sensors

MDPI

Article

Digital Twin Smart City Visualization with MoE-Based Personal Thermal Comfort Analysis

Hoang-Khanh Lam, Phuoc-Dat Lam, Soo-Yol Ok * and Suk-Hwan Lee *

Department of Computer Engineering, Dong-A University, Busan 49315, Republic of Korea; 2371049@donga.ac.kr (H.-K.L.); 2478532@donga.ac.kr (P.-D.L.)
* Correspondence: sooyol@dau.ac.kr (S.-Y.O.); skylee@dau.ac.kr (S.-H.L.)

Abstract: Digital twin technology us used to create accurate virtual representations of objects or systems. Digital twins span the object's life cycle and keep updated with real-time data. Therefore, their simulation capabilities can be combined with deep learning to create a system that simulates scenarios, enabling analysis. As cities continue to grow and the demand for sustainable development increases, digital twin technology, combined with AI-driven analysis, will play a critical role in shaping the future of urban environments. The ability to accurately simulate and manage complex systems in real time opens up new possibilities for optimizing energy usage, reducing costs, and improving the quality of life for urban residents. In this study, a digital twin application is built to visualize a smart area in South Korea, utilizing a deep learning model for personal thermal comfort analysis, which can be useful for managing and saving building and household energy consumption. Using Cesium for Unreal, a powerful tool for integrating 3D geospatial data, and leveraging DataSmith to convert 3D data into Unreal Engine format, this study also contributes a roadmap for smart city application development, which is currently considered to be lacking. By creating a robust framework for smart city applications, this research not only addresses current challenges but also lays the groundwork for future innovations in urban planning and management.

Keywords: digital twin; smart city; deep learning; thermal comfort; Cesium; Unreal Engine



Received: 27 September 2024
Revised: 17 December 2024
Accepted: 14 January 2025
Published: 24 January 2025

Citation: Lam, H.-K.; Lam, P.-D.; Ok, S.-Y.; Lee, S.-H. Digital Twin Smart City Visualization with MoE-Based Personal Thermal Comfort Analysis. *Sensors* **2025**, *25*, 705. https://doi.org/10.3390/s25030705

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

A city is a combination of many components, geometric objects (buildings, houses, and streets), semantics (security and network traffic), and sensors (temperature, humidity, and air quality sensors). With the rapid development of cities, monitoring and managing these components is essential for improving quality of life and saving energy. Smart cities (SCs) have been defined by International Business Machines Corporation (IBM) as cities that use information and communication technology (ICT) to sense, analyze, and integrate the key information of their core systems [1]. In recent years, many studies have shown interest in 3D city modeling and digital twins, as they allow for the creation smart city applications in which decision makers can perform analyses, simulations, planning, and monitoring in different domains (energy management, urban planning, and weather simulation). Many approaches have been proposed, such as CityGML, CityJson, and the combination of Building Information Modeling (BIM) and Geographic Information System (GIS) modeling to provide City Information Modeling (CIM) [2]. Lam et al. [3] successfully implemented a method for converting Industry Foundation Class (IFC) data to 3D city models and evaluated the results through visualization across multiple applications. This method lays

a promising foundation for future smart city applications, as it can comprehensively map all IFC data to 3D city models, enabling the seamless integration of building information modeling (BIM) into smart city platforms. As shown in Figure 1, a smart city includes four main components: natural components (e.g., world terrain and Sun), geometric objects (static and mobile objects), semantics (e.g., noise level, network traffic, etc.), and sensors (e.g., weather sensors, etc.). These are the basic elements required to monitor and manage a smart city. Cities: Skyline, a popular game, can be described as a single-player, open-ended city-building simulation [4]. Players engage in urban planning by establishing a road network; controlling zoning; and providing public services such as budgeting, education, employment, pollution management, etc. Based on this concept, leveraging a 3D city model, we focus on visualizing housing objectsand weather conditions to keep track of rapidly changing conditions.

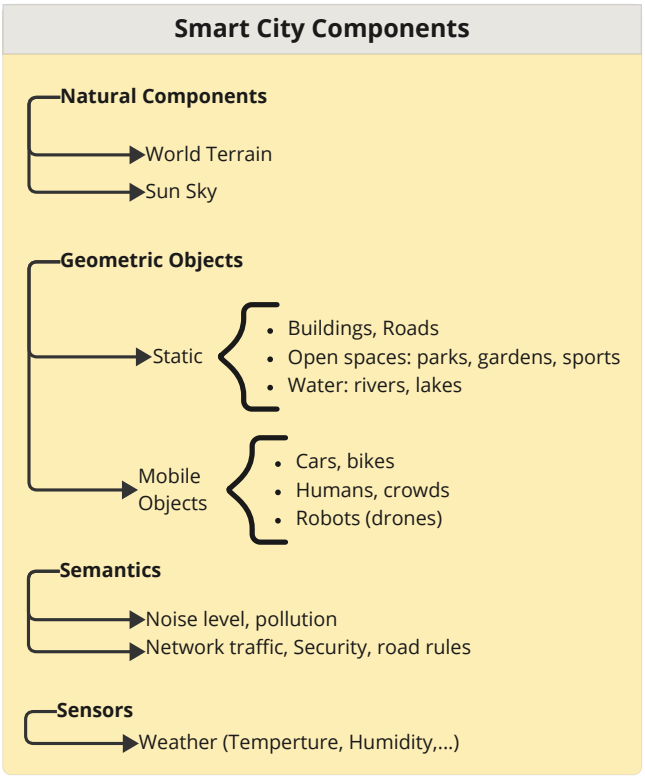


Figure 1. Main components of a smart city.

A contour map was built by Hao Wang et al. [5], with colors from cool to warm to reflecting low to high research intensity by topic, describing the distribution and concentration of studies based on the time scale and knowledge management. Figure 2 highlights a red area for transportation/mobility in 2020, indicating that this sector is the most extensively researched. Additionally, warm-toned areas are visible in other aspects of smart cities (SCs), such as energy/power systems, environmental and resource management, and public infrastructure management, suggesting frequent applications of digital twin (DT) technology in these domains. The figure also reveals that DTs have not yet been widely adopted in smart education (9) and the smart economy (10). Overall, research on DT technology has garnered significant attention in recent years. An early fire detection system was trained using synthetic data, following the method proposed in [6], demonstrating its potential as a use case within DT environments. Geographic Virtual City Miniature (GeoVCM) [7] demonstrates an application that integrates both real and virtual IoT devices into a digital twin (DT) platform, allowing city managers to interact with and monitor

urban environments in real time. A DT version of a two-story building in Larisa, northern Greece, was developed in [8], allowing managers to monitor sensor values in each room and track the building's electricity consumption. The Intelligent Transportation System [9] incorporates numerous IoT devices to integrate multi-dimensional and multi-state data into a collaborative data framework. This system enables the visualization of traffic flow and other details related to vehicles on the implemented roads, providing real-time insights into traffic conditions and vehicle movements.

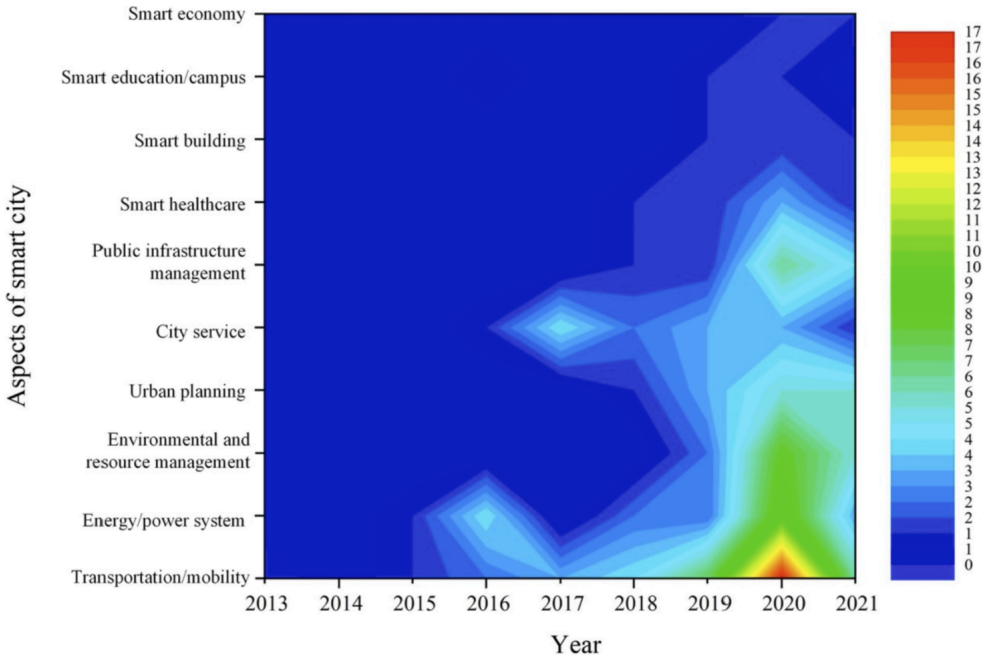


Figure 2. Distribution and concentration of studies on DT-supported SCs [5].

The indoor environment is the main factor affecting the health and well-being of residents/occupants. Thermal comfort refers to the condition of being physically and psychologically comfortable with the surrounding environment, especially the temperature. It is measured by various indices, such as Predicted Mean Vote (PMV), Predicted Percentage of Dissatisfied (PPD), and thermal sensation vote, which are the most prominent conventional methods [10]. Thermal comfort (TC) has been applied in buildings and houses by designing, suggesting, and controlling heating, ventilation, and air conditioning (HVAC) systems to maintain an optimal indoor temperature, relative humidity, and air movements. HVAC systems consume up to 50% of the total energy budget of a building, amounting to a staggering 20% of the total energy consumption in the USA [11]. There is a worldwide standard, ASHRAE 55 [12], determining a satisfactory thermal environment in occupied spaces and compliance documents. These indices assess thermal comfort levels by incorporating factors such as temperature, humidity, air velocity, clothing insulation, and metabolic rate. The development of personal comfort models requires the collection of extensive sensor-related measurements and user-labeled data, a process that can be highly intrusive and labor-intensive. To address this, Zeynep et al. [13] proposed a hybrid active learning framework aimed at reducing data collection costs while developing data-efficient and robust models. Many researchers have referenced their study on the three popular thermal comfort metrics shown in Figure 3: Thermal Sensation Vote (TSV), Thermal Preference Vote (TPC), and Thermal Comfort Vote (TCV). A deep neural network called DeepComfort [14] was proposed based on multi-task learning to predict the three most important thermal comfort metric, achieving 90% accuracy on a combination of the ASHRAE II dataset [15] and self-collected primary student dataset. We primarily rely

on TSV (thermal sensation vote) to evaluate our thermal comfort model, as this metric reflects how individuals perceive the current environmental conditions, aligning closely with our objective. Through the use of thermal comfort, we can analyze the effectiveness of HVAC systems, as well as make some suggestions to control the system to save on energy consumption. We build thermal comfort using a Mixture-of-Expert (MoE) model architecture with two main backbones, namely Long Short-Term Memory (LSTM) and a Liquid Neural Network (LNN), integrated with the transformer architecture for evaluation, analysis, and comparison.

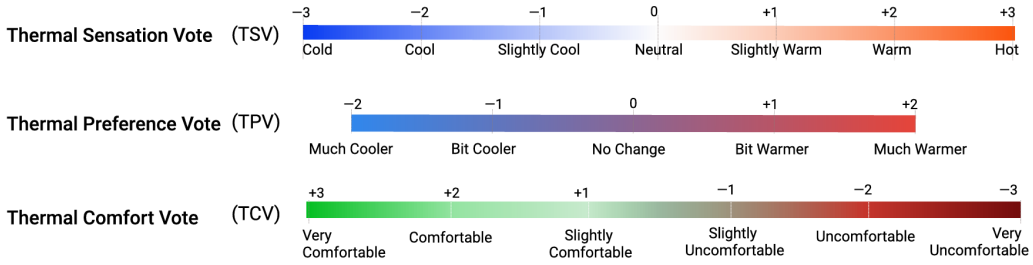


Figure 3. Popular thermal comfort metrics.

In this study, the smart city platform successfully maintained a high frame rate, ensuring smooth exploration of the virtual environment, including villages, houses, and building structures. Each component within the platform was embedded with detailed, queryable information, enhancing the user's ability to navigate and interact with the virtual space. The integration of the deep learning model into the platform allowed for real-time analysis of thermal comfort levels, considering individual characteristics such as height, weight, and clothing. This real-time prediction capability provided insights into each person's comfort based on the environmental conditions. The model demonstrated robust performance, delivering accurate thermal comfort predictions across various environmental scenarios. Additionally, the system effectively adjusted to diverse body conditions, providing highly personalized comfort assessments. The combination of the smart city platform and the deep learning model enabled a seamless integration of environmental modeling and real-time comfort analysis, offering significant potential for urban planning and smart city management by optimizing both infrastructure and individual well-being.

The remainder of this paper is structured as follows. Section 2 presents the technical details of the smart city platform and the personal thermal comfort model. This section covers how environmental measurements are visualized, the features required to develop the personal thermal comfort model, and the structural design of the model. Section 3 discusses the performance of both the thermal comfort model and the smart city platform, highlighting key evaluation metrics. Finally, the conclusions of the study and potential future work are outlined in Section 4.

2. Materials and Methods

As shown in Figure 4, our method integrates natural components, static geographic objects, and environmental sensor data (Section 2.1.1) into a platform built using Unreal Engine, as detailed in Section 2.2. OpenWeather is utilized to collect and transfer weather-related data to both the platform and the thermal comfort model. This model (Section 2.3) is powered by personal and environmental data (Section 2.1.2) and is connected to the platform through FastAPI (Section 2.4).

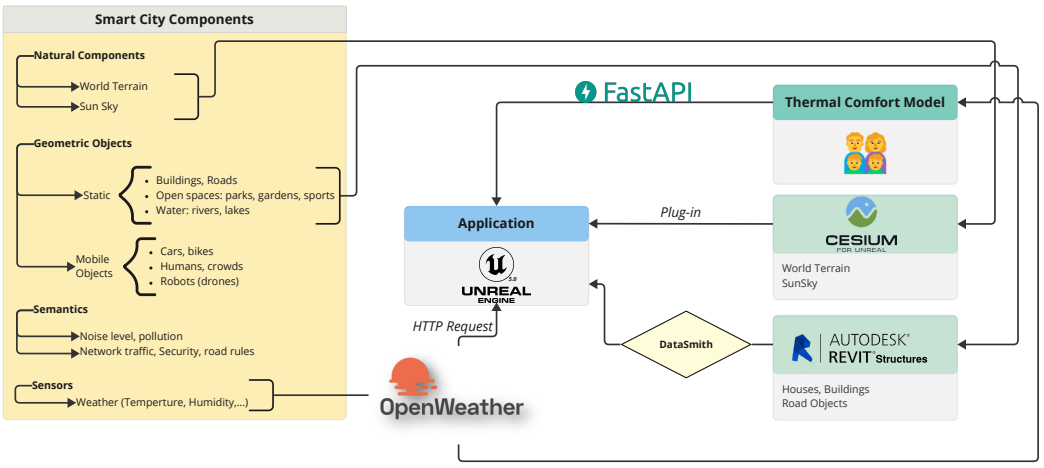


Figure 4. Overview of the proposed Smart City Platform.

2.1. Dataset Overview

2.1.1. Platform Data

The proposed approach can be conceptualized as a three-dimensional map incorporating detailed information and interactive elements, enabling users to interact with the environment and retrieve pertinent data in real time. A key component of such a map is the globe, which features 3D terrain, along with a dynamic solar illumination system that simulates daylight conditions. Cesium is a globally recognized platform for supporting 3D geospatial applications that has been widely utilized in numerous studies ([3,16–18]). With the assistance of Cesium, a geospatial system is implemented, enabling accurate visualization and management of spatial data in a 3D environment. Cesium facilitates the integration of geographic information, such as building locations and terrain data, into the smart city platform, allowing for real-time mapping and analysis of urban infrastructure within a global coordinate system, as shown in Figure 5. SunSky [19] is also integrated into the system to provide realistic daylight simulation based on the date, time, and time zone of the specific area. This feature enhances the accuracy of the environmental visualization by adjusting lighting conditions dynamically according to the geographical location and time of day, offering a more immersive and contextually accurate representation of the urban environment.

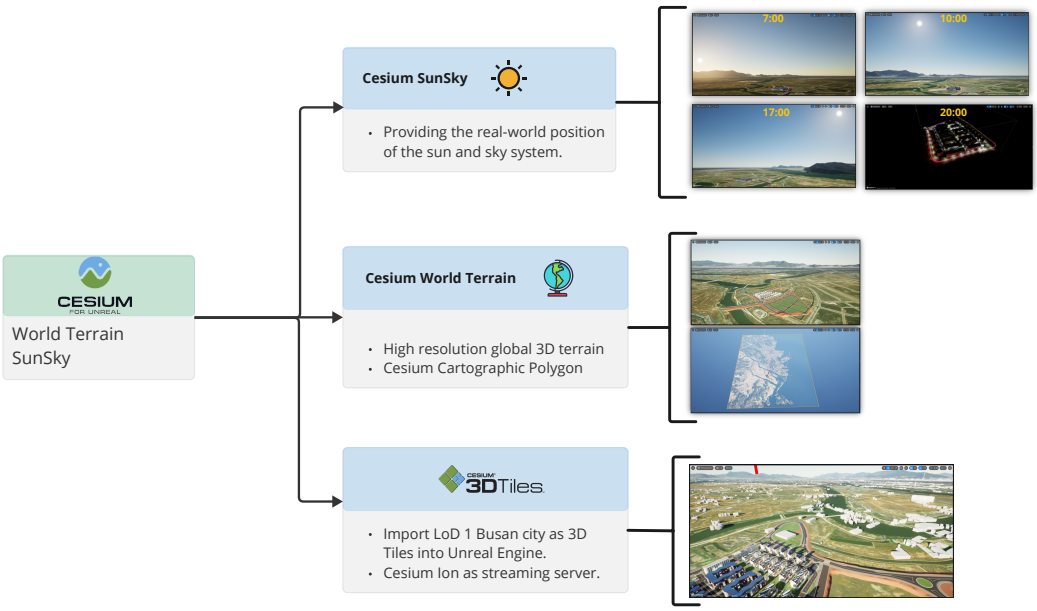


Figure 5. Leveraging Cesium support in Unreal Engine.

Among the important components that create smart city applications are houses and buildings. To import them into the application, we converted data from Revit (*.rvt*) to static mesh by leveraging the DataSmith plugin. As shown in Figure 6, Datasmith is a suite of tools and plugins designed to import complete pre-constructed scenes and complex assets from various industry-standard design applications into Unreal Engine. During the conversion of a Revit file to Datasmith format, the geographic data are preserved to position the houses accurately in the correct coordinates of the Cesium globe.

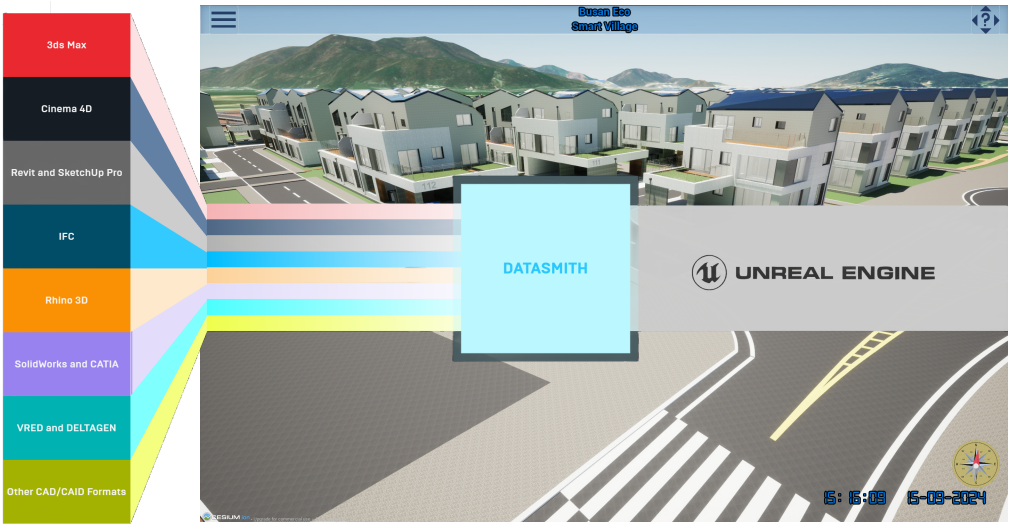


Figure 6. Three-dimensional houses and buildings visualized in Unreal Engine with DataSmith support.

The detailed information for each component is gathered, analyzed, and formatted in CityGML, which is then converted to 3D tiles for querying and visualization purposes. The conversion process begins with building information modeling (BIM) data in IFC format, which serve as the foundational input. These data are transformed into CityGML format at Level of Detail 4 (LOD4), capturing intricate architectural features. Subsequently, the CityGML data are converted into 3D Tiles format, optimized for efficient streaming and visualization. This streamlined process ensures a smooth transition from BIM to an interoperable 3D format, facilitating dynamic data utilization and real-time visualization. Cesium 3D Tiles is a geospatial data format and rendering engine crucial for visualizing and streaming 3D geospatial content over the web. It offers an efficient and scalable solution for delivering large-scale 3D geographic information, enabling users to interactively explore complex 3D environments in web-based applications and virtual globes. Cesium 3D Tiles excels in streaming and rendering detailed 3D models in real time, optimizing performance and allowing for seamless visualization of intricate data, even in large-scale or high-density scenarios.

2.1.2. Thermal Comfort Data

Most deep learning-based thermal comfort (DLTC) studies utilize feature parameters that encompass a combination of indoor environmental measurements (e.g., indoor temperature), outdoor environmental data (e.g., daily rainfall), and individual-specific characteristics (e.g., clothing) [20]. Accordingly, our thermal comfort model incorporates input features consisting of indoor environmental measurements (e.g., indoor temperature and relative humidity), individual-specific factors (e.g., clothing value), and weather data obtained from OpenWeather for the corresponding month during which the field experiments were conducted. The complete list of features in the dataset is presented in Table 1.

Table 1. Coding conventions include personal details of the subjects, indoor, outdoor measurements, and data on subjective thermal comfort assessments.

Field	Description
A. Individual-specific	
A1. Age	Age of the subjects
A2. Height	Height of the subjects (cm)
A3. Weight	Weight of the subjects (kg)
A4. Gender	Male, female, or undefined
A5. Clothing Insulation	Intrinsic clothing ensemble insulation of the subjects (clo)
A6. Metabolic Rate	Activity status at the time of the survey (met); refer to ASHRAE 55-2020 [21]
B. Indoor environmental measurements	
B1. Air temperature	Air temperature measured in the occupied zone (°C)
B2. Air Velocity	Air velocity measured in the occupied zone (m/s)
B3. Relative Humidity	Relative humidity measured in the occupied zone (%)
C. Outdoor environmental data	
C1. Season	Summer season, winter season, or transition season
C2. Climate zone	Severe Cold Zone (SC), Cold Zone (C), Hot Summer & Cold Winter Zone (HSCW), Hot Summer & Warm Winter Zone (HSWW), Mild Zone (M)
D. Thermal Comfort Information	
D1. Thermal Sensation Vote	ASHRAE thermal sensation vote, from −3 (cold) to +3 (hot)
D2. PMV	Predicted mean vote, it was calculated according to ASHRAE 55-2020 [21]

There have been multiple attempts to develop a unified and widely accepted thermal comfort model that can be received and adopted by large audiences. The most popular model is the Predicted Mean Vote (PMV) model, which was constructed by P. O. Fanger [22] and was later adapted to American Society of Heating, Refrigerating, and Air-Conditioning Engineers (ASHRAE) Standard 55 [12]—Thermal Environmental Conditions for Human Occupancy. Many recent works have adapted PMV calculation as a useful feature, showing promising results [23,24]. The PMV value can be directly calculated using a system of highly nonlinear and iterative equations, which were eventually adopted in ASHRAE Standard 55 [12]:

$$PMV = (0.028 + 0.3033e^{-0.036M}) \times L \tag{1}$$

$$\begin{aligned} L = (M - V) - 3.05 \times 10^{-3}(5733 - 6.99(M - V) - Pa) \\ -0.42(M - V - 58.15) - 1.7 \times 10^{-5}(5867 - Pa) \\ -0.0014M(34 - t_a) - f_{cl}h_c(t_{cl} - t_a) - 3.96 \\ \times 10^{-8} f_{cl}[(t_{cl} + 237)^4 - (t_r + 273)^4] \end{aligned} \tag{2}$$

where L defines the overall heat transfer around a single occupant in W/m^2, M is the metabolic rate in W/m^2, W is the work emitted by the occupant in W/m^2, Pa is the water vapor pressure, t_r is the mean radiant temperature in °C, t_a is the air temperature in °C, f_{cl} is the clothing insulation factor defined as the percentage of the total body surface area covered by clothing, I_{cl} is the clothing insulation in CLO, and h_c is the convective-rate heat transfer coefficient.

Table 2 provides a breakdown of the dataset used in this study, categorizing samples by building type and their respective heating, cooling, and naturally ventilated conditions.

This dataset provides a comprehensive basis for analyzing thermal comfort across diverse building environments.

Table 2. Number of samples in the dataset.

Building Type	Cooling	Heating	Naturally Ventilated	Total
Office	6330	6880	1735	14,945
Classroom	1202	1978	1647	4827
Residence	1010	3950	9043	14,003
Total	8542	12,808	12,425	33,775

2.2. Smart City Visualization Platform

Accurate visualization of urban infrastructure is crucial for informed decision making and effective city management. To support this, our objective is to develop a platform that can visualize each component of houses and buildings, along with associated data. Detailed information for house components is seamlessly transferred from Revit files into Datasmith format and stored as a *UASET* file, which Unreal Engine recognizes as *AssetUserData* for the respective objects. We implement ray tracing from the user's camera to the mouse collision point to detect which object the user clicks on and retrieve the corresponding component's ID. Once the component's ID is obtained, it is used to query all relevant data, which are then displayed on a custom-designed widget on the screen. This process is illustrated in Figure 7.

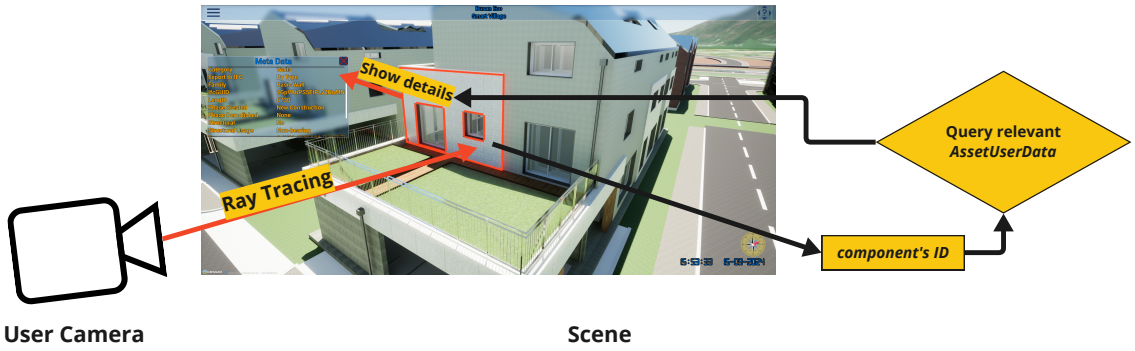


Figure 7. Details of house components shown by clicking (ray tracing is activated).

Visualizing environmental measurements, such as temperature, humidity, and wind, is critical for understanding the dynamic conditions affecting urban environments. By integrating real-time or simulated data, these variables can be represented as heat maps, color gradients, or vector fields within a 3D model. For instance, temperature variations across a city can be displayed using a color-coded overlay, while wind speed and direction can be visualized using animated vectors or particle systems. Humidity levels may be represented through shading or transparency effects, providing a clear, intuitive way for users to assess the environmental conditions of buildings and infrastructure. This visualization not only enhances decision making but also aids in climate analysis and urban planning [25].

In this study, we visualize three key environmental factors: temperature, humidity, and wind. The specific techniques for visualizing each factor are described as follows:

- **Temperature**: Color correction with dynamic color temperature is employed to map environmental temperature values to the corresponding color. As described in Equation (3), adjusting the white balance value directly influences the temperature

color, causing the visual representation of temperature to change accordingly. This allows for a more accurate depiction of environmental conditions by mapping temperature variations to specific color gradients. The color gradient ranges from orange–red (indicating hot temperatures) to blue (indicating cold temperatures), as illustrated in the legend in Figure 8.

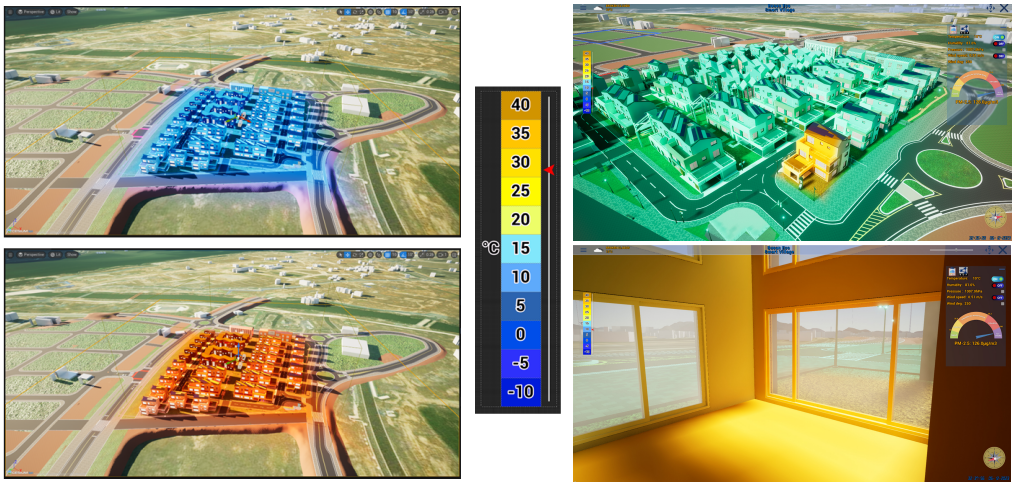


Figure 8. Temperature visualization in the smart city platform.

$$W = -270 \times (T - 40) + 1500 \tag{3}$$

where W is the white balance value and T is the air temperature value.

- **Humidity**: Since this measurement is represented as a scaled percentage value, we visualize humidity by creating a cylinder with a dynamic height corresponding to each humidity value. An example of this representation is shown in Figure 9.

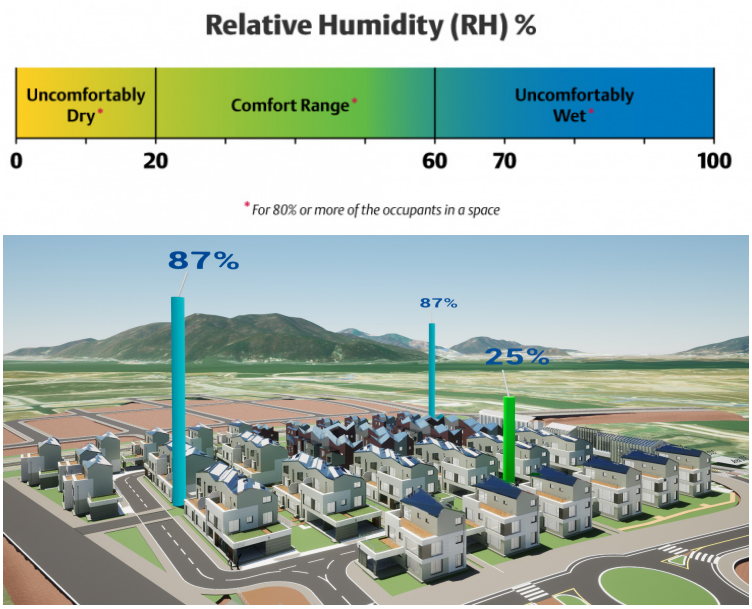


Figure 9. Humidity visualization in the smart city platform.

- **Wind**: The Niagara VFX System is the primary tool for creating visual effects (VFX) in Unreal Engine 5 (UE5) [26]. By utilizing this system, we guide particle movements to follow a predefined spline object, dynamically adjusting the movement speed and angle based on the wind speed and direction. This allows for a realistic representation

of environmental effects, such as wind flow, which enhances the overall visual accuracy and interactivity of the smart city platform as shown in Figure 10.

$$Particle_{speed} = Wind_{speed} \tag{4}$$

$$Particle_{yawDegree} = Wind_{degree} - 180 \tag{5}$$

Meteorological wind direction is defined as the direction from which wind originates. For example, a northerly wind blows from the north to the south. Wind direction is measured in degrees clockwise from due north. Hence, a wind coming from the south has a wind direction of 180 degrees, while one from the east has a direction of 90 degrees.



Figure 10. Wind visualization in the smart city platform.

2.3. Thermal Comfort Model

A mixture of experts (MoE) enables models to be trained with far less computing, which means you can dramatically scale up the model or dataset size with the same compute budget as a dense model. MoE models can help by allowing different experts to specialize in different tasks, improving performance on each. MoE models consist of two main elements:

- Sparse MoE layers are used instead of dense feed-forward network (FFN) layers. The experts are FFNs, which can also be more complex networks or even MoE models or hierarchical MoE models.
- A gate network or router determines which tokens are sent to which experts (there can be more than one expert).

$$y = \sum_{i=1}^{n} G(x)_i E_i(x) \tag{6}$$

$$G(x) = Softmax(x.W_g) \tag{7}$$

Although large batch sizes are usually better for performance, batch sizes in MoE models are effectively reduced as data flow through the active experts. For example, if

a batch consists of 10 tokens, 5 tokens might end in one expert, and the other 5 tokens might end in five different experts, leading to uneven batch sizes and underutilization. Shazeer et al. [27] explored other gating mechanisms, such as noisy top-k gating. This gating approach introduces some (tunable) noise, then keeps the top k values as follows:

$$KeepTopK(v,k)_i = \begin{cases} v_i \text{ if } v_i \text{ in top } k \text{ element of } v \\ -\infty \text{ otherwise} \end{cases} \tag{8}$$

$$G(x) = Softmax(KeepTopK(x,k)) \tag{9}$$

This sparsity introduces some interesting properties. Selecting a sufficiently low k can significantly accelerate both training and inference compared to activating multiple experts. The initial hypothesis was that routing to more than one expert is necessary for the gate to learn how to distribute tasks across different experts effectively. Therefore, at least two experts need to be selected to enable the gate to generalize its routing decisions.

We define the thermal comfort dataset in a time-series format to support future integration into real-time applications, allowing for continuous monitoring and prediction of thermal comfort levels over time. This format enables the analysis of temporal patterns and trends in environmental and individual comfort data. As a result, various well-established time-series models, such as LSTM [28] and liquid neural networks (LNNs) [29] have been considered for development. As mentioned in [29], an LNN is a specialized type of neural network that continues to learn and adapt in real time, not just during the initial training phase. LNNs are specifically designed to process and predict time-series data, making them highly suitable for dynamic environments where data patterns evolve continuously. These models are constructed using a mixture-of-experts (MoE) architecture [27], a neural network design that enhances efficiency and performance by dynamically selecting and activating a subset of specialized networks known as experts for each input. This approach allows the model to allocate computational resources more effectively, focusing on the most relevant experts for specific tasks, which leads to improved accuracy and reduced computational cost in time-series data processing and prediction.

As illustrated in Figure 11, the input data are simultaneously fed into multiple expert models and a gating model. The gating model is responsible for selecting the most appropriate expert(s) for the processing of each input based on the specific characteristics of the data. This process is managed by calculating the softmax probabilities from the gating model's output and retaining the top k expert models with the highest softmax values, as proposed in [30]. This ensures that only the most relevant expert models are activated, optimizing the model's performance and efficiency for each input.

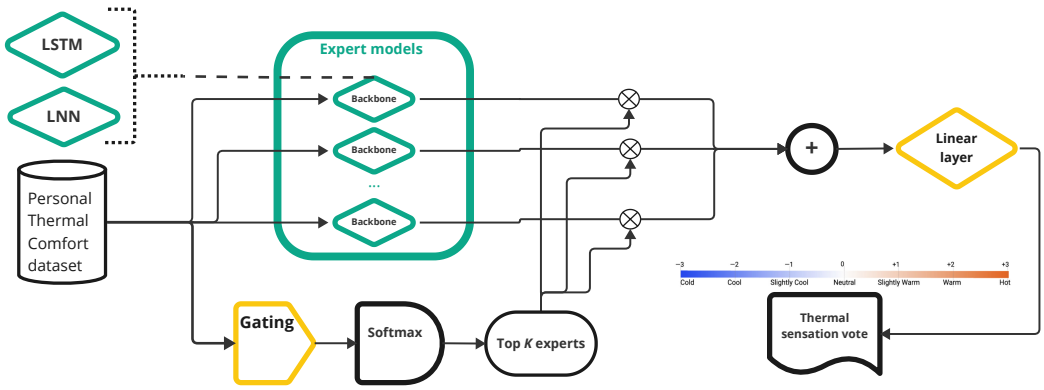


Figure 11. Personal thermal comfort model structure.

2.4. Integration

The thermal comfort model is integrated with the smart city platform using FastAPI, Figure 12, a modern framework designed for building fast and efficient APIs. This connection enables real-time data exchange and interaction between the thermal comfort model and the smart city infrastructure, facilitating seamless updates and automated adjustments based on environmental conditions.

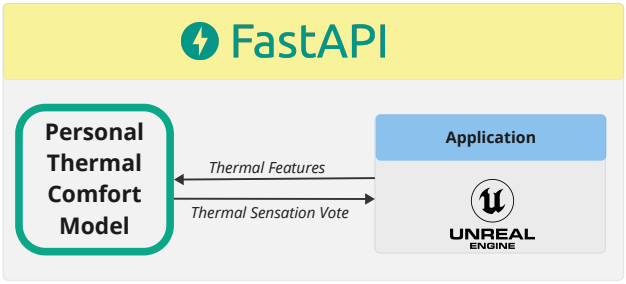


Figure 12. Overview of the connection between the personal thermal comfort model and the smart city platform.

All thermally related data are stored and transferred for processing, they are fed into the thermal comfort model to generate a predicted thermal sensation vote. This predicted value is then sent to and displayed on the smart city platform, providing real-time feedback on comfort levels across the urban environment and individual features.

3. Results and Discussion

3.1. Personal Thermal Comfort Model Evaluation

To evaluate the proposed model and verify its generalization ability, K-fold cross-validation is employed with $k = 5$. The dataset is divided into k equal subsets, i.e., folds. In each iteration, $k - 1$ folds are combined to form the training set, while the remaining fold is used as the validation set. This ensures that each data sample is included in the validation set exactly once and in the training set $k - 1$ times. After completing all iterations, the average error across all k folds is calculated and used to select the optimal model parameters. This approach helps mitigate the bias–variance trade-off by rotating the roles of the training and validation sets, leading to more robust and reliable model performance. The dataset was divided into three subsets for the training, validation, and testing processes using an 8:1:1 ratio. This means 80% of the data was allocated for training, 10% for validation to fine tune model parameters, and the remaining 10% for testing to evaluate the model's performance on unseen data. This approach ensures an effective balance between model development and assessment.

In this section, we evaluate the performance of the proposed model using two key criteria: accuracy and Mean Squared Error (MSE) with mean reduction. Accuracy measures the proportion of correct predictions made by the model, while MSE quantifies the average squared difference between the predicted and actual values, as described in Equation (10).

$$Accuracy = \frac{1}{m}\sum_{0}^{m}\begin{cases}1, & \text{if } abs(\hat{y}_i) = y_i\\ 0, & \text{otherwise}\end{cases} \tag{10}$$

$$MSE = \frac{1}{m}\sum_{i=0}^{m}(y_i - \hat{y}_i)^2 \tag{11}$$

where m represents the total number of samples, $\hat{y}_i$ denotes the predicted thermal sensation value, and y_i is the true thermal sensation vote.

Table 3 presents the training results of mixture of experts (MoE) models for predicting personal thermal comfort using LSTM and LNN. The models were evaluated based on the number of expert models (k), their accuracy, and mean squared error (MSE). For LSTM models, the best performance was achieved with $k = 9$, yielding an accuracy of 0.9403 and the lowest MSE of 0.3278. For LNN models, the optimal configuration was observed with $k = 13$, achieving an accuracy of 0.9387 and the lowest MSE of 0.3218. These results indicate that both LSTM and LNNs perform best when using nine expert models, with LNNs slightly outperforming LSTM in terms of MSE, reflecting their ability to capture complex temporal patterns effectively. However, LSTM marginally outperforms LNNs in accuracy in this configuration.

Table 3. MoE-based personal thermal comfort training results. Higher is better for Accuracy while lower is better for MSE.

k (Number of Used Expert Models)	Training Time (h)	Accuracy ↑	MSE ↓
LSTM			
1	0.31	0.9400	0.3398
5	0.42	0.9370	0.3315
9	0.51	**0.9403**	**0.3278**
13	0.64	0.9201	0.3405
17	0.89	0.9145	0.3511
21	1.2	0.9044	0.3689
LNN			
1	0.45	0.9154	0.4652
5	0.58	0.9215	0.3513
9	1.21	0.9269	0.3249
13	1.42	**0.9387**	**0.3218**
17	1.51	0.9251	0.3409
21	1.85	0.9163	0.4783

Table 4 highlights the performance of MoE-based models using various combinations of LSTM and LNN backbones. The configurations were evaluated based on accuracy and MSE. The best-performing combination was 13 LSTM-9 LNN, achieving the highest accuracy of 0.9613 and the lowest MSE of 0.3156. This indicates that this combination strikes an optimal balance between the strengths of both LSTM and LNNs, effectively leveraging their complementary features for thermal comfort prediction. Other configurations, such as 9 LSTM-13 LNN, also showed competitive performance, with an accuracy of 0.9517 and an MSE of 0.3218, suggesting that increasing the proportion of LSTM backbones generally enhances model performance. These results demonstrate the effectiveness of combining LSTM and LNN backbones to improve prediction accuracy and reduce errors in the MoE framework. The training time for both the LSTM and LNN models increases as the number of expert models (k) rises, with LSTM requiring slightly less time compared to LNNs for the same k. For example, at $k = 21$, the LSTM training time is 1.2 h, whereas LNNs take 1.85 h. When combining backbones, the training time is longer overall, reaching 1.93 h for the 1 LSTM-21 LNN combination. However, the combined models demonstrate a trade-off between training time and improved performance, as seen in the 13 LSTM-9 LNN model, which achieves the best accuracy (0.9613) and lowest MSE (0.3156) with a training time of 1.76 h. Thus, while training time increases with model complexity, the combined approach balances performance and time efficiency effectively.

Table 4. Performance of MoE-based models with different numbers of backbone combinations.

k (Combined Backbones)	Training Time (h)	Accuracy ↑	MSE ↓
1 LSTM-21 LNN	1.93	0.9367	0.3402
5 LSTM-17 LNN	1.88	0.9478	0.3265
9 LSTM-13 LNN	1.85	0.9517	0.3218
13 LSTM-9 LNN	1.76	**0.9613**	**0.3156**
17 LSTM-5 LNN	1.64	0.9483	0.3247
21 LSTM-1 LNN	1.52	0.9512	0.3226

3.2. Smart City Platform Performance

For interactive computer graphics applications, a suitable frame rate typically ranges from 30 to 60 FPS, which ensures smooth performance and responsiveness for general use cases like user interfaces and dashboards [31]. The hardware setup included high-performance machines equipped with NVIDIA RTX A4000 GPUs, 32GB of memory, and 500GB SSD storage to manage computationally intensive tasks such as rendering and machine learning model inference. The platform was built using Python for data processing, Unreal Engine for 3D visualization, Blueprint programming for the creation of functions, and Pytorch for deep learning model integration. Data from IoT sensors, such as temperature and humidity sensors, were transmitted via HTTP protocol, ensuring low-latency communication. Geographic data integration was achieved using CesiumJS and CityGML, allowing for precise mapping of urban infrastructure. With our proposed method, the platform maintains a frame rate of approximately 40 frames per second (fps), representing an improvement in performance for 3D smart city platforms compared to other studies [32–34]. This enhanced frame rate ensures smoother and more responsive visualization, contributing to a better user experience and more effective real-time data analysis. As shown in Figure 13, the platform allows users to easily navigate inside a house and query detailed information related to its components. Additionally, the material of each component is preserved and visualized, providing a clearer understanding of the house's structure and construction, further enhancing the user's interaction and comprehension of the built environment.

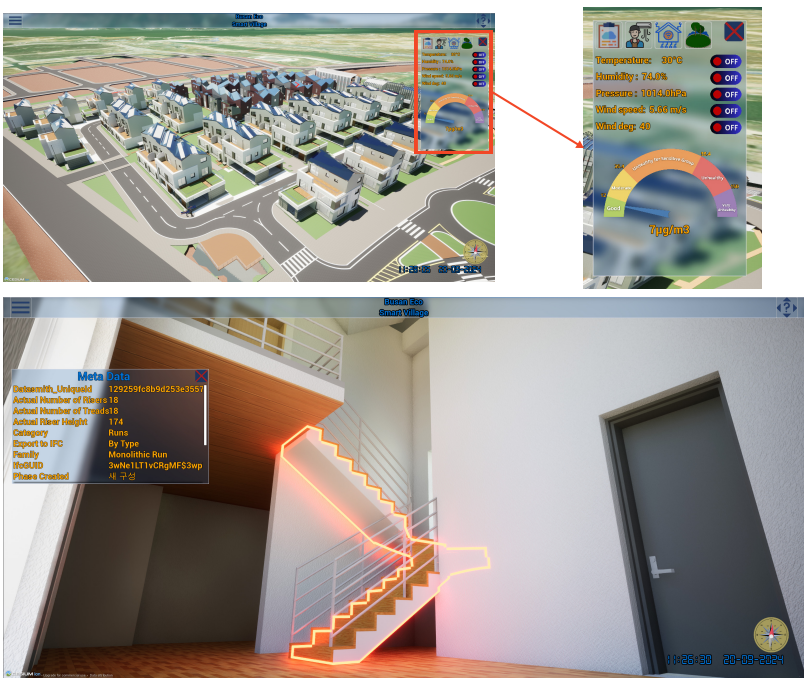


Figure 13. Overview of the smart city platform dashboard. Korean word in the Meta Data board means "new configuration" in English.

Thermal comfort in the platform is also displayed as a board for each character. This board provides the detailed information required to compute the thermal sensation vote for each character, including factors such as activity level, clothing, weight, and height. The thermal sensation vote is predicted by the thermal comfort model and transferred to the platform via FastAPI, where it is visualized on the board. This feature offers city managers valuable insights into individual thermal comfort levels, enabling more informed decisions regarding HVAC system design to optimize both comfort and energy efficiency. Some examples are illustrated in Figure 14.



Figure 14. Thermal comfort information board.

4. Conclusions

This study presents an end-to-end approach to building a smart city platform. With the proposed method, the rendering time and frame rate are optimized, maintaining performance at a level suitable for scaling into the production stage. Additionally, detailed information about each component is transferred to the platform, offering users a more comprehensive view of the smart city. Furthermore, weather data are integrated and visualized in multiple ways, enabling city managers to monitor environmental conditions and efficiently plan city infrastructure, enhancing decision making and urban management.

A comprehensive approach is proposed to integrate a thermal comfort prediction model into a smart city platform, enabling real-time analysis and visualization of environmental factors such as temperature, humidity, and wind. By utilizing a mixture-of-experts architecture, the system dynamically selects the most relevant models for data processing, enhancing both efficiency and prediction accuracy. The thermal comfort model is built on time-series data and incorporates individual-specific features, as well as environmental measurements, making it highly adaptable for real-time applications in smart cities.

The system is integrated with the smart city platform through FastAPI, allowing for seamless data transmission, processing, and the display of thermal sensation predictions. This integration ensures that city managers have access to detailed, real-time insights into thermal comfort levels across urban areas, which can inform decision making related to energy efficiency, urban planning, and building management.

This research highlights the potential for integrating advanced machine learning models into urban management systems, offering a scalable solution for monitoring and

predicting thermal comfort in dynamic environments. Future work will focus on enhancing model precision; expanding the range of environmental and individual parameters; and exploring the broader applicability of the system in areas such as energy management, climate resilience, and sustainable city development. Additionally, further exploration of hybrid models and real-time learning techniques, such as liquid neural networks, may offer avenues for improving the adaptability of the system in complex and evolving urban contexts.

Author Contributions: Conceptualization, H.-K.L.; methodology, H.-K.L.; software, P.-D.L. and H.-K.L.; validation, P.-D.L.; formal analysis, P.-D.L. and H.-K.L.; investigation, S.-Y.O. and S.-H.L.; resources, S.-Y.O. and S.-H.L.; writing—review and editing, H.-K.L.; visualization, P.-D.L. and H.-K.L.; supervision, S.-Y.O. and S.-H.L.; project administration, S.-Y.O. and S.-H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIP) (No.2022-0-00622, Digital Twin Testbed Establishment).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Arowoiya, V.A.; Moehler, R.C.; Fang, Y. Digital twin technology for thermal comfort and energy efficiency in buildings: A state-of-the-art and future directions. *Energy Built Environ.* **2024**, *5*, 641–656. [CrossRef]
2. Lafioune, N.; St-Jacques, M. Towards the creation of a searchable 3D smart city model. *Innov. Manag. Rev.* **2020**, *17*, 285–305. [CrossRef]
3. Lam, P.D.; Gu, B.H.; Lam, H.K.; Ok, S.Y.; Lee, S.H. Digital Twin Smart City: Integrating IFC and CityGML with Semantic Graph for Advanced 3D City Model Visualization. *Sensors* **2024**, *24*, 3761. [CrossRef]
4. Info Views—Cities: Skylines Wiki. Available online: https://skylines.paradoxwikis.com/Info_views (accessed on 9 September 2024).
5. Wang, H.; Chen, X.; Jia, F.; Cheng, X. Digital twin-supported smart city: Status, challenges and future research directions. *Expert Syst. Appl.* **2023**, *217*, 119531. [CrossRef]
6. Kim, H.C.; Lam, H.K.; Lee, S.H.; Ok, S.Y. Early Fire Detection System by Using Automatic Synthetic Dataset Generation Model Based on Digital Twins. *Appl. Sci.* **2024**, *14*, 1801. [CrossRef]
7. Ssin, S.; Cho, H.; Woo, W. GeoVCM: Virtual urban digital twin system augmenting virtual and real geo-spacial data. In Proceedings of the 2021 IEEE International Conference on Consumer Electronics (ICCE), Taiwan, China, 16–18 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–5.
8. Evangelou, T.; Gkeli, M.; Potsiou, C. Building digital twins for smart cities: A case study in Greece. *Isprs Ann. Photogramm. Remote. Sens. Spat. Inf. Sci.* **2022**, *10*, 61–68. [CrossRef]
9. Zhang, X.; Han, D.; Zhang, X.; Fang, L. Design and application of intelligent transportation multi-source data collaboration framework based on digital twins. *Appl. Sci.* **2023**, *13*, 1923. [CrossRef]
10. Song, C.; Liu, Y.; Liu, J. The sleeping thermal comfort model based on local thermal requirements in winter. *Energy Build.* **2018**, *173*, 163–175. [CrossRef]
11. Aguilera, J.J.; Kazanci, O.B.; Toftum, J. Thermal adaptation in occupant-driven HVAC control. *J. Build. Eng.* **2019**, *25*, 100846. [CrossRef]
12. *ASHRAE Standard 55*; Thermal Environmental Conditions for Human Occupancy. ASHRAE: Peachtree Corners, GA, USA, 2019.
13. Tekler, Z.D.; Lei, Y.; Peng, Y.; Miller, C.; Chong, A. A hybrid active learning framework for personal thermal comfort models. *Build. Environ.* **2023**, *234*, 110148. [CrossRef]
14. Lala, B.; Rizk, H.; Kala, S.M.; Hagishima, A. Multi-task learning for concurrent prediction of thermal comfort, sensation and preference in winters. *Buildings* **2022**, *12*, 750. [CrossRef]
15. Ličina, V.F.; Cheung, T.; Zhang, H.; De Dear, R.; Parkinson, T.; Arens, E.; Chun, C.; Schiavon, S.; Luo, M.; Brager, G.; et al. Development of the ASHRAE global thermal comfort database II. *Build. Environ.* **2018**, *142*, 502–512. [CrossRef]

16. Zhong, T.; Lin, J.; Wang, H.; He, J.; Zhong, L.; Chen, L. An Initial Exploration of Digital Twin Technology for Hydraulic Structures Based on Cesium. In Proceedings of the 2024 Guangdong-Hong Kong-Macao Greater Bay Area International Conference on Digital Economy and Artificial Intelligence, Dongguan, China, 19–21 January 2024; pp. 762–771.
17. Lin, Z.; Xu, H. Virtual surface mining based on cesium. In Proceedings of the Ninth International Symposium on Energy Science and Chemical Engineering (ISESCE 2024), Nanjing, China, 22–24 March 2024; SPIE: St Bellingham, WA, USA, 2024; Volume 13172, pp. 88–94.
18. Cho, J.; Kim, C.; Lim, K.J.; Kim, J.; Ji, B.; Yeon, J. Web-based agricultural infrastructure digital twin system integrated with GIS and BIM concepts. *Comput. Electron. Agric.* **2023**, *215*, 108441. [CrossRef]
19. Using a Geospatially Accurate Sun, 2024. Available online: https://cesium.com/learn/unreal/unreal-geospatially-accurate-sun/ (accessed on 19 September 2024).
20. Fard, Z.Q.; Zomorodian, Z.S.; Korsavi, S.S. Application of machine learning in thermal comfort studies: A review of methods, performance and challenges. *Energy Build.* **2022**, *256*, 111771. [CrossRef]
21. Standard, ASHRAE. *Thermal Environmental Conditions for Human Occupancy*; The American Society of Heating, Refrigerating and Air-Conditioning Engineers: Atlanta, GA, USA, 2020; Volume 5.
22. Fanger, P.O. *Thermal comfort. Analysis and Applications in Environmental Engineering*; Danish Technical Press: Raleigh, NC, USA, 1970.
23. Alam, N.; Zaki, S.A.; Ahmad, S.A.; Singh, M.K.; Azizan, A. Machine Learning Approach for Predicting Personal Thermal Comfort in Air Conditioning Offices in Malaysia. *Build. Environ.* **2024**, *266*, 112083. [CrossRef]
24. Shi, Z.; Zheng, R.; Zhao, J.; Shen, R.; Gu, L.; Liu, Y.; Wu, J.; Wang, G. Towards various occupants with different thermal comfort requirements: A deep reinforcement learning approach combined with a dynamic PMV model for HVAC control in buildings. *Energy Convers. Manag.* **2024**, *320*, 118995. [CrossRef]
25. Bouloukakis, M.; Partarakis, N.; Drossis, I.; Kalaitzakis, M.; Stephanidis, C. Virtual reality for smart city visualization and monitoring. In *Mediterranean Cities and Island Communities: Smart, Sustainable, Inclusive and Resilient*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 1–18.
26. Creating Visual Effects in Niagara For Unreal Engine | Unreal Engine 5.4 Documentation. Available online: https://dev.epicgames.com/documentation/en-us/unreal-engine/creating-visual-effects-in-niagara-for-unreal-engine (accessed on 17 September 2024).
27. Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv* **2017**, arXiv:1701.06538.
28. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
29. Hasani, R.; Lechner, M.; Amini, A.; Rus, D.; Grosu, R. Liquid time-constant networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 7657–7666.
30. Chen, Z.; Deng, Y.; Wu, Y.; Gu, Q.; Li, Y. Towards understanding mixture of experts in deep learning. *arXiv* **2022**, arXiv:2208.02813.
31. Andreev, D. Real-time frame rate up-conversion for video games: Or how to get from 30 to 60 fps for "free". In *ACM SIGGRAPH 2010 Talks*; Association for Computing Machinery: New York, NY, USA, 2010; p. 1.
32. Nesi, P.; Po, L.; Viqueira, J.R.; Trillo-Lado, R. An integrated smart city platform. In Proceedings of the Semantic Keyword-Based Search on Structured Data Sources: Third International KEYSTONE Conference, IKC 2017, Gdańsk, Poland, 11–12 September 2017; Revised Selected Papers and COST Action IC1302 Reports 3; Springer: Berlin/Heidelberg, Germany, 2018; pp. 171–176.
33. Dobrilović, D.; Malić, M.; Malić, D. Learning platform for smart city application development. *Interdiscip. Descr. Complex Syst. INDECS* **2019**, *17*, 430–437. [CrossRef]
34. Kikuchi, N.; Fukuda, T.; Yabuki, N. Future landscape visualization using a city digital twin: Integration of augmented reality and drones with implementation of 3D model-based occlusion handling. *J. Comput. Des. Eng.* **2022**, *9*, 837–856. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.