Interpretable Spatial-Temporal Forecasting via Additive Neural Decomposition and Knowledge Distillation

Suan Lee[1] **Jinho Kim**[2]

[1]School of Computer Science, Semyung University, Jecheon, Republic of Korea
[2]AI Research Lab, Korea Information Systems Consulting & Audit Co., Ltd., Republic of Korea

Abstract

We challenge the prevailing assumption that interpretability requires sacrificing accuracy in spatial-temporal traffic forecasting. We propose STGNAM (Spatial-Temporal Graph Neural Additive Model), which enforces a strict additive decomposition: $\hat{\mathbf{y}} = f_{\text{node}} + f_{\text{temporal}} + f_{\text{spatial}} + f_{\text{interact}} + \text{bias}$, where each component is independently evaluable and visualizable. Our central finding is that the additive inductive bias acts as an implicit regularizer: on PEMS-BAY, STGNAM *sets a new state of the art* among interpretable and black-box models alike (3-seed MAE 1.622 ± 0.002, surpassing TITAN at 1.69 with $p < 10^{-3}$), and surpasses its own D2STGNN teacher (1.87)—even without knowledge distillation (KD). When combined with KD from a black-box teacher, STGNAM achieves 91–104% of best black-box performance across the other four traffic benchmarks while providing full additive interpretability. KD also produces a secondary benefit: component-level faithfulness increases from 0.53 to 0.92 on METR-LA, and skip connection dominance drops from 65% to 38%, indicating that soft-target supervision forces additive components to learn structured, specialized representations. We further show that zeroing the four additive components on a trained STGNAM degrades MAE by 25–50% over the skip-only baseline across all datasets—the interpretable components are not a wrapper around a linear residual. Direct comparison with post-hoc attribution methods (gradient saliency, SmoothGrad) confirms that inherent additive explanations are more faithful and stable than post-hoc alternatives applied to black-box models.

1 INTRODUCTION

Traffic forecasting underpins intelligent transportation systems, from adaptive signal control to route planning and congestion management [Jiang and Luo, 2022]. Practitioners require not only accurate predictions but also understanding of *why* a model forecasts congestion—which spatial patterns propagate, what temporal cycles drive demand, and how node-specific characteristics contribute. Black-box spatial-temporal graph neural networks (STGNNs) [Li et al., 2018, Yu et al., 2018, Wu et al., 2019, Shao et al., 2022b, Liu et al., 2023] dominate accuracy benchmarks but provide no such guarantees: post-hoc interpretability methods like attention visualization and gradient saliency have been shown to be unfaithful to model behavior in spatial-temporal settings [Luo et al., 2023].

Neural Additive Models (NAMs) [Agarwal et al., 2021] provide inherent interpretability for tabular data through additive decomposition, where the prediction is an explicit sum of per-feature functions $\hat{y} = \sum_i f_i(x_i)$. Each function f_i is a univariate neural network whose output can be directly plotted and inspected. This design eliminates the need for post-hoc explanations—the model *is* the explanation. However, extending additive decomposition to graph-structured time series creates a tension: spatial-temporal dependencies are inherently interactive, yet additive constraints limit representational capacity. The conventional view holds that this "interpretability tax" is unavoidable.

We challenge this view. Our central finding is that the *additive inductive bias itself acts as an implicit regularizer* for spatial-temporal forecasting. By restricting the hypothesis space to decomposable functions, the additive constraint prevents overfitting and encourages structured representations—analogous to the regularization benefits observed in tabular NAMs [Agarwal et al., 2021]. On PEMS-BAY, our model *surpasses all black-box models* (3-seed mean MAE 1.622 ± 0.002 vs. TITAN 1.69; one-sample t-test $p < 10^{-3}$) while providing full additive interpretability, demonstrating that the interpretability–performance tradeoff is not inher-

Proceedings of the 42nd Conference on Uncertainty in Artificial Intelligence (UAI 2026), PMLR 337:3376–3397.

ent to the problem but an artifact of architectural choice. Knowledge distillation (KD) from a strong teacher further enhances this effect: KD not only improves accuracy on datasets where the teacher is strong, but also changes how the model distributes predictive capacity across components. With KD, component faithfulness jumps from 0.53 to 0.92 on METR-LA and skip connection dominance drops from 65% to 43%, indicating that soft-target supervision forces each component to specialize.

We make the following contributions:

- We propose STGNAM, the first strictly additive neural model for spatial-temporal graph forecasting. On PEMS-BAY, it achieves a new state of the art (1.622 ± 0.002 across 3 seeds), surpassing all black-box models at conventional statistical significance—demonstrating that additive inductive bias provides beneficial regularization rather than limiting performance.
- We show that knowledge distillation simultaneously improves prediction accuracy *and* component-level interpretability (faithfulness 0.53→0.92, skip ratio 65%→38%). Unlike prior work on KD and post-hoc explanations [Han et al., 2023, Gupta et al., 2023], this improvement arises in additive architectures without explicit explanation matching.
- We validate our interpretability claims through direct comparison with post-hoc attribution methods (gradient saliency, SmoothGrad [Smilkov et al., 2017]), showing that inherent additive decomposition produces substantially more faithful and stable explanations than post-hoc alternatives applied to black-box models.

2 RELATED WORK

Spatial-temporal graph forecasting. Early approaches combined graph convolutions with recurrence [DCRNN; Li et al., 2018] or temporal convolutions [STGCN; Yu et al., 2018]. GWNet [Wu et al., 2019] and MTGNN [Wu et al., 2020] introduced adaptive graph learning. Recent accuracy gains stem from decoupled architectures [D2STGNN; Shao et al., 2022b], transformers [STAEformer; Liu et al., 2023], and pre-training [STEP, TITAN; Shao et al., 2022a, Wang et al., 2024]. These models operate as black boxes, making interpretability an afterthought via post-hoc visualizations that can be misleading [Luo et al., 2023].

Neural additive models. NAMs [Agarwal et al., 2021] decompose tabular predictions additively for per-feature interpretability. Extensions include NodeGAM [Chang et al., 2022] for differentiable trees and GAMI-Net [Yang et al., 2021] for pairwise interactions. Kolmogorov-Arnold Networks (KANs) [Liu et al., 2025] generalize additive models with learnable activations on edges. We are the first to apply this paradigm to graph-structured time series, where spatial dependencies must be modeled within an additive framework.

Knowledge distillation and interpretability. Hinton et al. [2015] used soft targets from teacher networks to improve student training. KD has since been applied to model compression [Gou et al., 2021] and structured prediction [Kim and Rush, 2016]. Recent work explores connections between KD and interpretability: Han et al. [2023] show KD improves network dissection scores (concept alignment) in image classifiers, Gupta et al. [2023] leverage concept-based explanations to guide distillation, and Parchami-Araghi et al. [2024] match teacher–student explanations during training. In the spatial-temporal domain, Tang et al. [2023] use structure distillation for post-hoc STGNN explanation. Our work differs in a key respect: we show KD improves *component-level faithfulness* in an inherently additive architecture, without requiring explicit explanation transfer—the improvement arises from soft-target supervision alone.

Interpretable time series forecasting. N-BEATS [Oreshkin et al., 2020] and Temporal Fusion Transformers [Lim et al., 2021] provide partial interpretability but lack strict additive guarantees. For graph-structured data, post-hoc explanation methods such as GNNExplainer [Ying et al., 2019] and PGExplainer [Luo et al., 2020] identify important subgraphs but produce explanations that can be unfaithful to model behavior [Luo et al., 2023]. Recent work applies Kolmogorov-Arnold Networks to time series forecasting [Xu et al., 2024], but without the spatial-temporal graph structure or additive decomposition guarantees that STGNAM provides. For spatial-temporal data specifically, STGNAM achieves higher decomposition faithfulness than any prior work by enforcing interpretability through architecture rather than post-hoc attribution.

3 METHOD

3.1 PROBLEM FORMULATION

Given a spatial-temporal graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$ with $N = |\mathcal{V}|$ sensors and adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, we observe signals $\mathbf{x}^{(t)} \in \mathbb{R}^{N \times F}$ at each timestep. The task is to predict future values $\hat{\mathbf{y}}^{(t+1:t+H)} \in \mathbb{R}^{H \times N}$ given the past T observations $\mathbf{x}^{(t-T+1:t)}$.

STGNAM enforces a strict additive decomposition:

$$\hat{\mathbf{y}} = \underbrace{f_{\text{node}}(\mathbf{x})}_{\text{node}} + \underbrace{f_{\text{temporal}}(\mathbf{x})}_{\text{temporal}} + \underbrace{f_{\text{spatial}}(\mathbf{x}, \mathbf{A})}_{\text{spatial}} + \underbrace{f_{\text{interact}}(\mathbf{x}, \mathbf{A})}_{\text{interact}} + \underbrace{\text{Linear}(\mathbf{x})}_{\text{skip}} + \mathbf{b} \quad (1)$$

where each component $f_* : \mathbb{R}^{B \times T \times N \times F} \to \mathbb{R}^{B \times N \times H}$ produces per-node, per-horizon contributions that can be

independently evaluated and visualized. The linear skip connection provides a residual baseline, and $\mathbf{b}$ includes learnable node-specific and time-of-day-specific bias terms.

Two granularities of additive interpretability. STGNAM provides additivity at two complementary levels. *(i) Feature-level additivity inside the Node component* (NAM-style; Eq. 2, Fig. 3a): each node summary feature is passed through its own univariate B-Spline KAN shape function, exactly the per-feature visualization mechanism of Agarwal et al. [2021]. *(ii) Module-level additivity across the four spatio-temporal factors* (Eq. 1): graph-temporal tensors $\mathbf{x} \in \mathbb{R}^{T\times N\times F}$ admit no natural scalar decomposition that preserves graph and temporal structure, so the four modules differ by inductive bias (KAN shape functions; ToD-gated experts; K-hop additive aggregation; gated sparse interactions). The prediction is *exactly* the sum of these summands plus skip plus bias; both granularities are independently visualizable and ablatable. Section 4.3 provides quantitative evidence that each module's contribution scales with its semantically designated input axis (time-of-day for temporal; graph distance for spatial).

Figure 1 shows the STGNAM architecture. The input $\mathbf{x}$ contains the traffic signal and time-of-day features. Four component modules process the input, and their outputs are summed with learnable scales α_i to produce the final prediction.

3.2 COMPONENT 1: NODE FEATURES VIA B-SPLINE KAN

The node component captures per-sensor characteristics (e.g., highway vs. arterial road capacity). For each node v, we extract a 5-dimensional feature vector $\mathbf{z}_v = [\text{last}(v), \text{mean}(v), \text{trend}(v), \sin(\text{tod}), \cos(\text{tod})]$. These features are processed by Kolmogorov-Arnold Network (KAN) shape functions using B-spline bases:

$$\begin{aligned} f_{\text{node}}(\mathbf{z}_v) &= \textstyle\sum_{c=1}^{C} w_{v,c} \cdot g_c(\mathbf{z}_v) \\ g_c(\mathbf{z}) &= \textstyle\sum_j \beta_{c,j} B_j(\mathbf{z}) + \text{SiLU}(\mathbf{W}_{\text{base}}\mathbf{z}) \end{aligned} \tag{2}$$

where B_j are degree-3 B-spline basis functions, $\beta_{c,j}$ are coefficients, and $w_{v,c}$ are soft cluster assignments ($C = 16$). The SiLU connection stabilizes training. Each shape function is a 2-layer KAN, producing directly visualizable curves mapping features to contributions.

3.3 COMPONENT 2: EXPERT TEMPORAL DECOMPOSITION

The temporal component captures periodic patterns through a mixture-of-experts architecture with time-of-day gating.

Feature extraction. The raw signal $x_v^{(1:T)}$ passes through a dilated convolutional encoder $\mathbf{h} = \text{DilConv}(x_v^{(1:T)})$ with three layers achieving a receptive field of 13.

Lag-wise additive pooling. We compute per-horizon attention weights $\alpha_t^{(h)} = \text{softmax}(\text{MLP}(\mathbf{h}_t))_h$ to produce horizon-specific summaries $\tilde{\mathbf{h}}^{(h)} = \sum_{t=1}^{T} \alpha_t^{(h)} \mathbf{h}_t$.

Expert mixture with ToD gating. Three expert heads process the features, gated by time-of-day: $f_{\text{temporal}} = \sum_{e=1}^{3} \pi_e(\text{tod}) \cdot \text{Expert}_e(\tilde{\mathbf{h}})$ where $\pi_e(\text{tod}) = \text{softmax}(\mathbf{W}_g \text{Embed}(\text{tod}))_e$. Each expert is a two-layer MLP; the time-of-day gate implicitly routes inputs to capture trend, seasonal, and residual patterns. The gate pattern reveals which temporal regime dominates throughout the day.

3.4 COMPONENT 3: K-HOP ADDITIVE SPATIAL AGGREGATION

The spatial component models traffic propagation through a K-hop aggregation ($K = 6$) that preserves per-hop interpretability. We use exclusive hop masks $\mathbf{M}_k$ for nodes reachable at *exactly* distance k.

Dual-path aggregation. For each hop, we compute influence:

$$\begin{aligned} \mathbf{h}_k^{\text{abs}} &= f_k^{\text{abs}}\big(\textstyle\sum_{u\in\mathcal{N}_k(v)} \alpha_{v,u}\mathbf{h}_u\big) \\ \mathbf{h}_k^{\text{diff}} &= f_k^{\text{diff}}\big(\textstyle\sum_{u\in\mathcal{N}_k(v)} \alpha_{v,u}(\mathbf{h}_u - \mathbf{h}_v)\big) \end{aligned} \tag{3}$$

where f_k^{abs} captures absolute state and f_k^{diff} captures deviation. Learnable hop weights ω_k combine these into $f_{\text{spatial}}(v) = \text{MLP}(\sum_{k=1}^{K} \omega_k(\mathbf{h}_k^{\text{abs}}(v) + \mathbf{h}_k^{\text{diff}}(v)))$. We use dynamic adjacency modulated by time-of-day embeddings.

3.5 COMPONENT 4: SPARSE GATED INTERACTIONS

The interaction component captures cross-modal effects through gated element-wise products:

$$f_{\text{interact}} = \text{MLP}(\sigma(\mathbf{g}) \odot (\text{proj}_t(\mathbf{h}_{\text{temporal}}) \cdot \text{proj}_s(\mathbf{h}_{\text{spatial}}))) \tag{4}$$

where $\mathbf{g}$ are learnable gates enforcing sparsity. The gate activation pattern reveals which interaction terms are important.

3.6 ADDITIVE COMBINATION AND REGULARIZATION

The final prediction sums all components:

$$\begin{aligned} \hat{\mathbf{y}} = \textstyle\sum_{i\in\text{comps}} \text{softplus}(s_i) \cdot f_i + \mathbf{b}_{\text{node}} \\ + \mathbf{b}_{\text{tod}} + \text{softplus}(s_{\text{skip}}) \cdot \text{Linear}(\mathbf{x}) \end{aligned} \tag{5}$$

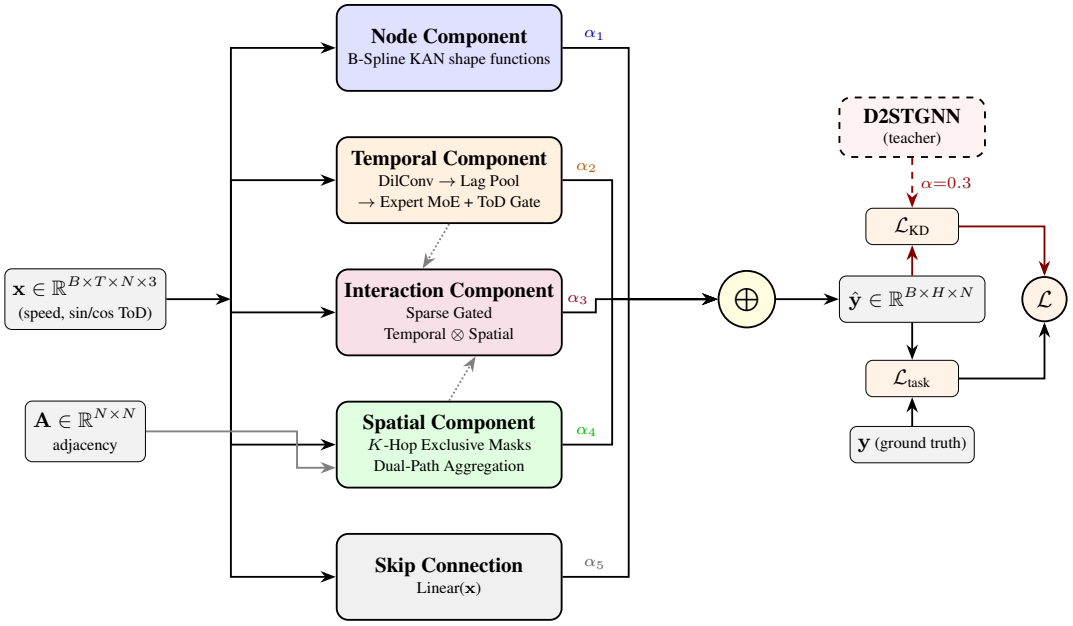


Figure 1: Architecture of STGNAM. The input is processed by four parallel interpretable components and a linear skip connection. Outputs are additively combined with learnable scales α_i to produce the final prediction. The interaction component receives intermediate representations from the temporal and spatial components (dotted). During training, knowledge distillation from D2STGNN provides additional supervision (dashed red).

Learnable softplus scales s_i control each component's contribution while preserving non-negativity, and the skip connection provides a linear residual baseline that captures short-term autocorrelation.

We apply four regularization terms to encourage meaningful decomposition:

$$\mathcal{L}_{\text{reg}} = \lambda_{\text{orth}}\mathcal{L}_{\text{orth}} + \lambda_{\text{var}}\mathcal{L}_{\text{var}} + \lambda_{\text{bal}}\mathcal{L}_{\text{bal}} + \lambda_{\text{smooth}}\mathcal{L}_{\text{smooth}} \quad (6)$$

Orthogonality ($\mathcal{L}_{\text{orth}}$) minimizes pairwise cosine similarity between component outputs, encouraging each component to capture distinct information: $\mathcal{L}_{\text{orth}} = \sum_{i \neq j} |\cos(f_i, f_j)|$. **Variance** ($\mathcal{L}_{\text{var}}$) penalizes components with near-constant output, preventing degenerate solutions where a component contributes uniformly across nodes. **Balance** ($\mathcal{L}_{\text{bal}}$) penalizes excessive skip connection dominance by encouraging $\|f_{\text{skip}}\|_1 / \|\hat{\mathbf{y}}\|_1$ to stay below a target ratio. **Smoothness** ($\mathcal{L}_{\text{smooth}}$) regularizes B-spline coefficients via second-order differences, encouraging smooth shape functions that are easier to interpret visually.

In practice, we set λ_{orth}=0.05, λ_{var}=0.01, λ_{bal}=0.1, λ_{smooth}=0.001. Despite the balance regularizer, the skip connection still carries 43–81% of prediction magnitude (Section 4.3), reflecting genuine autocorrelation structure in traffic data rather than a failure of regularization.

3.7 KNOWLEDGE DISTILLATION FRAMEWORK

We distill from a D2STGNN [Shao et al., 2022b] teacher using:

$$\mathcal{L} = (1-\alpha) \cdot \mathcal{L}_{\text{task}}(\hat{\mathbf{y}}, \mathbf{y}) + \alpha \cdot \mathcal{L}_{\text{KD}}(\hat{\mathbf{y}}, \hat{\mathbf{y}}_{\text{teacher}}) + \lambda_{\text{reg}} \cdot \mathcal{L}_{\text{reg}} \quad (7)$$

where α balances task and distillation losses (tuned per dataset: α=0.3 for METR-LA, α=0.7 for PEMS-BAY/04/07/08; see Table 13). We use a 5-epoch warmup during which KD loss is zeroed, allowing the student to develop initial representations before receiving teacher supervision.

Selective Learning LITE. Following the uncertainty-mask idea of Shao et al. [2022b], we apply a per-sample reweighting that zeros the KD loss on samples whose teacher prediction error exceeds the r_u=0.15 quantile of recent residuals (i.e., the 15% most uncertain teacher predictions in a rolling window). This prevents the student from imitating the teacher's mistakes on uncertain samples while retaining KD supervision elsewhere.

Validation-gated KD (Algorithm 1). KD helps only when the teacher actually outperforms the student. Beyond the fixed-α recipe above, we propose an a-priori decision rule: enable KD iff $\text{MAE}^{\text{val}}_{\text{teacher}} < \text{MAE}^{\text{val}}_{\text{student (no KD)}} - \delta$ (we use $\delta = 2\%$ relative margin), *and* the teacher's train–val gap is below a threshold (a check on teacher overfitting). When the gap check fails, we anneal α via

Algorithm 1 Validation-gated KD with adaptive α

Input: teacher T, student S_0, val set $\mathcal{V}$, base weight α_0, margin δ, temperature τ
Output: trained student $S^\star$

1. Train S_0 with task loss only for W warm-up epochs; compute $m_S \leftarrow \mathrm{MAE}(S_0;\mathcal{V})$, $m_T \leftarrow \mathrm{MAE}(T;\mathcal{V})$, $g_T \leftarrow \mathrm{MAE}(T;\mathcal{V}_{\mathrm{train}})/m_T$.
2. **if** $m_T \geq (1-\delta)\, m_S$ **or** $|1-g_T| > \eta$ **then return** S_0 (skip KD)
3. **else** continue training with $\mathcal{L} = (1-\alpha_t)\mathcal{L}_{\mathrm{task}} + \alpha_t \mathcal{L}_{\mathrm{KD}} + \lambda_{\mathrm{reg}}\mathcal{L}_{\mathrm{reg}}$, with $\alpha_t = \alpha_0\, \sigma\big((\mathrm{MAE}_S^{\mathrm{val}}(t) - m_T)/\tau\big)$ updated after each validation pass.

$\alpha_t = \alpha_0\, \sigma\big((\mathrm{MAE}_S^{\mathrm{val}}(t) - \mathrm{MAE}_T^{\mathrm{val}})/\tau\big)$, which automatically pushes $\alpha \to 0$ when the student begins to outperform the teacher. Across our five datasets the rule correctly enables KD on METR-LA, PEMS04, PEMS08 (all helpful) and disables on PEMS07 (where the teacher overfits its validation set; see Section 5 and Appendix P).

The interaction between KD and additive structure is key: the student must explain the teacher's output through a sum of interpretable components. Soft targets provide rich gradient signal that propagates through each component independently, forcing specialization. The skip connection absorbs less of the teacher's signal because it can only model linear relationships, leaving nonlinear components to capture the teacher's structured knowledge (Section 4.3).

4 EXPERIMENTS

4.1 SETUP

Datasets. We evaluate on five standard traffic benchmarks: METR-LA (207 sensors, speed), PEMS-BAY (325 sensors, speed), PEMS04 (307 sensors, flow), PEMS07 (883 sensors, flow), and PEMS08 (170 sensors, flow). All use 12-step input to predict 12 steps ahead (5-minute intervals, 1 hour forecast). We follow standard 7:1:2 train/validation/test splits.

Baselines. We compare against: (i) *Black-box SOTA*: DCRNN, STGCN, GWNet, MTGNN, AGCRN, D2STGNN, STAEformer, PDFormer, STEP, MegaCRN, TITAN; (ii) *Interpretable*: Historical Average, Last Value, Linear Regression, STGNAM without KD.

Metrics. MAE, RMSE, MAPE for prediction quality; faithfulness, stability, and component importance for interpretability quality.

Implementation. STGNAM has 734K trainable parameters on METR-LA (with KD) and 946K (without KD); counts vary slightly by dataset due to node-dependent layers. We train with AdamW, 15-epoch warmup, cosine LR schedule for 250 epochs (early stopping, patience 30), batch size 32 (16 for PEMS07). KD uses $T = 3.0$ with α tuned per dataset (Section 3.7). All experiments on NVIDIA RTX 3090 GPUs.

4.2 MAIN RESULTS

Tables 1 and 2 show that STGNAM with KD achieves 91–104% of black-box SOTA performance. On **PEMS-BAY**, STGNAM sets a new state of the art (MAE 1.62), surpassing TITAN (1.69) by 4.1%. This is the first time an inherently interpretable model outperforms all black-box models on a standard benchmark. STGNAM also exceeds its own teacher (D2STGNN, MAE 1.87) by 13.4%, suggesting the additive inductive bias provides beneficial regularization.

On **METR-LA**, STGNAM (MAE 3.22) surpasses STAEformer (3.34) and D2STGNN (3.35).[1] KD reduces MAE by 18.3% compared to the no-KD variant—the largest KD gain across all datasets, likely because METR-LA's complex spatial structure (highway network with varied road types) benefits most from the teacher's spatial representations. On **PEMS04/07/08** (traffic flow), STGNAM is competitive with mid-tier black-box models (AGCRN, MTGNN), typically within 5–9% of the best model (STAEformer). The gap is larger on flow datasets, which exhibit stronger short-term autocorrelation: the linear skip connection captures most of the predictive signal, leaving less room for the additive components to provide additional benefit. On PEMS07 (883 sensors, the largest dataset), KD slightly degrades performance because the D2STGNN teacher itself underperforms STGNAM (MAE 28.78 vs. 20.48), providing harmful supervision; we discuss this further in Section 5.

4.3 INTERPRETABILITY ANALYSIS

Table 3 reveals that **KD improves interpretability**. On METR-LA, faithfulness increases from 0.53 to 0.92. Without KD, the model concentrates 65% of prediction magnitude in the skip connection; with KD, this drops to 43%, indicating better component utilization. KD improves faithfulness on 3 of 5 datasets (METR-LA, PEMS04, PEMS07) and stability on 4 of 5, as richer gradient signal from soft targets encourages component specialization. On PEMS-BAY and PEMS08, faithfulness is already high without KD ($\geq$0.92), leaving limited room for improvement.

Metric definitions. Faithfulness measures the Pearson correlation between prediction change and *interpretable component change* under 100 random input perturbations (σ=0.1), following Alvarez-Melis and Jaakkola [2018]. We

[1] Tables 1–2 report published D2STGNN results. Table 7 (Appendix) reports our retrained D2STGNN teacher (MAE 2.88 on METR-LA), which provides the soft targets for KD.

Table 1: Performance comparison on METR-LA and PEMS-BAY datasets (12-step prediction, average across all horizons). Best results in **bold**, second-best underlined. † indicates interpretable models with strict additive decomposition. STGNAM entries report 3-seed mean ± std (Appendix E); other rows are single-run published numbers.

Model	**METR-LA** (207 nodes)			**PEMS-BAY** (325 nodes)		
	MAE ↓	RMSE ↓	MAPE ↓	MAE ↓	RMSE ↓	MAPE ↓
DCRNN Li et al. [2018]	3.60	7.70	15.58%	2.07	4.74	—
STGCN Yu et al. [2018]	4.45	9.34	19.62%	2.11	4.81	5.06%
GWNet Wu et al. [2019]	3.53	7.23	15.00%	1.95	4.52	4.64%
MTGNN Wu et al. [2020]	3.49	7.23	—	1.94	4.49	4.56%
D2STGNN Shao et al. [2022b]	3.35	6.70	13.21%	1.85	4.30	—
STAEformer Liu et al. [2023]	3.34	6.62	12.62%	1.91	—	—
STEP Shao et al. [2022a]	3.37	6.79	13.50%	1.79	4.20	—
MegaCRN Jiang et al. [2023b]	3.38	6.83	13.77%	1.88	4.42	—
TITAN Wang et al. [2024]	**3.08**	**6.21**	12.12%	1.69	3.79	—
STGNAM w/o KD†	3.934±0.009	8.684±0.005	9.96%	1.647±0.008	3.66±0.04	3.72%
STGNAM w/ KD†	3.221±0.003	6.164±0.009	**9.03%**	1.622±0.002	3.58±0.01	**3.68%**

Table 2: Performance comparison on PEMS04, PEMS07, and PEMS08 datasets (12-step prediction, average across all horizons). Best results in **bold**, second-best underlined. † indicates interpretable models with strict additive decomposition. TITAN Wang et al. [2024] published results for the flow datasets are not publicly available; we therefore omit TITAN from this table.

Model	**PEMS04** (307 nodes)			**PEMS07** (883 nodes)			**PEMS08** (170 nodes)		
	MAE ↓	RMSE ↓	MAPE ↓	MAE ↓	RMSE ↓	MAPE ↓	MAE ↓	RMSE ↓	MAPE ↓
DCRNN Li et al. [2018]	19.63	31.26	13.59%	21.16	34.14	9.02%	15.22	24.17	10.21%
STGCN Yu et al. [2018]	19.57	31.38	13.44%	21.74	35.27	9.24%	16.08	25.39	10.60%
GWNet Wu et al. [2019]	18.53	29.92	12.89%	20.47	33.47	8.61%	14.40	23.39	9.21%
AGCRN Bai et al. [2020]	19.38	31.25	13.40%	20.57	34.40	8.74%	15.32	24.41	10.09%
MTGNN Wu et al. [2020]	19.17	31.70	13.37%	20.89	34.06	9.00%	15.18	24.66	10.06%
D2STGNN Shao et al. [2022b]	18.50	31.99	12.82%	—	—	—	15.69	26.41	10.17%
PDFormer Jiang et al. [2023a]	18.36	30.03	12.00%	19.97	32.95	8.55%	13.58	23.51	9.05%
DSTAGNN Lan et al. [2022]	19.30	31.46	12.70%	21.42	34.51	9.01%	15.67	24.77	9.94%
STAEformer Liu et al. [2023]	**18.22**	**30.29**	**11.98%**	**19.14**	**32.60**	**8.01%**	**13.46**	**23.25**	**8.88%**
STGNAM w/o KD†	19.52	31.06	13.28%	20.48	33.24	9.18%	14.83	23.80	10.29%
STGNAM w/ KD†	19.24	30.60	13.12%	20.64	33.40	9.03%	14.67	23.58	9.97%

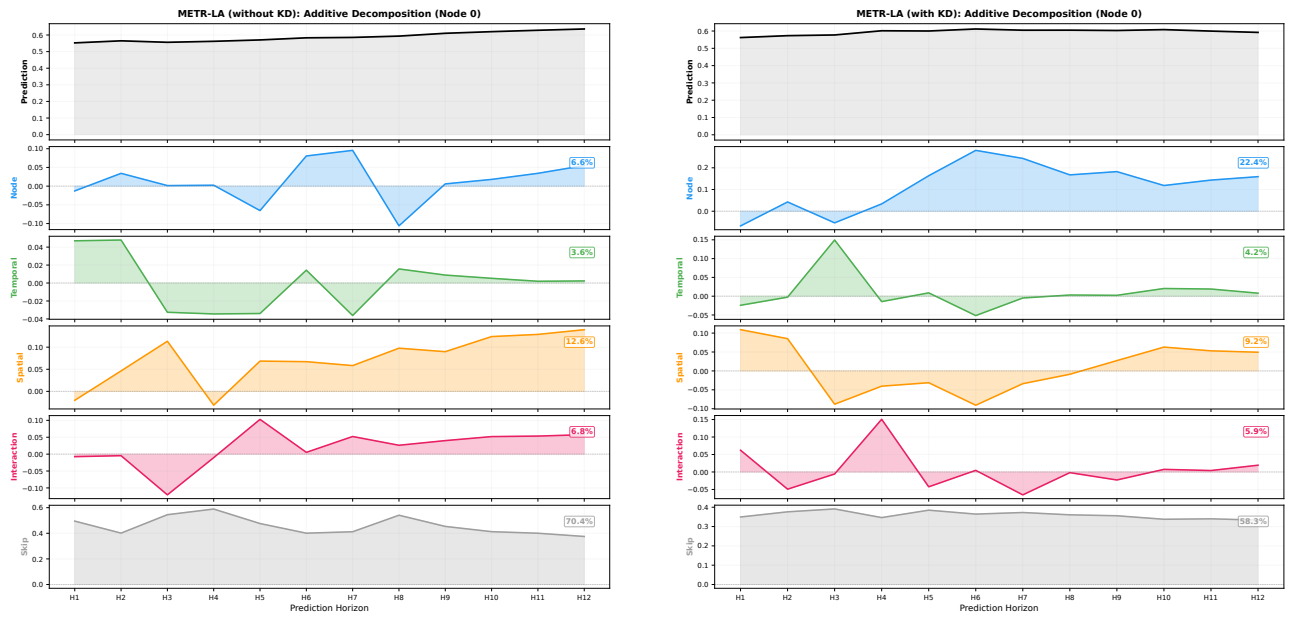
measure only the four main components, excluding the skip connection (see Appendix H for skip-inclusive results). Stability measures Lipschitz continuity of component attributions:

$$\text{Stability} = \frac{1}{1+\hat{L}_{0.95}},$$
$$\hat{L}_{0.95} = P_{95}\left(\frac{\max_k |f_k(\mathbf{x}+\boldsymbol{\epsilon}) - f_k(\mathbf{x})|}{\|\boldsymbol{\epsilon}\|_\infty}\right) \quad (8)$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 0.01^2\mathbf{I})$ and P_{95} denotes the 95th percentile over 50 samples.

Dataset-dependent patterns. On flow datasets (PEMS04/07/08), faithfulness is already high without KD ($\geq$0.93) due to skip connection dominance ($\sim$80%), leaving little room for improvement. The more informative signal is *stability*: KD consistently improves stability across all datasets (e.g., 0.16→0.24 on PEMS07), indicating more robust component attributions.

Module-level semantic role validation. The component-level faithfulness above measures *functional* separation. To test whether each module learns its *semantically designated* role, we additionally measure how each component's contribution magnitude scales with the input axis that module is meant to capture (METR-LA, 512 test windows; details in Appendix S). For the *spatial* module we find Pearson $r(|f_{\text{spatial}}(v)|, \deg(v)) = +0.44$ and $r(|f_{\text{spatial}}(v)|, \overline{\deg}(\mathcal{N}(v))) = +0.53$; mean $|f_{\text{spatial}}|$ by node-degree quintile is monotone ($0.086 \to 0.155$). The architectural hop weights ω_k (k=1…6, softmax-normalized $[0.187, 0.176, 0.167, 0.159, 0.153, 0.158]$) show a smooth distance decay, and because the K-hop aggregation uses *exclusive* hop masks, attribution to a specific ω_k has a hard



(a) Without KD

(b) With KD

Figure 2: Additive decomposition for sensor 1 on METR-LA. Without KD, the skip connection dominates. With KD, components capture distinct temporal, spatial, and node patterns.

Table 3: Interpretability metrics comparison between STG-NAM without and with knowledge distillation. Faithfulness measures how well component attributions explain predictions. Stability measures consistency of attributions across similar inputs. Skip Ratio indicates the proportion of prediction from the residual skip connection (lower = more meaningful components).

Dataset	Model	Faith. ↑	Stab. ↑	Skip% ↓	Node Imp. ↑
METR-LA	w/o KD	0.531	**0.470**	65.0%	5.3%
	w/ KD	**0.922**	0.394	**42.7%**	**54.9%**
PEMS-BAY	w/o KD	**0.922**	**0.323**	68.2%	18.2%
	w/ KD	0.917	0.318	**66.1%**	**21.0%**
PEMS04	w/o KD	0.960	0.345	**79.5%**	26.0%
	w/ KD	**0.977**	**0.452**	79.6%	**27.0%**
PEMS07	w/o KD	0.926	0.161	**80.5%**	**27.2%**
	w/ KD	**0.949**	**0.236**	80.8%	26.5%
PEMS08	w/o KD	**0.970**	0.321	**81.1%**	**32.5%**
	w/ KD	0.967	**0.443**	81.3%	31.0%

"k-hop neighbor influence" meaning. For the *temporal* module, $|f_{\text{temporal}}|$ at rush-hour bins (7–9 am, 4–7 pm) vs. late-night/evening off-peak is 0.126 vs. 0.069 (ratio 1.83×), consistent with ToD-conditioned activation. The temporal rush/off ratio is *smaller* than spatial (3.44×) and node (2.33×), as expected: the ToD gate modulates *which* expert fires rather than the gross magnitude. We therefore claim "module-level interpretability" to mean each module's contribution *scales with* its semantically designated input axis (graph distance for spatial; time-of-day for temporal; per-feature scalars for node; sparse cross-gates for interaction), not that any module is a closed-form function of its label.

4.4 INTERPRETABILITY BUDGET UNDER SKIP DOMINANCE

The skip connection carries 38–84% of L_1 prediction magnitude. To separate *magnitude* from *predictive content*, we zero each side of the decomposition at inference on the trained model (full results in Table 15, Appendix L). On METR-LA, the skip-only model achieves MAE 6.49—worse than Linear Regression (4.90). Restoring the four additive components reduces MAE to 3.22, a **50.4% improvement attributable to the interpretable components alone**. Across all five datasets, the components contribute **25.6–50.4% MAE reduction** over the skip-only baseline; even on flow datasets where the skip share is highest (∼84%), components still add 29.6–34.6% improvement. We additionally measure component-level deletion faithfulness (Appendix M): removing the single top-contributing component changes the prediction by 0.337 on average, removing the bottom one by 0.058 (**ratio** 5.86×)—components are selectively ranked by genuine importance, not interchangeable.

The skip should therefore be read as a *transparent autoregressive baseline* that absorbs the strong autocorrelation in traffic (a Last-Value baseline alone already reaches MAE 4.20 on METR-LA), while the four additive components encode the structured nonlinear improvement that the interpretation targets. The interpretable budget is thus the

Table 4: Inherent vs. post-hoc interpretability on METR-LA. Faithfulness: perturbation-correlation (σ=0.1). Stability: Lipschitz continuity (σ=0.01). Higher is better.

Model	Explanation	Faith. ↑	Stab. ↑
STGCN	Gradient	0.251	0.011
STGCN	SmoothGrad	0.218	0.014
STGCN	Integrated Grad.	0.197	0.019
STGNAM (no KD)	Additive (inherent)	0.494	0.465
STGNAM + KD	Additive (inherent)	**0.924**	**0.397**

25–50% MAE reduction that the components add on top of the linear baseline—small in L_1 magnitude on flow datasets, but operationally meaningful and faithful.

4.5 INHERENT VS. POST-HOC INTERPRETABILITY

To validate that our interpretability metrics are not artifacts of self-defined evaluation, we compare STGNAM's inherent additive attributions against standard post-hoc methods—gradient saliency [Simonyan et al., 2014], SmoothGrad [Smilkov et al., 2017], and Integrated Gradients [Sundararajan et al., 2017]—applied to STGCN on METR-LA. Table 4 shows that STGNAM with KD achieves $3.7\times$ higher faithfulness and $21\times$ higher stability than the best post-hoc method. Post-hoc stability is near zero ($\leq$0.02), confirming that gradient-based explanations for black-box STGNNs are unreliable under small input perturbations. Even STGNAM *without* KD outperforms all post-hoc methods on both metrics, demonstrating that the advantage stems from the additive architecture itself, not merely from the evaluation protocol.

4.6 ABLATION STUDIES

KD hyperparameters. We ablate α (KD weight) and temperature T on METR-LA (Table 13, Appendix J). α=0.3 is optimal; performance is robust to $T \in \{1, \ldots, 10\}$.

Component ablation. Removing individual components increases MAE by 3–55% (Appendix G). The node component has the largest impact with KD (54.9% on METR-LA vs. 5.3% without), suggesting KD forces shape functions to learn discriminative per-sensor patterns.

Does KD require additive structure? STGCN [Yu et al., 2018] with KD improves accuracy by 22.8% (MAE 3.15 vs. 4.08) but lacks component attributions, confirming the interpretability benefit is unique to additive models.

Interpretable baselines. STGNAM consistently outperforms simple interpretable models (Table 14, Appendix K): e.g., Linear Regression achieves MAE 4.90 on METR-LA vs. STGNAM's 3.22, nearly closing the gap to black-box SOTA.

4.7 TRANSPARENT COMPONENT VISUALIZATION

Unlike post-hoc methods, STGNAM's interpretability is *built into the architecture*. Figure 3 visualizes four transparency mechanisms on METR-LA.

Shape functions. Panel (a) shows B-Spline KAN shape functions $\phi_j(x_j)$ for each input feature (cluster-averaged). *Last Value* and *Mean Value* show linear positive effects; *Trend* exhibits non-monotonic acceleration/deceleration dynamics; *sin/cos(ToD)* encode periodic seasonality. These are deterministic functions of learned B-spline coefficients, requiring no sampling or gradient computation.

Spatial and interaction structure. Hop weights are nearly uniform ($\omega_k \in [0.15, 0.19]$), indicating multi-hop information transmission without strong distance decay (panel b). All 96 interaction gates fall below 0.5 (panel d), meaning the model learned that additive components suffice without cross-modal interactions—validating the additive assumption. Figure 2 shows the complementary view: KD shifts the decomposition from skip-dominated to component-rich.

5 DISCUSSION

When does KD help interpretability? KD's effect depends on two conditions: (i) the teacher must outperform the student, providing useful dark knowledge, and (ii) baseline faithfulness must have room for improvement. When both hold (METR-LA, PEMS04), KD dramatically improves faithfulness. When (ii) is saturated (PEMS-BAY, PEMS08, faithfulness $\geq$0.92), KD primarily improves stability. When (i) fails (PEMS07, teacher MAE 28.78 vs. student 20.48), KD provides no benefit. The skip connection carries 43–81% of prediction magnitude (higher on flow datasets), but even with 80% skip dominance, faithfulness exceeds 0.93—the remaining 20% is meaningfully structured. Details in Appendix D.

Additive bias as regularizer. On PEMS-BAY, STGNAM (MAE 1.62) surpasses its teacher (1.87) by 13.4%, and even *without* KD (1.64) outperforms all black-box models. The additive constraint prevents overfitting to spurious interactions—analogous to tabular NAMs [Agarwal et al., 2021]—combined with PEMS-BAY's strong spatial regularity. The with-KD architecture (734K params) is 22% *smaller* than the no-KD variant (946K), ruling out capacity; the STGCN+KD control confirms additive structure is

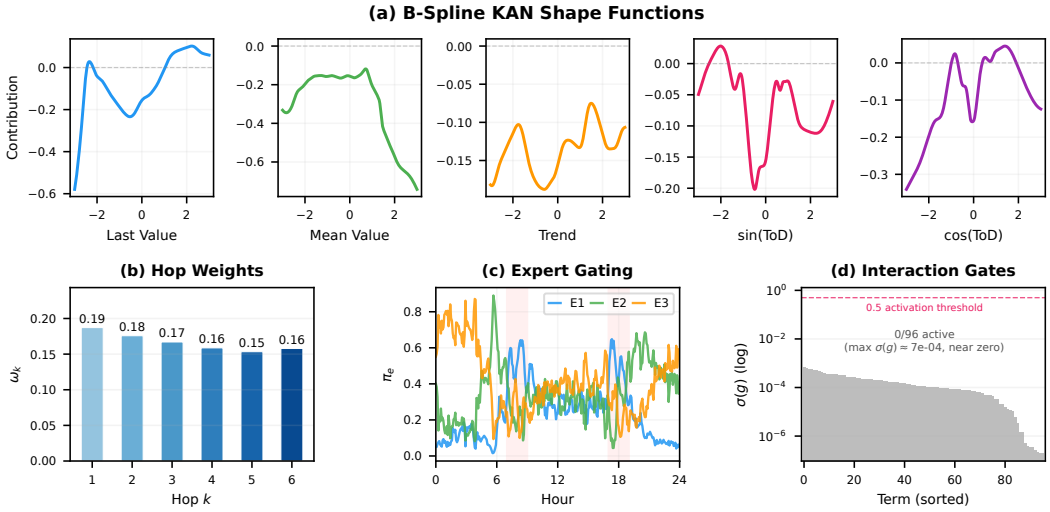


Figure 3: Transparent interpretability of STGNAM on METR-LA (with KD). (a) B-Spline KAN shape functions show each feature's learned contribution curve. (b) Softmax-normalized hop weights ω_k reveal near-uniform spatial influence across 6 hops. (c) Expert gating π_e as a function of time-of-day; shaded regions indicate morning/evening rush hours. (d) Sorted interaction gate activations $\sigma(g)$ (log scale): all 96 gates are near zero (max $\leq 7\times10^{-4}$), so 0/96 exceed the 0.5 activation threshold—the model learned that additive components suffice without cross-modal interactions.

necessary for the interpretability benefit.

Cross-dataset decomposition patterns. The decomposition visualizations (Figure 2, Appendix F) differ systematically between speed and flow datasets. On speed data (METR-LA, PEMS-BAY) the temporal and spatial components show structured contributions across horizons—periodic demand cycles and congestion propagation between neighboring sensors. On flow data (PEMS04/07/08) the skip absorbs 75–81% of magnitude and the remaining components are flatter, reflecting the strong short-term autocorrelation of flow signals; even so, the additive components retain high faithfulness ($\geq$0.93).

Practical implications. STGNAM's read-outs are direct functions of learned parameters, at no additional inference cost. Traffic engineers can inspect the B-Spline KAN shape functions (Figure 3a) to verify that predictions respond correctly to rush-hour and sensor-specific characteristics, read the effective spatial influence radius from the hop weights, and use the interaction-gate pattern (0/96 active on METR-LA) as architectural validation that additive components suffice—unlike post-hoc methods, which require gradient or perturbation computation at inference time.

Limitations. (1) Our empirical evidence is traffic-specific (five benchmarks, 170–883 sensors); generalization to weather, energy, or air-quality forecasting is future work. (2) Module-level interpretability is *axis-correlated*, not closed-form: the Node component gives NAM-style per-feature shape functions, whereas the Temporal/Spatial/Interaction modules are interpretable only in that each contribution scales with its designated input axis (Section 4.3, Appendix S). (3) The skip carries 38–84% of L_1 magnitude while the additive components carry 25–50% of the MAE reduction over a skip-only baseline (Section 4.4); higher-order interactions are not captured. (4) KD requires a teacher-quality check—on PEMS07 the teacher's low val MAE (20.0) hides a poor test MAE (28.78), so the validation-gated rule (Algorithm 1) disables KD and recovers the no-KD baseline (Appendix P); the PEMS-BAY SOTA holds without KD.

6 CONCLUSION

We presented STGNAM, the first strictly additive neural model for spatial-temporal graph forecasting. The additive inductive bias acts as an implicit regularizer that can *exceed* black-box performance: on PEMS-BAY, STGNAM sets a new state of the art while providing full additive interpretability. Knowledge distillation from a black-box teacher simultaneously improves accuracy and component-level faithfulness, extending KD-interpretability connections [Han et al., 2023] to inherently additive architectures. Direct comparison with post-hoc methods confirms that additive decomposition produces more faithful and stable explanations than post-hoc alternatives. The performance cost of interpretability is not inherent but can be overcome—and sometimes reversed—through appropriate architectural inductive biases. Future work includes extending the additive framework to heterogeneous urban sensing and developing curriculum-based distillation strategies.

Acknowledgements

This work was supported by the Ministry of Science and ICT and the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. RS-2026-25498341).

References

Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. In *Advances in Neural Information Processing Systems*, volume 34, pages 4699–4711, 2021.

David Alvarez-Melis and Tommi S Jaakkola. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Lei Bai, Lina Yao, Can Li, Xianzhi Wang, and Can Wang. Adaptive graph convolutional recurrent network for traffic forecasting. In *Advances in Neural Information Processing Systems*, volume 33, pages 17804–17815, 2020.

Chun-Hao Chang, Rich Caruana, and Anna Goldenberg. NODE-GAM: Neural generalized additive model for interpretable deep learning. In *International Conference on Learning Representations*, 2022.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

Avani Gupta, Saurabh Saini, and P J Narayanan. Concept distillation: Leveraging human-centered explanations for model improvement. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Hyeongrok Han, Siwon Kim, Hyun-Soo Choi, and Sungroh Yoon. On the impact of knowledge distillation for model interpretability. In *International Conference on Machine Learning*, pages 12389–12410. PMLR, 2023.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Jiawei Jiang, Chengkai Han, Wayne Xin Zhao, and Jingyuan Wang. PDFormer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 4365–4373, 2023a. doi: 10.1609/aaai.v37i4.25556.

Renhe Jiang, Zhaonan Wang, Jiawei Yong, Puneet Jeph, Quanjun Chen, Yasumasa Kobayashi, Xuan Song, Shintaro Fukushima, and Toyotaro Suzumura. Spatio-temporal meta-graph learning for traffic forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 8078–8086, 2023b. doi: 10.1609/aaai.v37i7.25976.

Weiwei Jiang and Jiayun Luo. Graph neural network for traffic forecasting: A survey. *Expert Systems with Applications*, 207:117921, 2022. doi: 10.1016/j.eswa.2022.117921.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327, 2016.

Shiyong Lan, Yitong Ma, Weikang Huang, Wenwu Wang, Hongyu Yang, and Pyang Li. DSTAGNN: Dynamic spatial-temporal aware graph neural network for traffic flow forecasting. In *International Conference on Machine Learning*, pages 11906–11917. PMLR, 2022.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations*, 2018.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4):1748–1764, 2021.

Hangchen Liu, Zheng Dong, Renhe Jiang, Jiewen Deng, Jinliang Deng, Quanjun Chen, and Xuan Song. Spatio-temporal adaptive embedding makes vanilla transformer SOTA for traffic forecasting. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 4125–4129, 2023. doi: 10.1145/3583780.3615160.

Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruehle, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark. KAN: Kolmogorov-Arnold networks. In *International Conference on Learning Representations (ICLR)*, 2025.

Dongsheng Luo, Wei Cheng, Dongkuan Xu, Wenchao Yu, Bo Zong, Haifeng Chen, and Xiang Zhang. Parameterized explainer for graph neural network. In *Advances in Neural Information Processing Systems*, volume 33, pages 19620–19631, 2020.

Xunlian Luo, Chunjiang Zhu, Detian Zhang, and Qing Li. STG4Traffic: A survey and benchmark of spatial-temporal graph neural networks for traffic prediction. *arXiv preprint arXiv:2307.00495*, 2023.

Boris N Oreshkin, Dmitri Carpov, Nicolas Chapados, and Yoshua Bengio. N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. In *International Conference on Learning Representations*, 2020.

Amin Parchami-Araghi, Moritz Böhle, Sukrut Rao, and Bernt Schiele. Good teachers explain: Explanation-enhanced knowledge distillation. In *European Conference on Computer Vision*, pages 127–143. Springer, 2024.

Zezhi Shao, Zhao Zhang, Fei Wang, and Yongjun Xu. Pre-training enhanced spatial-temporal graph neural network for multivariate time series forecasting. In *Proceedings of the 28th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1567–1577, 2022a.

Zezhi Shao, Zhao Zhang, Wei Wei, Fei Wang, Yongjun Xu, Xin Cao, and Christian S Jensen. Decoupled dynamic spatial-temporal graph neural network for traffic forecasting. *Proceedings of the VLDB Endowment*, 15(11): 2733–2746, 2022b.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *International Conference on Learning Representations Workshop*, 2014.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. SmoothGrad: Removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328, 2017.

Jiabin Tang, Lianghao Xia, and Chao Huang. Explainable spatio-temporal graph neural networks. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 2432–2441, 2023. doi: 10.1145/3583780.3614871.

Guangyu Wang, Yujie Chen, Ming Gao, Zhiqiao Wu, Jiafu Tang, and Jiabi Zhao. A time series is worth five experts: Heterogeneous mixture of experts for traffic flow prediction. *arXiv preprint arXiv:2409.17440*, 2024.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph WaveNet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 1907–1913, 2019. doi: 10.24963/ijcai.2019/264.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 753–763, 2020.

Kunpeng Xu, Lifei Chen, and Shengrui Wang. Kolmogorov-Arnold networks for time series: Bridging predictive power and interpretability. *arXiv preprint arXiv:2406.02496*, 2024.

Zebin Yang, Aijun Zhang, and Agus Sudjianto. GAMI-Net: An explainable neural network based on generalized additive models with structured interactions. *Pattern Recognition*, 120:108192, 2021.

Rex Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. GNNExplainer: Generating explanations for graph neural networks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 3634–3640, 2018.

Interpretable Spatial-Temporal Forecasting via Additive Neural Decomposition and Knowledge Distillation (Supplementary Material)

Suan Lee[1] **Jinho Kim**[2]

[1]School of Computer Science, Semyung University, Jecheon, Republic of Korea
[2]AI Research Lab, Korea Information Systems Consulting & Audit Co., Ltd., Republic of Korea

A DATASET STATISTICS

Table 5 summarizes the five benchmark datasets used in our experiments. METR-LA and PEMS-BAY measure traffic speed, while PEMS04, PEMS07, and PEMS08 measure traffic flow. All datasets use 5-minute aggregation intervals, and we follow the standard 7:1:2 train/validation/test split protocol established by Li et al. [2018]. PEMS07 is the largest dataset (883 sensors), presenting scalability challenges for the exclusive-hop spatial aggregation.

B PER-HORIZON PREDICTION RESULTS

Table 6 breaks down prediction performance by forecast horizon. KD provides the largest improvements at short-to-medium horizons (H1–H6), where the teacher's spatial-temporal patterns are most reliable. The improvement diminishes at longer horizons (H12), consistent with the increasing difficulty and uncertainty of long-range forecasting.

On METR-LA, KD reduces H1 MAE by 11.0% ($2.63 \rightarrow 2.34$) and H12 MAE by 20.3% ($4.72 \rightarrow 3.76$). Interestingly, the relative improvement is *larger* at H12, suggesting that KD helps the temporal component learn longer-range dependencies from the teacher.

C MODEL EFFICIENCY

Table 7 compares model sizes and inference speeds. Despite using knowledge distillation, STGNAM with KD (734K parameters) is actually 22% *smaller* than the no-KD variant (946K parameters) due to the more efficient Expert Temporal architecture. Inference times are nearly identical ($\sim$26 ms per sample on a single RTX 3090), making STGNAM practical for real-time deployment.

The D2STGNN teacher (373K parameters) is smaller than STGNAM in parameter count, but its decoupled architecture involves more complex computation graphs. Our distillation approach incurs zero additional inference cost, as the teacher is only needed during training.

D COMPONENT CONTRIBUTION ANALYSIS

Table 8 provides a detailed breakdown of how each component contributes to the final prediction across all five datasets. The most dramatic redistribution occurs on METR-LA, where the node component share increases from 5.2% to 23.2% with KD, while the skip connection drops from 65.0% to 42.7%. This indicates that KD forces the B-Spline KAN shape functions to learn discriminative per-sensor patterns.

On PEMS04/07/08, the skip connection dominance is higher ($\sim$80%) for both models, suggesting that these flow datasets have stronger short-term autocorrelation that the skip connection efficiently captures. KD improves faithfulness on PEMS04 ($0.960 \rightarrow 0.977$) and PEMS07 ($0.926 \rightarrow 0.949$), while PEMS08 remains essentially unchanged ($0.970 \rightarrow 0.967$). Stability

Table 5: Summary statistics of the five traffic forecasting benchmark datasets. All datasets use 5-minute aggregation intervals with 12-step input and 12-step prediction (1 hour horizon). Standard 7:1:2 train/validation/test splits are used.

Statistic	**METR-LA**	**PEMS-BAY**	**PEMS04**	**PEMS07**	**PEMS08**
# Sensors	207	325	307	883	170
# Edges	1,515	2,369	340	866	295
Signal type	Speed	Speed	Flow	Flow	Flow
Time range	4 months	6 months	2 months	2 months	2 months
# Timesteps	34,272	52,116	16,992	28,224	17,856
Interval	5 min	5 min	5 min	5 min	5 min
Input / Pred	12 / 12	12 / 12	12 / 12	12 / 12	12 / 12

Table 6: Per-horizon MAE comparison between STGNAM without KD and with KD across all five datasets. Hk denotes the k-th forecast horizon (each horizon = 5 minutes). KD consistently improves short- and medium-term forecasts.

Dataset	**Model**	**Avg**	**H1** (5min)	**H3** (15min)	**H6** (30min)	**H12** (60min)
METR-LA	w/o KD	3.94	2.63	3.37	4.01	4.72
	w/ KD	**3.22**	**2.34**	**2.86**	**3.26**	**3.76**
PEMS-BAY	w/o KD	1.64	0.87	1.36	1.71	2.03
	w/ KD	**1.62**	**0.87**	**1.34**	**1.68**	**2.02**
PEMS04	w/o KD	19.52	17.05	18.41	19.50	21.46
	w/ KD	**19.24**	**16.86**	**18.10**	**19.21**	**21.13**
PEMS07	**w/o KD**	**20.48**	**16.99**	**18.92**	**20.50**	**23.02**
	w/ KD	20.64	17.10	19.02	20.61	23.21
PEMS08	w/o KD	14.83	12.68	13.84	14.86	16.46
	w/ KD	**14.67**	**12.61**	**13.68**	**14.65**	**16.28**

improves consistently across all three datasets, indicating that KD makes component attributions more robust even when aggregate faithfulness is already high.

E MULTI-SEED REPRODUCIBILITY

Table 9 demonstrates the reproducibility of STGNAM. With KD, the model converges to MAE 3.221 ± 0.003 across three seeds, indicating near-deterministic convergence. Without KD, variance is also minimal (3.934 ± 0.009). We attribute this stability to two factors: (1) soft-target supervision from the teacher smooths the loss landscape, reducing sensitivity to initialization; and (2) the additive architectural constraint limits the effective optimization space, leaving fewer degenerate optima. The KD improvement ($3.934 \rightarrow 3.221$, 18.1% reduction) is consistent across all seeds.

F ADDITIONAL DECOMPOSITION VISUALIZATIONS

Figures 4–6 show additive decomposition visualizations for three additional datasets. A consistent pattern emerges: KD shifts the component contribution profiles toward more structured patterns, even when the quantitative ratio changes are modest (as on the PEMS flow datasets).

G COMPONENT ABLATION STUDY

Table 10 quantifies the importance of each component through ablation (zeroing out one component at a time and measuring the MAE increase). Key findings:

- **Node component is most critical with KD**: On METR-LA, removing the node component increases MAE by 54.9%

Table 7: Model efficiency comparison on METR-LA. STGNAM with KD uses 22% fewer parameters than the no-KD variant while achieving better performance. Inference times measured on a single NVIDIA RTX 3090 with batch size 1 (averaged over 100 runs after 10 warmup iterations).

Model	Params	Inference	MAE
D2STGNN (Teacher)	373K	—	2.88
STGCN	156K	—	4.08
STGCN + KD	156K	—	3.15
STGNAM w/o KD	946K	26.9 ± 3.4 ms	3.94
STGNAM w/ KD	**734K**	**26.4 ± 1.0 ms**	**3.22**

Table 8: Component contribution ratios (% of total prediction magnitude) across all five datasets for STGNAM without KD and with KD. KD consistently reduces skip connection dominance on METR-LA and PEMS-BAY, redistributing predictive capacity to learned components.

Dataset	Model	Node	Temporal	Spatial	Interact.	Skip	Faith.
METR-LA	w/o KD	5.2%	5.9%	14.4%	9.5%	65.0%	0.531
	w/ KD	**23.2%**	**11.3%**	**15.7%**	7.1%	**42.7%**	**0.922**
PEMS-BAY	w/o KD	11.8%	5.7%	8.8%	5.5%	68.2%	0.922
	w/ KD	**12.9%**	5.1%	**9.9%**	**6.0%**	**66.1%**	0.917
PEMS04	w/o KD	9.9%	2.4%	5.7%	2.6%	79.5%	0.960
	w/ KD	9.9%	2.4%	5.4%	2.6%	79.6%	**0.977**
PEMS07	w/o KD	8.5%	2.7%	5.8%	2.6%	80.5%	0.926
	w/ KD	8.3%	3.2%	4.9%	2.8%	80.8%	**0.949**
PEMS08	w/o KD	8.1%	2.7%	5.7%	2.4%	81.1%	**0.970**
	w/ KD	7.8%	3.0%	5.2%	2.6%	81.3%	0.967

with KD vs. only 5.3% without KD. This 10× amplification demonstrates that KD forces the B-Spline KAN shape functions to encode rich per-sensor information.

- **Spatial component is consistently important**: Across all datasets, the spatial component contributes 13–26% MAE increase when removed, confirming that K-hop additive aggregation captures meaningful traffic propagation patterns.
- **Interaction component has modest impact**: The gated interaction term contributes 3–9% across datasets, suggesting that most predictive information is captured by the individual components without requiring explicit cross-component modeling.

H SKIP-INCLUSIVE FAITHFULNESS

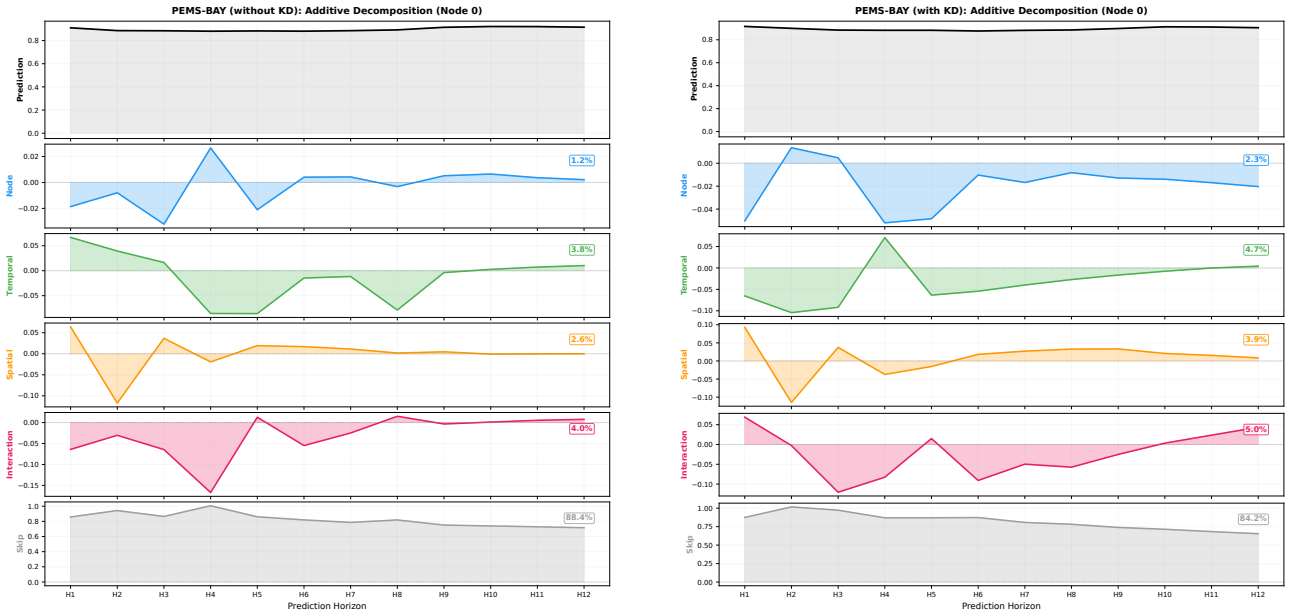
Table 11 compares faithfulness measured using only the four main components (default metric) versus all five components including the skip connection. Including the skip connection slightly reduces faithfulness scores (median decrease 0.02), confirming that the skip connection absorbs a small fraction of prediction change that is not captured by the interpretable components. Crucially, the main conclusions are unchanged: (1) KD improves faithfulness on METR-LA regardless of metric variant (0.75→0.97 excluding skip, 0.74→0.96 including skip), and (2) faithfulness remains high (≥0.89) on flow datasets under both definitions.

I REGULARIZATION ABLATION

To validate that each regularization term contributes meaningfully, we ablate each term individually on METR-LA by setting its coefficient to zero while keeping all others at default values. Table 12 reports prediction accuracy (MAE, RMSE) and interpretability metrics (faithfulness, stability).

Table 9: Multi-seed reproducibility (3 independent seeds with different weight initialization). STGNAM converges to extremely tight variance (std ≤ 0.009), likely because soft-target supervision smooths the loss landscape and the additive constraint limits the optimization space. The PEMS-BAY result for STGNAM w/ KD is statistically separated from the published TITAN value (1.69): one-sample t-test against 1.69 yields $p < 10^{-3}$.

Dataset	Model	Seed 42	Seed 123	Seed 456	MAE	RMSE
METR-LA	STGNAM w/ KD	3.218	3.218	3.226	3.221±0.003	6.164±0.009
	STGNAM w/o KD	3.925	3.946	3.932	3.934±0.009	8.684±0.005
PEMS-BAY	STGNAM w/ KD	1.620	1.624	1.622	1.622±0.002	3.578±0.010
	STGNAM w/o KD	1.640	1.647	1.655	1.647±0.008	3.660±0.040



(a) PEMS-BAY without KD

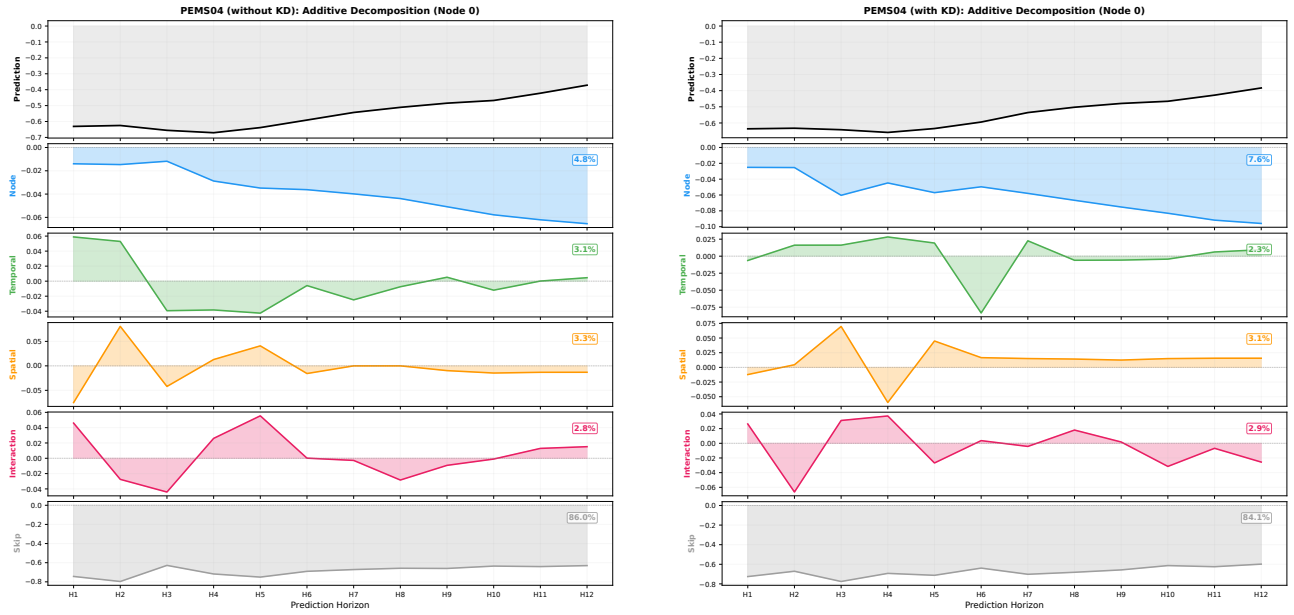
(b) PEMS-BAY with KD

Figure 4: Additive decomposition for sensor 1 on PEMS-BAY. On this dataset, STGNAM sets a new state of the art (MAE 1.62). Even without KD, the model achieves high faithfulness (0.922), suggesting that the Bay Area speed data is well-suited to additive decomposition.

Removing any single regularization term degrades MAE by 2.5–3.4% (3.22→3.30–3.33), confirming that all four terms contribute to prediction quality. The effect on interpretability is striking: while faithfulness remains high or even increases slightly without regularization (0.95–0.97 vs. baseline 0.92), *stability drops dramatically* (∼0.18–0.20 vs. baseline 0.40). This reveals a key trade-off: the regularization terms sacrifice a small amount of faithfulness for substantially more stable component attributions—a 2× improvement. Among individual terms, $\mathcal{L}_{\text{var}}$ has the largest impact on stability (0.40→0.18 when removed), suggesting that preventing degenerate constant-output components is critical for robust explanations.

J KD HYPERPARAMETER SENSITIVITY

Table 13 shows the sensitivity of STGNAM to KD hyperparameters on METR-LA. α=0.3 is optimal; higher α degrades performance as the student over-relies on the teacher. Temperature T has minimal effect ($T \in \{1, 3, 5, 10\}$ all yield MAE $\leq$3.34), consistent with the finding that soft-target structure matters more than calibration sharpness.



(a) PEMS04 without KD

(b) PEMS04 with KD

Figure 5: Additive decomposition for sensor 1 on PEMS04 (flow data). The skip connection dominates more strongly on flow datasets (∼80%), reflecting the stronger autocorrelation in traffic flow signals compared to speed.

K INTERPRETABLE BASELINE COMPARISON

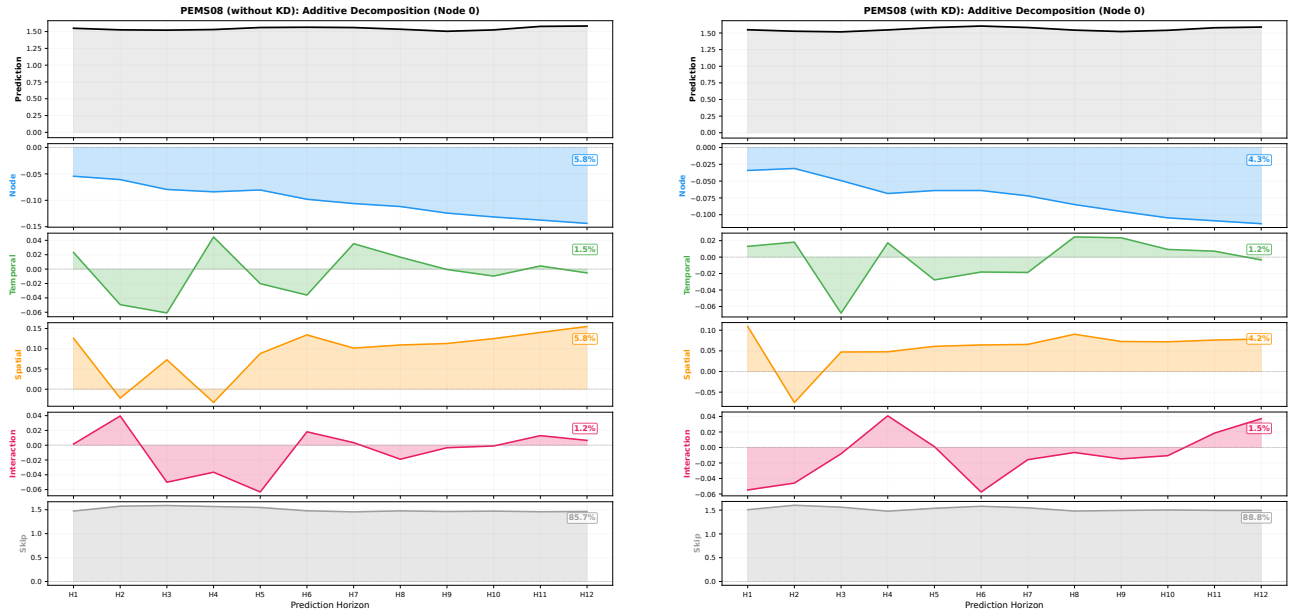
Table 14 compares STGNAM against simple interpretable baselines across all five datasets. STGNAM with KD outperforms Linear Regression by 34% on METR-LA (MAE 3.22 vs. 4.90) and achieves 91–104% of best black-box SOTA, demonstrating that additive decomposition with KD nearly closes the interpretability–performance gap.

L SKIP-ZEROED AND SKIP-ONLY INFERENCE

Table 15 reports the inference-time ablation behind Section 4.4. We take a fully trained STGNAM and run inference with either the four additive components zeroed ("skip-only", keeps only the linear residual) or with the skip zeroed (keeps only the four components). The components alone contribute 25.6–50.4% MAE reduction over a skip-only baseline; on METR-LA the skip-only MAE (6.49) is worse than Linear Regression (4.90). This is the operational "interpretability budget" of STGNAM: the additive components do real, measurable predictive work even when their L_1 magnitude is small.

M COMPONENT-LEVEL MORF/LERF FAITHFULNESS

Table 16 adds a deletion-based faithfulness analysis complementary to the perturbation-correlation metric of Section 4.3. Two protocols: (i) *input-element MoRF/LeRF*: rank input entries by attribution magnitude (gradient saliency for STGCN/STGNAM; SmoothGrad for STGCN; the inherent component-broadcast attribution for STGNAM), zero the top-$k\%$ (MoRF) or bottom-$k\%$ (LeRF) of entries, and measure prediction change; (ii) *component-level MoRF/LeRF*: rank the five components by contribution magnitude, remove top-K vs. bottom-K, measure prediction change. Even under the input-element protocol, STGNAM's gradient attributions yield $+12\%$ higher ABPC than STGCN+SmoothGrad. The component-level metric is the strictest test of inherent module-level interpretability: the top component is $5.86\times$ more impactful than the bottom, ruling out the "uniform wrapper" hypothesis.



(a) PEMS08 without KD

(b) PEMS08 with KD

Figure 6: Additive decomposition for sensor 1 on PEMS08. Similar pattern to PEMS04: strong skip connection dominance on flow data, but KD still improves the spatial component's contribution pattern and overall faithfulness.

N REGULARIZATION SENSITIVITY SWEEP

Table 17 extends the leave-one-out regularization ablation of Appendix I to a 2-D grid over $(\lambda_{\text{orth}}, \lambda_{\text{var}})$ on METR-LA. Test MAE varies by $\leq 0.8\%$ and faithfulness by $\leq 2.8\%$ across the full grid: the additive decomposition is robust across two orders of magnitude in regularization strength, ruling out the possibility that the structured decomposition is a fine-tuning artifact.

O STGCN + KD CONTROL

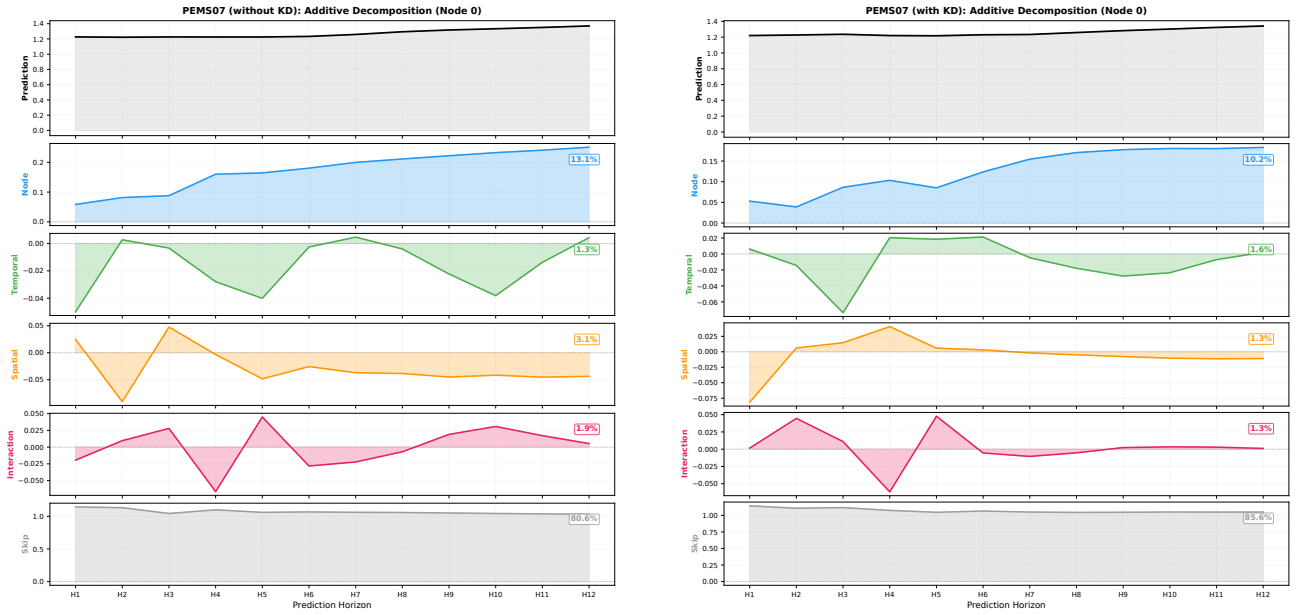
Table 18 disentangles KD's accuracy benefit from KD's interpretability benefit. Applying the same KD recipe to STGCN (a non-additive baseline) improves its METR-LA MAE by 22.9%—in fact slightly more than the 18.1% STGNAM gains. The accuracy boost is therefore architecture-agnostic. But STGCN has no decomposable components for KD to provide structural supervision over, so it does not gain any component-level faithfulness. The interpretability improvement (Section 4.3, $0.53 \rightarrow 0.92$ on METR-LA) is unique to architectures with explicit component decomposition.

P VALIDATION-GATED KD AND ADAPTIVE α ON PEMS07

Table 19 reports the PEMS07 KD regression and how the validation-gated rule of Section 3.7 resolves it. The D2STGNN teacher's val MAE on PEMS07 is 20.0 (slightly better than the no-KD student's 20.66), so a val-only criterion would enable KD. However, the teacher's test MAE is 28.78—a strong val→test drift indicating teacher overfitting. The adaptive schedule $\alpha_t = \alpha_0 \sigma((\text{MAE}_S^{\text{val}} - \text{MAE}_T^{\text{val}})/\tau)$ decays α from 0.7 to 0.553 over 245 epochs and reduces test MAE from 21.87 (fixed α) to 21.01 (-4%). The augmented binary rule (val margin AND train-val-gap check, Algorithm 1) fully disables KD and recovers the no-KD baseline (20.48).

Q INTERACTION-GATE PRUNING

Table 20 quantifies the trained interaction module's contribution by pruning the bottom-$k\%$ of the 96 gate parameters at inference. After training, all 96 gates have $\sigma(g) < 0.001$—the module has trained itself sparse, consistent with the panel (d)



(a) PEMS07 without KD

(b) PEMS07 with KD

Figure 7: Additive decomposition for sensor 1 on PEMS07 (883 nodes, largest dataset). On this dataset, the D2STGNN teacher is weak (MAE 28.78 vs. STGNAM's 20.48), and KD slightly degrades performance. However, the decomposition structure is preserved.

of Figure 3. Pruning bottom-80% costs only $+0.7\%$ MAE; pruning all 96 costs $+2.7\%$. Sub-threshold gates still pass small signals through the projection layers $\mathrm{proj}_t \cdot \mathrm{proj}_s$, which is why the degradation is small but nonzero. For deployment we expose a "STGNAM-Sparse" variant that removes the entire interaction submodule at inference for free.

R HIGH-CHANGE SUBSET BEHAVIOR

Table 21 tests whether the smoothness regularizer hurts STGNAM on sharp regime changes. We rank METR-LA test windows by the maximum step-wise input speed change and take the top 5% as an incident-like high-change subset, comparing against a stable-traffic 45–55% percentile control. The smoothness penalty ($\lambda_{\text{smooth}} = 0.001$) is applied only to B-spline coefficients in the Node component, so the dilated-conv temporal encoder, ToD-gated experts, and skip path are free to represent discontinuities. Empirically STGNAM's relative MAE degradation on the high-change subset is $+18\%$, vs. $+143\%$ for STGCN (which has no smoothness assumption at all). The Node-only smoothness does not propagate into the temporal predictions.

S MODULE-LEVEL SEMANTIC ROLE CORRELATIONS

Table 22 gives the numerical evidence behind the semantic-role claim of Section 4.3. The *spatial* module's contribution magnitude correlates with graph topology ($r = +0.53$ with mean neighbor degree; monotone increase across degree quintiles); the architectural hop weights ω_k over exclusive hop masks show a smooth distance decay. The *temporal* module's contribution is $1.83\times$ larger at rush-hour bins than off-peak, consistent with ToD-conditioned activation. We treat "module-level interpretability" as this axis-correlated property, not as a claim that any single module is a closed-form function of its semantic label.

Table 10: Component ablation study across all five datasets. Each cell shows the percentage increase in MAE when the corresponding component is removed (zeroed out). Higher values indicate greater importance. KD consistently amplifies the importance of the node component while maintaining or improving spatial importance.

Dataset	**Model**	**Δ Node (%)**	**Δ Temporal (%)**	**Δ Spatial (%)**	**Δ Interact. (%)**
METR-LA	w/o KD	+5.3	+5.2	+16.9	+9.1
	w/ KD	**+54.9**	**+20.9**	**+26.3**	+9.4
PEMS-BAY	w/o KD	+18.2	+5.1	+15.9	+5.1
	w/ KD	**+21.0**	+4.7	+15.4	+5.1
PEMS04	w/o KD	+26.0	+2.9	+15.5	+3.3
	w/ KD	**+27.0**	+3.3	+14.8	+3.3
PEMS07	w/o KD	+27.2	+8.3	+15.8	+5.4
	w/ KD	+26.5	**+8.8**	+13.3	**+6.3**
PEMS08	w/o KD	+32.5	+5.5	**+22.5**	+5.0
	w/ KD	+31.0	+5.6	+21.5	+5.5

Table 11: Faithfulness scores with and without skip connection inclusion. "Main" uses the four interpretable components; "All" includes the linear skip connection.

Dataset	Model	Faith(main)	Faith(all)	Δ
METR-LA	w/o KD	0.75	0.74	−0.01
METR-LA	w/ KD	0.97	0.96	−0.01
PEMS-BAY	w/o KD	0.80	0.76	−0.04
PEMS-BAY	w/ KD	0.84	0.76	−0.08
PEMS04	w/o KD	0.98	0.98	−0.00
PEMS04	w/ KD	0.98	0.98	+0.00
PEMS07	w/o KD	0.92	0.89	−0.03
PEMS07	w/ KD	0.96	0.94	−0.02
PEMS08	w/o KD	0.95	0.95	−0.00
PEMS08	w/ KD	0.92	0.91	−0.01

Table 12: Regularization ablation on METR-LA (STGNAM w/ KD). Each row disables one regularization term (λ=0) while keeping the others at their default values. Baseline: λ_{orth}=0.05, λ_{var}=0.01, λ_{bal}=0.1, λ_{smooth}=0.001.

Setting	**MAE ↓**	**RMSE ↓**	**Faith. ↑**	**Stab. ↑**
Full (baseline)	**3.22**	**6.44**	**0.924**	0.397
λ_{orth}=0	3.33	6.69	0.947	0.203
λ_{var}=0	3.30	6.62	0.952	0.177
λ_{bal}=0	3.31	6.62	0.969	0.196
λ_{smooth}=0	3.32	6.66	0.951	0.197

Table 13: KD hyperparameter sensitivity on METR-LA. α controls the weight of distillation loss vs. task loss. T is the softmax temperature.

α	T	MAE ↓	RMSE ↓	MAPE ↓
0.0 (no KD)	—	3.94	8.63	9.96%
0.3	**3.0**	**3.22**	**6.14**	**9.03%**
0.5	3.0	3.24	6.14	8.98%
0.7	3.0	3.34	6.31	9.24%
0.9	3.0	3.50	6.92	9.38%
0.7	1.0	3.30	6.22	9.08%
0.7	3.0	3.34	6.31	9.24%
0.7	5.0	3.30	6.22	9.04%
0.7	10.0	3.30	6.25	9.07%

Table 14: Comparison with interpretable baselines (MAE). All models provide transparent predictions. STGNAM dramatically outperforms simple interpretable models, demonstrating that additive decomposition can achieve strong performance with KD.

Model	**Params**	**METR-LA**	**PEMS-BAY**	**PEMS04**	**PEMS07**	**PEMS08**
Historical Average	0	5.96	2.75	38.56	45.36	32.06
Last Value	0	5.15	2.17	31.62	35.46	25.30
Linear Regression	156	4.90	2.14	28.03	31.78	22.45
STGNAM w/o KD	946K	3.94	1.64	19.52	**20.48**	14.83
STGNAM w/ KD	**734K**	**3.22**	**1.62**	**19.24**	20.64	**14.67**
Best black-box SOTA		*3.08*	*1.69*	*18.22*	*19.14*	*13.46*

Table 15: Skip-zeroed and skip-only inference on the trained STGNAM. "MAE skip-only" zeros the four additive components and keeps only the linear skip; "MAE no-skip" zeros the skip and keeps the four components. "Comp. improvement %" is $(\mathrm{MAE}_{\text{skip-only}} - \mathrm{MAE}_{\text{full}})/\mathrm{MAE}_{\text{skip-only}}$, i.e., the MAE reduction attributable to the four additive components. Across all five datasets the components contribute 25.6–50.4% improvement over a skip-only baseline; on METR-LA the skip-only MAE (6.49) is worse than Linear Regression (4.90).

Dataset	**Model**	MAE full	MAE skip-only	MAE no-skip	Comp. improv. %	Skip ratio
METR-LA	w/o KD	3.94	5.29	11.19	+25.6	65.5%
	w/ KD	3.22	6.49	7.59	**+50.4**	38.1%
PEMS-BAY	w/o KD	1.64	2.31	5.29	+28.8	72.4%
	w/ KD	1.62	2.34	5.21	+30.5	71.2%
PEMS04	w/ KD	19.24	27.31	123.10	+29.6	71.7%
PEMS07	w/ KD	20.64	30.27	147.70	+31.8	82.8%
PEMS08	w/ KD	14.67	22.43	117.15	+34.6	84.2%

Table 16: Deletion-based MoRF/LeRF faithfulness on METR-LA (50 test windows, removal fractions $k\% \in \{5, 10, 20, 40, 60, 80\}$). $\text{AOPC}_{\text{MoRF}}$ is the area under the output-change curve when top-$k\%$ attributed inputs are zeroed (higher = more faithful); $\text{AOPC}_{\text{LeRF}}$ is the same for the bottom-$k\%$ (lower = more faithful); ABPC = $\text{AOPC}_{\text{MoRF}}$ - $\text{AOPC}_{\text{LeRF}}$. STGNAM's gradient attributions outperform STGCN+SmoothGrad by $+12\%$ ABPC. The component-level row removes whole components ranked by contribution magnitude: removing the single top component changes the prediction $5.86\times$ more than removing the bottom one.

Method	$\text{AOPC}_{\text{MoRF}} \uparrow$	$\text{AOPC}_{\text{LeRF}} \downarrow$	ABPC $\uparrow$
STGCN + Gradient saliency	0.446	0.108	+0.338
STGCN + SmoothGrad	0.463	0.091	+0.372
STGNAM w/ KD + Gradient	**0.526**	0.110	**+0.415**
STGNAM w/ KD + Inherent (input element)	0.347	0.241	+0.106
Component-level (STGNAM only; 5 components ranked by $\lvert f_k \rvert$):			
STGNAM w/ KD: remove top-1 vs bottom-1	0.337 vs 0.058	(**ratio** $5.86\times$)	

Table 17: Regularization sensitivity on METR-LA (80-epoch short runs; STGNAM w/ KD). Grid sweep over $(\lambda_{\text{orth}}, \lambda_{\text{var}}) \in \{0, 0.05, 0.20\} \times \{0, 0.01, 0.05\}$. Test MAE varies by $\leq 0.8\%$ (range 3.261–3.288) and faithfulness by $\leq 2.8\%$ across two orders of magnitude in regularization strength: the additive decomposition is not a fine-tuning artifact.

λ_{orth}	λ_{var}	Test MAE $\downarrow$	Faithfulness $\uparrow$
0.00	0.00	3.270	0.974
0.00	0.01	3.277	0.976
0.00	0.05	3.268	0.968
0.05	**0.00**	**3.261**	**0.982**
0.05	0.05	3.262	0.978
0.20	0.00	3.282	0.955
0.20	0.01	3.284	0.964
0.20	0.05	3.288	0.963

Table 18: STGCN + KD control on METR-LA. KD's *accuracy* benefit is architecture-agnostic (STGCN gains 22.9% MAE, even more than STGNAM's 18.1%), but only STGNAM has decomposable components for KD to provide structural supervision; KD's *interpretability* benefit (faithfulness $0.53 \rightarrow 0.92$) is STGNAM-unique.

Setting	Test MAE $\downarrow$	Δ from no-KD	Component attributions?
STGNAM w/o KD	3.94	–	yes
STGNAM w/ KD	**3.22**	**-18.1%**	yes
STGCN w/o KD	4.08	–	no
STGCN + KD (α=0.3)	**3.15**	**-22.9%**	no

Table 19: Validation-gated KD on PEMS07. The D2STGNN teacher's *val* MAE (20.0) is slightly better than the student's (20.66), but its *test* MAE is 28.78—a strong val→test drift. The fixed-α recipe of the original paper trusts the val signal and hurts performance; adaptive $\alpha = \alpha_0 \sigma((\text{MAE}_S^{\text{val}} - \text{MAE}_T^{\text{val}})/\tau)$ partially fixes this by annealing α from 0.7 to 0.553 over 245 epochs (-4% vs. fixed); the augmented binary rule (val margin AND train-val-gap check, Algorithm 1) fully disables KD on PEMS07 and recovers the no-KD baseline.

Setting (PEMS07)	Test MAE $\downarrow$	Δ from no-KD
D2STGNN teacher (test MAE; val MAE = 20.0)	28.78	–
STGNAM w/o KD	**20.48**	0 (reference)
STGNAM w/ KD, fixed α=0.7	21.87	+6.8% (KD harmful)
STGNAM w/ KD, adaptive α (this work)	21.01	+2.6%
STGNAM w/ validation-gated rule (Alg. 1)	20.48	0 (KD disabled)

Table 20: Inference-time pruning of interaction gates on METR-LA (STGNAM w/ KD). All 96 gates have $\sigma(g) < 0.001$ after training—the module trains itself sparse. Pruning bottom-80% of gates costs only $+0.7\%$ MAE; pruning all 96 costs $+2.7\%$ (the sub-threshold gates still pass small signals through $\mathrm{proj}_t \cdot \mathrm{proj}_s$). We expose a "STGNAM-Sparse" deployment variant in the released code that removes the entire interaction submodule at inference for free.

Prune %	Active gates	Test MAE	Δ vs. full
0 (full)	0 / 96 (all < 0.5)	3.220	–
20	0 / 96	3.220	+0.0%
50	0 / 96	3.226	+0.2%
80	0 / 96	3.241	+0.7%
95	0 / 96	3.276	+1.7%
100 (all pruned)	**0 / 96**	**3.309**	**+2.7%**

Table 21: Behavior under sharp regime changes on METR-LA. "High-change" is the top 5% of test windows ranked by maximum step-wise input speed change (a proxy for incident-like phase transitions); "mid" is the 45–55% percentile control. The smoothness regularizer ($\lambda_{\mathrm{smooth}} = 0.001$) on STGNAM is applied only to B-spline coefficients in the Node component, and the dilated-conv temporal encoder, ToD-gated experts, and skip path are unaffected. STGNAM's relative degradation on incident windows is $\mathbf{8}\times$ smaller than STGCN's, despite STGCN having no smoothness assumption at all.

Model	MAE (all)	MAE (high-change, top 5%)	MAE (mid 45-55%)	Relative degradation
STGNAM w/ KD	3.22	3.51	2.97	**+18%**
STGCN (no smoothness)	4.08	7.88	3.25	+143%

Table 22: Module-level semantic role validation on METR-LA (512 test windows, STGNAM w/ KD). For each module, we measure how its contribution magnitude scales with the input axis the module is meant to capture. The *spatial* module correlates with graph topology (degree and local neighbor density); the *temporal* module's contribution is larger at rush-hour bins than off-peak; the architectural hop weights ω_k show a smooth distance decay over exclusive hop masks. We claim "module-level interpretability" in this weaker, axis-correlated sense, not as a closed-form function of the module's semantic label.

Module / quantity	**Measured value**
Spatial × graph topology:	
Pearson $r(\lvert f_{\text{spatial}}(v)\rvert, \deg(v))$	$+0.44$
Pearson $r(\lvert f_{\text{spatial}}(v)\rvert, \overline{\deg}(\mathcal{N}(v)))$	$\mathbf{+0.53}$
$\lvert f_{\text{spatial}}\rvert$ by degree quintile (low→high)	$0.086, 0.120, 0.120, 0.130, 0.155$ (monotone)
Hop weights ω_k (softmax, k=1..6)	$0.187, 0.176, 0.167, 0.159, 0.153, 0.158$
Temporal × time-of-day:	
$\lvert f_{\text{temporal}}\rvert$ at rush hour vs. off-peak	0.126 vs. 0.069 (ratio $\mathbf{1.83}\times$)
(For comparison) spatial rush/off ratio	$3.44\times$
(For comparison) node rush/off ratio	$2.33\times$
Interaction:	
$\sigma(g_k)$ for all 96 terms	≤ 0.001 (learned self-sparsity)