn from six dataset families: the four ETT benchmarks (ETTh1, ETTh2, ETTm1, and ETTm2), Weather and Exchange (Zhou et al., 2021; Shih et al., 2019), four stations from the Beijing Multi-Site Air-Quality dataset (Aotizhongxin, Changping, Dingling, and Guanyuan) (Zhang et al., 2017), Appliances Energy (Candanedo et al., 2017), and Bike Sharing (Fanaee-T & Gama, 2013). Only real-valued attributes are retained for Bike Sharing. The suite spans 5–28 variables, sampling rates from 10 minutes to one day, and sequence lengths from 7,588 to 69,680; complete statistics appear in Table 3.

Baselines. We compare against MLP/linear forecasters (DLinear and TimeMixer), Transformer forecasters (PatchTST and iTransformer), convolutional or recurrent models (TimesNet and RWKV-TS), KAN-based models (TimeKAN and HaKAN), and the lightweight time-series foundation model SEMPO (Zeng et al., 2023; Wang et al., 2024; Nie et al., 2023; Liu et al., 2024; Wu et al., 2023; Hou & Yu, 2024; Huang et al., 2025; Hasan et al., 2026; He et al., 2025).

Forecasting protocol. The experiment uses look-back length $L = 512$ and evaluates $T \in \{96, 192, 336, 720\}$ for every series. CAW-KAN uses a Mexican-Hat dictionary with $N = 8$ basis functions, initialized over a bounded grid range of $[-S_g, S_g]$ with $S_g = 3.0$. To ensure seamless coverage without dead zones, the midpoint initialization response (the overlap amplitude between adjacent wavelets) is set to $\rho = 0.5$. Furthermore, to capture both broad trends and local fluctuations, the narrow-to-wide initial scale ratio is fixed at $\rho_a = 0.5$, meaning the interleaved narrow wavelets are exactly half the width of the wide ones. (When evaluating the Morlet wavelet family, its base frequency parameter is statically set to $\omega_0 = 5.0$ as a non-learnable buffer). The CP rank R and temporal context kernel k are treated as dataset-dependent hyperparameters. It is trained with the Log-Cosh objective (Saleh & Saleh, 2022)

$$\mathcal{L}_{\mathrm{LC}}(y, \widehat{y}) = \sum_{i=1}^{n} \log(\cosh(\widehat{y}_i - y_i)), \tag{29}$$

where $n = B \times M \times T$ denotes the total number of flattened elements across the batch size B, number of variables M, and forecast horizon T.

Implementation details. All experiments are conducted on a single NVIDIA A100 40GB GPU. To ensure a fair comparison CAW-KAN is optimized using a Log-Cosh loss, while baselines use standard MSE. The multi-seed results of the models trained with the same Log-Cosh loss are provided in Appendix B.5. Thus, our primary results reflect the combined benefits of the CAW-KAN architecture and robust optimization. All accuracy values in the main tables are point estimates obtained using a single fixed initialization seed (2021).

Table 1: Representative reported comparisons. Each parenthesis identifies the strongest other method for that metric. Lower is better.

Dataset	T	MSE		MAE	
		CAW	Best other	CAW	Best other
ETTh1	96	**0.360**	0.364 (TimeKAN)	**0.388**	0.395 (HaKAN)
ETTm1	96	**0.288**	0.295 (TimeKAN)	**0.341**	0.347 (DLinear)
Exchange	96	**0.088**	**0.088** (TimeMixer)	0.213	**0.210** (TimeMixer)
Aotizhongxin	96	0.637	**0.634** (TimeKAN)	**0.450**	0.459 (TimeKAN)
ETTh1	336	0.435	**0.419** (SEMPO)	0.431	**0.421** (HaKAN)
ETTh2	96	0.269	**0.254** (HaKAN)	0.331	**0.323** (HaKAN)
Weather	720	0.313	**0.309** (TimeKAN)	**0.325**	0.336 (TimeKAN)
Exchange	720	1.016	**0.957** (TimeKAN)	0.771	**0.734** (TimeKAN)

4.2 Long-horizon forecasting

Table 1 highlights representative comparisons where CAW-KAN exhibits strong performance at shorter forecasting horizons ($T = 96$). It achieves the lowest errors on ETTh1 and ETTm1, matches the leading baseline on the Exchange dataset, and secures the best MAE on Aotizhongxin. However, the results also reveal specific limitations: competing models such as HaKAN and TimeKAN demonstrate superior performance on ETTh2 ($T = 96$) and across longer horizons ($T \in \{336, 720\}$) for ETTh1, Weather, and Exchange. This performance divergence suggests that the context-aware value-space dictionary is not fully optimal; while it manages to capture local variations in short-term predictions, its overall effectiveness remains limited. Furthermore, the channel independence assumption and shared direct head restrict the model's capacity to resolve complex, long-range dependencies.

4.3 Ablation studies

We ablate CAW-KAN's design choices across two functional groups: architecture and objective and dictionary and latent mixing, with average performance across forecasting horizons summarized in Table 2.

Architecture and Objective Analysis. Our ablation study validates the essential contribution of each proposed component. Removing the 1D convolution (w/o Conv1D) degrades performance, confirming its necessity in extracting local contexts and smoothing temporal fluctuations prior to KAN processing. Eliminating residual connections (w/o Residual) disrupts stable gradient flow, thereby hampering the optimization process. Replacing the Wavelet-KAN core with a standard MLP (KAN $\rightarrow$ MLP) limits representational capacity, proving that KAN's edge-parameterized functions are vastly superior at capturing complex, non-linear time-series dynamics. Finally, substituting the Log-Cosh loss with a standard MSE objective (Log-Cosh $\rightarrow$ MSE) yields the most significant performance drop. This highlights that KANs are highly sensitive to outliers; thus, the Log-Cosh

Table 2: Reported component ablations. The full model is repeated as the common reference. Lower is better; the best value in each column is bold.

	ETTh1		ETTh2		ETTm1		ETTm2		AVG	
Variant	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Original	**0.414**	**0.423**	**0.346**	0.386	**0.342**	0.374	**0.254**	0.311	**0.339**	**0.373**
w/o Conv1D	0.422	0.429	0.346	**0.384**	0.353	0.377	0.254	0.312	0.344	0.376
w/o Residual	0.422	0.426	0.352	0.389	0.349	0.376	0.255	0.312	0.344	0.376
KAN $\rightarrow$ MLP	0.420	0.424	0.352	0.390	0.344	**0.374**	0.255	0.312	0.343	0.375
Log-Cosh $\rightarrow$ MSE Loss	0.418	0.427	0.360	0.401	0.344	0.383	0.257	0.317	0.345	0.382
Single-grid	0.424	0.427	0.356	0.394	0.346	0.375	0.254	0.312	0.345	0.377
Dual-grid	0.417	0.424	0.353	0.391	0.347	0.377	0.254	**0.311**	0.343	0.376
Single-grid + CP	0.428	0.433	0.346	0.388	0.343	0.374	0.254	0.311	0.343	0.377
Diagonal only	0.420	0.427	0.349	0.387	0.346	0.374	0.254	0.311	0.342	0.375

objective is crucial for regularizing training by combining MAE's robustness with MSE's smooth convergence.

Dictionary and Latent Mixing Analysis. We next isolate the impact of grid initialization and tensor factorization. Initializing a *Single-grid* that utilizes only wide groups and mixes latent features within a single variable noticeably degrades overall performance, proving the necessity of complementary dual-grid coverage. The *Single-grid + CP* variant, combining this single grid with CP factorization, improves upon the dense single-grid baseline but remains inferior to the original model, indicating that factorization alone cannot compensate for a limited basis set. Furthermore, utilizing the *Dual-grid* without CP factorization falls short of the proposed architecture, demonstrating that CP factorization acts as a crucial regularizer rather than a representational bottleneck. Finally, the *Diagonal only* setting that entirely removes the mixing of latent features within a single variable degrades average accuracy despite a marginal edge on ETTm2. This empirically confirms that latent feature mixing is broadly essential for effective multivariate forecasting.

4.4 Sensitivity and efficiency

Sensitivity Analysis. The table 8 shows the sensitivty of models with different parameters. Both the number of wavelets N and the grid range S_g demonstrate that moderate capacities generally yield the best generalization by avoiding high frequency noise, though highly complex spectra necessitate larger and wider basis sets. Similarly, the CP factorization rank R and the number of blocks B underscore that maintaining moderate structural complexity effectively regularizes the learning process without creating representational bottlenecks. This need for balanced capacity extends to the temporal context, where moderate kernels smooth stochastic fluctuations while narrow contexts preserve rapid and localized shifts. Finally, the choice of basis function clearly reflects distinct signal properties. Complex wavelets excel at explicitly modeling phase and periodic oscillations, whereas real valued functions better capture sharp and transient variations.

Efficiency Analysis. Table 9 presents the complete parameter counts and reported MAC values across all evaluated models. In this resource study, while ultra-lightweight baselines exist, CAW-KAN operates with significantly fewer parameters than current state-of-the-art models, ranging from 52.651K to 54.991K. Furthermore, its computational cost remains remarkably low at 28.029M to 48.303M MACs which is fewer than other models such as TimesNet, PatchTST, or HaKAN.

5 Conclusion

We presented CAW-KAN, a multi-horizon forecaster separating temporal context extraction (via dense Conv1D) from nonlinear edge-function mixing (via a dual-grid wavelet dictionary). A directly trained CP parameterization drastically reduces complexity while preserving cross-feature interactions, yielding the lowest parameter count and fastest inference among evaluated baselines. Empirically, CAW-KAN leads 59 of 96 MSE/MAE evaluations across 12 datasets. Theoretical analyses and ablations further validate our dual-grid initialization, temporal contextualization, and robust Log-Cosh training.

Limitations. Channel independence precludes direct cross-variable interactions, dense temporal convolution scales quadratically with latent width, and the direct $L \to T$ head dominates long-window costs. Optimal grid ranges, dictionary sizes, and CP ranks remain data-dependent.

References

Zavareh Bozorgasl and Hao Chen. Wav-KAN: Wavelet kolmogorov–arnold networks. *arXiv preprint arXiv:2405.12832*, 2024. doi: 10.48550/arXiv.2405.12832. URL https://arxiv.org/abs/2405.12832.

Luis M. Candanedo, Véronique Feldheim, and Dominique Deramaix. Data driven prediction models of energy use of appliances in a low-energy house. *Energy and Buildings*, 140:81–97, 2017. doi: 10.1016/j.enbuild.2017.01.083.

Hadi Fanaee-T and João Gama. Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence*, 2:113 – 127, 2013. URL https://api.semanticscholar.org/CorpusID:256282956.

Md Zahidul Hasan, Abdessamad Ben Hamza, and Nizar Bouguila. Time series forecasting with hahn kolmogorov–arnold networks. In *Proceedings of the 29th International Conference on Artificial Intelligence and Statistics*, volume 300 of *Proceedings of Machine Learning Research*, pp. 2188–2196. PMLR, 2026. URL https://proceedings.mlr.press/v300/hasan26a.html.

Hui He, Kun Yi, Yuanchi Ma, Qi Zhang, Zhendong Niu, and Guansong Pang. Sempo: Lightweight foundation models for time series forecasting. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 161846–161879. Curran Associates, Inc., 2025. doi: 10.52202/085713-5405. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/ecfb69ce6be017deb5a926c2718f6bc1-Paper-Conference.pdf.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1026–1034, December 2015. doi: 10.1109/ICCV.2015.123.

Haowen Hou and F. Richard Yu. RWKV-TS: Beyond traditional recurrent neural network for time series tasks. *arXiv preprint arXiv:2401.09093*, 2024. doi: 10.48550/arXiv.2401.09093. URL https://arxiv.org/abs/2401.09093.

Songtao Huang, Zhen Zhao, Can Li, and LEI BAI. Timekan: Kan-based frequency decomposition learning architecture for long-term time series forecasting. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 28514–28529, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/46362971bfc3a97e6a271f2eb90fba17-Paper-Conference.pdf.

Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *The Tenth International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=cGDAkQo1C0p.

Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009. doi: 10.1137/07070111X. URL https://doi.org/10.1137/07070111X.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 11116–11140, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/2ea18fdc667e0ef2ad82b2b4d65147ad-Paper-Conference.pdf.

Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljacic, Thomas Hou, and Max Tegmark. Kan: Kolmogorov–arnold networks. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 70367–70413, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/afaed89642ea100935e39d39a4da602c-Paper-Conference.pdf.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Jbdc0vTOcol.

Resve A. Saleh and A. K. Md. Ehsanes Saleh. Statistical properties of the log-cosh loss function used in machine learning. *ArXiv*, abs/2208.04564, 2022. URL https://api.semanticscholar.org/CorpusID:251442781.

Shun-Yao Shih, Fan-Keng Sun, and Hung-yi Lee. Temporal pattern attention for multivariate time series forecasting. *Mach. Learn.*, 108(8–9):1421–1441, September 2019. ISSN 0885-6125. doi: 10.1007/s10994-019-05815-0.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Zhang, and JUN ZHOU. Timemixer: Decomposable multiscale mixing for time series forecasting. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 38626–38652, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/a7ac8a21e5a27e7ab31a5f42a0117bdb-Paper-Conference.pdf.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2d-variation modeling for general time series analysis. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ju_Uqw384Oq.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(9):11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317. URL https://ojs.aaai.org/index.php/AAAI/article/view/26317.

Shuyi Zhang, Bin Guo, Anlan Dong, Jing He, Ziping Xu, and Song Xi Chen. Cautionary tales on air-quality improvement in beijing. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 473(2205), 2017. doi: 10.1098/rspa.2017.0457.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(12):11106–11115, 2021. doi: 10.1609/aaai.v35i12.17325. URL https://ojs.aaai.org/index.php/AAAI/article/view/17325.

A Detailed Proofs and Implementation Checks

Conventions. Vector norms are Euclidean. Matrix and sequence norms are Frobenius unless a subscript 2 explicitly denotes the spectral norm. The parameters of every map are fixed when a Lipschitz bound is stated, and dropout is the identity at inference time. Hidden sequences belong to $\mathbb{R}^{L\times d}$, token maps have the form $\boldsymbol{F} : \mathbb{R}^d \to \mathbb{R}^{d'}$, and the horizon predictor satisfies $\boldsymbol{W}_h \in \mathbb{R}^{T\times L}$. None of the deterministic inequalities below is a statement about optimization, generalization, robustness to distribution shift, or forecast accuracy.

A.1 Implemented RevIN inversion

For one observed variable, fix window statistics $\mu \in \mathbb{R}$ and $\sigma > 0$, affine parameters $\gamma, \beta \in \mathbb{R}$, and an inverse offset $\delta_{\mathrm{R}} \geq 0$ such that $\gamma + \delta_{\mathrm{R}} \neq 0$. Define

$$\mathcal{N}(x) = \gamma \frac{x-\mu}{\sigma} + \beta, \qquad \mathcal{D}(u) = \sigma \frac{u-\beta}{\gamma+\delta_{\mathrm{R}}} + \mu. \tag{30}$$

Then direct substitution gives

$$\begin{aligned} \mathcal{D}(\mathcal{N}(x)) - x &= \frac{\gamma}{\gamma+\delta_{\mathrm{R}}}(x-\mu) + \mu - x \\ &= -\frac{\delta_{\mathrm{R}}}{\gamma+\delta_{\mathrm{R}}}(x-\mu). \end{aligned} \tag{31}$$

Consequently, for a vector $\boldsymbol{x}$ normalized with the same fixed statistics,

$$\|\mathcal{D}(\mathcal{N}(\boldsymbol{x})) - \boldsymbol{x}\|_2 = \frac{\delta_{\mathrm{R}}}{|\gamma+\delta_{\mathrm{R}}|} \|\boldsymbol{x} - \mu\mathbf{1}\|_2 . \tag{32}$$

For any normalized forecasts $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^T$,

$$\|\mathcal{D}(\boldsymbol{u}) - \mathcal{D}(\boldsymbol{v})\|_2 = \frac{\sigma}{|\gamma + \delta_{\mathrm{R}}|} \|\boldsymbol{u} - \boldsymbol{v}\|_2 . \tag{33}$$

Equation (33) is conditional on the same look-back-window statistics. It must not be interpreted as a bound for two raw windows whose means and standard deviations also change.

A.2 Dense temporal convolution and LayerNorm

Let the padding map in equation (8) satisfy

$$\nu_b = \max_{\tau \in \{-q,\dots,q\}} \max_{s \in \{1,\dots,L\}} \#\{t : p(t-\tau) = s\}. \tag{34}$$

This multiplicity records how often one input location can be reused at a fixed kernel offset; its value depends on the implemented padding rule. For $\boldsymbol{\Delta} = \boldsymbol{H} - \boldsymbol{H}'$ and a fixed offset τ,

$$\begin{aligned}\sum_{t=1}^{L} \left\| \boldsymbol{Q}_{b,\tau} \boldsymbol{\Delta}^{\top}_{p(t-\tau),:} \right\|_2^2 &\le \|\boldsymbol{Q}_{b,\tau}\|_2^2 \sum_{t=1}^{L} \left\| \boldsymbol{\Delta}_{p(t-\tau),:} \right\|_2^2 \\ &\le \nu_b \|\boldsymbol{Q}_{b,\tau}\|_2^2 \|\boldsymbol{\Delta}\|_F^2 .\end{aligned} \tag{35}$$

The triangle inequality over kernel offsets therefore yields

$$\|\mathcal{A}_b(\boldsymbol{H}) - \mathcal{A}_b(\boldsymbol{H}')\|_F \le \underbrace{\sqrt{\nu_b} \sum_{\tau=-q}^{q} \|\boldsymbol{Q}_{b,\tau}\|_2}_{K_{\mathrm{conv},b}} \|\boldsymbol{H} - \boldsymbol{H}'\|_F . \tag{36}$$

The bias cancels in the difference.

We next derive a global upper bound for affine LayerNorm. Let $\boldsymbol{P} = \boldsymbol{I} - d^{-1}\mathbf{1}\mathbf{1}^{\top}$ and

$$g(\boldsymbol{u}) = \frac{\boldsymbol{u}}{\sqrt{d^{-1}\|\boldsymbol{u}\|_2^2 + \varepsilon_{\mathrm{LN}}}}. \tag{37}$$

Writing $\alpha^2 = d^{-1}\|\boldsymbol{u}\|_2^2 + \varepsilon_{\mathrm{LN}}$, the Jacobian is

$$\boldsymbol{J}_g(\boldsymbol{u}) = \alpha^{-1}\boldsymbol{I} - d^{-1}\alpha^{-3}\boldsymbol{u}\boldsymbol{u}^{\top}. \tag{38}$$

Its eigenvalue in the direction of $\boldsymbol{u}$ is $\varepsilon_{\mathrm{LN}}\alpha^{-3}$, and every orthogonal eigenvalue is α^{-1}. Both are at most $\varepsilon_{\mathrm{LN}}^{-1/2}$. Because $\|\boldsymbol{P}\|_2 = 1$ and the spectral norm of $\mathrm{diag}(\boldsymbol{\lambda})$ is $\|\boldsymbol{\lambda}\|_\infty$, the mean-value theorem gives

$$\|\mathrm{LN}_{\boldsymbol{\lambda},\boldsymbol{\rho}}(\boldsymbol{x}) - \mathrm{LN}_{\boldsymbol{\lambda},\boldsymbol{\rho}}(\boldsymbol{y})\|_2 \le \frac{\|\boldsymbol{\lambda}\|_\infty}{\sqrt{\varepsilon_{\mathrm{LN}}}} \|\boldsymbol{x} - \boldsymbol{y}\|_2 . \tag{39}$$

Applying the same token map to every row preserves this constant under the Frobenius norm, because the squared row inequalities can be summed.

A.3 Proof of LayerNorm coverage

Proof of lemma 3.1. Let $v = d^{-1}\sum_i (x_i - \bar{x})^2$. The normalized coordinates satisfy

$$\frac{1}{d}\sum_{i=1}^{d} q_i^2 = \frac{v}{v + \varepsilon_{\mathrm{LN}}} \le 1. \tag{40}$$

If $n_S = \#\{i : |q_i| > S\}$, then $n_S S^2 < \sum_i q_i^2 \le d$, which implies $n_S/d \le \min\{1, S^{-2}\}$. For affine coordinates $c_i = \lambda_i q_i + \rho_i$, the assumptions $|\lambda_i| \le \Lambda$, $|\rho_i| \le P$, $S > P$, and $\Lambda > 0$ imply

$$|c_i| > S \implies \Lambda\,|q_i| + P > S \implies |q_i| > (S - P)/\Lambda.$$

Substituting this threshold into the first result gives $d^{-1}\#\{i : |c_i| > S\} \le \min\{1, \Lambda^2/(S - P)^2\}$. □

The result applies separately to every token and makes no distributional assumption. The empirical statistic in equation (15) is still necessary because the deterministic bound can be loose and because learned affine LayerNorm parameters change the effective range.

A.4 Proof of the CP rank–error–cost result

Proof of Theorem 3.2. Let $\boldsymbol{E} = \boldsymbol{W} - \boldsymbol{W}_R$ and collect the basis responses in $\boldsymbol{\Phi}(\boldsymbol{z}) \in \mathbb{R}^{d \times N}$. For output j,

$$F_j(\boldsymbol{z}) - F_{R,j}(\boldsymbol{z}) = \langle \boldsymbol{E}_{j,:,:}, \boldsymbol{\Phi}(\boldsymbol{z}) \rangle_F.$$

Frobenius Cauchy–Schwarz, followed by summation over j, gives

$$\begin{aligned} \|\boldsymbol{F}(\boldsymbol{z}) - \boldsymbol{F}_R(\boldsymbol{z})\|_2^2 &\le \sum_{j=1}^{d'} \|\boldsymbol{E}_{j,:,:}\|_F^2 \, \|\boldsymbol{\Phi}(\boldsymbol{z})\|_F^2 \\ &= \|\boldsymbol{E}\|_F^2 \, \|\boldsymbol{\Phi}(\boldsymbol{z})\|_F^2 . \end{aligned} \tag{41}$$

If $|\psi(u)| \le B_\psi$, then $\|\boldsymbol{\Phi}(\boldsymbol{z})\|_F \le B_\psi \sqrt{dN}$, proving equation (26).

For exactness, let $\boldsymbol{e}_i$ and $\boldsymbol{e}_k$ be standard basis vectors. The construction

$$\boldsymbol{W} = \sum_{i=1}^{d} \sum_{k=1}^{N} \boldsymbol{W}_{:,i,k} \circ \boldsymbol{e}_i \circ \boldsymbol{e}_k \tag{42}$$

has at most dN rank-one terms. Constructing rank-one terms by indexing the other two pairs of modes gives the bounds $d'd$ and $d'N$, hence an exact representation exists at some $R \le \min\{d'd, d'N, dN\}$. This is exactness for a fixed dictionary and its coefficient tensor, not a representation theorem for arbitrary multivariate functions at smaller rank.

The dense tensor contributes $d'dN$ coefficients and the shared translations and scales contribute $2N$ parameters. The three CP factors contribute $R(d' + d + N)$ parameters, with the same $2N$ dictionary parameters. An optional output bias adds the same d' term to both alternatives. Excluding the dN basis evaluations, the dense contraction in equation (17) uses $d'dN$ MACs. Equations (19)–(21) use $dNR + dR + d'R = R(dN + d + d')$ MACs. Solving the corresponding strict inequalities gives the distinct parameter- and contraction-saving thresholds stated after Theorem 3.2. □

A.5 Edge-function and conditional network stability

Assume the selected basis is differentiable with $\sup_u |\psi'(u)| \le G$ and every scale satisfies $a_k \ge a_{\min} > 0$. For the conceptual coefficient tensor in equation (17), define

$$c_{j,i} = \sum_{k=1}^{N} |W_{j,i,k}| \, \frac{G}{a_k}, \qquad K_F = \left(\sum_{j=1}^{d'} \|\boldsymbol{c}_j\|_2^2 \right)^{1/2}. \tag{43}$$

The scalar mean-value theorem implies

$$|\psi((z_i - b_k)/a_k) - \psi((z_i' - b_k)/a_k)| \le (G/a_k) \, |z_i - z_i'| .$$

Therefore

$$|F_j(\boldsymbol{z}) - F_j(\boldsymbol{z}')| \le \sum_i c_{j,i} \, |z_i - z_i'| \le \|\boldsymbol{c}_j\|_2 \, \|\boldsymbol{z} - \boldsymbol{z}'\|_2 , \tag{44}$$

$$\|\boldsymbol{F}(\boldsymbol{z}) - \boldsymbol{F}(\boldsymbol{z}')\|_2 \le K_F \, \|\boldsymbol{z} - \boldsymbol{z}'\|_2 . \tag{45}$$

When $\boldsymbol{F}$ is applied independently to the rows of two sequences, summing the squared form of equation (45) proves

$$\|\boldsymbol{F}(\boldsymbol{C}) - \boldsymbol{F}(\boldsymbol{C}')\|_F \le K_F \, \|\boldsymbol{C} - \boldsymbol{C}'\|_F . \tag{46}$$

Let $\boldsymbol{\lambda}_{b,1}, \boldsymbol{\lambda}_{b,2}$ and $\varepsilon_{b,1}, \varepsilon_{b,2}$ denote the two LayerNorm scales and floors in block b. Combining residual addition with equations (36), (39) and (46) gives

$$K_{C,b} = \frac{\|\boldsymbol{\lambda}_{b,1}\|_\infty}{\sqrt{\varepsilon_{b,1}}} (1 + K_{\mathrm{conv},b}), \tag{47}$$

$$K_{B,b} = \frac{\|\boldsymbol{\lambda}_{b,2}\|_\infty}{\sqrt{\varepsilon_{b,2}}} (1 + K_{F,b}) K_{C,b}. \tag{48}$$

At inference time, a stack of B blocks consequently satisfies

$$\|\boldsymbol{H}_B - \boldsymbol{H}'_B\|_F \le \left(\prod_{b=1}^{B} K_{B,b}\right) \|\boldsymbol{H}_0 - \boldsymbol{H}'_0\|_F \,. \tag{49}$$

For one variable, the predictor and projector in equations (23) and (24) add the factors $\|\boldsymbol{W}_h\|_2$ and $\|\boldsymbol{w}_o\|_2$. If the two forecasts use the same RevIN statistics and affine parameters, equation (33) then yields

$$\|\widehat{\boldsymbol{y}} - \widehat{\boldsymbol{y}}'\|_2 \le \frac{\sigma}{|\gamma + \delta_{\mathrm{R}}|} \|\boldsymbol{W}_h\|_2 \|\boldsymbol{w}_o\|_2 \left(\prod_{b=1}^{B} K_{B,b}\right) \|\boldsymbol{H}_0 - \boldsymbol{H}'_0\|_F \,. \tag{50}$$

This is a potentially loose, conditional inference-time sensitivity bound. In particular, it is not an accuracy guarantee and does not establish robustness when the raw input—and hence RevIN statistics—changes.

A.6 Odd-N dual grid and Mexican-Hat overlap

Suppose $N = 2r + 1 \ge 5$. Then $N_{\mathrm{w}} = r + 1$, $N_{\mathrm{n}} = r$, and equations (10) and (11) give

$$\Delta_{\mathrm{w}} = \frac{2S_g}{r}, \qquad \Delta_{\mathrm{n}} = \frac{2S_g - 2S_g/r}{r-1} = \frac{2S_g}{r} = \Delta_{\mathrm{w}}.$$

It follows that

$$b^{(0)}_{\mathrm{n},j} = -S_g + \left(j + \frac{1}{2}\right)\Delta_{\mathrm{w}} = \frac{1}{2}\left(b^{(0)}_{\mathrm{w},j} + b^{(0)}_{\mathrm{w},j+1}\right), \tag{51}$$

so every narrow center is exactly the midpoint of an adjacent wide pair. This identity is not generally true for even N under equations (11) and (12).

Proof of Proposition 3.3. At the midpoint, the standardized distance from either center is $z = \Delta/(2a) = 1/(2c)$. Requiring positive-lobe response $\rho \in (0, 1)$ gives

$$(1 - z^2)e^{-z^2/2} = \rho, \qquad z \in (0, 1). \tag{52}$$

The left side decreases continuously from 1 to 0 on this interval because its derivative is $z(z^2 - 3)e^{-z^2/2} < 0$; the solution is unique. Set $y = 1 - z^2$. Rearrangement gives

$$\rho e^{1/2} = y e^{y/2}, \qquad \frac{\rho e^{1/2}}{2} = \frac{y}{2} e^{y/2}.$$

Thus $y = 2W_0(\rho e^{1/2}/2)$, $z = \sqrt{1 - 2W_0(\rho e^{1/2}/2)}$, and substituting $c = (2z)^{-1}$ proves equation (28). □

Each of the two equally scaled atoms contributes ρ at the midpoint; their equally weighted sum is 2ρ. A feasible raw-scale initialization for equation (13) additionally requires $a^{(0)} > a_{\min}$ and

$$e_a^{(0)} = \mathrm{softplus}^{-1}\left(a^{(0)} - a_{\min}\right). \tag{53}$$

The overlap derivation does not select ρ_a, does not establish an optimal grid size or range, and does not apply unchanged to Morlet or DoG.

B Experimental Details and Full Results

B.1 Dataset statistics

The original description groups these 12 series into six families. For Bike Sharing, only the five real-valued features are retained. All preprocessing and splitting are chronological; the explicitly reported ratios are 6:2:2 for ETT and 7:1:2 for other datasets. Complete dataset statistics are summarized in Table 3.

Table 3: Dataset statistics used in the reported evaluation. N is the number of time steps and M is the number of retained variables.

Dataset	Frequency	N	M	Observed interval
ETTh1	1 hour	17,420	7	2016-07-01 00:00 to 2018-06-26 19:00
ETTh2	1 hour	17,420	7	2016-07-01 00:00 to 2018-06-26 19:00
ETTm1	15 minutes	69,680	7	2016-07-01 00:00 to 2018-06-26 19:45
ETTm2	15 minutes	69,680	7	2016-07-01 00:00 to 2018-06-26 19:45
Weather	10 minutes	52,696	21	2020-01-01 00:10 to 2021-01-01 00:00
Exchange	1 day	7,588	8	1990-01-01 00:00 to 2010-10-10 00:00
Aotizhongxin	1 hour	35,064	11	2013-03-01 00:00 to 2017-02-28 23:00
Changping	1 hour	35,064	11	2013-03-01 00:00 to 2017-02-28 23:00
Dingling	1 hour	35,064	11	2013-03-01 00:00 to 2017-02-28 23:00
Guanyuan	1 hour	35,064	11	2013-03-01 00:00 to 2017-02-28 23:00
Appliances Energy	10 minutes	19,735	28	2016-01-11 17:00 to 2016-05-27 18:00
Bike Sharing	1 hour	17,379	5	2011-01-01 00:00 to 2012-12-31 23:00

Table 4: Full reported MSE comparison (part 1). Lower is better; best values are bold and second-best values are underlined. Ties receive the same rank.

Dataset	T	DLinear	PatchTST	TimesNet	RWKV-TS	TimeMixer	iTrans.	TimeKAN	SEMPO	HaKAN	CAW-KAN
ETTh1	96	0.367	0.386	0.429	0.395	0.387	0.398	0.364	0.389	0.365	**0.360**
	192	0.403	0.460	0.490	0.447	0.419	0.431	0.399	0.402	**0.398**	0.402
	336	0.431	0.473	0.479	0.515	0.471	0.443	0.430	0.419	**0.423**	0.435
	720	0.472	0.689	0.570	0.620	0.482	0.522	**0.436**	0.464	0.445	0.459
ETTh2	96	0.303	0.316	0.394	0.337	0.282	0.304	0.280	0.278	**0.254**	0.269
	192	0.394	0.388	0.401	0.405	0.352	0.376	0.328	0.350	**0.310**	0.342
	336	0.490	0.427	0.404	0.387	0.377	0.418	0.386	0.358	**0.320**	0.373
	720	0.811	0.474	0.467	0.433	0.438	0.441	0.433	0.398	**0.390**	0.398
ETTm1	96	0.304	0.296	0.355	0.300	0.318	0.310	0.295	0.304	0.300	**0.288**
	192	0.336	0.357	0.399	0.348	0.360	0.348	0.329	0.338	0.334	**0.319**
	336	0.367	0.374	0.410	0.387	0.429	0.380	0.363	0.374	0.359	**0.351**
	720	0.420	0.445	0.464	0.451	0.511	0.438	0.420	0.427	0.413	**0.405**
ETTm2	96	0.168	0.176	0.191	0.181	0.175	0.180	0.167	0.174	0.169	**0.163**
	192	0.226	0.246	0.259	0.251	0.225	0.242	0.223	0.234	0.224	**0.219**
	336	0.299	0.302	0.315	0.313	0.282	0.292	0.275	0.290	**0.264**	0.271
	720	0.415	0.395	0.396	0.399	0.364	0.373	0.375	0.368	**0.341**	0.363
Weather	96	0.170	0.156	0.159	0.150	0.147	0.171	**0.146**	0.151	0.149	**0.146**
	192	0.214	0.222	0.224	0.198	0.200	0.213	**0.187**	0.194	0.193	0.189
	336	0.260	0.263	0.275	0.256	0.247	0.266	**0.238**	0.245	0.243	0.239
	720	0.322	0.336	0.347	0.329	0.324	0.339	**0.309**	0.322	0.311	0.313
Exchange	96	0.096	0.098	0.221	0.122	0.088	0.105	0.093	0.095	0.089	**0.088**
	192	0.186	0.220	0.373	0.238	0.206	0.210	0.183	0.208	0.191	**0.182**
	336	**0.271**	0.380	0.668	0.532	0.375	0.409	0.366	0.373	0.378	0.389
	720	1.003	1.347	1.465	1.329	1.016	1.018	**0.957**	1.124	1.031	1.016

B.2 Full reported forecasting comparison

This section provides the complete forecasting results across all evaluated datasets, horizons, and baselines. Specifically, Tables 4, 5, 6, and 7 present the full metric values that correspond to the representative comparison summarized in Table 1 of Section 4.1.

B.3 Sensitivity analyses

Table 8 presents the complete sensitivity analysis across all evaluated datasets and hyperparameter configurations, supplementing the summary provided in Section 4.4.

B.4 Reported resource measurements

Table B.4 details the parameter counts and MACs. Additionally, empirical profiling shows CAW-KAN achieves the lowest peak memory on ETTh2, ETTm1, and ETTm2, and is the fastest on four of the five datasets, complementing the efficiency claims in Section 4.4.

Table 5: Full reported MSE comparison (part 2). Lower is better; best values are bold and second-best values are underlined. Ties receive the same rank.

Dataset	T	DLinear	PatchTST	TimesNet	RWKV-TS	TimeMixer	iTrans.	TimeKAN	SEMPO	HaKAN	CAW-KAN
AQAotizhongxin	96	0.639	0.681	0.682	0.710	0.682	0.649	0.634	0.650	0.637	0.637
	192	0.678	0.733	0.773	0.813	0.763	0.692	0.676	0.695	0.682	**0.675**
	336	0.705	0.749	0.740	0.837	0.784	0.709	0.705	0.718	0.703	**0.704**
	720	0.760	0.784	0.766	0.899	0.783	0.773	0.763	0.773	0.767	0.770
AQChangping	96	0.639	0.596	0.615	0.633	0.611	0.649	0.571	0.579	0.571	0.578
	192	0.678	0.633	0.695	0.672	0.705	0.692	0.609	0.616	0.612	**0.608**
	336	0.701	0.652	0.660	0.722	0.699	0.709	**0.632**	0.639	0.634	0.638
	720	0.760	0.710	**0.660**	0.747	0.720	0.773	0.682	0.693	0.697	0.684
AQDingling	96	0.639	0.672	0.683	0.734	0.718	0.652	0.643	0.650	0.640	**0.631**
	192	0.691	0.741	0.778	0.812	0.842	0.703	0.697	0.701	0.698	**0.684**
	336	0.726	0.769	0.786	0.856	0.828	0.736	0.726	0.733	0.729	**0.724**
	720	0.799	0.822	0.803	0.948	0.850	0.811	**0.794**	0.799	0.803	**0.794**
AQGuanyuan	96	0.678	0.703	0.766	0.768	0.752	0.685	0.667	0.689	0.677	**0.636**
	192	0.723	0.765	0.786	0.844	0.832	0.731	0.723	0.739	0.729	**0.689**
	336	0.747	0.784	0.810	0.890	0.848	0.755	0.747	0.764	0.750	**0.723**
	720	0.811	0.834	0.815	0.921	0.848	0.825	0.808	0.829	0.822	**0.796**
Appliances energy	96	0.485	0.448	0.582	0.471	0.500	0.449	0.425	0.435	0.460	**0.420**
	192	0.649	0.624	0.877	0.672	0.584	0.626	0.575	0.610	0.636	**0.574**
	336	0.856	0.825	0.988	0.906	0.834	0.835	0.804	0.834	0.850	**0.803**
	720	1.374	1.280	1.415	1.374	1.267	1.317	1.269	1.320	**1.240**	1.259
Bike sharing	96	0.503	0.538	0.513	0.511	0.524	0.537	0.494	0.519	0.499	**0.483**
	192	0.547	0.538	0.574	0.568	0.575	0.597	0.559	0.572	**0.536**	0.547
	336	**0.590**	0.646	0.638	0.620	0.640	0.654	0.615	0.620	0.605	0.613
	720	**0.720**	0.786	0.818	0.768	0.773	0.792	0.786	0.754	0.785	0.758

Table 6: Full reported MAE comparison (part 1). Lower is better; best values are bold and second-best values are underlined. Ties receive the same rank.

Dataset	T	DLinear	PatchTST	TimesNet	RWKV-TS	TimeMixer	iTrans.	TimeKAN	SEMPO	HaKAN	CAW-KAN
ETTh1	96	0.396	0.414	0.450	0.417	0.415	0.423	0.397	0.412	0.395	**0.388**
	192	0.420	0.466	0.490	0.453	0.435	0.445	0.415	0.422	**0.412**	**0.412**
	336	0.442	0.470	0.484	0.490	0.475	0.456	0.434	0.435	**0.421**	0.431
	720	0.493	0.587	0.536	0.542	0.482	0.521	**0.457**	0.472	0.462	0.462
ETTh2	96	0.368	0.372	0.425	0.380	0.350	0.359	0.343	0.339	**0.323**	0.331
	192	0.426	0.418	0.433	0.420	0.392	0.405	0.382	0.384	**0.364**	0.378
	336	0.489	0.452	0.444	0.420	0.420	0.435	0.432	0.400	**0.382**	0.402
	720	0.638	0.489	0.480	0.456	0.458	0.463	0.455	0.438	0.431	**0.430**
ETTm1	96	0.347	0.351	0.385	0.355	0.364	0.364	0.351	0.354	0.354	**0.341**
	192	**0.366**	0.395	0.414	0.387	0.391	0.387	0.372	0.374	0.371	**0.360**
	336	0.386	0.402	0.426	0.410	0.435	0.405	0.397	0.395	0.385	**0.382**
	720	0.417	0.447	0.462	0.445	0.472	0.440	0.427	0.424	**0.412**	**0.412**
ETTm2	96	0.263	0.268	0.277	0.271	0.267	0.272	0.257	0.259	0.259	**0.249**
	192	0.306	0.309	0.325	0.318	0.298	0.313	0.298	0.300	0.296	**0.288**
	336	0.362	0.351	0.358	0.355	0.339	0.345	0.329	0.337	0.323	**0.322**
	720	0.438	0.407	0.412	0.413	0.388	0.394	0.395	0.387	**0.373**	0.382
Weather	96	0.230	0.212	0.214	0.201	0.199	0.221	0.198	0.201	0.200	**0.189**
	192	0.272	0.274	0.272	0.245	0.250	0.259	0.239	0.243	0.243	**0.230**
	336	0.311	0.300	0.307	0.290	0.285	0.296	0.280	0.282	0.284	**0.270**
	720	0.363	0.347	0.354	0.340	0.339	0.344	0.336	0.336	0.336	**0.325**
Exchange	96	0.228	0.224	0.353	0.248	0.210	0.230	0.215	0.221	0.217	**0.213**
	192	**0.307**	0.341	0.457	0.352	0.332	0.330	0.310	0.327	0.312	0.325
	336	**0.397**	0.457	0.635	0.526	0.449	0.473	0.441	0.443	0.450	0.446
	720	0.749	0.854	0.924	0.855	0.749	0.765	**0.734**	0.788	0.778	0.771

B.5 Sensitivity of Baselines to the Log-Cosh Objective.

In section 4.3, we observed that replacing the standard MSE loss with the Log-Cosh objective significantly improves CAW-KAN's average performance. To determine whether this optimization benefit generalizes across different architectural families or is specific to our design, we evaluate two representative baselines: DLinear and TimeKAN which were trained with the same Log-Cosh loss with three different seeds (2021, 2022, and 2023). The results are presented in the table 10. Notably, these multi-seed evaluations demonstrate that CAW-KAN achieves highly competitive performance

Table 7: Full reported MAE comparison (part 2). Lower is better; best values are bold and second-best values are underlined. Ties receive the same rank.

Dataset	T	DLinear	PatchTST	TimesNet	RWKV-TS	TimeMixer	iTrans.	TimeKAN	SEMPO	HaKAN	CAW-KAN
AQAotizhongxin	96	0.474	0.482	0.485	0.496	0.485	0.477	0.459	0.473	0.464	**0.450**
	192	0.496	0.517	0.523	0.538	0.519	0.497	0.484	0.498	0.484	**0.473**
	336	0.512	0.536	0.511	0.550	0.527	0.507	0.498	0.508	0.500	**0.489**
	720	0.543	0.529	0.518	0.575	0.535	0.536	0.523	0.532	0.528	**0.515**
AQChangping	96	0.474	0.462	0.473	0.481	0.474	0.477	0.452	0.459	0.452	**0.441**
	192	0.496	0.495	0.513	0.506	0.519	0.497	0.472	0.484	0.472	**0.459**
	336	0.512	0.504	0.495	0.522	0.517	0.507	0.486	0.492	0.486	**0.475**
	720	0.543	0.520	**0.495**	0.539	0.528	0.536	0.503	0.513	0.513	0.496
AQDingling	96	0.481	0.488	0.483	0.509	0.501	0.483	0.468	0.478	0.469	**0.452**
	192	0.505	0.519	0.521	0.543	0.549	0.504	0.492	0.504	0.493	**0.479**
	336	0.523	0.537	0.528	0.556	0.547	0.520	0.510	0.517	0.512	**0.495**
	720	0.557	0.554	0.534	0.587	0.558	0.556	0.540	0.540	0.539	**0.525**
AQGuanyuan	96	0.487	0.492	0.512	0.523	0.508	0.491	0.477	0.488	0.478	**0.455**
	192	0.511	0.527	0.531	0.549	0.541	0.511	0.500	0.513	0.500	**0.479**
	336	0.526	0.547	0.535	0.565	0.550	0.524	0.512	0.523	0.517	**0.501**
	720	0.557	0.546	0.533	0.578	0.557	0.553	0.539	0.548	0.544	**0.522**
Appliances energy	96	0.505	0.454	0.557	0.474	0.501	0.465	0.436	0.448	0.469	**0.430**
	192	0.601	0.558	0.688	0.592	0.536	0.569	0.533	0.556	0.574	**0.525**
	336	0.698	0.659	0.748	0.697	0.661	0.669	0.639	0.661	0.670	**0.636**
	720	0.908	0.845	0.905	0.875	0.845	0.868	0.841	0.861	0.834	**0.827**
Bike sharing	96	0.521	0.541	0.519	0.527	0.529	0.534	0.512	0.527	0.517	**0.501**
	192	0.547	0.541	0.557	0.560	0.561	0.566	0.553	0.563	0.542	**0.541**
	336	**0.571**	0.609	0.589	0.592	0.597	0.594	0.583	0.591	0.582	0.580
	720	**0.636**	0.679	0.673	0.665	0.669	0.662	0.661	0.660	0.679	0.658

Table 8: Sensitivity analysis of the grid range, number of wavelets, CP factorization rank, temporal context kernel size, number of encoder blocks, and wavelet basis function. Lower is better; bold marks the best value and underline marks the second best within each analysis group and dataset.

Analysis	Setting	ETTh1		ETTh2		ETTm1		ETTm2	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Grid range S_g	1.5	0.422	0.427	0.351	0.392	0.350	0.378	0.258	0.314
	3.0	0.413	0.423	**0.351**	0.390	**0.342**	**0.374**	0.257	0.313
	4.5	0.414	0.423	**0.351**	0.390	0.346	0.376	0.257	0.313
	6.0	0.413	0.424	0.351	0.390	0.344	0.376	0.256	0.311
	7.5	**0.411**	**0.421**	0.351	**0.390**	0.343	0.375	**0.254**	**0.310**
Number of wavelets N	4	0.412	0.422	0.345	0.387	0.347	0.376	0.253	0.310
	6	0.415	0.424	0.347	0.386	0.352	0.380	**0.252**	**0.309**
	8	**0.410**	**0.421**	0.346	**0.386**	**0.343**	**0.375**	0.253	0.311
	10	0.414	0.423	**0.345**	0.387	0.347	0.376	0.255	0.312
	12	0.415	0.425	0.350	0.389	0.345	0.375	0.254	0.311
CP factorization rank R	2	0.416	**0.424**	**0.347**	**0.386**	0.346	**0.376**	**0.252**	0.310
	4	0.418	**0.424**	0.350	0.388	**0.342**	**0.374**	0.254	0.311
	8	**0.414**	**0.424**	**0.347**	**0.386**	0.346	0.377	**0.252**	**0.309**
	16	0.418	0.426	0.350	0.389	0.348	0.378	0.254	0.311
	32	0.416	0.425	0.353	0.390	0.347	0.377	0.256	0.314
Temporal context kernel size k	1	0.427	0.429	0.349	0.389	0.354	0.377	0.254	0.312
	3	0.414	**0.423**	**0.345**	**0.384**	0.344	**0.374**	0.255	0.312
	7	**0.411**	0.426	0.346	0.386	**0.342**	0.374	**0.253**	**0.311**
	15	0.418	0.429	0.348	0.386	0.351	0.380	0.257	0.312
Blocks B	1	0.419	0.426	0.351	0.391	0.342	0.375	0.254	0.312
	2	0.415	**0.422**	**0.346**	**0.386**	**0.342**	**0.374**	**0.253**	**0.310**
	3	**0.414**	0.423	0.350	0.389	0.343	0.376	0.256	0.312
	4	0.427	0.428	0.350	0.389	0.343	0.374	0.254	0.310
Wavelet basis function	DoG	0.417	0.426	0.350	0.388	0.345	0.376	**0.254**	**0.311**
	Morlet	**0.416**	**0.424**	**0.346**	**0.386**	**0.345**	**0.376**	0.255	0.312

compared to HaKAN. Although HaKAN maintains a marginal lead on certain datasets, CAW-KAN delivers comparable accuracy while operating with significantly fewer parameters and MACs.

Table 9: Reported parameters and MACs at $L = 512$, $T = 96$, and batch size 1. The original export does not specify which operators are included in MACs; these counts are reproduced without using them for full-model complexity claims. Lower is better; bold marks the best value and underline marks the second best.

	ETTh1		ETTh2		ETTm1		ETTm2		Weather	
Model	Params	MAC	Params	MAC	Params	MAC	Params	MAC	Params	MAC
TimesNet	898.759K	1.788G	1.485M	3.699G	5.001M	14.300G	1.485M	3.807G	1.487M	3.589G
PatchTST	6.307M	1.440G	12.612M	4.266G	6.307M	1.440G	12.612M	4.266G	9.460M	8.560G
TimeKAN	54.937K	**24.354M**	70.809K	71.122M	60.761K	40.353M	93.721K	124.681M	60.873K	121.059M
HaKAN	3.213M	243.091M	3.213M	243.091M	3.213M	243.091M	3.213M	243.091M	3.220M	729.272M
CAW-KAN	**54.047K**	28.143M	**54.927K**	**28.029M**	**54.991K**	**28.029M**	**54.991K**	**28.029M**	**52.651K**	**48.303M**

Table 11: Aggregated test-set activation overflow rates (mean ± std) across three random seeds.

Dataset	**Average Overflow Rate (%)**
ETTh1	0.325 ± 0.578
ETTh2	0.112 ± 0.280
ETTm1	1.454 ± 1.511
ETTm2	0.268 ± 0.614

Table 10: Multi-seed evaluation results (mean ± sample std). Lower is better. Best results are highlighted in bold and second best results are underlined. All evaluated models in this comparison are trained using the Log-Cosh loss.

Dataset	T	CAW-KAN		DLinear		TimeKAN		HaKAN	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.363 ± 0.004	0.390 ± 0.002	0.364 ± 0.002	**0.389 ± 0.002**	**0.359 ± 0.004**	0.390 ± 0.002	0.361 ± 0.005	**0.389 ± 0.007**
	192	0.402 ± 0.003	0.413 ± 0.002	0.403 ± 0.003	0.416 ± 0.004	**0.396 ± 0.003**	0.411 ± 0.002	0.399 ± 0.008	**0.410 ± 0.004**
	336	0.432 ± 0.003	0.433 ± 0.002	0.431 ± 0.002	0.436 ± 0.002	0.417 ± 0.006	0.425 ± 0.002	**0.402 ± 0.006**	**0.421 ± 0.009**
	720	0.462 ± 0.008	0.463 ± 0.004	0.472 ± 0.005	0.492 ± 0.004	0.460 ± 0.023	0.465 ± 0.013	**0.443 ± 0.005**	**0.458 ± 0.007**
ETTh2	96	0.272 ± 0.004	0.333 ± 0.002	0.275 ± 0.005	0.339 ± 0.005	0.278 ± 0.005	0.338 ± 0.002	**0.259 ± 0.004**	**0.325 ± 0.006**
	192	0.340 ± 0.003	0.376 ± 0.005	0.360 ± 0.008	0.399 ± 0.007	0.345 ± 0.012	0.384 ± 0.005	**0.317 ± 0.008**	**0.365 ± 0.005**
	336	0.371 ± 0.005	0.402 ± 0.002	0.443 ± 0.010	0.455 ± 0.007	0.371 ± 0.009	0.411 ± 0.007	**0.320 ± 0.007**	**0.378 ± 0.009**
	720	0.397 ± 0.005	0.429 ± 0.003	0.638 ± 0.086	0.561 ± 0.035	0.417 ± 0.028	0.449 ± 0.017	**0.387 ± 0.004**	**0.427 ± 0.003**
ETTm1	96	**0.289 ± 0.002**	**0.341 ± 0.002**	0.305 ± 0.002	0.350 ± 0.003	0.293 ± 0.004	0.345 ± 0.001	0.298 ± 0.006	0.349 ± 0.008
	192	**0.326 ± 0.009**	**0.363 ± 0.004**	0.339 ± 0.002	0.372 ± 0.003	0.329 ± 0.006	0.366 ± 0.003	0.336 ± 0.005	0.369 ± 0.004
	336	**0.355 ± 0.003**	**0.383 ± 0.001**	0.368 ± 0.001	0.387 ± 0.002	0.359 ± 0.005	0.386 ± 0.002	0.363 ± 0.010	0.386 ± 0.007
	720	**0.404 ± 0.003**	**0.413 ± 0.002**	0.422 ± 0.001	0.420 ± 0.002	0.417 ± 0.010	0.421 ± 0.005	0.411 ± 0.006	0.418 ± 0.004
ETTm2	96	**0.163 ± 0.002**	**0.249 ± 0.001**	0.168 ± 0.002	0.262 ± 0.002	0.198 ± 0.051	0.259 ± 0.002	0.165 ± 0.006	0.255 ± 0.004
	192	**0.219 ± 0.001**	**0.289 ± 0.001**	0.229 ± 0.002	0.310 ± 0.001	0.249 ± 0.043	0.298 ± 0.005	0.220 ± 0.009	0.290 ± 0.005
	336	**0.272 ± 0.001**	**0.325 ± 0.001**	0.296 ± 0.010	0.355 ± 0.010	0.296 ± 0.029	0.332 ± 0.003	0.274 ± 0.003	0.326 ± 0.008
	720	0.360 ± 0.005	0.380 ± 0.004	0.428 ± 0.049	0.438 ± 0.030	0.364 ± 0.003	0.386 ± 0.002	**0.356 ± 0.007**	**0.378 ± 0.006**

B.6 Empirical Overflow Analysis

To explicitly validate the dual-grid initialization discussed in Proposition 3.3 and the distribution coverage formalized in Lemma 3.1, we tracked the pre-KAN activation overflow rate (Equation 15) during inference.

A critical concern when fixing the dictionary range to $[-S_g, S_g]$ is the potential for latent activations to fall outside this bounded interval, especially under real-world distribution shifts. To assess this, we measured the exact percentage of latent activations exceeding the grid bounds on the test sets of the ETT benchmarks. To ensure rigorous uncertainty quantification, all evaluations were conducted across three random initialization seeds (2021, 2022, and 2023).

As summarized in Table 11 and detailed in Table 12, the empirical overflow rates are consistently negligible. The average overflow rate remains well below 1.5% across all datasets, with ETTh2 and ETTm2 demonstrating exceptionally stable coverage (averaging 0.1117% and 0.2678%, respectively). These measurements provide strong empirical evidence that our chosen initialization bounds successfully cover the vast majority of the latent value distribution. This ensures stable dictionary conditioning, prevents gradient explosion outside the local support of the wavelets, and verifies the practical relevance of the midpoint response calibration proposed in our design.

Table 12: Detailed pre-KAN activation overflow rates across prediction horizons and random seeds.

Dataset	T	Seed 2021	Seed 2022	Seed 2023	Overflow Rate (mean ± std)
ETTh1	96	0.003%	0.001%	0.111%	**0.038%** ± 0.052%
	192	0.000%	0.000%	0.712%	0.237% ± 0.335%
	336	0.000%	0.000%	1.329%	0.443% ± 0.627%
	720	0.000%	0.010%	1.728%	0.579% ± 0.813%
ETTh2	96	0.001%	0.000%	0.000%	**0.001%** ± 0.001%
	192	0.000%	0.340%	0.006%	0.116% ± 0.159%
	336	0.001%	0.001%	0.004%	0.002% ± 0.001%
	720	0.988%	0.000%	0.000%	0.329% ± 0.466%
ETTm1	96	3.915%	0.554%	2.706%	2.392% ± 1.390%
	192	0.063%	4.105%	2.871%	2.346% ± 1.691%
	336	0.311%	0.057%	2.043%	0.804% ± 0.883%
	720	0.012%	0.643%	0.169%	**0.275%** ± 0.268%
ETTm2	96	0.004%	0.039%	0.989%	0.344% ± 0.456%
	192	0.000%	0.000%	0.001%	**0.001%** ± 0.001%
	336	0.000%	0.001%	0.033%	0.012% ± 0.015%
	720	0.007%	2.097%	0.042%	0.716% ± 0.977%

B.7 Visualization of Wavelets KAN

Initialization. The figure 2 shows the original state of the dual-grid wavelet dictionary. Every wavelet is normalized to a peak amplitude of 1.0, with a symmetric shape based on the wide and narrow grids. This panel acts as a baseline. The remaining figures show how the model adapts this initial dictionary after training on data.

Adapting amplitude. In Figures 3, 4, 5, 6, the plotted curves represent the individual fixed-center basis functions multiplied by their effective learned scalar weights derived from the CP-factorized layer. Because the original centers and scales remain strictly fixed after initialization, the individual basis functions do not shift horizontally or change their fundamental shape. Instead, the model adapts entirely through vertical scaling and inversion (e.g., a negative weight inverts a central peak into a valley). The relative amplitude of these scaled curves demonstrates how the network selectively amplifies or suppresses specific regions of the latent value space to fit the unique characteristics of each dataset.

Dataset Adaptation. Comparing the hourly datasets (ETTh1, ETTh2) with the 15-minute datasets (ETTm1, ETTm2) shows that the weight allocation adapts to dataset characteristics. At a forecast horizon of 96, wavelet 5 is most important for ETTh1, while wavelet 7 dominates for ETTm1. This adaptation is enabled by CP factorization, which directly trains the factors to flexibly adjust the importance of each basis function in the latent value space, allowing Wavelet KAN to better capture dataset-specific patterns.

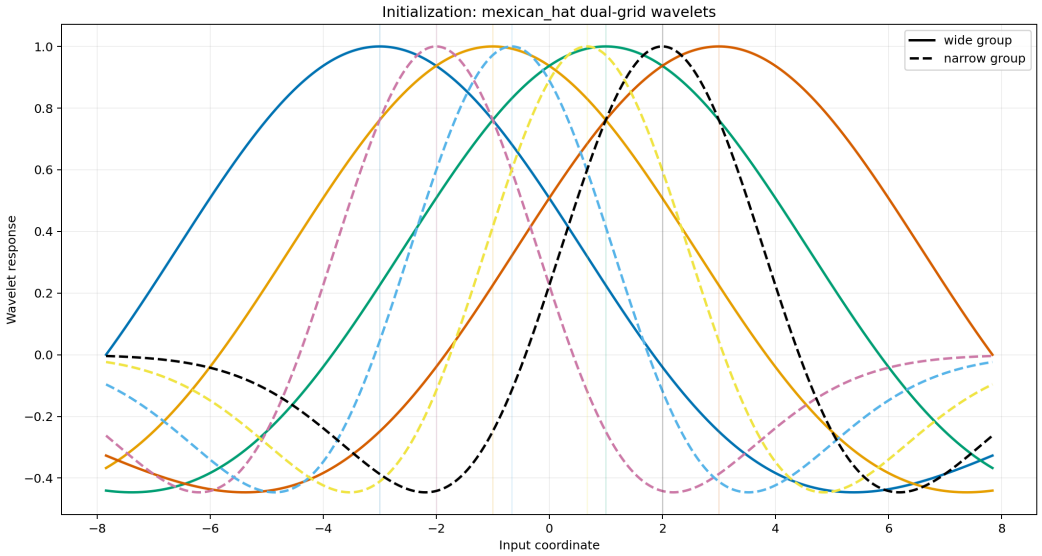

Figure 2: Initialization of the mexican_hat dual-grid wavelets

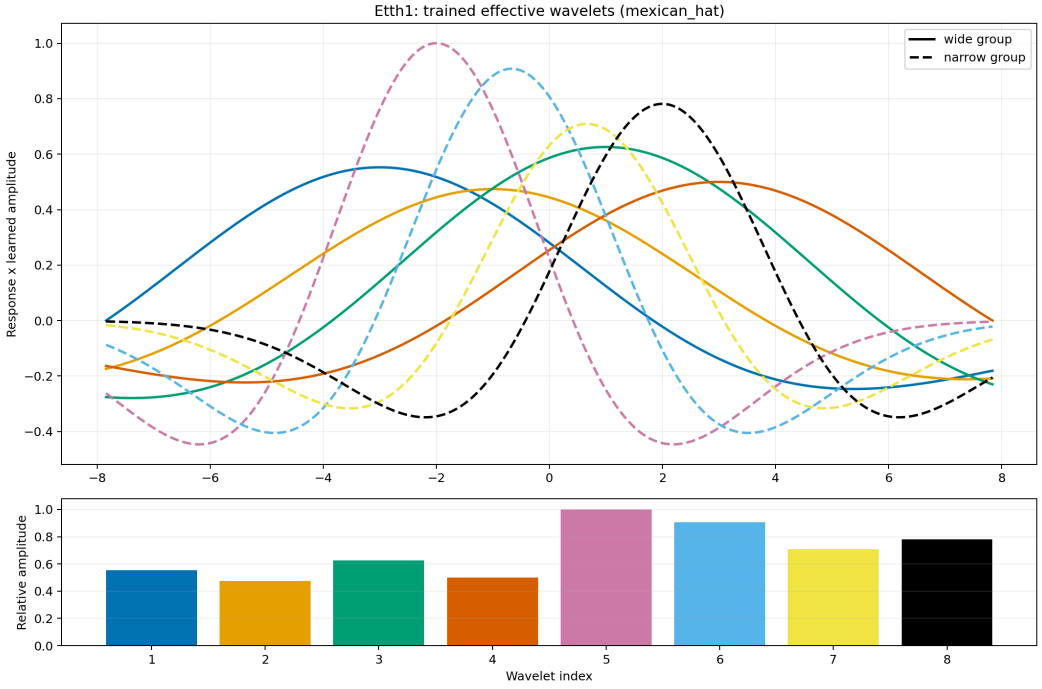

Figure 3: Trained effective wavelets and relative amplitudes on the ETTh1 dataset (prediction length = 96)

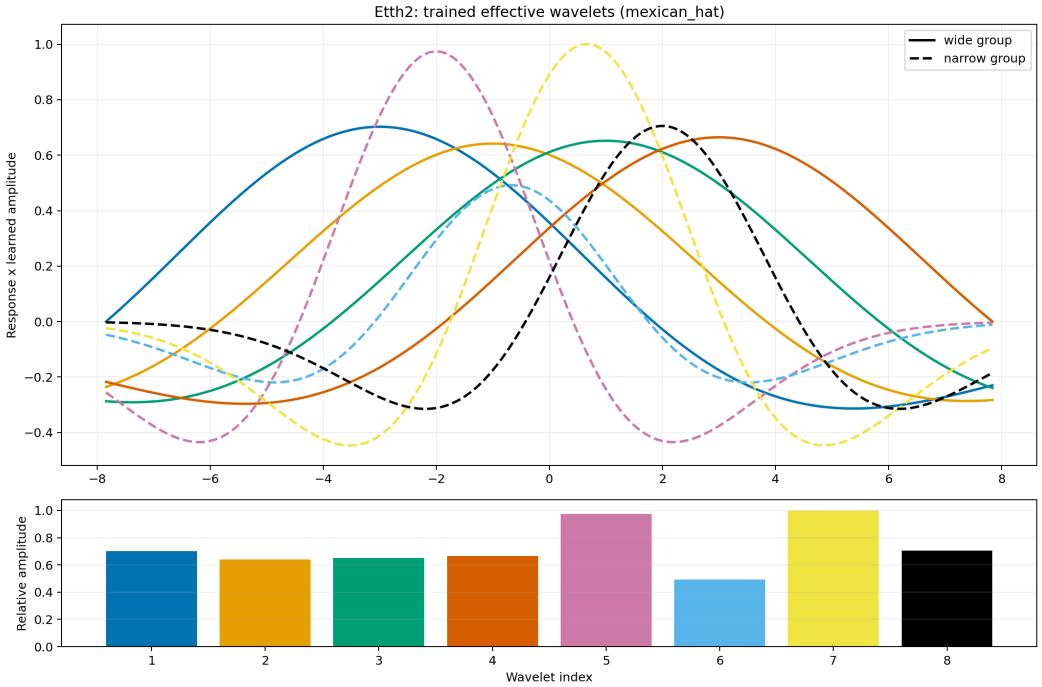

Figure 4: Trained effective wavelets and relative amplitudes on the ETTh2 dataset (prediction length = 96)

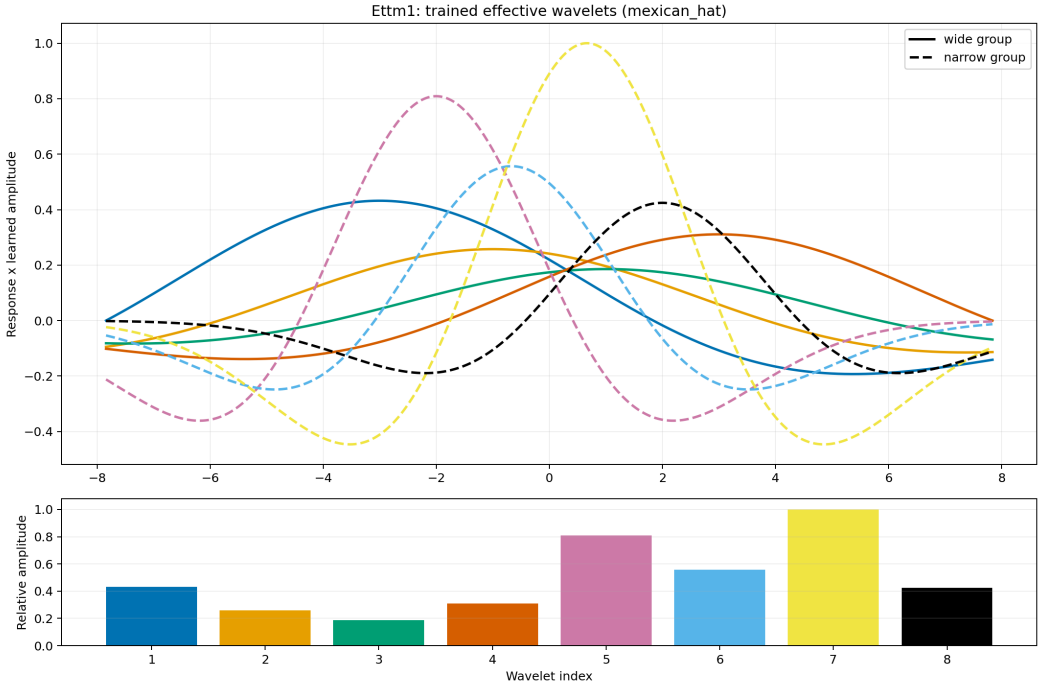

Figure 5: Trained effective wavelets and relative amplitudes on the ETTm1 dataset (prediction length = 96)

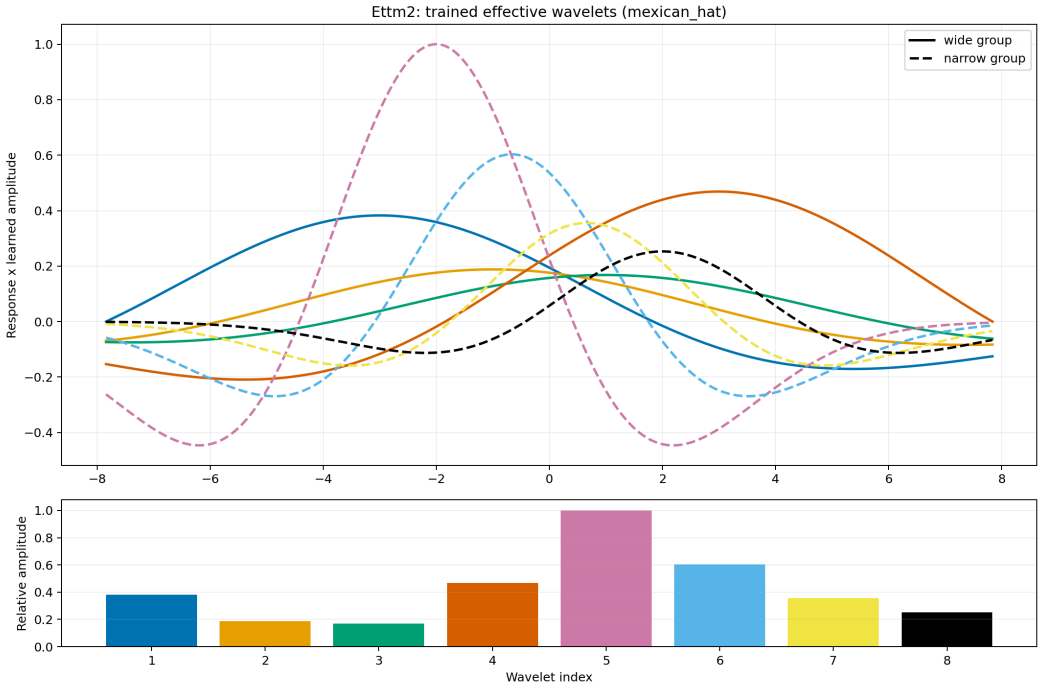

Figure 6: Trained effective wavelets and relative amplitudes on the ETTm2 dataset (prediction length = 96)