AI-Enabled Digital Twin Framework for Intelligent Residential Energy Monitoring, Forecasting, and Advisory

Sujal Karoshi[1], Anmol Lohar[2], Shravani Gundawade[3], Pratiksha Ghorapade[4], Prof. Dr. R. N. Patil[5]
[1,2,3,4,5]*DKTE Society's Textile & EngineeringInstitute of technology in Ichalkaranji*

***Abstract*—Residential energy users often don't have information about how much energy they use, when they use the most and how much it costs. This paper introduces a framework that uses the Internet of Things and artificial intelligence to manage energy in homes. The system uses ESP32-based sensors, MQTT communication, AI analysis and a digital energy twin that works like a real-time copy of the homes energy use. The system offers real-time tracking, analysis of how energy's used, detecting when energy use is too high finding problems estimating bills and giving tips on saving energy. A web-based dashboard called VoltWatch lets people watch their devices and see how energy is used. A Smart Energy Assistant helps users understand what is going on. The system brings together monitoring, predicting, simulating and giving advice to help people make more efficient choices, about their home energy.**

***Index Terms*—Internet of Things, Artificial Intelligence, Digital Energy Twin, Edge Computing, MQTT, Residential Energy Management, Smart Energy Monitoring, Energy Advisory**

I. INTRODUCTION

The rise in the use of devices and distributed energy technologies has led to a growing need for smart ways to manage energy in homes. Traditional energy monitoring systems usually measure voltage, current, power and total energy use. These systems do not help much in understanding how energy is used over time predicting future needs spotting unusual usage or helping users make cost-efficient choices. IoT-based energy management systems solve these problems by allowing real-time sensing, monitoring and control of energy use in homes.

Still most IoT systems depend on cloud computing for data processing. This can cause delays and high communication costs when large amounts of sensor data are sent over the network. Edge computing offers a solution by processing data near the source close to the sensors. The system we are proposing includes processing, data buffering, prediction, event detection and selective communication using the MQTT protocol. Artificial Intelligence or AI plays a role in making energy systems smarter. AI can help with forecasting energy use detecting patterns finding peak load times estimating bills and giving personalized tips to save energy. Our system uses AI. Adds a user-focused advisory layer that turns complex data into clear and useful guidance for users. Digital Twin technology creates a model of a real-world system. This model can be used for monitoring, simulation and helping with decisions. The framework we propose includes a digital energy twin. This twin represents each appliance in the home. Shows its current operating state. It allows users to test energy-saving actions in a virtual setting before making changes in the real world. This paper introduces an AI-powered digital energy twin framework, for better residential energy management. The system combines ESP32 sensors, edge computing, MQTT communication, a Flask-based backend, AI analytics, a VoltWatch dashboard and a Smart Energy Assistant. Together these components provide real-time monitoring, energy analysis, bill forecasting, predictive insights and smart recommendations to help users manage their energy efficiently.

II. RELATED WORK

The development of residential energy

management has moved from simple monitoring to systems that can predict, optimize and make automated decisions for energy management. Existing research can broadly be split into IoT-based monitoring, AI-assisted energy management, edge-powered analytics and digital-twin systems for energy management.

A. IoT-Based Residential Energy Monitoring

IoT technology lets us keep track of data from home appliances and smart meters. This gives users real-time insight into how much energy's used and supports energy management. Usually these systems use sensors, small controllers, communication protocols and web or mobile screens to watch and control usage. Traditional IoT designs often send amounts of sensor data to central servers. This can raise the amount of data that must travel and cause delays for real-time use. Using protocols such as MQTT has become important for efficient data transfer in limited-resource IoT settings. The proposed framework expands the monitoring concept by adding real-time sensing, local processing and smart analysis. It does not just show measured values. The design uses an ESP32-based sensor layer and MQTT communication to move key energy information for energy management.

B. AI-Based Home Energy Management

Artificial Intelligence is now used more in home energy management for forecasting controlling appliances balancing demand and making choices. For example, an AI-based Home Energy Management System shown in the IEEE Internet of Things Journal used prediction tools and smart appliance controls while looking at both energy efficiency and resident comfort. The test study showed that AI can improve home energy management beyond monitoring. Newer studies look at deep-learning smart-home systems that predict energy use with models such as EMA, ARIMA and LSTM. A system published in Discover Internet of Things used multimodal sensing, local computing, MQTT and energy prediction showing that edge computing can boost energy analytics for energy management. Even though these methods show how AI can predict and improve energy use the proposed system also turns analysis into advice for energy management. It gives bill estimates, energy-saving tips, peak-hour guidance and chat-style help.

C. Edge Computing for Energy Management

Cloud-based IoT designs can become slow when much sensor data is sent all the time. Edge computing solves this by moving some processing to where data is made. This cuts down on data travel. Let’s decisions happen faster for energy management. Recent research in smart-home energy management shows that edge processing can predict energy use and run tasks reducing the need for far-off services while still keeping real-time monitoring and control for energy management. In the framework edge processing sits inside the monitoring chain. ESP32 sensors can store data locally. Analyze it to pull out helpful facts before sending only what matters to the backend. The method blends sensing local analysis, selective data transfer and central analysis for energy management.

D. Digital Twin-Based Energy Management

Digital Twin tech has become a way to model real energy systems with virtual replicas that help watch, test, analyze and decide for energy management. Bayer and Pruckner built a twin of an energy system using real smart-meter data and building details showing how virtual parts and tests can help decision making for energy management. Recent IEEE work also finds that mixing AI with twins is a new path for smart building energy management especially for self-learning systems, instant optimization and decisions that fit occupants for energy management. Building on these ideas the new system has a digital twin that shows home appliances and their states. Of just drawing the model the system lets users change virtual loads and run what-if tests so they can see how energy use might change before making real appliance changes for energy management.

E. Research Gap

The studies above show progress in separate areas like IoT monitoring, AI prediction, edge computing and digital twins. These ideas are often treated alone or focus on one specific optimization. There is still a chance to build a platform that

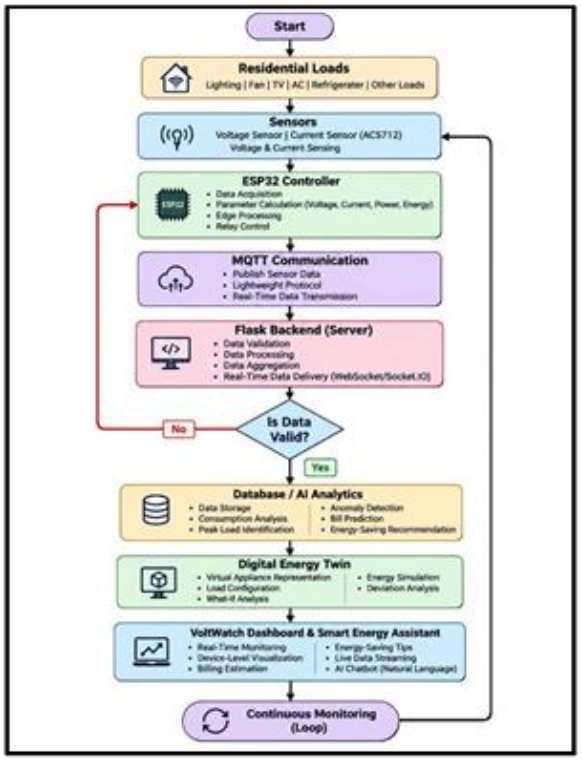
mixes real-time monitoring, edge help good IoT communication AI insight, interactive twin simulation and user advice, for home energy management.

III. PROPOSED METHODOLOGY

The proposed system is an IoT and AI enabled framework designed to improve energy monitoring and management. It mixes real-time sensing, edge-level processing, MQTT communication, AI based analytics, an interactive digital energy twin and a web based advisory interface. The system goes beyond energy monitoring by giving users useful information, about consumption patterns, peak demand, energy cost and possible energy saving action.

A. System Architecture

The system is built using connected layers: physical, communication, processing, analytics and user interaction. Electrical data from home appliances are collected using voltage and current sensors linked to an ESP32 controller. The ESP32 does processing and computes electrical values like voltage, current, power and total energy used.

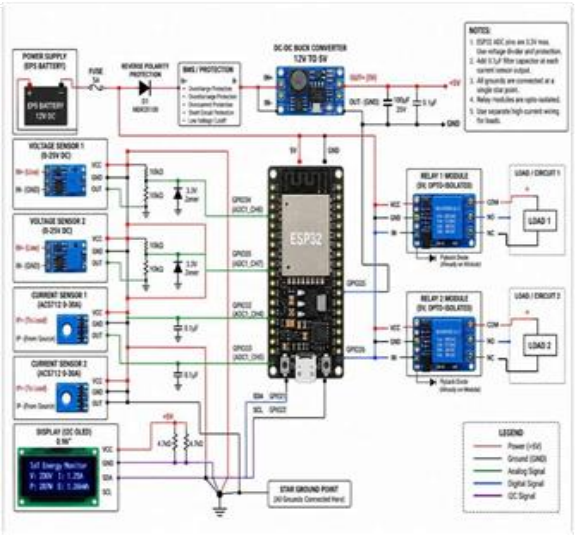


This data is then sent using the MQTT protocol to the backend server. The backend, built with Flask checks the data for accuracy cleans it groups it stores it and sends it in time when needed. The cleaned data is passed on to the AI analytics module the energy twin and the user interface, for further use.

B. Hardware Design

The hardware part is built around an ESP32 microcontroller. This microcontroller is the part that senses and controls. Voltage and current sensors are used to check the parts of the things that are connected. The current is measured using ACS712 sensors. The voltage is measured using voltage sensors. The ESP32 gets the signals from the sensors through its input channels. It works on the measurements to figure out the voltage the current, the power and the total energy used. Relay modules are added to let the system switch the loads on and off. The prototype also has a power supply that's steady parts that protect the system and things that make the signals cleaner so the system works well.

The way the hardware is set up lets the system get information, from many things that are connected. It sends the information that has been worked on to the software part.



C. IoT Communication and Data Processing

MQTT is used as the communication protocol between the ESP32 sensing unit and the backend system. Its lightweight publish-subscribe

architecture allows for transfer of energy data with relatively low communication overhead. At the backend the Flask server receives the data that comes in and does validation, cleaning, aggregation and feature extraction before storing the information. Real-time updates are sent to the web interface using Socket. IO. Local processing and buffering can also be used to keep data safe during communication interruptions.

The processed data are the foundation, for energy visualization, historical analysis, prediction, billing estimation and intelligent recommendations.

D. AI-Based Energy Analytics

The AI analytics part turns the energy data that is gathered into information that helps with making decisions. Data from the past and data that is happening now is looked at to find how energy is used and how the need for energy changes.

This analytics part helps with functions like:

- Energy consumption analysis
- Peak load identification
- Finding strange usage
- Estimating and predicting electricity bills
- Forecasting how much energy will be used
- Suggestions for saving energy
- Guidance that takes cost into account when using energy

The information that comes from this is given to the user in an easy-to-understand form of making the user deal with just the basic electrical numbers.

E. Interactive Digital Energy Twin

An interactive digital energy twin is used to create a version of home appliances and how they are working. This digital twin lets people set up and check the state of appliances and virtual loads without changing the equipment. The platform lets users do what-if analysis, where they can see what happens if they turn appliances on or off or change how they work. This helps people understand how energy is spread out how much is used and what might happen with demand. This digital energy twin works as a simulation. Help tool between the energy data and what the user does.

F. Energy Advisory and User Interface

The user part of the system has mainly the VoltWatch dashboard and the Smart Energy Assistant. The VoltWatch dashboard shows real-time pictures of voltage, power, energy use, device status and how much the electricity might cost. It shows past data and live updates so users can see how energy use changes over time. The Smart Energy Assistant talks in language and gives advice that is right for each user. It can answer questions about electricity bills how much is being used now times when energy use is high what is expected to be used and ways to save energy.

By using real-time checking with AI advice and interactive simulation the user interface helps people go from watching how much energy is used to making smart choices, about managing energy.

Hardware Components

Sr. No.	Component	Function / Specification
1	ESP32 Microcontroller	Data acquisition, parameter calculation, communication (MQTT) and load control
2	Voltage Sensor (0–25 V DC)	Measures voltage across the load
3	Current Sensor (ACS712, 0–30 A)	Measures load current
4	Relay Module (5 V, Opto-isolated)	Switching and control of connected loads
5	12 V DC EPS Battery	Primary power supply (12 V DC)
6	BMS / Protection Circuit	Overcharge, over-discharge, overcurrent, short-circuit and low-voltage protection
7	DC-DC Buck Converter (12 V to 5 V)	Provides regulated 5 V supply for ESP32 and modules
8	I²C OLED Display (0.96")	Displays real-time electrical parameters
9	Fuse (5 A) & Reverse Polarity Protection	Overcurrent and reverse connection protection
10	Capacitors (100 µF, 0.1 µF)	Supply filtering and noise reduction
11	Voltage Divider (10 kΩ)	Scales and protects ADC input for voltage measurement

Software Components

Sr. No.	Software / Technology	Function / Description
1	ESP32 Firmware (Arduino IDE)	Sensor data acquisition, parameter calculation, relay control and MQTT communication.
2	MQTT (Mosquitto Broker)	Lightweight publish-subscribe protocol for real-time data transmission.
3	Flask Backend (Python)	Data reception, validation, processing, aggregation and API services.
4	Database (SQLite / MySQL)	Storage of real-time and historical energy data.
5	Socket.IO	Real-time data delivery to the web dashboard.
6	AI Analytics (Python, ML)	Energy consumption analysis, peak-load detection, anomaly identification, bill prediction and energy-saving recommendations.
7	Digital Energy Twin (Simulation Module)	Virtual appliance representation, load configuration, what-if analysis and energy simulation.
8	VoltWatch Dashboard (Web Interface)	Real-time monitoring, device-level visualization, billing estimation and historical analysis.
9	Smart Energy Assistant (Natural Language Interface)	User query handling, personalized energy guidance and energy-saving suggestions.

IV. RESULT AND DISCUSSION

A. Hardware Implementation

The energy monitoring system that was suggested was built using an ESP32 microcontroller, voltage

sensors, ACS712 sensors and relay modules. Light bulbs were used as test loads to see how the system worked when it was running. The voltage and current sensors checked the values of the bulbs that were connected.

The ESP32 took the sensor data. Sent it to the monitoring system. The setup was used to show how to measure voltage how to measure current how to calculate power and how to watch loads in real time. The relay modules let the bulbs be turned on and off. The data that was gathered was sent to the end and shown on the VoltWatch Pro dashboard so people could look at it and study it.



Fig. 1. Experimental Hardware Setup Using Electric Bulbs as Test Loads.

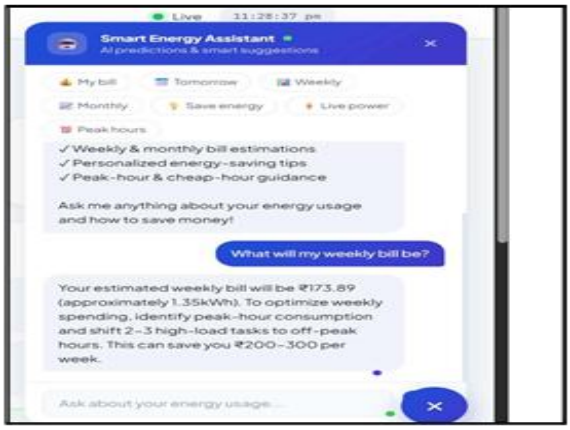
B. Software Implementation

The software component contains the VoltWatch Pro dashboard, the backend processing, an interactive digital energy twin and a chatbot-based Smart Energy Assistant. The VoltWatch Pro dashboard shows parameters such as voltage, current, power consumption, energy usage and estimated electricity cost. MQTT communication moves sensor data from the ESP32 to the backend. The Flask server then handles the measurements. Socket. IO lets the dashboard update in time. The software interface lets users see how electrical parameters change when test bulbs are turned on or off. The interactive digital energy twin offers a simulation space for household appliances letting users try load setups and study virtual energy consumption and estimated electricity costs.

The chatbot-based Smart Energy Assistant is built into the software platform. It supplies users with energy facts, consumption insights and tips for cutting electricity use. The conversational interface lets users ask questions about their energy consumption and receive advice drawn from the system data. By combining real-time monitoring, virtual energy simulation and chatbot help the software component gives a place, for understanding and controlling residential energy usage.



Fig. 2. VoltWatch Pro Dashboard with the Smart Energy Assistant.



C. Interactive Digital Energy Twin Simulation



Fig. 3. Interactive Energy Coach: Digital Energy Twin Simulation and Energy Analytics

The Interactive Energy Coach offers a simulation-based energy twin that lets you see how a home's energy use changes when you turn appliances on or off. The Interactive Energy Coach shows you numbers such as how much energy you use each month what your electricity bill might be, the highest load you reach the average load and the load right now. The Interactive Energy Coach also breaks down each appliance uses prediction to guess future use spots odd patterns and lets you try different scenarios to help you decide how to save energy.

In Fig. 3. The Interactive Energy Coach shows that the simulated monthly energy consumption is 764. 37 kWh and the bill would be about ₹6,115 if the rate is ₹8 per kWh. The current virtual load shown by the Interactive Energy Coach is 1. 655 kW while the peak load recorded was 3. 17 kW. These numbers, from the Interactive Energy Coach prove that the dashboard can display energy demand clearly and let you explore ways to cut energy use.

D. Appliance-Wise Energy Consumption Analysis

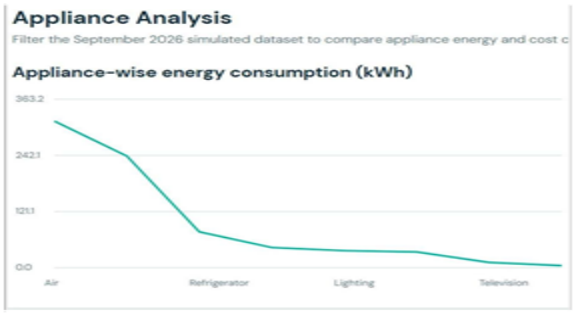


Fig. 4. Appliance-Wise Energy Consumption Analysis Using the Digital Energy Twin.

The appliance-wise energy analysis looks at how energy different household appliances use, in September 2026. The data is shown in graphs to make it easy to see how energy use varies between devices. The results show that the air conditioner uses the energy. After that the refrigerator and other household devices come next. This comparison helps point out which appliances use a lot of power. It also gives a way to decide which appliances should be targeted first for energy-saving actions. By studying this analysis users can better understand their energy use. It helps them make choices about how to manage energy at home.

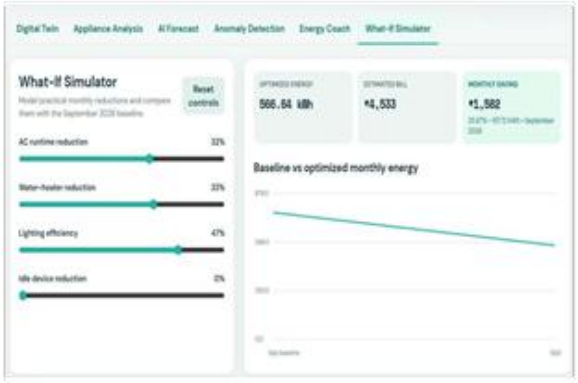
E. Simulated 24-Hour Load Profile Analysis



Fig. 5. Simulated 24-Hour Residential Load Profile for September 2026.

The 24-hour load profile shows how electricity use in a home change during the last 24 hours of September 2026. The graph displays changes in how much power's needed during the day. There are times when more energy is used and times when less is needed. A clear rise happens in the evening. The highest amount of power used is shown around 20:00. The graph also includes details about energy use and costs for each time period. This helps people see how changes, in power use affect the cost of electricity. This kind of study shows how the digital energy twin can help find times when power use's highest and look at how energy is used over time.

F. What-If Energy Simulation



The What-If Simulator helps people see how changing how they use household appliances can affect the amount of energy they use and how much it costs to power their home. People can try out combinations of appliances and how they are used in a pretend setting to see what kind of effect that might have on energy use. For instance,

turning off an appliance that uses a lot of power or using it for time can show how much energy use and electricity bills might go down. This tool offers a real-world way to look at ways to save energy before making any changes, to how appliances are used in a house.

G. Electrical Parameter Calculations

Power Calculation

$$P = V_{rms} \times I_{rms} \times PF \quad (1)$$

where P is power in watts, V_{rms} is the RMS voltage in volts, I_{rms} is the RMS current in amperes, and PF represents the power factor.

Energy Consumption Calculation

$$E = \frac{P \times t}{1000} \quad (2)$$

where E is energy in kWh, P is power in watts, and t is the operating time in hours.

Electricity Cost Estimation

$$C = E \times T \quad (3)$$

where C is the estimated cost in rupees, E is energy consumption in kWh, and T is the electricity tariff in ₹/kWh.

The electrical parameters measured during bulb testing were used to evaluate the system's monitoring functionality. The power consumed by a test load can be estimated using the measured voltage and current.

H. Summary of Simulated Residential Energy Dataset

Parameter	Value
Power supply	12 V DC
Call interval	10 s
Sampling time	1 s
Communication delay	< 0.5 s
Packet size	≈ 256 B
Test duration (simulation)	24 h (86,400 s)
Total data points	86,400
Database size	≈ 45.6 MB
System availability rate	99.38%

Appliance	Energy Consumption (kWh)	Percentage (%)
Air Conditioner	379.00	44.63
Water Heater	246.00	28.98
Refrigerator	78.50	9.25
Computer	48.50	5.71
Lighting	36.75	4.32
Fan	36.00	4.25
Washing Machine	16.00	1.89
Television	14.48	1.70
Total	**848.13**	**100**

Appliance-Wise Energy Consumption

I. Overall Discussion

The proposed IoT and AI enabled energy management framework brings together hardware-based energy monitoring, software-based visualization, appliance wise energy analysis and digital twin simulation. The physical prototype uses bulbs as test loads to show electrical parameter monitoring and switching control. The ESP32 sends the collected measurements to the backend while the VoltWatch Pro dashboard shows parameters and estimated electricity costs. The chatbot based Smart Energy Assistant gives an interface for energy related guidance.

The simulated dataset allows analysis of residential energy consumption. The appliance wise results point out that the air conditioner and water heater are the contributors to simulated energy demand. The 24-hour load profile shows how demand changes while the twin and What If Simulator help explore different appliance operating conditions. These features help users spot energy loads and look at possible energy saving opportunities.

However, the physical prototype and the simulation serve purposes. The bulb-based experiment shows hardware functionality while the digital twin gives an environment, for analyzing household energy scenarios. The consistency of the simulated results and the performance of the AI forecasting module need evaluation before their accuracy can be confirmed.

V. CONCLUSION

This paper introduces a system that uses Internet of Things and artificial intelligence to keep track of energy use in homes and manage it in a way. The system uses hardware like ESP32 for sensing MQTT for communication Flask for processing and VoltWatch Pro for showing data. Testing with bulbs shows that the system can monitor energy use and control devices. The Interactive Energy Coach adds features to the system. It uses a version of the home's energy use looks at how much each appliance uses shows energy use over 24 hours lets users try different scenarios and has a chatbot to help with energy tips. These tools help people understand their energy use and find ways to save energy. The data from the simulation helps analyze how energy each appliance uses and how much it

costs. Even though the system shows how hardware and digital tools can work together more work is needed to test how well it predicts energy use check if the calculations are accurate and see how it works with types of home appliances.
Overall, the system is a start, for creating better energy management tools that are easier to use and based on data.

REFERENCES

[1]. V. Sathappan, R. Rupashri, M. Sakthi Mahadevan, R. Praveen Rajesh, et al., "IoT-Based Home Power Monitoring with Smart Forecasting," in *Proc. 6th Int. Conf. Data Intelligence and Cognitive Informatics (ICDICI)*, 2025, doi: 10.1109/ICDICI66477.2025.11135333.

[2]. U. Ali, M. U. Ramzan, W. Ali, M. E. Rana, and A. Qayyum, "IoT-Driven Smart Energy Monitoring: Real-Time Insights and AI-Based Unit Predictions," in *Proc. IEEE 21st Student Conf. Research and Development (SCOReD)*, 2023, pp. 672–677, doi: 10.1109/SCOReD60679.2023.10563825.

[3]. M. Boukaf, F. Fadli, and N. Meskin, "A Comprehensive Review of Digital Twin Technology in Building Energy Consumption Forecasting," *IEEE Access*, vol. 12, pp. 187778–187799, 2024, doi: 10.1109/ACCESS.2024.3498107.

[4]. S. Agostinelli, F. Cumo, G. Guidi, and C. Tomazzoli, "Cyber-Physical Systems Improving Building Energy Management: Digital Twin and Artificial Intelligence," *Energies*, vol. 14, no. 8, Art. no. 2338, 2021, doi: 10.3390/en14082338.

[5]. F. H. Aghdam, M. Rasti, E. Pongracz, and A. Anvari-Moghaddam, "Digital Twin Assisted Real-Time Energy Management System for Smart Homes," *Energy and Buildings*, vol. 355, Art. no. 116978, 2026, doi: 10.1016/j.enbuild.2026.116978.