Energy 334 (2025) 137503

Contents lists available at ScienceDirect

Energy

journal homepage: www.elsevier.com/locate/energy

Interpretable spatiotemporal urban energy forecasting

Siyuan Jia [a,1], Xiufeng Liu [b,1], Letian Zhao [a], Chaofan Wang [a], Jieyang Peng [c], Xiang Li [d], Zhibin Niu [a,*]

[a] College of Intelligence and Computing, Tianjin University, 300350 Haihe Education Park, Tianjin, China
[b] Department of Technology, Management and Economics, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark
[c] Department of Electronic Engineering, Tsinghua University, Beijing, China
[d] China Iron and Steel Research Institute Group, 100081, Haidian District, Beijing, China

ARTICLE INFO

Keywords:
Urban energy demand forecasting
Deep learning
Time series imaging
Spatial analysis
Sustainable urban development

ABSTRACT

Accurate and spatially detailed urban energy consumption forecasting is crucial for sustainable urban development. Existing methods often fail to capture the complex interplay of spatial and temporal factors influencing energy demand, hindering interpretability and limiting their effectiveness for targeted interventions. This paper presents a novel deep learning model for interpretable, multi-scale urban energy demand forecasting. Our approach leverages time series imaging to transform discrete energy consumption data into continuous spatial representations, generating energy consumption density maps. These maps are input to a deep learning encoder–forecaster architecture, enabling the model to learn intricate spatiotemporal dependencies. Crucially, by preserving the 2D spatial structure throughout the prediction process, our model offers enhanced interpretability compared to methods that reduce spatial information to 1D. We validate our model with real-world electricity data from Shanghai, demonstrating superior performance against traditional and state-of-the-art benchmarks across various spatial granularities and forecasting horizons. For a 7-day forecast, our model achieves a Mean Squared Error (MSE) of 6.032. The resulting interpretable forecasts, visualized as density maps, provide actionable insights for urban planners, policymakers, and utility operators, promoting energy efficiency and facilitating the integration of renewable energy sources into the urban fabric.

1. Introduction

Urban centers, while engines of economic growth and innovation, are also disproportionate consumers of energy and significant contributors to global greenhouse gas emissions [1]. As the world's population increasingly concentrates in urban areas, understanding and managing energy demand in these complex environments becomes paramount to achieving global sustainability goals [2]. Buildings, in particular, represent a substantial portion of urban energy consumption, accounting for up to 40% of total energy use and a significant share of CO_2 emissions [3]. To mitigate the environmental impact of cities and transition towards more sustainable urban environments, accurate, granular, and robust energy demand forecasting methods are crucial. Accurate energy demand forecasting is fundamental for efficient urban energy management, impacting decisions for utilities, consumers, and urban planners. For utilities, precise forecasts optimize power generation, balance supply and demand, and ensure grid stability, particularly with increasing renewable energy integration [4,5]. For consumers, accurate forecasts enable effective demand response, energy conservation, and cost savings [6,7]. Moreover, reliable predictions inform long-term urban planning, guiding infrastructure investments, land use optimization, and energy-efficient building development [8,9].

Traditionally, energy demand forecasting has relied heavily on time-series analysis methods, utilizing historical consumption data to predict future trends [10]. While these methods have proven valuable for capturing temporal dependencies in energy consumption patterns, they often fail to adequately account for the significant spatial heterogeneity inherent to urban environments. Factors such as population density, building typology, socio-economic activity, and micro-climatic variations can significantly impact energy consumption patterns across different locations within a city [11]. Neglecting these spatial dynamics can lead to inaccurate forecasts, hindering the effectiveness of energy management strategies and potentially undermining efforts toward building more sustainable and resilient cities. Furthermore, energy consumption data is characterized by a high degree of multivariate complexity and high latitude, which presents a significant

* Corresponding author.
E-mail addresses: jsiyuan@tju.edu.cn (S. Jia), xiuli@dtu.dk (X. Liu), zlt030814@tju.edu.cn (L. Zhao), wcf@tju.edu.cn (C. Wang), pengjieyang1991@gmail.com (J. Peng), xli@berkeley.edu (X. Li), zniu@tju.edu.cn (Z. Niu).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.energy.2025.137503
Received 11 January 2025; Received in revised form 1 March 2025; Accepted 8 July 2025
Available online 24 July 2025

challenge in fully capturing its inherent features. Recognizing the limitations of traditional approaches, recent research has explored the potential of deep learning techniques, particularly convolutional neural networks (CNNs), to address the challenges of spatiotemporal data analysis in energy demand forecasting [12]. CNNs, originally developed for image recognition tasks, have shown remarkable capabilities in extracting complex spatial patterns from data [13]. Shallow convolutional networks excel at capturing local features like individual electricity consumption patterns, as the network deepens, each layer of convolution further abstracts the features extracted by the previous layer, enabling multi-scale urban energy consumption predictions for both individuals and regions. Coupled with time series imaging techniques, which convert traditional time-series data into image-like representations, CNNs can be leveraged to effectively capture both spatial and temporal dependencies in energy consumption data. This approach has been successfully applied in various domains, including precipitation nowcasting [14], traffic flow prediction [15], and, more recently, energy consumption forecasting [16–18].

Building on these advancements, this paper proposes a novel deep learning model for interpretable, multi-scale urban energy demand forecasting. Unlike existing methods that often reduce spatial information to one-dimensional representations, our model explicitly incorporates spatiotemporal information through a two-dimensional time series imaging approach, preserving crucial spatial context for enhanced interpretability. Our approach leverages kernel density estimation (KDE) to transform discrete, spatially distributed energy consumption data into continuous spatial representations, generating energy consumption density maps. These maps serve as input to a deep learning encoder–forecaster architecture, enabling the model to learn complex spatiotemporal dependencies and generate accurate forecasts across a range of spatial granularities.

Building on these advancements, this paper proposes a novel deep learning model for interpretable, multi-scale urban energy demand forecasting. Unlike existing methods that often reduce spatial information to one-dimensional representations, our model explicitly incorporates spatiotemporal information through a two-dimensional spatial representation time series imaging approach, preserving crucial spatial context for enhanced interpretability. Our approach leverages kernel density estimation (KDE) to transform discrete, spatially distributed energy consumption data into continuous spatial representations, generating energy consumption density maps. These maps serve as input to a deep learning encoder–forecaster architecture, enabling the model to learn complex spatiotemporal dependencies and generate accurate forecasts across a range of spatial granularities. A key advantage of our approach lies in its enhanced interpretability. By preserving the 2D spatial structure of the data throughout the prediction process, our model generates forecasts that can be readily visualized and understood in their spatial context. This allows stakeholders to gain a deeper understanding of how geographical factors influence energy consumption patterns and to develop more targeted and effective energy management strategies.

This work makes the following key contributions to the field of sustainable urban development:

- **A novel framework for interpretable, multi-scale urban energy demand forecasting using 2D spatiotemporal deep learning:** We introduce a novel framework that integrates time series imaging, KDE, and a deep learning encoder–forecaster architecture. This framework explicitly incorporates spatial heterogeneity by working directly with 2D density maps, enabling more accurate and granular forecasts at multiple spatial scales, from individual buildings to city districts, while maintaining crucial spatial context for enhanced interpretability.
- **End-to-end prediction and reconstruction for flexible, area-specific energy demand forecasting:** Our model employs a novel end-to-end prediction approach using an encoder–forecaster architecture trained to directly predict future energy consumption density maps. We introduce novel reconstruction methods that extract energy demand values from these density maps for any area of interest, providing a powerful tool for multi-scale targeted energy management and planning.
- **Rigorous validation demonstrating superior performance and enhanced interpretability for informed urban planning:** We rigorously evaluate our model using real-world electricity consumption data from Shanghai, China. Our results demonstrate superior accuracy compared to traditional forecasting methods and other state-of-the-art spatiotemporal prediction models. Furthermore, we highlight the enhanced interpretability of our approach, demonstrating how the generated density maps provide valuable spatial insights for urban planners and policymakers.

The remainder of this paper is structured as follows: Section 2 reviews the related work. Section 3 presents the dataset used in this study; Section 4 presents the methods; Section 5 conducts experiments to evaluate the model; Section 6 discusses the related issues and Section 7 concludes the paper and presents the future work.

2. Related work

Accurately forecasting urban energy demand is crucial for achieving a more sustainable future for cities. It enables informed planning, optimized resource allocation, and informed policies to promote energy efficiency and the integration of renewable energy sources [5]. This section reviews existing energy demand prediction methods, highlighting their strengths, limitations, and relevance to sustainable urban development. We begin by exploring traditional approaches, focusing on their ability to address the unique challenges of forecasting in complex urban environments. Subsequently, we delve into the emerging field of deep learning-based spatiotemporal forecasting, which holds significant promise for capturing the intricate interplay of spatial and temporal factors driving energy consumption in cities.

2.1. Traditional energy demand forecasting approaches

Energy consumption forecasting methods can be categorized based on their underlying principles and complexity. Somu et al. [11] classify these methods into three main categories: engineering methods, statistical methods, and artificial intelligence (AI) methods. Another common classification within the building energy domain distinguishes between black-box, white-box, and gray-box models [19]. Although these classifications offer valuable perspectives, we have categorized them into statistical, machine learning, and artificial intelligence (AI) methods due to their comprehensive representation of the diverse methodologies employed in energy demand forecasting.

Table 1 presents a selection of state-of-the-art methods within these three categories, along with a summary of their strengths and weaknesses.

Statistical methods offer a computationally efficient approach for short-term energy demand forecasting, particularly when dealing with relatively stable consumption patterns. However, their limitations in capturing non-linear relationships and external influences hinder their accuracy for long-term predictions and their ability to adapt to the dynamic nature of urban environments [45]. Machine learning methods offer a promising approach for energy demand forecasting by detecting patterns in data without extensive domain knowledge. Techniques like regression, support vector machines, and decision trees can model both linear and nonlinear relationships, enabling more accurate short and medium-term predictions than traditional methods. ML models can also adapt to changing consumption patterns and integrate external factors like weather and socio-economic data, making them useful for dynamic urban environments. However, handling complex, high-dimensional, and long-term dependent time-series data often requires significant

Table 1
List of some state-of-the-art methods for energy demand forecasting.

Ref.	Methods	Description	Strengths and Weaknesses
Statistical methods			
Ramanathan et al. [20]	Multiple regression	This study develops multiple regression models for short-term forecasting of hourly system loads, utilizing historical data on electrical loads and weather conditions. These models aim to capture the relationship between energy demand and multiple explanatory variables, improving forecast accuracy.	Strengths: Computationally efficient, interpretable. Weaknesses: Limited in capturing non-linear relationships, less adaptable to dynamic environments.
Pappas et al. [21]	ARIMA	This study utilizes ARIMA models to simulate and forecast energy production and consumption in Asturias, Spain, highlighting the model's applicability for regional energy planning and policy analysis.	Strengths: Effective for time series with temporal dependencies, accounts for seasonality. Weaknesses: Assumes data stationarity, struggles with complex patterns.
Reikard [22], Huang et al. [23], Atique et al. [24], Alsharif et al. [25]	ARMA, ARIMA, SARIMA	Autoregressive Moving Average (ARMA), Autoregressive Integrated Moving Average (ARIMA), and Seasonal ARIMA (SARIMA) models are widely used statistical methods for energy demand forecasting, particularly in applications involving seasonal or cyclical patterns. These models have been successfully applied to solar forecasting and other renewable energy applications.	Strengths: Widely applicable, captures linear temporal dependencies and seasonality. Weaknesses: Limited in capturing non-linearities and spatial variations, requires stationarity.
Server et al. [26], Walter and Sohn [27]	Linear regression	Linear regression models are a simple yet widely used statistical method for electricity load forecasting and building energy monitoring, providing a baseline for assessing the performance of more complex methods.	Strengths: Simple, computationally efficient, interpretable baseline. Weaknesses: Oversimplifies complex relationships, low accuracy for long-term forecasts.
Lazos et al. [28]	Time series analysis	Time series analysis techniques, particularly autoregressive models, are widely employed in energy demand forecasting. These models correlate the future value of a variable with its past values, capturing temporal dependencies in energy consumption patterns.	Strengths: Computationally efficient, captures temporal autocorrelation. Weaknesses: Ignores spatial heterogeneity, limited non-linear pattern recognition.
Xu et al. [29]	Grey model, Time response function (TRF) and nonlinear optimization method	This study proposes a novel grey model incorporating an optimized Time Response Function (TRF) to enhance short-term electricity consumption forecasting. The authors utilize a nonlinear optimization method to fine-tune the model's parameters, improving its accuracy and adaptability for predicting electricity demand.	Strengths: Effective with limited data, captures non-linear trends to some extent. Weaknesses: Less accurate with large datasets, limited in capturing complex spatial–temporal dynamics.
Wu et al. [30]	Grey convex relational analysis, GMC(1,N) model	This study develops a novel multi-variable grey forecasting model for electricity consumption forecasting that explicitly considers the influence of population growth. The model employs grey convex relational analysis and is optimized using the GMC(1,N) model with fractional-order accumulation, enhancing its predictive capabilities.	Strengths: Incorporates multiple variables, improved accuracy over basic grey models. Weaknesses: Still limited by grey model assumptions, less adaptable than AI methods.
Mui et al. [31]	Bayesian regularization and a genetic algorithm	This study focuses on estimating annual cooling energy consumption for diverse building types in subtropical regions, using a hybrid simulation approach that combines Bayesian regularization with a genetic algorithm to optimize model parameters. The authors propose a generalized method for reducing energy consumption and greenhouse gas emissions in buildings.	Strengths: Accounts for uncertainty, optimizes model parameters effectively. Weaknesses: Computationally intensive, complexity in implementation.
Wang et al. [32]	A structural adaptive Caputo fractional grey prediction model (FCSAGM)	This study introduces a novel grey prediction model for energy consumption forecasting, employing Caputo fractional derivatives to capture the dynamics of energy consumption patterns. The FCSAGM model's parameters are adjusted using the particle swarm optimization (PSO) method, enhancing its adaptability and accuracy for forecasting energy demand.	Strengths: Captures complex dynamics, adaptive parameter tuning. Weaknesses: Grey model limitations, computational cost of PSO.

(*continued on next page*)

Table 1 (*continued*).

Machine learning methods			
Voulis et al. [33]	K-means clustering and logistic regression	This study employs K-means clustering and logistic regression to systematically analyze electricity demand patterns at different city scales in the Netherlands. The research highlights the spatial variations in energy consumption and provides insights into the factors influencing demand across different urban areas.	Strengths: Captures spatial variations, interpretable clusters. Weaknesses: Limited to linear relationships, clustering accuracy depends on feature selection.
Dubey et al. [34]	ARIMA, SARIMA, LSTM	This study compares the performance of ARIMA, SARIMA, and Long Short-Term Memory (LSTM) models for predicting daily energy consumption using smart meter data in London. The research provides a comprehensive evaluation of these methods, highlighting their strengths and limitations for urban energy demand forecasting.	Strengths: Captures temporal dependencies, LSTM handles non-linearities. Weaknesses: Ignores spatial data, LSTM computationally intensive.
Feng et al. [35]	A stochastic shading building model	This study investigates the uncertainty associated with shading in building energy models, using a stochastic shading model that incorporates time, temperature, solar radiation, and shading coefficients. The research employs machine learning algorithms, the Shapley Value Method, and hyperparameter optimization to refine and optimize the model, aiming to improve the accuracy of energy consumption predictions for buildings.	Strengths: Quantifies uncertainty, physics-informed approach. Weaknesses: Complexity in model development, computationally intensive.
Artificial intelligence methods			
Peng et al. [36]	Empirical Wavelet Transform (EWT)-attention-LSTM	This study proposes a novel hybrid model for long-term energy consumption prediction, combining Empirical Wavelet Transform (EWT), an attention mechanism, and an LSTM network. The EWT module decomposes the input data into multiple frequency components, enhancing the model's ability to capture complex temporal patterns.	Strengths: Hybrid approach, captures complex temporal patterns effectively. Weaknesses: Black-box nature, computationally intensive.
Jin et al. [37]	Singular spectrum analysis (SSA) and parallel long short term memory (PLSTM)	This study introduces a parallel LSTM model for energy consumption forecasting at both the appliance level and the individual appliance level. The model utilizes Singular Spectrum Analysis (SSA) to decompose the input data into multiple sub-signals, improving prediction accuracy by capturing different frequency components of the energy consumption data.	Strengths: Appliance-level forecasting, captures multi-frequency temporal patterns. Weaknesses: High data demand, computational complexity.
Elbeltagi and Wefki [38]	ANNs	This study employs Artificial Neural Networks (ANNs) to improve the prediction of energy usage for residential buildings. The research highlights the effectiveness of ANNs in capturing non-linear relationships and complex patterns in building energy consumption data.	Strengths: Captures non-linear relationships, adaptable to various data types. Weaknesses: Black-box nature, requires large datasets.
Jin et al. [39]	Deep Reinforcement Learning (DRL)	This study proposes a novel method for building energy consumption prediction that leverages Deep Reinforcement Learning (DRL). The model focuses on improving prediction accuracy at fluctuation points, which are critical for energy management and demand response applications.	Strengths: Optimized for fluctuation points, potential for real-time control. Weaknesses: Complex to train, high computational cost.

(*continued on next page*)

Table 1 (*continued*).

Liu et al. [40]	A3C, DDPG and RDPG	This study investigates the use of three prominent Deep Reinforcement Learning (DRL) techniques – A3C, DDPG, and RDPG – for energy consumption forecasting in an office building. The research compares their performance to traditional supervised learning models, highlighting the potential of DRL for optimizing energy management strategies in buildings.	Strengths: DRL for energy management, adaptive learning. Weaknesses: High complexity, requires extensive hyperparameter tuning.
Zhong et al. [41]	Vector field-based support vector regression (SVR)	This study introduces a novel vector field-based Support Vector Regression (SVR) model for energy demand prediction. The model transforms the highly non-linear relationship between input and output variables into a linear relationship, enhancing prediction accuracy, robustness, and generalization ability.	Strengths: Handles non-linearities, robust and generalizable. Weaknesses: Computational cost with large datasets, less effective than deep learning for complex patterns.
Ozcan et al. [42]	RNN with dual-stage attention mechanism	This study proposes a Recurrent Neural Network (RNN) model for electric load prediction, incorporating a dual-stage attention mechanism in both the encoding and forecasting stages. This mechanism enables the model to selectively focus on the most relevant temporal features, improving prediction accuracy.	Strengths: Attention mechanism for feature selection, captures temporal dependencies. Weaknesses: RNN limitations with long sequences, computationally intensive.
Muralitharan et al. [43]	Neural network + GA + PSO	This study explores a novel approach for optimizing neural networks for energy demand prediction using Genetic Algorithms (GA) and Particle Swarm Optimization (PSO). The research finds that the GA-based method performs better for short-term prediction, while the PSO-based method is more suitable for long-term forecasting.	Strengths: Optimized neural networks, GA/PSO for parameter tuning. Weaknesses: Increased complexity, computational cost of optimization.
Le et al. [44]	CNN + Bi-LSTM	This study proposes a hybrid model combining a Convolutional Neural Network (CNN) and a Bidirectional Long Short-Term Memory (Bi-LSTM) network to enhance electricity consumption prediction accuracy. The model leverages the strengths of both CNNs for spatial feature extraction and Bi-LSTMs for capturing temporal dependencies.	Strengths: Hybrid CNN-BiLSTM, captures spatiotemporal features. Weaknesses: Complexity in model design, computationally intensive.

feature engineering and intervention, limiting generalization. Artificial intelligence methods have emerged as a powerful tool for energy demand forecasting, demonstrating significant advancements in capturing complex patterns and improving prediction accuracy. Within AI methods, Zeroing Neural Networks (ZNNs) have been explored for dynamic system problem-solving, with research focusing on enhancing their convergence speed and robustness [46,47]. Hybrid approaches, combining multiple AI techniques and incorporating advanced data processing methods, offer promising avenues for addressing the challenges of forecasting in urban settings [35,48–51]. Despite these advancements, traditional methods often treat energy consumption as a single time series, overlooking the significant spatial variations inherent to urban environments. This approach limits their ability to provide the granular, location-specific forecasts necessary for effective urban energy management and planning in the context of sustainable development goals.

2.2. Deep learning-based spatiotemporal forecasting for sustainable cities

Recognizing the importance of spatial heterogeneity, recent research has focused on developing deep learning-based spatiotemporal forecasting models, particularly those leveraging convolutional neural networks (CNNs) to extract complex spatial patterns from data [52,53]. These models offer a promising avenue for providing accurate, granular, and dynamically adaptable forecasts crucial for building sustainable and resilient urban energy systems. However, many existing approaches, while incorporating spatial features, often reduce the problem to one-dimensional representations, effectively flattening the spatial context and limiting the interpretability of the resulting forecasts.

Several studies have demonstrated the effectiveness of CNN-based approaches for spatiotemporal energy demand forecasting. Peng et al. [52] introduced a potential flow-based spatiotemporal model for urban energy demand. Bu and Cho [54] proposed a deep learning model combining multi-headed attention with a convolutional recurrent neural network to predict residential energy consumption. Other research has leveraged CNNs in conjunction with recurrent neural networks (RNNs), such as Long Short-Term Memory (LSTM) networks, to capture both spatial and temporal dependencies [11,44,55–58]. Recent advancements have incorporated attention mechanisms to enhance performance. Ozcan et al. [42] proposed a deep-learning model using dual-stage attention-based RNNs, and Lilhore et al. [59] presented a hybrid deep learning model with two-way attention and multi-objective particle swarm optimization for short-term load prediction. Additional studies explicitly modeled spatial autocorrelation effects [60–63]. Furthermore, Peng et al. (2024) proposed a spatiotemporal feature fusion model based on graph neural networks for user-level energy consumption forecasting. These studies highlight the importance of considering spatiotemporal information for accurate energy demand prediction.

Despite these advancements, a key limitation of many existing spatiotemporal forecasting models is their reduced interpretability. By transforming spatial data into 1D representations, these models obscure the spatial relationships and distributions crucial for understanding the factors driving energy demand and for developing targeted interventions. Moreover, existing methods often overlook the multi-scale nature of urban energy demand, which varies significantly across different spatial granularities, from individual buildings to neighborhoods and city-wide levels.

This research addresses these limitations by proposing a novel deep learning model for interpretable, multi-scale urban energy demand forecasting. Our model integrates time series imaging, kernel density estimation, and a deep learning encoder–forecaster architecture to capture the complex spatiotemporal dynamics of urban energy consumption. Critically, our model preserves the 2D representation of spatial data, allowing for direct visualization of predicted energy consumption patterns and enabling more informed decision-making. Furthermore, we introduce reconstruction methods to calculate energy demand values for specific areas of interest from the predicted density maps, facilitating informed decision-making at various spatial scales. By providing accurate, granular, and interpretable energy demand forecasts, our model contributes to the development of more sustainable, efficient, and resilient cities.

3. Materials

In this study, we utilize real-world electricity consumption data from the Pudong District of Shanghai, China. Shanghai, a rapidly growing megacity and a global economic hub, faces significant challenges related to energy sustainability and efficient resource management [64]. We selected Shanghai as the case study city due to its status as a representative megacity with substantial energy consumption and well-developed smart grid infrastructure, ensuring data availability and relevance to global urban energy challenges. The Pudong District, located east of the Huangpu River, is characterized by a diverse mix of land uses, ranging from high-density commercial and financial centers to residential neighborhoods, making it an ideal case study for exploring the spatiotemporal dynamics of urban energy consumption. The electricity consumption data, collected from smart meters by the State Grid Corporation of China (SGCC), was preprocessed using Kernel Density Estimation (KDE) to transform discrete point readings into continuous energy density maps, which then serve as input to our deep learning model for spatiotemporal forecasting. This process allows us to capture both spatial and temporal variations in urban energy demand.

Our dataset encompasses a three-year period from July 1, 2015, to June 23, 2018. It includes electricity consumption data from 9,333 anonymized sample customers within Pudong, recorded at a 12-hour resolution via smart meters, ensuring relatively high accuracy and completeness. Each customer is associated with specific longitude and latitude coordinates, enabling spatial analysis of consumption patterns across the district. To highlight the spatial variations in energy demand, we focus on two representative areas: one primarily commercial and the other residential. Fig. 1 illustrates the distinct load curves for these areas in 2017. As depicted, commercial customer consumption is notably higher and more stable annually compared to residential consumption, which exhibits seasonal variations. This difference is attributed to the continuous operation of commercial buildings, which often rely on energy-intensive systems for lighting, HVAC, and other equipment. In contrast, residential electricity consumption is influenced by seasonal factors and displays cyclical variations, with higher consumption during the summer and winter months likely due to increased use of air conditioning and heating systems.

Prior to model training, the raw data underwent several preprocessing steps to ensure quality and suitability. Missing values, less than 1% of the dataset, were imputed using linear interpolation. Outlier detection was performed using a z-score threshold of 3, and identified outliers were capped to the 99th percentile. Energy consumption values were normalized to a [0, 1] range using min–max scaling for improved model stability. Finally, spatial coordinates were transformed to a local Cartesian system to facilitate KDE processing. The dataset exhibits high quality and completeness due to the use of smart meter technology and the rigorous data collection process by SGCC. Smart meters provide accurate and reliable measurements of electricity consumption. The completeness is high, with missing values less than 1%, which were handled using linear interpolation. However, like any real-world dataset, potential issues such as sensor errors or communication disruptions might exist, although expected to be minimal. SGCC, as a major utility provider, maintains strict quality control measures, further ensuring data reliability. The resulting preprocessed dataset, derived from real-world smart meter readings in Shanghai Pudong, provides a robust foundation for evaluating our deep learning model's ability to capture complex spatiotemporal urban energy consumption dynamics and contribute to sustainable urban development.

The above analysis demonstrates the significant influence of both spatial and temporal information on electricity consumption, highlighting the importance of incorporating both aspects into our prediction model. This enables accurate electricity consumption forecasting for any area of interest, contributing to more effective urban energy management and planning towards sustainable urban development. This real-world data from Shanghai's Pudong District will be used to train and evaluate our proposed deep learning model in the subsequent sections, allowing us to assess its effectiveness in capturing the complex spatiotemporal dynamics of urban energy consumption and its potential for contributing to the development of more sustainable cities.

4. Methodology

This section will first define the research problem, then present an overview of our proposed deep learning-based model for interpretable, multi-scale urban energy demand forecasting, and finally explain the modules in detail.

4.1. Problem formulation

Accurate urban energy demand forecasting is essential for sustainable urban development, yet existing methods struggle to provide both high accuracy and interpretability, particularly when considering the complex interplay of spatial and temporal factors. Traditional time series forecasting approaches, while effective at capturing temporal patterns, often neglect the significant spatial heterogeneity inherent in urban environments. Many spatiotemporal methods, in their attempt to incorporate spatial data, reduce it to one-dimensional representations, sacrificing valuable spatial context and consequently limiting the interpretability of forecasts. This lack of interpretability hinders the ability to gain actionable insights for targeted energy management interventions and informed urban planning. Therefore, the core problem we address is the need for an energy forecasting model that is not only accurate and multi-scale but also inherently interpretable, preserving spatial information to enable effective decision-making.

To address this problem, we propose a novel approach focusing on interpretable, multi-scale spatiotemporal forecasting. We begin by defining the conventional time series forecasting problem, where the goal is to predict future energy consumption (X) based on historical data: The time sequence of energy consumption can be defined as $X_{1:t} = [X_1, X_2, \dots, X_t]$, where $X_t \in \mathbb{R}^C$ represents the 1D tensor of energy consumption measurements for C customers at time t. The time series forecasting problem of energy consumption can be expressed as:

$$\hat{X}_{t+1}, \dots, \hat{X}_{t+K} = \underset{X_{t+1},\dots,X_{t+K}}{\arg\max}\ p(X_{t+1}, \dots, X_{t+K} | X_{t-T+1}, \dots, X_t), \tag{1}$$

where $X_{t-T+1}, \dots, X_t$ denotes the given tensor sequence with a time step T. Time prediction based on the historical tensor sequence involves forecasting the most probable future sequence of length K.

However, as noted, this approach lacks spatial context. To incorporate spatial information and enhance interpretability, we transform the discrete spatial consumption readings into a continuous 2D representation using Kernel Density Estimation (KDE) (illustrated in Fig. 2 and detailed in Section 4.3). This generates a grid of $N \times N$ cells, where each cell represents a pixel in a density map. This allows us to model the spatiotemporal energy consumption problem (now represented by

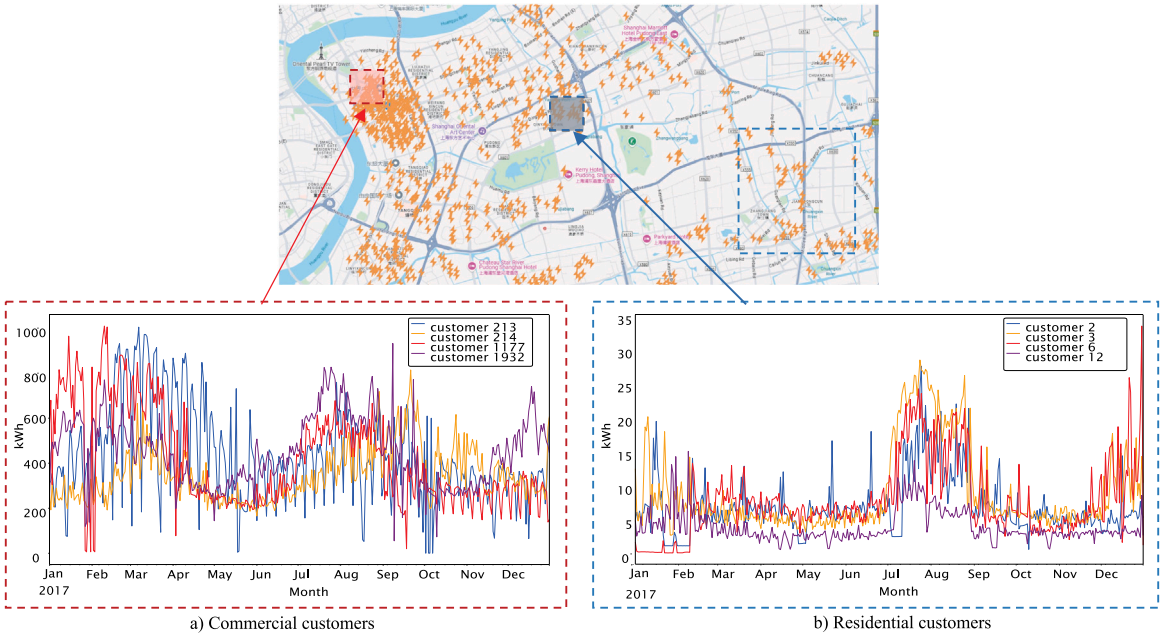

Fig. 1. Electricity consumption patterns in commercial and residential areas within Shanghai's Pudong District (2017). The figure displays the 12-hour resolution load curves for representative commercial and residential areas in Pudong, Shanghai, for the year 2017. It clearly illustrates the higher overall consumption and stability in commercial areas compared to the fluctuating, seasonally-influenced residential consumption.

$\mathcal{X}$) as a two-dimensional dense prediction problem in image processing: A spatiotemporal sequence of measurements, denoted as $\mathcal{X}$ (distinct from the temporal data X), is defined as $\mathcal{X}_{1:t} = [\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_t]$, where $\mathcal{X}_t \in \mathbb{R}^{N^2 \times (C+D)}$. Here, D is the number of random sampling measures within a coordinate, and C is the dimension of the coordinate (longitude, latitude), i.e., $C = 2$. The energy consumption density at grid cell (x, y) at time slot t is represented by $\mathcal{X}_t(x, y)$. The energy demand forecasting task translates to predicting energy consumption density for all map grid cells over K time steps after time t, using historical data with a time step T. The spatiotemporal sequence prediction is defined as:

$$\hat{\mathcal{X}}_{t+1}(x,y), \ldots, \hat{\mathcal{X}}_{t+K}(x,y) = \underset{\mathcal{X}_{t+1}(x,y),\ldots,\mathcal{X}_{t+K}(x,y)}{\arg\max} p(\mathcal{X}_{t+1}(x,y), \ldots, \mathcal{X}_{t+K}(x,y) | \mathcal{X}_{t-T+1}(x,y), \ldots, \mathcal{X}_t(x,y)) \tag{2}$$

This 2D representation is central to our model's enhanced interpretability, enabling direct visualization of the predicted energy distribution across the urban landscape.

4.2. Overview of the proposed model

This section presents an overview of our proposed novel deep learning model for interpretable, multi-scale urban energy demand forecasting. The model addresses the limitations of existing approaches by incorporating spatial heterogeneity through a unique combination of time series imaging, kernel density estimation (KDE), and a robust encoder–forecaster architecture. Critically, by preserving the 2D spatial context of the data, our model offers enhanced interpretability compared to 1D methods. This architecture, trained end-to-end, enables accurate and granular predictions across various spatial scales, providing valuable insights for diverse stakeholders. Figs. 3 and 4 provide visualizations of the framework and its procedure.

The process begins with raw energy consumption data collected from various sources, such as smart meters, building management systems, and publicly available datasets. This raw data, often discrete and irregularly distributed in space, is first passed through a data preprocessing layer. Here, we employ KDE to create a continuous spatial representation of energy consumption for each time step, generating a series of energy consumption density maps. This transformation to continuous density maps is crucial for enabling the subsequent application of convolutional neural networks (CNNs), which are particularly effective at extracting spatial features. To further enhance the model's robustness and ability to generalize to unseen data, we employ a noise injection technique during training. This involves adding random perturbations to the density maps, which helps to prevent overfitting and improves the model's ability to handle noisy or incomplete real-world data.

The resulting sequence of energy consumption density maps serves as input to the core of our model: the AI-based spatiotemporal forecasting engine. This engine utilizes a novel encoder–forecaster architecture based on stacked neural networks. This design is specifically chosen for its ability to capture the intricate spatiotemporal dependencies inherent in urban energy consumption data. The encoder network effectively compresses the input sequence into a lower-dimensional latent representation, capturing the essential spatial and temporal features. This compressed representation is then passed to the forecaster network, which decodes the latent information to generate a sequence of predicted energy consumption density maps. Within this encoder–forecaster framework, we explore and compare four prominent AI-based spatiotemporal prediction models — ConvLSTM, ConvGRU, PredRNN, and SA-ConvLSTM — each tailored to leverage the unique characteristics of urban energy consumption data. This comparative analysis allows us to identify the most effective model for this specific application.

Finally, our novel reconstruction module processes the predicted density maps, bridging the gap between the continuous spatial representations and the desired output: time series data for specific locations or areas of interest. This module employs a time series reconstruction technique based on bilinear interpolation. This allows us to accurately estimate energy demand values at arbitrary spatial locations within the predicted density maps, offering a high degree of flexibility in generating forecasts for specific areas of interest. This enables our model to provide granular, location-specific forecasts that can inform targeted energy management strategies and support data-driven decision-making for sustainable urban planning. The final output is a predicted time series of energy demand, tailored to the specific spatial scale and location desired by the user, facilitating informed decision-making for a variety of stakeholders.

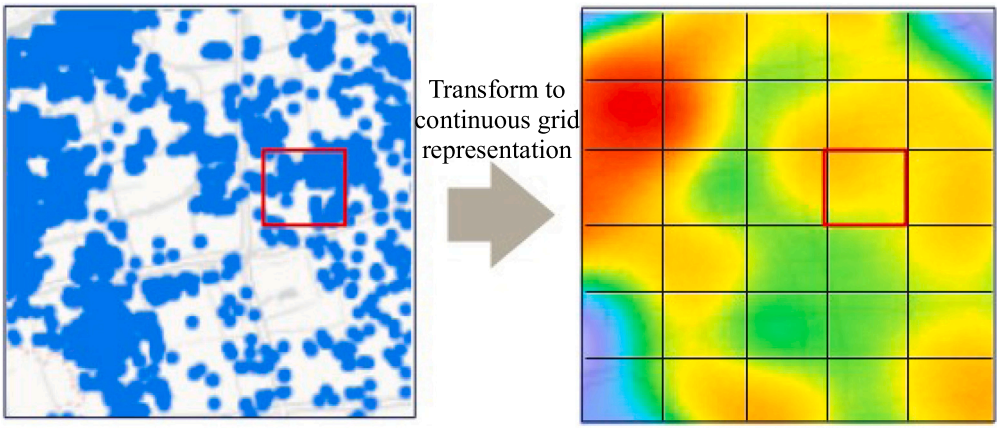

Fig. 2. Transforming discrete spatial data into a continuous 2D representation via Time Series Imaging using KDE. The left panel shows discrete energy consumption data points. The right panel represents the generated energy consumption density map via Kernel Density Estimation (KDE), illustrating the conversion of point-based time series data into a continuous spatial representation for each time step, effectively creating a "time series image".

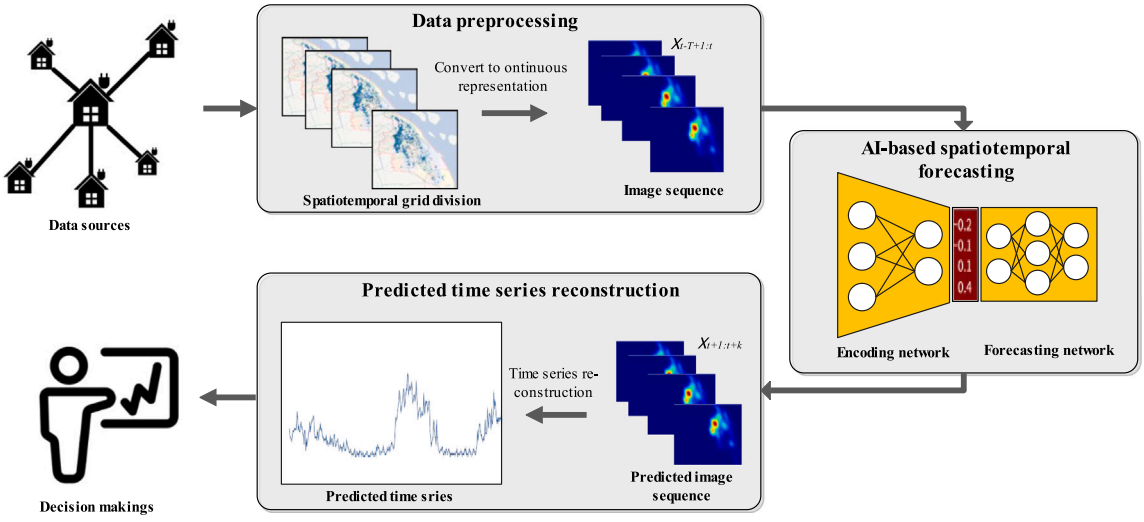

Fig. 3. Overview of the proposed spatiotemporal prediction model. The framework consists of four main stages: Data Preprocessing, Spatiotemporal Forecasting Engine, Reconstruction, and Output. The Data Preprocessing stage transforms raw energy consumption data into density maps using KDE. The Forecasting Engine, based on an encoder–forecaster architecture with stacked AI-based spatiotemporal prediction models, predicts future density maps. The Reconstruction module extracts energy demand forecasts for specific areas of interest. Finally, the Output provides interpretable, multi-scale energy demand forecasts.

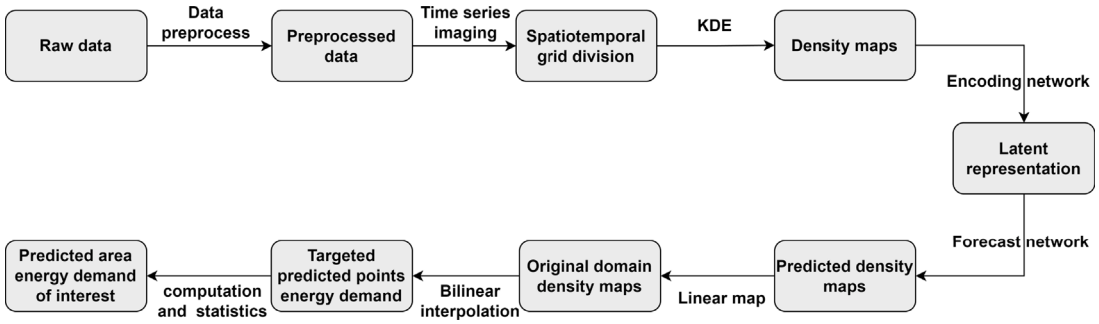

Fig. 4. Step-by-step procedure of the proposed framework. This flowchart details the sequential steps of our proposed framework, from raw data input to final interpretable forecast output. It illustrates the flow through data preprocessing using KDE, encoding and forecasting with AI-based models, and reconstruction via bilinear interpolation to obtain area-specific energy demand predictions.

4.3. Data preprocessing

As the data are spatially discrete and irregularly distributed, we employ a kernel density estimation (KDE)-based approach to encode discrete household energy consumption into a continuous representation suitable for processing by convolutional networks. Kernel Density Estimation (KDE) serves as our time series imaging technique to transform the raw time series data into a sequence of spatial density maps. Specifically, for each time step in our energy consumption time series data, we apply KDE to convert the discrete customer consumption points into a continuous spatial density map, representing the energy consumption distribution across the urban area at that specific time. This process is repeated for every time step, resulting in a sequence of density maps that capture the spatiotemporal evolution of urban energy demand in an image-like format, suitable for input to our CNN-based deep learning model. The KDE function is defined by:

$$\hat{f}_h(x) = \sum_{i=1}^{n} c_i K_h(x - x_i) \tag{3}$$

where x_i denotes the spatial location of customer i in terms of longitude and latitude (lon_i, lat_i), c_i is a vector of $(pap_r, pap_{r1}, pap_{r2})$ which are normalized energy consumption values representing total energy demand, peak period demand and off-peak period demand, respectively. This vector is used to reweight the demand strength with respect to geographic distribution. In this paper, we use the Gaussian kernel to estimate the energy consumption density at each location, defined as:

$$K_h(x - x_i) = e^{-\frac{\|x - x_i\|^2}{2h^2}} \tag{4}$$

where h is the kernel bandwidth that controls the smoothness of the density estimate.

The generated density maps will serve as input data for the training process. The spatial resolution of the density map will influence the accuracy of the forecasting. Low-resolution density maps may lose the detail of spatial energy consumption distribution. However, a high-resolution density map will increase the computational complexity and may reduce the spatial granularity of the prediction, which damages the robustness of forecasting. To balance these factors, we set the width of the grid cell $W = 0.04$ and the kernel bandwidth $h = 0.015$ to obtain the density map and adjust the size of the input to the neural network to 64×64 (the neural network types will be described later). The reason for this choice is that it provides a reasonable trade-off between accuracy and efficiency. We choose W and h based on the average distance between the data points and the desired level of smoothness of the density estimate. We choose the input size of 64×64 based on the optimal performance of the convolution network on image data.

4.4. Proposed model

In this section, we detail our proposed deep learning model, first presenting the encoder–forecaster architecture and then describing the four types of AI-based spatiotemporal modules (ST-Prediction) used for sequence prediction within this architecture. All four AI-based spatiotemporal prediction models (ConvLSTM, ConvGRU, PredRNN and SA-ConvLSTM) utilize convolutional operators rather than matrix multiplication in their recurrent layers to process spatiotemporal data. This approach enables the models to efficiently capture spatial features and temporal dependencies simultaneously. We selected these four representative models to comprehensively evaluate the performance of different spatiotemporal recurrent architectures within our encoder–forecaster framework and identify the most effective approach for urban energy demand forecasting.

4.4.1. Encoder–forecaster architecture

The proposed spatiotemporal prediction model uses an encoder–forecaster architecture [65], a type of neural network commonly used for sequence-to-sequence tasks such as machine translation, summarization, and image captioning. It consists of two main components:

- **Encoder Architecture:** The encoder processes the input sequence of energy consumption density maps and converts them into a compact, latent representation. This representation encodes the essential spatial and temporal features of the input. Our encoder consists of four convolutional layers. The first two convolutional layers have 64 filters with a kernel size of 5×5, followed by max pooling layers with a kernel size of 2×2 and a stride of 2. The third and fourth convolutional layers have 128 filters with a kernel size of 5×5, also followed by max pooling layers with the same parameters. We use ReLU activation functions after each convolutional layer. The output of the final convolutional layer is flattened and passed through a fully connected layer to produce a 128-dimensional latent representation. This compressed representation serves as input to the forecaster.
- **Forecaster Architecture:** The forecaster takes the latent representation produced by the encoder and generates a sequence of predicted energy consumption density maps. Our forecaster mirrors the encoder's structure but in reverse, using transposed convolutional layers for upsampling. It begins with a fully connected layer that reshapes the latent vector to match the encoder's output dimensions. This is followed by four transposed convolutional layers. The first two transposed convolutional layers have 128 filters with a 5×5 kernel and a stride of 2 for upsampling. The next two transposed convolutional layers have 64 filters with the same kernel size and stride. Each transposed convolutional layer is followed by a ReLU activation function. The output of the final transposed convolutional layer is the predicted density map sequence.

This encoder–forecaster architecture is particularly well-suited for urban energy demand forecasting due to its ability to effectively capture both the temporal evolution of energy consumption patterns and the underlying spatial dependencies across the urban landscape. Energy demand is inherently sequential, exhibiting temporal autocorrelation and trends, which encoder–decoder models are designed to handle. Simultaneously, spatial factors like building density and land use significantly influence energy consumption, requiring models to process spatial information. Compared to purely RNN-based approaches, which primarily focus on temporal dependencies, and purely CNN-based models, which excel in spatial feature extraction but may struggle with long-term temporal dynamics, the encoder–forecaster architecture provides a balanced approach. It allows the encoder to compress the input spatiotemporal sequence into a rich latent representation capturing both aspects, while the forecaster decodes this representation to generate future density maps, effectively learning the complex spatiotemporal transitions in urban energy demand.

According to studies [52,66], spatiotemporal shifts in energy demand exhibit fluid dynamics characteristics, often correlated with temperature or social activity changes. Spatial correlation in these shifts provides a basis for spatiotemporal analysis of urban energy demand. We model this as a spatiotemporal sequence prediction problem, taking a sequence of coded energy consumption data (density maps) as input and generating a fixed number of images as output. Our encoder and decoder networks employ stacked AI-based spatiotemporal prediction models (Fig. 5). The input image is encoded, and the last encoded state initializes the prediction network. To obtain a density map with the same dimensionality as the input, all forecasting network states are concatenated and fed into a 1×1 convolutional layer.

The encoder–forecaster architecture (Eq. (5)) maximizes the probability of the ground truth sequence given the input sequence, using the

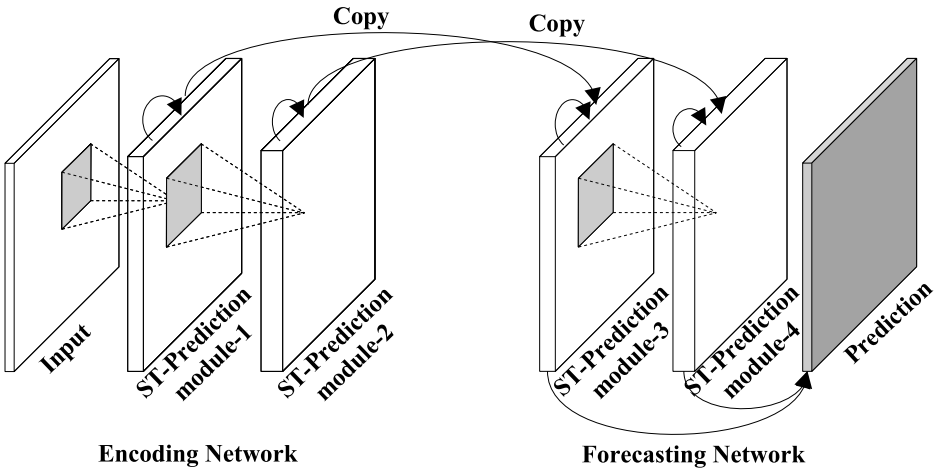

Fig. 5. The encoding and forecasting networks for spatiotemporal sequence prediction. The figure illustrates the encoder–forecaster architecture, detailing the encoding network's transformation of input density maps into a latent representation and the forecasting network's generation of future density maps from this latent representation. Both networks utilize stacked AI-based spatiotemporal prediction modules to capture complex spatiotemporal dynamics.

latent representation and forecaster output. The encoder converts input density maps into a compact latent representation encoding essential information. The forecaster then uses this to generate predicted density maps reflecting temporal dependencies. It predicts the density map evolution based on the encoded input and latent representation.

$$\begin{aligned}\hat{\mathcal{X}}_{t+1:t+L} &= \underset{\mathcal{X}_{t+1:t+L}}{\arg\max} p\left(\mathcal{X}_{t+1:t+L} \mid \mathcal{X}_{t-J+1:t}\right) \\ &\approx \arg\max p\left(\mathcal{X}_{t+1::t+L} \mid f_{\text{encoding}}\left(\mathcal{X}_{t-J+1:t}\right)\right) \\ &\approx g_{\text{forecasting}}\left(f_{\text{encoding}}\left(\mathcal{X}_{t-J+1:t+L}\right)\right)\end{aligned} \tag{5}$$

where the input and output are 3D tensors preserving spatial information.

We train the model using the Mean Squared Error (MSE) loss function (Eq. (6)) and optimize it using stochastic gradient descent (SGD).

$$L = \frac{1}{mn}\sum_{i=1,j=1}^{m,n}(\mathcal{X}_{i,j} - \hat{\mathcal{X}}_{i,j}), \tag{6}$$

where m and n are the height and width of the energy consumption density map, $\mathcal{X}$ is the ground truth, and $\hat{\mathcal{X}}$ is the prediction.

4.4.2. AI-based spatiotemporal prediction models

In this study, we employed four classic AI-based spatiotemporal prediction modules within the encoder–forecaster architecture: ConvLSTM [67], ConvGRU [68], PredRNN [69], and SA-ConvLSTM [70]. These models are selected for their proven effectiveness in spatiotemporal sequence prediction tasks, particularly in video prediction and weather forecasting, which share similarities with urban energy demand forecasting. A key characteristic shared by these models is their use of convolutional operations within their recurrent units, enabling them to efficiently process spatial information while capturing temporal dynamics. Fig. 6 illustrates the process flow within each of these modules. The network structures of the four models are shown in Fig. 6 and described in the following.

Convolutional LSTM (ConvLSTM) is an extension of the long short-term memory (LSTM) network, which was originally proposed to predict future precipitation and precipitation intensity. ConvLSTM replaces matrix multiplications in standard LSTM with convolution operations, enabling it to process spatiotemporal data. It uses convolution operators rather than matrix multiplication to process data and is able to use input from local neighbors and previous states to predict the future state. The network structure includes a memory unit that is updated at each time step by three sigmoid gates: the input gate, the forget gate, and the output gate. These gates control whether the input will accumulate in the memory unit, whether the past state will be forgotten, and whether the output will propagate to the final state. The ConvLSTM model is able to maintain gradients and preserve long-term dependencies.

The update equations for the gates in the ConvLSTM network structure are defined as follows:

$$\begin{aligned} i_t &= \sigma(W_i \otimes [h_{t-1}, \mathcal{X}_t] + b_i) \\ f_t &= \sigma(W_f \otimes [h_{t-1}, \mathcal{X}_t] + b_f) \\ o_t &= \sigma(W_o \otimes [h_{t-1}, \mathcal{X}_t] + b_o) \\ g_t &= \tanh(W_c \otimes [h_{t-1}, \mathcal{X}_t] + b_c) \\ c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\ h_t &= o_t \odot \tanh(c_t) \end{aligned} \tag{7}$$

ConvLSTM is included in our model suite due to its proven capability in capturing spatiotemporal dependencies in sequential data, particularly in domains like precipitation forecasting and video prediction, which share similarities with urban energy demand forecasting. Its convolutional layers effectively extract spatial features from the density maps, while the LSTM units capture temporal dynamics and long-range dependencies. Compared to a simpler ConvGRU, ConvLSTM's explicit memory cell and three gates (input, forget, output) may offer a more nuanced control over information flow and potentially better capture complex temporal patterns in energy consumption.

In these equations, $\otimes$ denotes the convolution operator; $\odot$ denotes the Hadamard product; σ is the logistic sigmoid function; $\mathcal{X}_t$ is the input at time step t; i_t, f_t, and o_t are the input gate, forget gate, and output gate, respectively, at time step t; g_t is the internal state at time step t; c_t is the memory unit at time step t; and h_t is the output at time step t. The ConvLSTM network structure is shown in Fig. 6(a).

Convolutional Gated Recurrent Unit (ConvGRU) is an extension of the long short-term memory (LSTM) network that uses convolution operators rather than matrix multiplication to process data and is able to incorporate input from local neighbors and previous states to predict the future state. ConvGRU simplifies the LSTM architecture by using only two gates, reducing computational complexity. The network structure of ConvGRU includes a memory unit that is updated at each time step by two sigmoid gates: the reset gate and the update gate. These gates control whether the input will accumulate in the memory unit and whether the past state will be forgotten. The ConvGRU model is able to maintain gradients and preserve long-term dependencies.

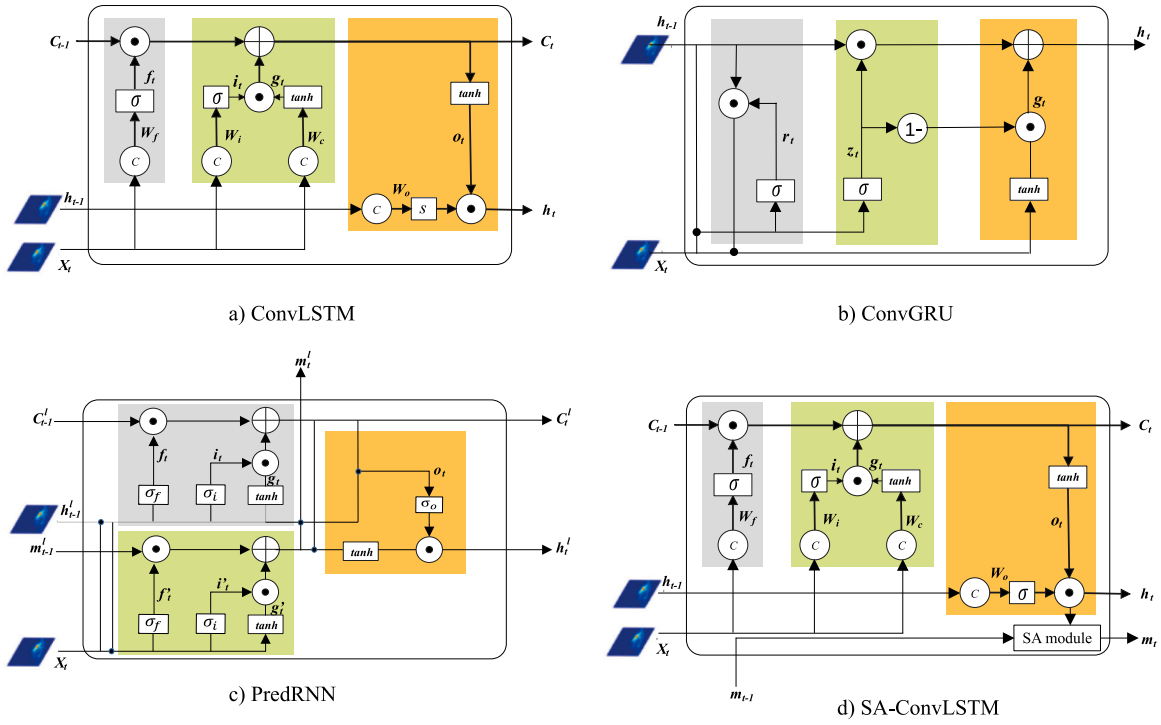

Fig. 6. AI-based spatiotemporal prediction modules for the cells in the encoder–forecaster architecture. (a) ConvLSTM module: Processes input and hidden state through convolutional LSTM units to update memory and hidden state. (b) ConvGRU module: Employs convolutional GRU units with reset and update gates for efficient spatiotemporal sequence modeling. (c) PredRNN module: Integrates horizontal and vertical LSTM units to capture both spatial and temporal dependencies with zigzag memory flow. (d) SA-ConvLSTM module: Incorporates a spatial attention mechanism to weight spatial locations and enhance feature extraction within the ConvLSTM structure.

The update equations for the gates in the ConvGRU network structure are defined as follows:

$$\begin{aligned} z_t &= \sigma(W_z \otimes [h_{t-1}, \mathcal{X}_t] + b_z) \\ r_t &= \sigma(W_r \otimes [h_{t-1}, \mathcal{X}_t] + b_r) \\ g_t &= \tanh(W_g \otimes [r_t \odot h_{t-1}, \mathcal{X}_t] + b_g) \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot g_t \end{aligned} \tag{8}$$

ConvGRU is chosen as another recurrent module due to its computational efficiency and effectiveness in capturing temporal dependencies. Similar to ConvLSTM, it utilizes convolutional operations for spatial feature extraction. However, ConvGRU has a simpler gated structure with only two gates (reset and update), making it computationally less expensive than ConvLSTM. While potentially sacrificing some fine-grained control over memory compared to ConvLSTM, ConvGRU can be advantageous in scenarios with limited computational resources or when faster training times are desired, while still effectively capturing the essential spatiotemporal dynamics of energy demand.

In these equations, $\otimes$ denotes the convolution operator; $\odot$ denotes the Hadamard product; σ is the logistic sigmoid function; $\mathcal{X}_t$ is the input at time step t; r_t and z_t are the reset gate and update gate, respectively, at time step t; g_t is the internal state at time step t; and h_t is the output at time step t. The ConvGRU network structure is shown in Fig. 6(b).

Predictive Recurrent Neural Network (PredRNN) is an extension of the long short-term memory (LSTM) network that uses convolution operators rather than matrix multiplication to process data and is able to incorporate input from local neighbors and previous states to predict the future state. PredRNN introduces a unique zigzag memory flow and decoupled memory cell structure to better capture long-range spatiotemporal dependencies. The network structure of PredRNN includes a memory unit that is updated at each time step by two sigmoid gates: the horizontal gate and the vertical gate. These gates control how much information is passed from previous states and from local neighbors, respectively. The PredRNN model is able to maintain gradients and preserve long-term dependencies.

The update equations for the gates in the PredRNN network structure are defined as follows:

$$\begin{aligned} g_t &= tanh(W_{xg} \otimes \chi_t + W_{hg} \otimes H_{t-1}^l + b_g) \\ i_t &= \sigma(W_{xi} \otimes \chi_t + W_{hi} \otimes H_{t-1}^l + b_i) \\ f_t &= \sigma(W_{xf} \otimes \chi_t + W_{hf} \otimes H_{t-1}^l + b_f) \\ C_t^l &= f_t \odot C_{t-1}^l + i_t \odot g_t \\ g_t' &= tanh(W_{xg}' \otimes \chi_t + W_{mg} \otimes M_t^{l-1} + b_g') \\ i_t' &= \sigma(W_{xi}' \otimes \chi_t + W_{mi} \otimes M_t^{l-1} + b_i') \\ f_t' &= \sigma(W_{xf}' \otimes \chi_t + W_{mf} \otimes M_t^{l-1} + b_f') \\ M_t^l &= f_t' \odot M_t^{l-1} + i_t' \odot g_t' \\ o_t &= \sigma(W_{xo} \otimes \chi_t + W_{ho} \otimes H_{t-1}^l + W_{co} \otimes C_t^l + W_{mo} \otimes M_t^l + b_o) \\ H_t^l &= o_t tanh(W_{1\times1} \otimes [C_t^l, M_t^l]). \end{aligned} \tag{9}$$

PredRNN is included to explore the benefits of its predictive recurrent network structure for energy demand forecasting. Its zigzag memory flow and decoupled memory cell are designed to improve the capture of long-range temporal dependencies, which may be crucial for forecasting energy consumption over extended horizons. By explicitly modeling both spatial and temporal predictive states, PredRNN aims to enhance prediction accuracy, particularly in complex spatiotemporal scenarios.

In these equations, g_t is the horizontal gate, i_t and f_t are the input and forget gates, respectively, $C^l t$ is the memory unit, $g' t$, $i' t$, and $f' t$ are the vertical gate, input gate, and forget gate, respectively, $M^l t$ is the memory unit from the lower layer, o_t is the output gate, and $H^l t$ is the output of the model. The symbols $\otimes$ and $\odot$ represent the convolution operator and the Hadamard product, respectively, and σ is the logistic sigmoid function. The variables W, b, and χ are the weights, biases, and input data, respectively. The PredRNN network structure is shown in Fig. 6(c).

Spatial Attention Convolutional LSTM (SA-ConvLSTM) is an extension of the ConvLSTM model that includes a spatial attention mechanism to weight the importance of different spatial locations. SA-ConvLSTM integrates a spatial attention mechanism into the ConvLSTM

architecture, allowing the model to focus on spatially relevant features. This allows the model to focus on the most relevant spatial features and improve the prediction accuracy. The SA-ConvLSTM model consists of three components: a ConvLSTM module, a spatial attention module, and an output module. The ConvLSTM module processes the input data and generates a hidden state, which is passed to the spatial attention module. The spatial attention module assigns a weight to each element of the hidden state, indicating its importance for the prediction. The weighted hidden state is then passed to the output module, which generates the final prediction.

The ConvLSTM module in the SA-ConvLSTM model is similar to the ConvLSTM model described earlier. It consists of a memory unit that is updated at each time step using three sigmoid gates (input, forget, and output gates) and a tanh function. The update equations for the gates are defined as follows:

$$\begin{aligned}
&\hat{\chi}_t = SA(\chi_t)\\
&\hat{H_{t-1}} = SA(H_{t-1})\\
&i_t = \sigma^i(W_i \otimes \left[H_{t-1}, \chi_t\right] + b_i)\\
&f_t = \sigma^f(W_f \otimes \left[H_{t-1}, \chi_t\right] + b_f)\\
&o_t = \sigma^o(W_o \otimes \left[H_{t-1}, \chi_t\right] + b_o)\\
&g_t = tanh(W_c \otimes \left[H_{t-1}, \chi_t\right] + b_c)\\
&c_t = f_t \odot c_{t-1} + i_t \odot g_t\\
&H_t = o_t \odot tanh(c_t)
\end{aligned} \tag{10}$$

The Spatial Attention (SA) module in our SA-ConvLSTM implementation is a self-attention mechanism, specifically designed to capture spatial dependencies within each density map. It operates on the input density map χ_t and the previous hidden state H_{t-1} separately before they are fed into the ConvLSTM unit. The SA module consists of three convolutional layers (1×1 convolutions for computational efficiency) followed by a sigmoid activation function to generate spatial attention weights. Specifically, for input $\mathcal{X}$ (either χ_t or H_{t-1}), the Spatial Attention module computes:

$$SA(\mathcal{X}) = \sigma(Conv_{1\times1}(ReLU(Conv_{1\times1}(ReLU(Conv_{1\times1}(\mathcal{X})))))) \odot \mathcal{X} \tag{11}$$

where σ is the sigmoid function, $ReLU$ is the Rectified Linear Unit activation function, and $Conv_{1\times1}$ denotes a convolutional layer with a 1×1 kernel. The use of 1×1 convolutions allows for efficient computation of spatial attention weights, focusing on channel-wise feature recalibration at each spatial location without increasing the model's complexity significantly. While self-attention mechanisms are used in other spatiotemporal models, our specific SA block is tailored for density map inputs and integrated directly before the ConvLSTM unit to refine the spatial features at each time step, enhancing the model's focus on spatially salient regions within the urban energy consumption maps.

SA-ConvLSTM is included to investigate the impact of spatial attention mechanisms on urban energy forecasting. By incorporating a self-attention module, SA-ConvLSTM can adaptively weight the importance of different spatial locations in the density maps when making predictions. This is particularly relevant in urban environments where energy consumption patterns are spatially heterogeneous and certain areas may exert more influence on future demand than others. Compared to the standard ConvLSTM, the spatial attention mechanism aims to enhance the model's ability to focus on the most salient spatial features, potentially leading to improved accuracy and interpretability by highlighting important spatial regions.

In these equations, $SA(\cdot)$ is the self-attention module, which helps the model to learn more informative and expressive features from the input by applying attention mechanisms on different parts of the input. σ^i, σ^f, and σ^o are sigmoid activation functions. W_i, W_f, W_o, and W_c are the weights of the input, forget, output, and cell gates, respectively. b_i, b_f, b_o, and b_c are the biases of the input, forget, output, and cell gates, respectively. $\otimes$ is the convolution operator, and $\odot$ is the Hadamard product. c_t and H_t are the cell state and hidden state at time step t, respectively. c_{t-1} and H_{t-1} are the cell state and hidden state at the previous time step, respectively. χ_t is the input at time step t, and i_t, f_t, and o_t are the input, forget, and output gates at time step t, respectively. g_t is the candidate cell state at time step t. The PredRNN network structure is shown in Fig. 6(d).

We summarize the four spatiotemporal prediction models in the comparison Table 2. The models are designed to process sequential data with both spatial and temporal dependencies. They are strong at capturing both temporal and spatial dependencies in the data and handling long-range dependencies and complex dynamics. However, they may have different levels of efficiency in terms of training and optimization. In the experiment section of this study, we evaluated and compared the performance of these four spatiotemporal prediction models on the task of predicting urban energy consumption.

4.5. Reconstructing energy demand forecasts based on density maps

The output of the spatiotemporal prediction model is a series of energy density maps. To obtain actual consumption values, we reconstruct energy consumption from these density maps using a reverse transformation process. As we use imaging-based methods for prediction, the pixel scale may be different between the original domain to the predicted domain. Therefore, we must first map the predicted pixel values back to the original domain. We assume that the pixel values from the original domain to the predicted domain follow a simple linear relationship, such that $f(x) = ax+b$. Given the minimum and maximum pixel values of the density map in the original domain, $\mathcal{X}^{min}$ and $\mathcal{X}^{max}$, and the minimum and maximum pixel values of the predicted density map, $\hat{\mathcal{X}}_k^{min}$ and $\hat{\mathcal{X}}_k^{max}$, we can obtain the corresponding value in the original domain for a predicted pixel value using the following formula:

$$\mathcal{X}_k(x,y) = \frac{(\hat{\mathcal{X}}_k(x,y) - \hat{\mathcal{X}}_k^{min})(\mathcal{X}^{max} - \mathcal{X}^{min})}{\hat{\mathcal{X}}_k^{max} - \hat{\mathcal{X}}_k^{min}} + \mathcal{X}^{min} \tag{12}$$

where $\hat{\mathcal{X}}_k(x,y)$ represents the predicted pixel value at a location (x,y), k represents the k-time step ahead for the prediction and $\mathcal{X}_k(x,y)$ represents the pixel value in the original domain. Based on this formula, we obtained the k time step ahead predicted density map in the k time step ahead in the original domain. We now need to calculate the consumption values for the point or area of interest. We grid the density map with a fixed step size to obtain the spatiotemporal measurements, which are the KDE values, and then generate the consumption value using the inverse transform sampling approach. In order to obtain a finer resolution forecast, we use bilinear interpolation [71] to calculate the energy demand of a customer. Bilinear interpolation is a method of interpolation that is used to estimate the value of a function at a point within a two-dimensional grid based on the values of the function at the surrounding grid points. It is called "bilinear" because it uses linear interpolation in both dimensions. Formally, given a set of points $(x_1, y_1, f(x_1, y_1)), (x_2, y_1, f(x_2, y_1)), (x_1, y_2, f(x_1, y_2))$, and $(x_2, y_2, f(x_2, y_2))$, where (x_1, y_1) and (x_2, y_2) are the coordinates of the corner points of a grid cell and $f(x,y)$ is the function defined at those points (see Fig. 7), bilinear interpolation estimates the value of the function at an arbitrary point (x,y) within the grid cell using the following formula:

$$f(x,y) = \frac{1}{(x_2-x_1)(y_2-y_1)} \begin{bmatrix} x_2 - x & x - x_1 \end{bmatrix} \begin{bmatrix} f(x_1,y_1) & f(x_1,y_2) \\ f(x_2,y_1) & f(x_2,y_2) \end{bmatrix} \begin{bmatrix} y_2 - y \\ y - y_1 \end{bmatrix} \tag{13}$$

To perform bilinear interpolation, we first need to determine the grid cell that contains the point for which we want to interpolate the function. We then use the values of the function at the four corners of the grid cell to estimate the value of the function at the desired point. For the customer located at the grid point, the kth day of energy demand is calculated as follows:

$$f_k(x,y) = \frac{\mathcal{X}_k(x,y)}{\mathcal{N}(x,y)} \tag{14}$$

where $\mathcal{N}(x,y)$ is the KDE density of the customer number at the point (x,y). Note that the KDE density map for the customer number

Table 2
Summary of the four spatiotemporal prediction models.

Model	Network structure	Description
ConvLSTM	Convolutional layers + LSTM layers	Uses convolutional layers to extract spatial features and LSTM layers to capture temporal dependencies
ConvGRU	Convolutional layers + GRU layers	Uses convolutional layers to extract spatial features and GRU layers to capture temporal dependencies
PredRNN	Convolutional layers + PredRNN layers	Uses convolutional layers to extract spatial features and PredRNN layers to capture long-range temporal dependencies
SA-ConvLSTM	Convolutional layers + self-attention mechanism + LSTM layers	Uses convolutional layers to extract spatial features, self-attention mechanism to capture global dependencies, and LSTM layers to capture temporal dependencies

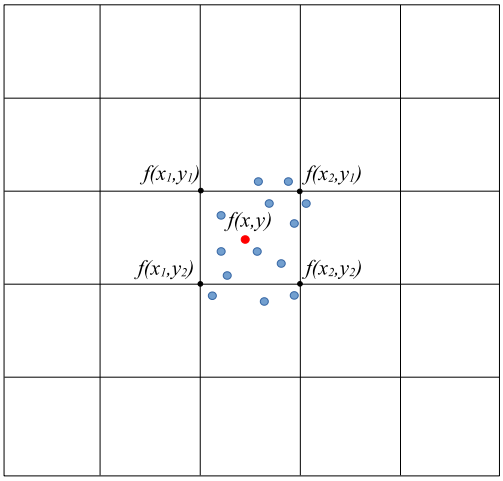

Fig. 7. Illustrative point energy demand calculation using bilinear interpolation. The figure shows a grid cell with four known points $(x_1, y_1), (x_2, y_1), (x_1, y_2), (x_2, y_2)$ and their corresponding function values $f(x_1, y_1), f(x_2, y_1), f(x_1, y_2), f(x_2, y_2)$. Bilinear interpolation estimates the function value at an arbitrary point (x, y) within this cell based on a weighted average of the corner point values.

distribution can be obtained by the following, which is similar to the energy demand density map in Eq. (3).

$$\mathcal{N} = \sum_{i=1}^{n} K_h(x - x_i). \tag{15}$$

The total energy demand of an area A can be calculated as follows:

$$\mathcal{X}_k(A) = \sum_{(x,y)\in A} f_k(x, y), \tag{16}$$

where $\mathcal{X}_k(A)$ denotes the total predicted energy demand of the area A at the kth time step.

5. Experiments and results

This section details the experimental evaluation of our proposed interpretable, multi-scale urban energy demand forecasting model. We detail the experimental setup, including baseline models, implementation details of our approach, and evaluation metrics. We then present the results of our experiments, comparing the performance of our model against the baselines across various spatial granularities and forecasting horizons. We analyze the impact of key hyperparameters on prediction accuracy and demonstrate the model's ability to generate spatially detailed, interpretable forecasts.

5.1. Baseline models and settings

To comprehensively evaluate our proposed model, we compared its performance against a range of baseline models, including traditional time series forecasting methods and advanced spatiotemporal deep learning models.

5.1.1. Traditional time series forecasting models

- **Multiple Linear Regression (MLR):** Models the relationship between past energy consumption and future demand using a linear equation. We tuned the number of past time steps (n) included in the model, selecting the optimal value based on validation set performance.
- **ARIMA:** Captures temporal dependencies using autoregressive (AR), integrated (I), and moving average (MA) components. We used an ARIMA(1,0,2) model, determined by analyzing the autocorrelation and partial autocorrelation functions of the time series.
- **SARIMA:** Extends ARIMA by incorporating seasonal patterns with seasonal AR, I, and MA components. We initialized the non-seasonal (p, d, q) order with the optimal ARIMA values and tuned the seasonal parameters (P, D, Q, m) via grid search for each forecasting horizon.
- **Long Short-Term Memory (LSTM):** An RNN architecture designed for capturing long-term dependencies. We tuned the number of hidden units and layers in the LSTM network.
- **Optical Flow (Farneback and ROVER):** We applied optical flow algorithms (Gunnar Farneback and ROVER [66,72–74]) to the density maps, treating energy demand shifts as spatial "motion". For ROVER, we used the initialization from [75]. We used default parameters for both algorithms due to their limited number of hyperparameters.

5.1.2. Advanced spatiotemporal deep learning models

- **PhyDNet** [76]: A physics-informed model using PhyCell for dynamics modeling. We used a patch size of 4 and a learning rate of 0.001. Other parameters were kept at their default values.
- **PredRNN-V2** [77]: Uses decoupled memory cells and zigzag memory flow. We used four hidden layers with 128 units, kernel size 5, stride 1, patch size 4, decoupling coefficient 0.1, reverse scheduled sampling, and a cosine annealing learning rate schedule (initial learning rate 0.001).
- **SimVP** [78] **and TAU** [79]: Both are CNN-based models. We used kernel size 3, spatial hidden size 32, temporal hidden size 256, 8 temporal and 2 spatial convolution layers, DropPath with a coefficient of 0.1, and cosine annealing learning rate scheduling (initial learning rate 0.001).

This selection of baselines provides a comprehensive benchmark, encompassing traditional statistical methods, optical flow techniques for motion-based prediction, and state-of-the-art deep learning models for spatiotemporal forecasting. The tuned hyperparameters, reported in the supplementary material, ensure a fair comparison and facilitate reproducibility.

5.2. Proposed model and settings

Our proposed model utilizes a deep learning encoder–forecaster architecture (Section 4.4) with four different spatiotemporal prediction modules: ConvLSTM, ConvGRU, PredRNN, and SA-ConvLSTM. The

Table 3
Optimal Hyperparameters for Deep Learning Models. The table summarizes the optimal hyperparameter configurations identified for our proposed models and the advanced baseline deep learning models through validation-set based hyperparameter tuning.

Model	Hidden units	Dropout rate	Batch size	Learning rate schedule
ConvLSTM	128	0.2	32	Fixed (0.001)
ConvGRU	128	0.25	32	Fixed (0.001)
PredRNN	128	0.15	32	Cosine Annealing (Initial LR 0.001)
SA-ConvLSTM	128	0.2	32	Fixed (0.001)
PhyDNet	(Default)	(Default)	(Default)	Fixed (0.001)
PredRNN-V2	128	0.1	32	Cosine Annealing (Initial LR 0.001)
SimVP	256	0.1	32	Cosine Annealing (Initial LR 0.001)
TAU	256	0.1	32	Cosine Annealing (Initial LR 0.001)

encoder and forecaster components of the architecture, along with details about each spatiotemporal module, are described in Section 4.4. Within this architecture, we explore the performance of each of the four modules.

- **Baseline Configuration:** We initialized all modules with ReLU activation functions and 5 × 5 filters. ConvLSTM, ConvGRU, and SA-ConvLSTM used 3 unit cell layers, while PredRNN used 2. The ReLU activation was selected for computational efficiency and to mitigate vanishing gradients, while the 5 × 5 kernel size effectively balanced capturing local spatial features with computational efficiency for 64 × 64 density maps. The layer configuration (3 layers for most models, 2 for the more complex PredRNN) was optimized through validation experiments to balance model capacity against overfitting.
- **Training Process:** Each model is trained for 30 epochs using the Adam optimizer with an initial learning rate of 0.001. We use a sliding window approach to generate training batches, where the input and output sequence lengths are determined by the forecasting time scale (T = 1, 7, 14, 21, or 28 days). The sliding window moves by one day to create subsequent batches. Early stopping, based on validation loss, is employed to prevent overfitting. The 30-epoch limit was established based on preliminary experiments where performance typically plateaued within this range. Batch size was set to 32 after grid search, balancing training stability and memory usage. Regularization included dropout (rate 0.2) and batch normalization to improve generalization performance. During hyperparameter tuning, we explored dynamic learning rate scheduling techniques including cosine annealing and step decay, but found that the fixed rate of 0.001 with Adam optimizer and early stopping provided the most stable training across different forecasting horizons and spatial scales. Table 3 summarizes the optimal hyperparameters for our proposed models and the advanced baseline deep learning models. These were determined through an efficient two-step process: initial grid search over a coarse hyperparameter space followed by manual refinement in promising regions based on validation performance.
- **Hyperparameter Tuning:** We tuned the following hyperparameters for each module using a combination of grid search and manual tuning, optimizing performance on a validation set:
  - Number of hidden units in recurrent layers (search range: 64 to 256)
  - Dropout rate (search range: 0 to 0.5)
  - Batch size (search range: 16 to 64)
  - Learning rate schedule (options: fixed, cosine annealing, step decay)

5.3. Evaluation metrics

To evaluate the performance of our proposed model and the baseline models, we used three widely recognized metrics for image and video prediction tasks:

- **Mean Squared Error (MSE):** MSE is a measure of the difference between the predicted value $\hat{y}$ and the true value y. It is defined as:

$$MSE(y,\hat{y}) = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\|y(i,j)-\hat{y}(i,j)\|^2$$

  where m and n denote the height and width of the energy consumption map, respectively.
- **Peak Signal-to-Noise Ratio (PSNR):** PSNR is a measure of the quality of signal reconstruction in fields such as image compression. It is defined as the ratio of the maximum possible power of a signal to the destructive noise power that affects its representation accuracy. PSNR is often expressed in logarithmic decibel units and is calculated as:

$$PSNR(y,\hat{y}) = 10\times\log_{10}\frac{max(\hat{y})^2}{MSE(y,\hat{y})}$$

- **Structural Similarity (SSIM):** SSIM is an indicator used to measure the similarity between two images. It is defined as:

$$SSIM(x,y) = \frac{(2u_y u_{\hat{y}}+c_1)(2\sigma_{y\hat{y}}+c_2)}{(u_y^2+u_{\hat{y}}^2+c_1)(\sigma_y^2+\sigma_{\hat{y}}^2+c_2)}$$

  where u_y and $u_{\hat{y}}$ represent the mean values used to estimate brightness, σ_y and $\sigma_{\hat{y}}$ represent the standard deviation values used to estimate contrast, and $\sigma_{\hat{y}}$ is the covariance used to measure structural similarity. The predicted value $\hat{y}$ is modeled as a combination of brightness, contrast, and structure. The value of SSIM ranges from −1 to 1, with a value of 1 indicating that the two images are identical.

5.4. Comparison study

Table 4 presents a comprehensive comparison of the forecasting performance of traditional methods and deep learning-based spatiotemporal prediction models across different time scales, ranging from 1 day to 28 days. The table reports the performance metrics PSNR, SSIM, and MSE for each model and forecasting horizon. The time scales are represented by the sequence length of the input energy consumption density maps and the generated energy demand prediction maps.

As shown in Table 4, traditional methods like OpticalFlow (ROVER) and OpticalFlow (Farneback) exhibit strong performance in ultra-short-term forecasting ($K = 1$), achieving high PSNR and SSIM and low MSE values. Based on the results in the table, it is clear that the traditional methods, such as OpticalFlow (ROVER) and OpticalFlow (Farneback), performed well in ultra-short-term forecasting (e.g., $K = 1$), achieving high values for PSNR and SSIM and low values for MSE. However, as the time span increases, these methods show a significant drop in prediction accuracy, with the MSE values increasing dramatically. In contrast, almost all deep learning-based spatiotemporal prediction models are able to maintain a high level of accuracy even for longer time spans.

Among the deep learning models, SA-ConvLSTM demonstrates superior performance for shorter time spans (e.g., $K = 1$), while PredRNN

Table 4
Comparison of traditional methods and deep learning-based spatiotemporal prediction models.

	1 Day			7 Days			14 Days			21 Days			28 Days		
	PSNR	SSIM	MSE	PSNR	SSIM	MSE	PSNR	SSIM	MSE	PSNR	SSIM	MSE	PSNR	SSIM	MSE
OpticalFlow (ROVER)	34.580	0.7329	22.649	34.436	0.7306	23.443	34.438	0.7237	23.434	34.412	0.7189	23.578	34.408	0.7138	23.600
OpticalFlow (Farneback)	42.654	0.9903	**4.359**	39.067	0.9624	9.356	37.667	0.9006	12.361	37.001	0.8355	14.141	36.611	0.7807	15.252
ARIMA	34.190	0.9652	24.774	32.067	0.9435	26.334	31.106	0.9311	29.869	30.593	0.9240	33.004	30.036	0.9132	34.368
SARIMA	34.190	0.9652	24.774	32.336	0.9482	25.997	32.057	0.9470	28.878	30.972	0.9346	33.026	31.184	0.9198	33.872
MLR	40.123	0.9940	7.004	32.855	0.8101	33.696	35.578	0.9024	18.001	28.9154	0.7885	83.471	37.164	0.9080	134.909
LSTM	34.214	0.9621	24.642	34.254	0.9651	24.113	34.001	0.9574	25.889	33.837	0.9603	26.872	33.731	0.9565	27.538
PhyDNet	–	–	–	34.218	0.7347	56.194	33.405	0.6886	63.691	32.863	0.6548	69.348	32.175	0.6199	77.914
PredNN-V2	39.829	0.9920	32.180	37.986	0.9901	33.784	37.244	0.9853	34.884	36.727	0.9840	35.947	36.705	0.9860	33.900
SimVP	39.232	0.9923	32.191	38.411	0.9898	34.090	37.948	0.9882	34.635	37.364	0.9894	33.472	36.875	0.9884	33.610
TAU	39.614	0.9920	30.886	38.464	0.9897	33.791	37.936	0.9884	34.514	37.231	0.9886	34.336	36.917	0.9880	33.679
ConvGRU	41.035	0.9889	11.490	41.154	0.9919	8.831	40.145	0.9910	9.727	39.944	0.9903	10.722	39.894	0.9905	10.447
ConvLSTM	40.520	0.9887	10.253	41.430	0.9922	8.496	41.144	0.9914	9.461	40.707	**0.9915**	9.758	40.052	**0.9907**	10.555
PredRNN	41.118	0.9923	7.200	**43.017**	0.9917	**6.032**	**42.321**	0.9916	**6.515**	**41.568**	0.9879	**7.446**	**40.990**	0.9882	**8.682**
SA-ConvLSTM	**42.665**	**0.9942**	6.547	42.331	**0.9928**	7.631	41.298	**0.9917**	8.873	40.886	0.9903	9.793	39.933	0.9902	10.422

and SA-ConvLSTM outperform others for longer horizons ($K = 14$ and $K = 28$). Among the deep learning-based models, SA-ConvLSTM showed the best performance for short time spans of input and predicted sequences, while SA-ConvLSTM and PredRNN significantly outperformed the other models for longer time spans (e.g., $K = 14$ and $K = 28$). In particular, PredRNN achieved the highest PSNR and lowest MSE values for longer time spans, achieving the highest PSNR and lowest MSE values for time spans of 14 and 28 days. This may be due to the zigzag information flow of PredRNN, which is helpful in capturing the time dependence over long distances.

Table 4 shows that advanced baseline models generally outperform traditional methods, but our proposed models (ConvLSTM, ConvGRU, PredRNN, SA-ConvLSTM) achieve the best overall performance across all forecasting horizons. The advanced baseline models outperform traditional methods, likely due to their sophisticated network designs, which excel at capturing energy consumption patterns and complex relationships within the data. PhyDNet, as a deep learning model incorporating physical constraints, exhibits relatively low performance. This may be attributed to the significant differences between the temporal characteristics of energy consumption data and physical dynamics (e.g., optical flow or motion fields), making it challenging for the model to effectively capture the specific patterns of energy consumption. However, the latest advanced models fall short compared to deep learning models utilizing encoding and forecasting networks, which highlights the enhanced representational capability of encoder–forecaster architecture with multi-layer stacking. The encoder–forecaster architecture design effectively integrates spatial information and temporal dependencies, enabling the modeling of highly nonlinear and complex patterns, such as the intricate dynamic changes in energy consumption.

It is worth noting that traditional statistical forecasting methods, such as ARIMA and SARIMA, and machine learning-based methods, including MLR and LSTM, also struggle in long-term forecasting and generally perform worse than the AI-based spatio-temporal prediction models. This suggests that the AI-based methods may be more suitable for handling the complexity and non-stationarity of long-term spatio-temporal data. This is likely due to the fact that traditional statistical forecasting methods rely on the stationarity and linearity of the raw data, which may not hold for longer time spans, while the deep learning-based models are able to capture more complex relationships and trends in the data. Additionally, the performance of traditional methods may be limited by the fact that they rely on relatively small amounts of historical data, while the deep learning-based models can take advantage of larger amounts of data to better capture long-term dependencies.

Overall, the results suggest that the deep learning-based spatiotemporal prediction models are promising for accurately forecasting energy demand at various time scales, with SA-ConvLSTM and PredRNN particularly well-suited for longer time spans. In contrast, traditional methods tend to perform well in ultra-short-term forecasting, but their accuracy decreases significantly as the time span increases. Advanced spatio-temporal prediction methods perform smoothly over a variety of time spans but with an overall lower accuracy compared to our model.

5.5. Impact of input parameters on prediction accuracy

In this section, we present a sensitivity analysis to evaluate the impact of key input parameters on the prediction accuracy of our proposed model. These parameters can influence the model's ability to capture the spatio-temporal patterns and dynamics of urban energy consumption. We conduct a series of experiments to evaluate the performance of our model under different settings of these parameters and compare it with other spatio-temporal prediction models.

The impact of different lengths of input and output sequences. In previous experiments, the length of both the input and predicted frame sequences remained constant. In the following, we investigate the effect on the results when the input and predicted frame sequences have different lengths. To do this, we fixed the length of the input image sequence to 7 and predicted the energy consumption for the next $K = 1, 4, 7, 10, 14$ days, respectively. Table 5 summarizes the results for varying prediction horizons (K) while keeping the input sequence length fixed at 7 days. The results, shown in Table 5, reveal that almost all of the spatio-temporal prediction models achieve their best performance at $K = 1$. As the value of K increases, there is a decreasing trend in both PSNR and SSIM, while the value of MSE increases. This suggests that these models perform better at shorter prediction times with the same length of input frame sequence.

The impact of varying input sequence lengths on prediction accuracy. We then consider the question of whether the accuracy of the results would be higher if the sequence of input energy density maps were longer. To test this, we used images for $T = 1, 4, 7, 10, 14$ days as input to predict energy demand for the next $K = 7$ days. Table 6 shows the impact of varying input sequence lengths (T) on prediction accuracy for a fixed prediction horizon of 7 days (K=7). The results indicate that increasing the input sequence length generally improves prediction performance. The results, presented in Table 6, show that as the length of the input series increases, all performance indicators tend to improve. In particular, the highest SSIM is obtained for all models when $T = 14$. This aligns with our expectations, as more training samples can improve the fit and robustness of the model, and the energy consumption data tends to have trends that the model can learn from longer input series. Therefore, when making predictions for the same number of days, having more raw data helps the model to make more accurate predictions. In contrast, predicting future energy demand beyond the length of the input degrades accuracy when the length of the time series fed into the model is fixed.

The impact of grid granularity on prediction accuracy. In addition, we recognize that the granularity of the grid division can also

Table 5
Impact of predicted time span K on accuracy (input sequence length: $T = 7$). The table shows PSNR, SSIM, and MSE values for different prediction horizons ($K = 1$, 4, 7, 10, 14 days) with a fixed input sequence length of 7 days. Results indicate that prediction accuracy decreases as the prediction horizon increases.

	PSNR				SSIM				MSE			
	ConvLSTM	ConvGRU	PredRNN	SA-ConvLSTM	ConvLSTM	ConvGRU	PredRNN	SA-ConvLSTM	ConvLSTM	ConvGRU	PredRNN	SA-ConvLSTM
$K = 1$	**42.872**	**42.523**	44.281	**44.613**	**0.9942**	**0.9936**	**0.9960**	**0.9945**	**6.768**	**6.505**	**3.296**	**5.959**
$K = 4$	41.629	41.719	**44.443**	43.159	0.9925	0.9927	0.9949	0.9931	8.087	8.203	4.010	6.453
$K = 7$	41.430	41.154	43.018	42.331	0.9922	0.9919	0.9917	0.9928	8.496	8.831	6.032	7.331
$K = 10$	41.101	41.318	42.407	42.105	0.9920	0.9920	0.9925	0.9919	8.839	8.556	6.755	8.023
$K = 14$	41.218	41.282	41.734	41.625	0.9913	0.9915	0.9916	0.9915	9.369	9.289	7.252	7.865

Table 6
Impact of input time span T on accuracy (output sequence length: $K = 7$). The table presents PSNR, SSIM, and MSE values for different input sequence lengths ($T = 1$, 4, 7, 10, 14 days) with a fixed prediction horizon of 7 days. Results demonstrate that increasing the input sequence length generally improves prediction accuracy.

	PSNR				SSIM				MSE			
	ConvLSTM	ConvGRU	PredRNN	SA-ConvLSTM	ConvLSTM	ConvGRU	PredRNN	SA-ConvLSTM	ConvLSTM	ConvGRU	PredRNN	SA-ConvLSTM
$T = 1$	39.543	38.979	39.247	40.012	0.9873	0.9835	0.9856	0.9893	12.617	13.767	11.634	10.815
$T = 4$	40.763	40.580	42.358	41.975	0.9919	0.9912	0.9919	0.9921	8.984	9.811	6.547	8.012
$T = 7$	41.430	41.154	43.018	42.331	0.9922	0.9919	0.9917	0.9928	8.496	8.831	6.032	7.331
$T = 10$	**41.809**	41.646	42.907	42.491	0.9911	0.9916	0.9918	0.9922	9.897	**7.988**	**5.856**	7.336
$T = 14$	41.637	**41.716**	**43.828**	**42.759**	**0.9928**	**0.9926**	**0.9935**	**0.9930**	**8.484**	8.174	6.329	**6.549**

Table 7
Impact of grid granularity on accuracy (input sequence length: $T = 7$, output sequence length: $K = 7$). The table compares PSNR, SSIM, and MSE values for different grid granularities (64 × 64, 200 × 200, 256 × 256 pixels) with fixed input and output sequence lengths of 7 days. Results indicate that finer grid granularity generally improves prediction accuracy but increases computational cost.

	64 × 64			200 × 200			256 × 256		
	PSNR	SSIM	MSE	PSNR	SSIM	MSE	PSNR	SSIM	MSE
ConvLSTM	40.614	0.9906	10.729	41.430	0.9922	8.496	41.835	0.9929	7.967
ConvGRU	40.437	0.9871	11.033	41.154	0.9919	8.831	41.369	0.9915	8.486
PredRNN	**42.943**	0.9921	**6.401**	**43.018**	0.9917	**6.032**	**43.341**	0.9935	**5.864**
SA-ConvLSTM	42.385	**0.9936**	7.247	42.646	**0.9931**	6.698	42.510	**0.9939**	6.775

impact the accuracy of the predictions. To examine this, we conducted comparison experiments with grid granularities of 64 × 64, 200 × 200, and 256 × 256, using an input sequence of length 7 to predict 7 future frames. Table 7 summarizes the impact of grid granularity on prediction accuracy. The results, shown in Table 7, indicate that almost all performance evaluation metrics of the model are best when the grid is divided into 256 × 256. This is because dividing the latitude and longitude coordinates into smaller blocks can improve the accuracy of the predictions by providing each pixel with more specific information and reinforcing the spatial correlation with its neighbors. However, it is important to note that this improvement in accuracy comes at the cost of increased computational cost for model training, so the grid division strategy should consider both accuracy and performance factors. In summary, we explored the impact of various factors on the accuracy of spatiotemporal prediction models for energy demand. The length of the input and predicted frame sequences, as well as the granularity of the grid division, were all found to affect the performance of the models. It was found that shorter prediction times and longer input sequences generally led to better accuracy, while finer grid granularity also improved accuracy but at the cost of increased computational cost.

In addition, based on the experimental results, it can be concluded that all four models – ConvLSTM, ConvGRU, PredRNN, and SA-ConvLSTM – show good performance in forecasting energy demand. However, the SA-ConvLSTM model consistently outperforms the other three models in terms of accuracy, as demonstrated by its higher PSNR, SSIM, and MSE values. The effect of the predicted time span and the input time span on accuracy varied among the four models. In general, shorter predicted time spans and longer input time spans were found to result in higher accuracy for all four models. However, the exact relationship between these variables and accuracy differed among the models. The impact of the kernel size on accuracy also varied among the four models. While larger kernel sizes were generally found to result in higher accuracy for the ConvLSTM and ConvGRU models, the PredRNN model showed the highest accuracy with a kernel size of 5, and the SA-ConvLSTM model showed the highest accuracy with a kernel size of 3. Overall, the SA-ConvLSTM model appears to be the most effective among the four models in forecasting energy demand, achieving the highest levels of accuracy across all experimental conditions.

5.6. Prediction at different urban spatial scales

One of the key advantages of our proposed model is its ability to provide flexible energy demand forecasts at different urban spatial scales. Fig. 8 demonstrates the model's multi-scale forecasting capability. The figure shows an example of selecting four urban areas of interest, labeled *A-D*, at four different spatial scales on a generated energy density map, and generating the corresponding load profiles. Note that, on the map, we can zoom in/out and select different area sizes to produce the prediction result at different spatial scales. In Fig. 8, it can be seen that the predicted load profiles almost match the ground truth. In fact, the accuracy of the predicted load profiles is dependent on the length of the input sequence and the grid granularity, as we have studied in Section 5.5. Therefore, these hyperparameters can be adjusted to meet different accuracy requirements.

Forecasting energy demand at different urban spatial scales is important for various sectors, such as generation, transmission, and distribution, particularly in regions of interest. Accurate energy demand forecasts at the regional level can help power plants optimize their operations and increase efficiency, while accurate energy demand forecasts at the local level can help distribution companies better manage their resources and improve service quality. By forecasting energy demand at different spatial scales, a more comprehensive understanding of energy consumption patterns can be gained and informed decisions can be made to optimize the energy system. While the proposed model is capable of predicting energy demand at the individual customer

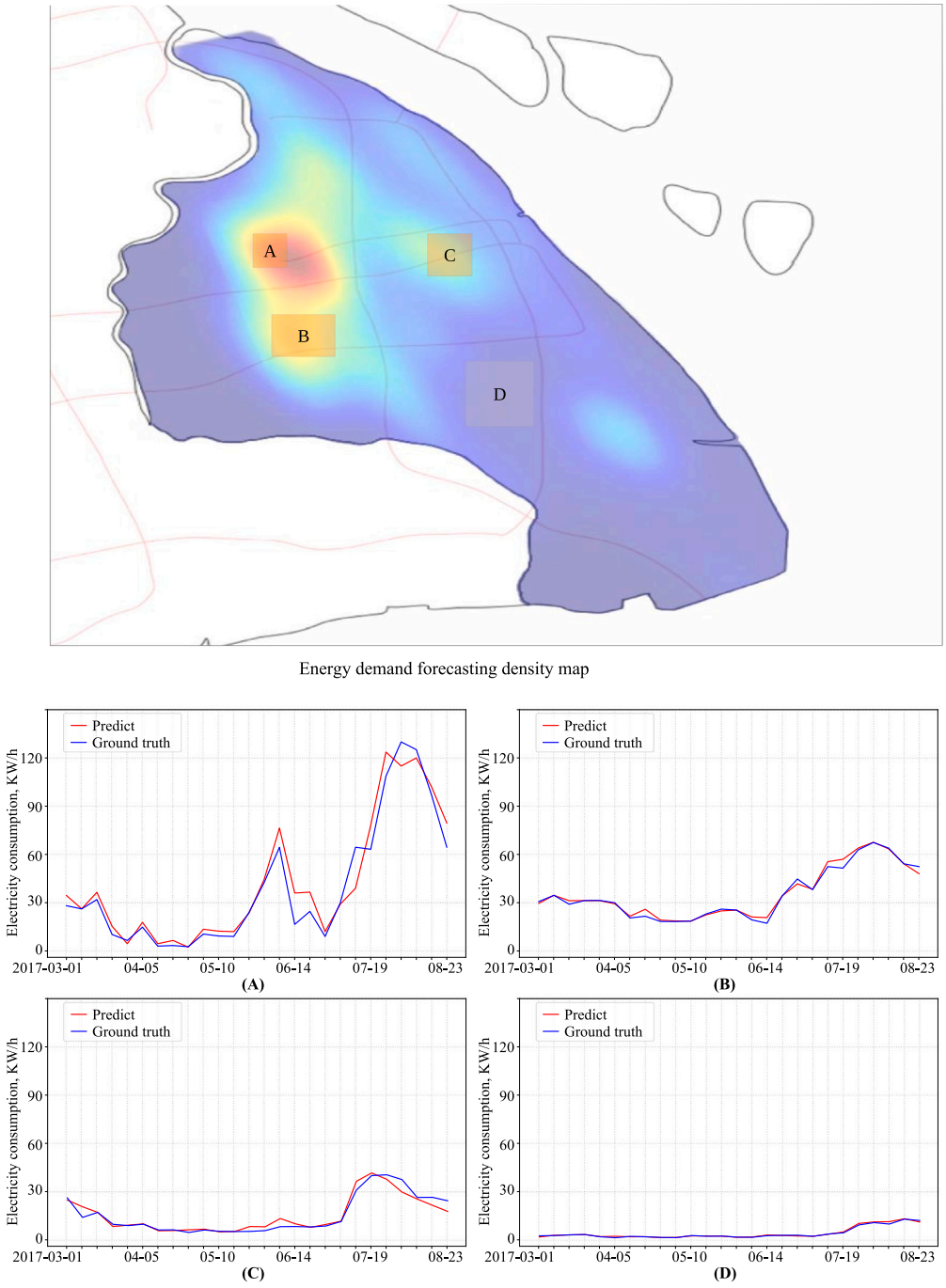

Fig. 8. Urban energy demand forecasting at different spatial scales. The figure illustrates the model's ability to forecast energy demand at varying spatial granularities. Areas A, B, C, and D represent different urban regions selected from the density map, demonstrating multi-scale forecasting. The corresponding load profiles for each area show that the predicted demand closely matches the ground truth, highlighting the model's effectiveness across spatial scales.

level, the high degree of variability at this level may make forecasts at the neighborhood or larger spatial scale more useful for utilities, especially when the distribution of customers is irregular.

6. Discussion

This study unequivocally demonstrates the critical importance of spatial information for achieving accurate and interpretable urban energy demand forecasting, a cornerstone for sustainable urban development. By integrating spatial heterogeneity into our deep learning framework, we achieve significant improvements in prediction accuracy and flexibility compared to traditional time-series methods. Our imaging-based approach, transforming discrete energy consumption data into continuous density maps, allows the model to capture spatial dependencies and correlations across the urban landscape. This holistic view, unlike individual predictors limited to metered locations, provides a comprehensive understanding of how factors such as population density, building typology, and micro-climatic variations influence energy demand. The use of kernel density estimation (KDE) enhances the model's robustness by addressing data gaps common in real-world applications. KDE effectively imputes missing data, ensuring consistent

performance even with incomplete records. Furthermore, our model's ability to reconstruct energy consumption time series from predicted density maps facilitates multi-scale forecasting, offering flexibility for diverse decision-making processes across various spatial granularities, from individual buildings to city-wide levels. The potential integration of an interactive user interface could further enhance accessibility, allowing stakeholders to intuitively select specific areas and generate tailored forecasts.

The encoder–forecaster architecture enables end-to-end predictions capturing both temporal patterns and spatial dynamics. This architecture, combined with time series imaging and KDE, addresses traditional forecasting limitations and improves the model's capability to handle nonlinear relationships. The rigorous validation against real-world electricity data from Shanghai demonstrates the model's superior performance across spatial scales and time horizons. Importantly, the preservation of 2D spatial information throughout the prediction process allows for direct visualization of the forecasted energy distribution, significantly enhancing interpretability compared to methods that reduce spatial information to 1D representations. This enhanced interpretability is not merely a theoretical advantage; it translates directly into actionable insights for urban stakeholders. These capabilities empower targeted energy management and provide actionable insights for urban planning, policy development, and energy management. These outcomes contribute to global sustainability goals by promoting efficient energy production and distribution, reducing emissions, and supporting the integration of renewable energy sources [80,81]. For instance, visualizing predicted energy demand as density maps allows urban planners to identify high-consumption zones, optimize infrastructure investments, and evaluate the impact of urban design choices on energy efficiency. Policymakers can leverage these interpretable forecasts to develop targeted energy conservation programs and incentives, while utility operators can enhance grid management and demand response strategies with spatially granular predictions.

6.1. Model assumptions, limitations, and uncertainty

While our model offers a robust and interpretable solution, it is crucial to acknowledge its inherent assumptions, limitations, and the uncertainties associated with forecasting complex real-world systems like urban energy demand. While our model presents a significant advancement in interpretable urban energy demand forecasting, it is important to acknowledge its assumptions, limitations, and inherent uncertainties.

- **Dataset Considerations:** While the Shanghai Pudong dataset provides a valuable foundation for our study, it is essential to recognize its inherent limitations which could influence the broader applicability of our findings.
- **Generalizability:** The dataset, being specific to Pudong District, Shanghai, may not fully encapsulate the diverse urban energy consumption patterns observed globally. Variations in urban morphology, climate, socio-economic factors, and building stock across cities could affect the direct transferability of the model without adaptation and retraining on local data. Therefore, applying our model to cities with vastly different characteristics should be approached with caution without proper recalibration.
- **Spatiotemporal Data Integration, not Segregation:** It is s important to clarify that our model is designed to capture the inherent interplay between spatial and temporal variations in urban energy demand, rather than segregating them. Urban energy consumption is fundamentally a spatiotemporal phenomenon, where spatial patterns evolve over time due to complex interactions of various factors. Attempting to strictly "segregate" spatial and temporal variations would oversimplify the problem and likely lead to a loss of valuable information and predictive accuracy. Our approach, using time series imaging and 2D-CNN based recurrent networks, is explicitly designed to model these intricate spatiotemporal dependencies in a holistic manner, recognizing that spatial and temporal aspects are inherently intertwined in urban energy dynamics.
- **Sampling Bias:** Despite SGCC's efforts towards representative customer sampling, potential biases related to customer demographics (e.g., income levels, household size) or building types (e.g., prevalence of older buildings, types of HVAC systems) might exist within the dataset. Such biases, if present, could subtly skew the learned patterns and affect forecast accuracy in scenarios with differing demographic or building characteristics.
- **Contextual Data Scarcity:** The absence of granular contextual information within the dataset, such as detailed building characteristics (e.g., insulation levels, building age, occupancy schedules), high-resolution weather data, or real-time socio-economic indicators, represents a limitation. Integrating such multi-source data could enrich the model's input features, potentially leading to more accurate and nuanced predictions by capturing a wider spectrum of influencing factors. Future research should prioritize incorporating such readily available urban datasets to enhance model fidelity and predictive power.

Our model operates under the assumptions of spatial autocorrelation (nearby locations exhibit similar energy consumption patterns due to shared factors) and temporal dependence (past consumption informs future consumption). While generally reasonable for urban energy modeling, deviations from these assumptions can occur, particularly in areas with unique characteristics (e.g., industrial zones or areas undergoing rapid development). We also assume that the KDE process effectively transforms discrete spatial data into continuous representations, and that a linear transformation adequately maps predicted pixel values back to the original domain. While KDE handles data gaps and smooths the representation, it may not perfectly capture highly localized variations. Similarly, the linear mapping, while generally accurate, could introduce minor errors in regions with non-linear relationships. Lastly, the model assumes the training data is representative of the target urban environment. However, variations in climate, population density, and socio-economic activity across different cities necessitate adaptation through retraining or fine-tuning with local data.

Furthermore, several limitations should be considered. The model's reliance on large training datasets, though mitigated by the increasing prevalence of smart meters [82], remains a practical constraint. More significantly, forecasting urban energy demand involves inherent uncertainties stemming from abrupt changes in consumption patterns (due to factors like extreme weather, policy shifts, or socio-economic changes), technology disruptions, and evolving regulatory landscapes. While our model incorporates robustness through KDE and noise injection during training, it does not explicitly quantify uncertainty in its predictions. The lack of uncertainty quantification can limit the model's usefulness for decision-making, as stakeholders need to understand the range of potential outcomes and the confidence level associated with the forecasts. The model's generalizability to other urban contexts requires further exploration. Direct application to cities with different characteristics may require adjustments to model parameters, spatial granularity, time steps, and the incorporation of city-specific data [83].

6.2. Model generalization and adaptability

While our validation using Shanghai Pudong data is promising, the broader generalizability of our model to diverse urban environments remains an important consideration for future deployment. Urban energy consumption is shaped by a complex interplay of factors that vary significantly across cities worldwide. Climate zones, urban layouts (ranging from grid-based to organic street patterns), building stock characteristics (age, insulation, prevalent building types), and socio-economic profiles all contribute to unique energy demand dynamics. These inter-city variations imply that a model trained on Shanghai

data may encounter performance degradation when directly applied to cities with substantially different characteristics. For example, cities in colder climates with a higher proportion of older, less energy-efficient buildings might exhibit different seasonal demand peaks and spatial consumption patterns compared to Shanghai.

To effectively adapt our model for broader urban applicability, several strategic approaches can be pursued. Retraining the model using local energy consumption data from the target city is a primary and essential step to capture city-specific dynamics. Furthermore, transfer learning techniques offer a promising avenue, allowing for fine-tuning pre-trained models (initially trained on Shanghai data or other large urban datasets) with smaller, city-specific datasets, potentially accelerating adaptation and improving performance even with limited local data availability. Crucially, incorporating city-specific contextual features as additional input channels to the model represents a key direction for future development. This includes integrating readily available datasets such as local weather forecasts (temperature, humidity, solar radiation), high-resolution population density maps, detailed building type distributions (residential, commercial, industrial mix), and even relevant socio-economic indicators (average income, employment rates). By enriching the model with such multi-source urban data, we can enhance its sensitivity to local conditions and improve its predictive accuracy and robustness across diverse urban landscapes. Future research should systematically evaluate the model's performance across a wider range of cities with varying characteristics, empirically assess the effectiveness of different adaptation strategies (retraining, fine-tuning, contextual feature integration), and develop robust guidelines for model deployment in diverse urban contexts.

In summary, this study contributes a practical solution for interpretable, multi-scale urban energy demand forecasting. However, acknowledging these assumptions, limitations, and the inherent uncertainty associated with energy forecasting is crucial for guiding future research and model refinement, paving the way for more robust and reliable decision-support tools.

7. Conclusions and future work

Accurate, spatially detailed, and interpretable energy demand forecasting is essential for effective energy management, infrastructure planning, and the transition to sustainable urban environments. This study addressed this critical need by developing a novel deep learning model integrating time series imaging, kernel density estimation (KDE), and an encoder–forecaster architecture to capture the complex spatiotemporal dynamics of urban energy consumption. By transforming discrete data into continuous density maps, our model leverages CNNs to extract spatial features and learn intricate dependencies. Importantly, the preservation of the 2D spatial structure throughout the prediction process enhances the interpretability of the forecasts, providing valuable spatial insights. Validation using real-world data from Shanghai's Pudong District demonstrated superior performance compared to traditional and state-of-the-art methods, empowering stakeholders to make informed decisions toward more sustainable energy systems.

While our model demonstrates promising results, future research will focus on enhancing its capabilities and addressing its limitations. We plan to incorporate uncertainty quantification techniques, such as probabilistic forecasting, Bayesian deep learning, and ensemble methods, to provide more robust and informative predictions. Furthermore, integrating additional data sources, including weather forecasts, socio-economic indicators, building characteristics, and technology trends, will improve predictive accuracy and provide deeper insights into the factors driving energy consumption. Enhancing the model's interpretability through advanced visualization tools, enabling interactive exploration of the spatial distribution of predicted demand and analysis of input variable influence, is another key objective. Finally, we aim to integrate our model with real-time energy management systems for dynamic adaptation and explore methods for adaptive learning and model updating to maintain accuracy in evolving urban environments. These advancements will contribute to more effective and robust urban energy planning, supporting the development of sustainable and resilient cities.

CRediT authorship contribution statement

Siyuan Jia: Writing – original draft, Methodology, Writing – review & editing, Software, Data curation. **Xiufeng Liu:** Writing – review & editing, Methodology, Supervision, Writing – original draft, Formal analysis. **Letian Zhao:** Writing – review & editing, Software, Methodology, Writing – original draft. **Chaofan Wang:** Software, Writing – review & editing, Methodology, Writing – original draft. **Jieyang Peng:** Methodology, Writing – review & editing, Writing – original draft. **Xiang Li:** Writing – original draft, Writing – review & editing, Methodology. **Zhibin Niu:** Project administration, Writing – original draft, Investigation, Supervision, Methodology, Writing – review & editing, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used chatGPT to verify the results of this experiment. After using this tool/service, the authors reviewed and edited the content as needed and takes full responsibility for the content of the publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

References

[1] Woodley L, Rossetti P, Nunes A. Targeted electric vehicle procurement incentives facilitate efficient abatement cost outcomes. Sustain Cities Soc 2023;96:104627.
[2] Shahmohammad M, Salamattalab MM, Sohn W, Kouhizadeh M, Aghamohmmadi N. Opportunities and obstacles of blockchain use in pursuit of sustainable development goal 11: A systematic scoping review. Sustain Cities Soc 2024;105620.
[3] Ahmad AS, Hassan MY, Abdullah MP, Rahman HA, Hussin F, Abdullah H, et al. A review on applications of ANN and SVM for building electrical energy consumption forecasting. Renew Sustain Energy Rev 2014;33:102–9.
[4] Hassan S, Khosravi A, Jaafar J, Khanesar MA. A systematic design of interval type-2 fuzzy logic system using extreme learning machine for electricity load demand forecasting. Int J Electr Power Energy Syst 2016;82:1–10.
[5] Ahmad T, Zhang H, Yan B. A review on renewable energy and electricity requirement forecasting models for smart grid and buildings. Sustain Cities Soc 2020;55:102052.
[6] Peng S, Chen Q, Zheng C, Liu E. Analysis of particle deposition in a new-type rectifying plate system during shale gas extraction. Energy Sci Eng 2020;8(3):702–17.
[7] Meng F, Lu Z, Li X, Han W, Peng J, Liu X, et al. Demand-side energy management reimagined: A comprehensive literature analysis leveraging large language models. Energy 2024;291:130303.
[8] Xiao J, Li Y, Xie L, Liu D, Huang J. A hybrid model based on selective ensemble for energy consumption forecasting in China. Energy 2018;159:534–46.
[9] Wang H, Wen C, Duan L, Li X, Liu D, Guo W. Sustainable energy transition in cities: A deep statistical prediction model for renewable energy sources management for low-carbon urban development. Sustain Cities Soc 2024;107:105434.
[10] Hu Y, Liu H, Wu S, Zhao Y, Wang Z, Liu X. Temporal collaborative attention for wind power forecasting. Appl Energy 2024;357:122502.
[11] Somu N, MR GR, Ramamritham K. A deep learning framework for building energy consumption forecast. Renew Sustain Energy Rev 2021;137:110591.

[12] Heidari A, Navimipour NJ, Unal M. Applications of ML/DL in the management of smart cities and societies based on new trends in information technologies: A systematic literature review. Sustain Cities Soc 2022;85:104089.
[13] Song J, Gao S, Zhu Y, Ma C. A survey of remote sensing image classification based on CNNs. Big Earth Data 2019;3(3):232–54.
[14] Samsi S, Mattioli CJ, Veillette MS. Distributed deep learning for precipitation nowcasting. In: 2019 IEEE high performance extreme computing conference. IEEE; 2019, p. 1–7.
[15] Zhang W, Yu Y, Qi Y, Shu F, Wang Y. Short-term traffic flow prediction based on spatio-temporal analysis and CNN deep learning. Transp A: Transp Sci 2019;15(2):1688–711.
[16] Fahim M, Fraz K, Sillitti A. TSI: Time series to imaging based model for detecting anomalous energy consumption in smart buildings. Inform Sci 2020;523:1–13.
[17] Huang Y, Zhao Y, Wang Z, Liu X, Liu H, Fu Y. Explainable district heat load forecasting with active deep learning. Appl Energy 2023;350:121753.
[18] Li X, Kang Y, Li F. Forecasting with time series imaging. Expert Syst Appl 2020;160:113680.
[19] Arendt K, Jradi M, Shaker HR, Veje C. Comparative analysis of white-, gray- and black-box models for thermal simulation of indoor environment: Teaching building case study. In: Proceedings of the 2018 building performance modeling conference and simBuild co-organized by ASHRAE and IBPSA-USA, chicago, IL, USA. 2018, p. 26–8.
[20] Ramanathan R, Engle R, Granger CW, Vahid-Araghi F, Brace C. Short-run forecasts of electricity loads and peaks. Int J Forecast 1997;13(2):161–74.
[21] Pappas SS, Ekonomou L, Karamousantas DC, Chatzarakis G, Katsikas S, Liatsis P. Electricity demand loads modeling using AutoRegressive moving average (ARMA) models. Energy 2008;33(9):1353–60.
[22] Reikard G. Predicting solar radiation at high resolutions: A comparison of time series forecasts. Sol Energy 2009;83(3):342–9.
[23] Huang R, Huang T, Gadh R, Li N. Solar generation prediction using the ARMA model in a laboratory-level micro-grid. In: 2012 IEEE third international conference on smart grid communications. IEEE; 2012, p. 528–33.
[24] Atique S, Noureen S, Roy V, Subburaj V, Bayne S, Macfie J. Forecasting of total daily solar energy generation using ARIMA: A case study. In: 2019 IEEE 9th annual computing and communication workshop and conference. IEEE; 2019, p. 0114–9.
[25] Alsharif MH, Younes MK, Kim J. Time series ARIMA model for prediction of daily and monthly average global solar radiation: The case study of Seoul, South Korea. Symmetry 2019;11(2):240.
[26] Server F, Kissock JK, Brown D, Mulqueen S. Estimating industrial building energy savings using inverse simulation. American Society of Heating, Refrigerating and Air-Conditioning Engineers; 2011.
[27] Walter T, Sohn MD. A regression-based approach to estimating retrofit savings using the building performance database. Appl Energy 2016;179:996–1005.
[28] Lazos D, Sproul AB, Kay M. Optimisation of energy management in commercial buildings with weather forecasting inputs: A review. Renew Sustain Energy Rev 2014;39:587–603.
[29] Xu N, Dang Y, Gong Y. Novel grey prediction model with nonlinear optimized time response method for forecasting of electricity consumption in China. Energy 2017;118:473–80.
[30] Wu L, Gao X, Xiao Y, Yang Y, Chen X. Using a novel multi-variable grey model to forecast the electricity consumption of Shandong Province in China. Energy 2018;157:327–35.
[31] Mui KW, Satheesan MK, Wong LT. Building cooling energy consumption prediction with a hybrid simulation approach: Generalization beyond the training range. Energy Build 2022;276:112502.
[32] Wang Y, Yang Z, Wang L, Ma X, Wu W, Ye L, et al. Forecasting China's energy production and consumption based on a novel structural adaptive Caputo fractional grey prediction model. Energy 2022;259:124935.
[33] Voulis N, Warnier M, Brazier FM. Understanding spatio-temporal electricity demand at different urban scales: A data-driven approach. Appl Energy 2018;230:1157–71.
[34] Dubey AK, Kumar A, García-Díaz V, Sharma AK, Kanhaiya K. Study and analysis of SARIMA and LSTM in forecasting time series data. Sustain Energy Technol Assess 2021;47:101474.
[35] Feng Y, Yao J, Li Z, Zheng R. Uncertainty prediction of energy consumption in buildings under stochastic shading adjustment. Energy 2022;254:124145.
[36] Peng L, Wang L, Xia D, Gao Q. Effective energy consumption forecasting using empirical wavelet transform and long short-term memory. Energy 2022;238:121756.
[37] Jin N, Yang F, Mo Y, Zeng Y, Zhou X, Yan K, et al. Highly accurate energy consumption forecasting model based on parallel LSTM neural networks. Adv Eng Inform 2022;51:101442.
[38] Elbeltagi E, Wefki H. Predicting energy consumption for residential buildings using ANN through parametric modeling. Energy Rep 2021;7:2534–45.
[39] Jin W, Fu Q, Chen J, Wang Y, Liu L, Lu Y, et al. A novel building energy consumption prediction method using deep reinforcement learning with consideration of fluctuation points. J Build Eng 2023;63:105458.
[40] Liu T, Tan Z, Xu C, Chen H, Li Z. Study on deep reinforcement learning techniques for building energy consumption forecasting. Energy Build 2020;208:109675.
[41] Zhong H, Wang J, Jia H, Mu Y, Lv S. Vector field-based support vector regression for building energy consumption prediction. Appl Energy 2019;242:403–14.
[42] Ozcan A, Catal C, Kasif A. Energy load forecasting using a dual-stage attention-based recurrent neural network. Sensors 2021;21(21):7115.
[43] Muralitharan K, Sakthivel R, Vishnuvarthan R. Neural network based optimization approach for energy demand prediction in smart grid. Neurocomputing 2018;273:199–208.
[44] Le T, Vo MT, Vo B, Hwang E, Rho S, Baik SW. Improving electric energy consumption prediction using CNN and Bi-LSTM. Appl Sci 2019;9(20):4237.
[45] Ghalehkhondabi I, Ardjmand E, Weckman GR, Young WA. An overview of energy demand forecasting methods published in 2005–2015. Energy Syst 2017;8:411–47.
[46] Xiao L, Tan H, Jia L, Dai J, Zhang Y. New error function designs for finite-time ZNN models with application to dynamic matrix inversion. Neural Comput Appl 2020;32(16):12419–32.
[47] Xiao L, Zhang Y, Dai J, Chen K, Yang S, Li W, et al. A new noise-tolerant and predefined-time ZNN model for time-dependent matrix inversion. Neural Netw 2019;117:124–34.
[48] Jiang W, Wu X, Gong Y, Yu W, Zhong X. Holt–winters smoothing enhanced by fruit fly optimization algorithm to forecast monthly electricity consumption. Energy 2020;193:116779.
[49] Somu N, MR GR, Ramamritham K. A hybrid model for building energy consumption forecasting using long short term memory networks. Appl Energy 2020;261:114131.
[50] Zhang G, Tian C, Li C, Zhang JJ, Zuo W. Accurate forecasting of building energy consumption via a novel ensembled deep learning method considering the cyclic feature. Energy 2020;201:117531.
[51] Lu H, Cheng F, Ma X, Hu G. Short-term prediction of building energy consumption employing an improved extreme gradient boosting model: A case study of an intake tower. Energy 2020;203:117756.
[52] Peng J, Kimmig A, Niu Z, Wang J, Liu X, Ovtcharova J. A flexible potential-flow model based high resolution spatiotemporal energy demand forecasting framework. Appl Energy 2021;299:117321.
[53] Peng J, Kimmig A, Wang J, Liu X, Niu Z, Ovtcharova J. Dual-stage attention-based long-short-term memory neural networks for energy demand prediction. Energy Build 2021;249:111211.
[54] Bu S-J, Cho S-B. Time series forecasting with multi-headed attention-based deep learning for residential energy consumption. Energies 2020;13(18):4722.
[55] Wang T, Zhao J, Liu Q, Wang W. Granular-based multilayer spatiotemporal network with control gates for energy prediction of steel industry. IEEE Trans Instrum Meas 2021;70:1–12.
[56] Kim T-Y, Cho S-B. Predicting residential energy consumption using CNN-LSTM neural networks. Energy 2019;182:72–81.
[57] Alhussein M, Aurangzeb K, Haider SI. Hybrid CNN-LSTM model for short-term individual household load forecasting. IEEE Access 2020;8:180544–57.
[58] Ageng D, Huang C-Y, Cheng R-G. A short-term household load forecasting framework using LSTM and data preparation. IEEE Access 2021;9:167911–9.
[59] Lilhore UK, Dalal S, Radulescu M, Barbulescu M. Smart grid stability prediction model using two-way attention based hybrid deep learning and MPSO. Energy Explor Exploit 2024;01445987241266892.
[60] D'Aversa A, Polimena S, Pio G, Ceci M. Leveraging spatio-temporal autocorrelation to improve the forecasting of the energy consumption in smart grids. In: International conference on discovery science. Springer; 2022, p. 141–56.
[61] Ohtsuka Y, Oga T, Kakamu K. Forecasting electricity demand in Japan: A Bayesian spatial autoregressive ARMA approach. Comput Statist Data Anal 2010;54(11):2721–35.
[62] Shu J, Zhang X, Yao Y, Yi D, Gu B. Graph spatio-temporal attention network-based electricity demand forecasting. In: 2021 6th international conference on power and renewable energy. IEEE; 2021, p. 792–7.
[63] Jiang H, Dong Y, Dong Y, Wang J. Power load forecasting based on spatial-temporal fusion graph convolution network. Technol Forecast Soc Change 2024;204:123435.
[64] Wu J, Niu Z, Li X, Huang L, Nielsen PS, Liu X. Understanding multi-scale spatiotemporal energy consumption data: A visual analysis approach. Energy 2023;263:125939.
[65] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. Adv Neural Inf Process Syst 2014;27.
[66] Niu Z, Wu J, Liu X, Huang L, Nielsen PS. Understanding energy demand behaviors through spatio-temporal smart meter data analysis. Energy 2021;226:120493.
[67] Xingjian S, Chen Z, Wang H, Yeung D-Y, Wong W-K, Woo W-c. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: Advances in neural information processing systems. 2015, p. 802–10.
[68] Ballas N, Yao L, Pal C, Courville A. Delving deeper into convolutional networks for learning video representations. 2015, arXiv preprint arXiv:1511.06432.
[69] Wang Y, Long M, Wang J, Gao Z, Yu PS. Predrnn: Recurrent neural networks for predictive learning using spatiotemporal lstms. Adv Neural Inf Process Syst 2017;30.
[70] Lin Z, Li M, Zheng Z, Cheng Y, Yuan C. Self-attention convlstm for spatiotemporal prediction. In: Proceedings of the AAAI conference on artificial intelligence, vol. 34, (07):2020, p. 11531–8.

[71] Sacchi MD, Ulrych TJ, Walker CJ. Interpolation and extrapolation using a high-resolution discrete Fourier transform. IEEE Trans Signal Process 1998;46(1):31–8.
[72] Woo W, Wong W. Application of optical flow techniques to rainfall nowcasting. In: The 27th conference on severe local storms. 2014.
[73] Farnebäck G. Two-frame motion estimation based on polynomial expansion. In: Scandinavian conference on image analysis. Springer; 2003, p. 363–70.
[74] Woo W-c, Wong W-k. Operational application of optical flow techniques to radar-based rainfall nowcasting. Atmosphere 2017;8(3):48.
[75] Shi X, Chen Z, Wang H, Yeung D-Y, Wong W-K, Woo W-c. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. Adv Neural Inf Process Syst 2015;28.
[76] Guen VL, Thome N. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 11474–84.
[77] Wang Y, Wu H, Zhang J, Gao Z, Wang J, Philip SY, et al. Predrnn: A recurrent neural network for spatiotemporal predictive learning. IEEE Trans Pattern Anal Mach Intell 2022;45(2):2208–25.
[78] Gao Z, Tan C, Wu L, Li SZ. Simvp: Simpler yet better video prediction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 3170–80.
[79] Tan C, Gao Z, Wu L, Xu Y, Xia J, Li S, et al. Temporal attention unit: Towards efficient spatiotemporal predictive learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2023, p. 18770–82.
[80] Liu X, Shi X-Q, Peng Z-R, He H-D. Quantifying the effects of urban fabric and vegetation combination pattern to mitigate particle pollution in near-road areas using machine learning. Sustain Cities Soc 2023;93:104524.
[81] Abu-Rayash A, Dincer I. Development of an integrated sustainability model for resilient cities featuring energy, environmental, social, governance and pandemic domains. Sustain Cities Soc 2023;92:104439.
[82] Liu X, Golab L, Golab W, Ilyas IF, Jin S. Smart meter data analytics: Systems, algorithms, and benchmarking. ACM Trans Database Syst 2016;42(1):1–39.
[83] Acosta MP, Dikkers M, Vahdatikhaki F, Santos J, Dorée AG. A comprehensive generalizability assessment of data-driven urban heat Island (UHI) models. Sustain Cities Soc 2023;96:104701.