Computers & Graphics 124 (2024) 104091

Contents lists available at ScienceDirect

Computers & Graphics

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

Special Section on IMET 2023

Executing realistic earthquake simulations in unreal engine with material calibration

Yitong Sun [a], Hanchun Wang [b], Zhejun Zhang [a], Cyriel Diels [a], Ali Asadipour [a,*]

[a] Royal College of Art, London, SW11 4NL, UK
[b] Imperial College London, London, SW7 2BX, UK

ARTICLE INFO

Keywords:
Earthquake
Simulation
Unreal engine
Synthetic data generation
Urban disaster modeling

ABSTRACT

Earthquakes significantly impact societies and economies, underscoring the need for effective search and rescue strategies. As AI and robotics increasingly support these efforts, the demand for high-fidelity, real-time simulation environments for training has become pressing. Earthquake simulation can be considered as a complex system. Traditional simulation methods, which primarily focus on computing intricate factors for single buildings or simplified architectural agglomerations, often fall short in providing realistic visuals and real-time structural damage assessments for urban environments. To address this deficiency, we introduce a real-time, high visual fidelity earthquake simulation platform based on the Chaos Physics System in Unreal Engine, specifically designed to simulate the damage to urban buildings. Initially, we use a genetic algorithm to calibrate material simulation parameters from Ansys into the Unreal Engine's fracture system, based on real-world test standards. This alignment ensures the similarity of results between the two systems while achieving real-time capabilities. Additionally, by integrating real earthquake waveform data, we improve the simulation's authenticity, ensuring it accurately reflects historical events. All functionalities are integrated into a visual user interface, enabling zero-code operation, which facilitates testing and further development by cross-disciplinary users. We verify the platform's effectiveness through three AI-based tasks: similarity detection, path planning, and image segmentation. This paper builds upon the preliminary earthquake simulation study we presented at IMET 2023, with significant enhancements, including improvements to the material calibration workflow and the method for binding building foundations.

1. Introduction

Earthquakes are frequent natural disasters that significantly affect human life and economic activities [1,2]. To mitigate their adverse effects and improve the effectiveness of post-earthquake rescue operations, a paradigm shift toward integrating artificial intelligence (AI) and robotics has been observed [3]. These technologies, which include tasks such as path planning, automatic obstacle avoidance, and image recognition, significantly enhance the efficiency of responses to earthquakes [4]. Iterative training of AI in simulation environments has been widely recognized as an effective method [5]. However, traditional earthquake simulation systems often focus on incorporating complex multiple factors in their calculations, which fails to meet the real-time responsiveness required for a variety of AI models and the high fidelity needed for computer vision recognition. Consequently, there is an urgent need to create a user-friendly simulation platform capable of generating countless realistic earthquake scenarios for iterative training of AI models.

While game engines have become powerful tools for simulating various disaster scenarios, such as firestorms or flash floods [6], their potential for earthquake simulations has yet to be fully explored. Earthquake simulations are regarded as inherently complex systems, particularly in terms of material simulation. The diversity of building materials and structures presents significant challenges. The primary optimization goal of current game engines is to ensure real-time performance, which has led to parameter simplification. This results in a conflict with scientific applications that involve intensive calculations of multiple parameters. Consequently, these engines cannot be directly used for scientific simulations without modifications or calibrations. Despite this, compared to traditional simulation methods, game engines not only offer simplified operations and efficiency but also provide richer plugins and reusable virtual assets. Additionally, the adoption of advanced realistic rendering techniques, including ray tracing, is beginning to replace the need for real-world data, revolutionizing

* Corresponding author.
E-mail addresses: yitong.sun@network.rca.ac.uk (Y. Sun), stephenx.wang@mail.utoronto.ca (H. Wang), ezrealzhang1030@gmail.com (Z. Zhang), cyriel.diels@rca.ac.uk (C. Diels), ali.asadipour@rca.ac.uk (A. Asadipour).

https://doi.org/10.1016/j.cag.2024.104091
Received 19 May 2024; Received in revised form 17 August 2024; Accepted 20 September 2024
Available online 24 September 2024

research in visual recognition [7]. However, the application of these sophisticated techniques and tools specifically for earthquake simulations within virtual environments remains an underdeveloped area in current research.

To address this gap, we introduce RESenv — A Realistic Earthquake Simulation Environment, utilizing the Chaos physics system within Unreal Engine 5 (UE5). The RESenv workflow begins by simulating ideal building material parameters using Ansys Explicit Dynamics for fracture simulations. Subsequently, the results from Ansys are aligned with the parameters of the UE Destruction system through a genetic algorithm, creating a material library. This parameter alignment step not only brings UE's material fracture results comparable with Ansys but also does not increase the computational load, laying a foundation for executing real-time high-fidelity simulations. Furthermore, RESenv employs UE's Physics Constraint Actor (UEPCA) to automatically bind building foundations, accurately transmitting terrain vibrations and stresses to the structures, thus mirroring real earthquake phenomena. In practical tests, we introduce a random pre-fracture parameter in UE to simulate various outcomes, thereby covering as many potentialities of building damage as possible. Additionally, RESenv utilizes a user interface (UI) plugin to retrieve real earthquake waveform data from online databases and applies it to the virtual terrain in UE to accurately reproduce the terrain movements observed in actual earthquake events. The aim of RESenv is to leverage UE's high-performance, real-time simulation capabilities to closely emulate the destruction caused by real earthquakes. The overarching goal is to provide a training platform for AI, VR, and robotics, offering real-time interactive, high-resolution, and high-fidelity earthquake scenario simulations. This not only aids in search and rescue mission planning but also serves as a valuable synthetic data reservoir for AI training in various applications, such as path planning and visual recognition.

This paper is an extension of our conference paper, "RESenv: A realistic earthquake simulation environment based on Unreal Engine" [8]. Significant improvements include a detailed discussion of the material calibration process integrated with Ansys Explicit Dynamics, and enhancements to the method of binding building foundations in UE.

Our key contributions in this study include:

1. **Enhanced Simulation Accuracy**: Our RESenv environment utilizes a designed genetic algorithm to align Ansys material simulation results with the UE Fracture system, substantially enhancing the realism and accuracy of earthquake simulations executed by UE.
2. **Real Earthquake Data Integration**: Our RESenv binds real-time earthquake data to UE's virtual terrain and automates the physical binding of building foundations, accurately replicating the terrain movements and stress transmission observed in actual earthquakes.
3. **Simplified and Automated Workflow**: The visualization and automation features of RESenv significantly reduce the complexity of conducting simulations. Empirical validations have demonstrated RESenv's effectiveness in mimicking architectural earthquake damage, as well as in training AI for visual recognition and path planning tasks.

2. Related work

This section reviews research related to earthquake simulation and AI training for rescue missions, which form the basis for our proposed UE-based earthquake simulation approach.

2.1. Earthquake simulation

Earthquake simulation, a longstanding research focus in geophysics, geology, and engineering [9], has seen significant advancements due to recent progress in computer hardware. This has enabled more sophisticated modeling of earthquakes and consequent building damage using numerical simulation techniques [10,11]. Stress simulations of individual buildings, initially aimed at analyzing earthquake stress-induced deformation and structural optimization, have matured, and some researchers have employed finite element analysis (FEA) for assessing earthquake-induced building damage and exploring risk mitigation strategies [12,13]. However, due to real-world buildings' structural complexity, material diversity, and computational constraints, most simulations only model primary load-bearing structures and facades, resulting in discrepancies between simulated and actual outcomes. Current earthquake platforms, constrained by the complexity of the physics engine limits and simulating only single or two degree-of-freedom (DOF) vibrations, fail to mimic the three DOF motions of actual earthquakes [14,15].

Urban multi-building simulations, compared to individual building simulations, emerged much later. One of the most widely used frameworks is HAZUS, developed by the United States Federal Emergency Management Agency (FEMA) [16]. Based on standardized Geographic Information System (GIS) methodologies, HAZUS is employed for estimating the impact of earthquakes, post-earthquake fires, floods, and hurricanes, among other disasters. To overcome the limitations of HAZUS's single DOF model, Japanese researchers introduced the Integrated Earthquake Simulation (IES) framework, utilizing multi-dimensional data fusion calculation methods [17]. Subsequently, Turkish researchers developed a regional building simulation method for Istanbul using MATLAB, based on the IES framework [18]. Similar to individual building simulations, multi-building simulations are also constrained by software limitations in terms of physical collisions and building structural complexity. Although a study by David et al. employed large-scale computing to simulate the motion of geological faults and measure building responses [19], the focus of this research predominantly lies in calculating the complexity of geological structures, with scant attention paid to the fidelity of building structures.

Ansys has been proven to offer high precision and reliability in material benchmark testing and dynamic earthquake simulations [20]. Ansys Explicit Dynamics is specifically designed to handle complex, highly nonlinear problems, essential for accurate fracture simulations that require analysis of rapid impact interactions within materials, effectively simulating the impacts experienced by building structures during earthquakes [21]. Additionally, it efficiently calculates material responses under short-duration, severe loading conditions encountered during earthquakes, making it highly suitable for simulating earthquake-induced material damage [22]. Furthermore, its prebuilt comprehensive material library allows for detailed testing of diverse building materials, significantly easing the workload for cross-disciplinary developers.

In our approach, we utilize the Chaos destruction system [23] and Nanite visualization system [24] within UE5 game engine. This integration facilitates previously challenging fracture and fragmentation simulations for various materials and complex hybrid structures, enables accurate physical collisions, and supports three degrees of freedom (DOF) in geological motion. To further enhance the accuracy and realism of these simulations, we have incorporated the RESenv Material Calibration Bridge, which seamlessly integrates Ansys Explicit Dynamics simulation results into the UE Destruction system. This bridge allows for the precise calibration of material properties based on real-world data, significantly enhancing the fidelity of fracture simulations within UE5. Our method demonstrates a significant improvement in computational efficiency and cost compared to conventional techniques, enabling real-time and accelerated calculations on consumer-grade computers.

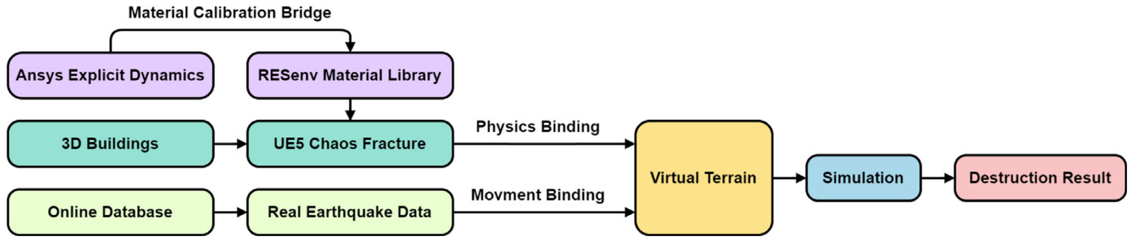

Fig. 1. Flowchart of RESenv for earthquake simulation, encompassing four stages: *Material calibration*, *Scenario preparation*, *Data binding*, and *Simulation*. *Material calibration*: The RESenv Material Calibration Bridge utilizes data simulated in Ansys Explicit Dynamics as a benchmark to optimize the UE Fracture settings. The calibrated materials are stored in the RESenv material library. *Scenario preparation*: 3D building models are imported into Unreal Engine and assigned fracture settings from the RESenv material library. Actual earthquake wave data is acquired from the IRIS online database and imported through a graphical user interface system. *Data binding*: 3D buildings are bound to the virtual terrain using UEPCA via an automated analysis program. The earthquake wave data is converted into terrain displacements for binding. *Simulation*: RESenv operates at specified frame rates: 40 FPS for desktop computer simulation, 90 FPS for VR training, and 240 FPS for high frame rate sensor training. As the simulation commences, the earthquake wave data displaces the terrain, which in turn causes the pre-fractured 3D buildings to be destroyed. RESenv remains interactive throughout the simulation.

2.2. AI training for rescue missions

AI applications in search and rescue operations, as well as the robotics domain, are becoming increasingly widespread. Numerous researchers are dedicated to employing deep learning and reinforcement learning techniques for complex terrain path planning, image recognition, and other related tasks [25,26]. For instance, the study by LinLin et al. utilized the SBMPC algorithm to investigate path planning problems for search and rescue robots [27]. Xuexi et al. explored indoor search and rescue using Simultaneous Localization and Mapping (SLAM) and Light Detection and Ranging (LiDAR) methods [28]. Farzad's research introduced the application of deep reinforcement learning (DRL) methods in search and rescue robot tasks [29]. These studies underscore the potential of artificial intelligence in enhancing the effectiveness and efficiency of search and rescue missions.

The success of AI approaches largely depends on the availability of ample and high-quality data as training inputs, which can accurately represent the complexities and dynamics of environments affected by earthquakes [30]. However, in earthquake rescue scenarios, collecting and obtaining real-world data poses significant challenges. To address data limitations, researchers have developed various virtual environments, such as the RoboCup Rescue Simulation Environment [31], USARSim [32], and the BCB environment developed by Laurea University of Applied Sciences [33], for creating training data for deep learning and reinforcement learning algorithms in search and rescue operations. Nonetheless, these frameworks currently lack the level of texture rendering realism and detail richness required for training AI models that rely on image recognition and depth data inputs. This also results in substantial discrepancies between the volume and complexity of simulated scenarios and actual search and rescue missions.

Our proposed simulation environment fills this void by aiming to provide a highly realistic and detailed virtual earthquake damage environment using a ray-tracing system and authentic scanned textures. The environment allows for generating high-quality training data that can be directly used for AI algorithm visual recognition and depth data scanning. Weather phenomena, lighting conditions, and post-earthquake dust will be effectively simulated.

3. Method

RESenv executes the earthquake simulation in a four-stage process: **material calibration**, **data preparation**, **data binding**, and **simulation**. Fig. 1 illustrates the diagrammatic representation of the aforementioned stages.

3.1. Material calibration

The Issue of UE Chaos Destruction System The UE Destruction system utilizes the latest Chaos physics engine, enabling precise pre-fracturing of geometries to define their destruction behavior during run-time simulations [23]. This pre-fracturing facilitates the pre-generation of fragment objects complete with collision boundaries and cluster connections, thereby enabling real-time interactions with collisions and physical constraints. Compared to traditional physical simulation methods, which demand substantial computational resources, the UE Destruction system offers a real-time interactable platform foundation for simulations.

However, there are two main limitations in the current UE Destruction system that hinder accurate physical simulations. Firstly, the pre-fracturing feature only allows designers to visually fracture geometries into fragment groups using predefined graphic patterns, rather than employing calculations based on actual material properties [34]. This approach results in deformations and fracture shapes during collisions that do not accurately represent real-world phenomena. Secondly, the stress values recorded in the cluster links between fragment groups are not expressed in standard physical units, thereby precluding calculations based on physical formulas. Consequently, for applications like earthquake simulations, where precise physical calculations are crucial to accurately mimic real-world dynamics, the UE Destruction System cannot be applied directly without calibrations.

To address the aforementioned limitations and achieve realistic earthquake fracture simulations in UE, we propose a material calibration workflow for the UE Destruction system. This workflow initially employs the Ansys Explicit Dynamics system to conduct physical simulations on various calibrated building materials, strictly adhering to real-world testing standards. The outcomes of these simulations are then exported to serve as calibration benchmark data for UE Fracture. Subsequently, We have specifically designed a genetic algorithm (developed in C++ as a UE plugin) that quantifies the differences between the simulation results of Ansys and UE, and automatically iterates and optimizes the fracture parameters of materials in UE. This process ensures that the material fracture behavior approximates the results of Ansys simulations without altering the UE physics engine. Ultimately, the calibrated material parameters are stored in the RESenv material library for future application to building geometry assets for simulation.

Fig. 2 demonstrates an example of the calibration of two construction materials using the RESenv Material Calibration Workflow, and compares the destruction results with the default parameters of UE Fracture. The left material, a 35 MPa concrete is subjected to a fracture test in Ansys using the ASTM C293 standard [35], followed by iterative optimization via the calibration bridge. On the right side, a masonry assemblage (brick wall) is optimized using the workflow with the ASTM E519 standard [36]. Compared to the default parameters of UE Fracture, which nearly fail to accurately represent the physical material's fracture behavior, the geometric assets calibrated through the RESenv workflow achieved comparable results in UE fracture simulation to those of Ansys simulations, particularly in terms of crack patterns and stress distribution.

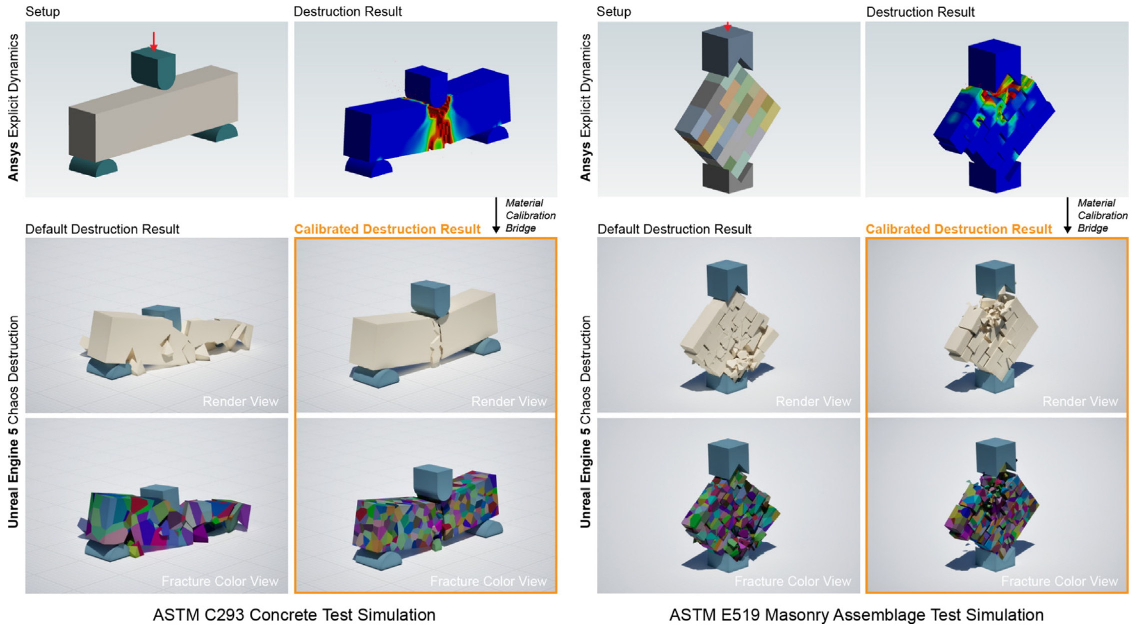

Fig. 2. Comparative Visualization of Material Fracture Simulations between Default UE Fracture Settings and RESenv Material Calibration. The image showcases two sets of material tests. On the left, a concrete fracture test based on ASTM C293; on the right, a masonry assemblage fracture test based on ASTM E519. For each material, the top panels display the setup and results visualized using Ansys Explicit Dynamics, tailored to specific test standards and material properties. The bottom left panels show the results of the simulations using default UE Fracture settings, while the bottom right panels illustrate the outcomes after iterative optimization via the RESenv Material Calibration Bridge. Each pair demonstrates a significant enhancement in the fidelity of destruction simulations, particularly in crack patterns and stress distribution, through calibrated fracture settings compared to the default.

Ansys Simulation The RESenv workflow is implemented in Ansys Workbench 2024 R1, utilizing the Explicit Dynamics simulation environment. The workflow in Ansys includes setting material properties according to different real-world testing standards, importing test geometries, defining simulation conditions, and exporting the results to the Material Calibration Bridge.

Materials can be prepared through three channels based on different testing standards, facilitating rapid development for developers:

1. Importing from the built-in Explicit Material library.
2. Importing from the Ansys GRANTA Material Database.
3. Creating custom materials based on tested parameters.

In the example shown in Fig. 2, the 35 MPa concrete is sourced from the Explicit Material library [37], while the masonry assemblage composite material is set based on research by Parker et al. [38]. The materials used for the support and pressure bodies are the default Structural Steel.

Geometries are imported or created using Ansys modeling tools according to specific dimensions and test requirements of the standards. Once placed in the model, the prepared materials are assigned to geometries and Ansys mesh segmentation is executed. Subsequent steps include defining initial simulation conditions according to standards, such as constant support, and the velocity and direction of the pressure body, before commencing the simulation.

After the simulation concludes, the Ansys Python Result tool is used to export the solved simulation parameters from the solution worksheet to the RESenv Material Calibration Bridge, enabling subsequent optimization and calibration of UE Fracture settings for materials.

RESenv Material Calibration Bridge The RESenv Material Calibration Bridge aims to iteratively optimize the UE Fracture settings for different materials, using Ansys simulation parameters as a benchmark. After synchronizing the common parameters in UE with Ansys test conditions (including gravity, friction, damping, mass, inertia tensor, elasticity, constraints, collision detection, initial velocity, and test duration), the optimization focuses on two key areas: 1. the morphology and density of the fractured segment groups within the UE geometries; 2. the initial link stress values among the fractured segments.

The calibration workflow can be mathematically formalized as a non-convex optimization problem. That is we would like to find the optimal value of damage threshold σ and fracturing numbers N for each material. When using UE's default random Voronoi tessellation, we randomly a set of points $\{x_i\}$ in the interior of the object, and denote their position under Ansys simulation $\phi_{\text{ansys}}(t, x_i)$ as ground truth. We want to optimize the UE fracture simulation $\phi_{\text{UE}}(t, x_i; N, \sigma)$ on the following metric:

$$F(\sigma, N) = -\sum_i \left|\phi_{\text{UE}}\left(T, x_i; \sigma, N\right) - \phi_{\text{ansys}}\left(T, x_i\right)\right|^2 \tag{1}$$

To optimize the parameters of the UE fracture simulation, we employ a genetic algorithm that integrates Rank Selection and Two-Point Crossover with a Fitness Function and Mutation strategy. The Fitness Function, $F(\sigma, N)$, is designed to quantify the accuracy of the UE simulation compared to the Ansys benchmark. Specifically, $F(\sigma, N)$ is defined as the negative sum of squared errors between the UE simulation results $\phi_{\text{UE}}(T, x_i; \sigma, N)$ and the corresponding Ansys results $\phi_{\text{ansys}}(T, x_i)$ over a set of points $\{x_i\}$. The objective is to maximize $F(\sigma, N)$, which is equivalent to minimizing the discrepancy between the two simulations.

We utilize Rank Selection to select individuals from the population based on their fitness. In this method, individuals are ranked according to their fitness scores, and selection probabilities are assigned proportionally to their rank, not directly to their fitness values. This approach mitigates the issue of premature convergence by maintaining population diversity, especially when fitness differences are minimal. The selection probability for an individual with rank r is given by $P(r) = \frac{2\times(N_{\text{pop}}-r+1)}{N_{\text{pop}}\times(N_{\text{pop}}+1)}$, where N_{pop} denotes the population size.

Once individuals are selected, Two-Point Crossover is applied to generate new offspring. In this crossover strategy, two crossover points p_1 and p_2 are randomly selected along the gene sequences of the parent individuals $\{\sigma^1, N^1\}$ and $\{\sigma^2, N^2\}$. The segments of the genes between these points are exchanged to produce a new offspring $\{\sigma^{\text{child}}, N^{\text{child}}\}$, thereby allowing a robust exploration of the solution space. For a more general case where σ and N are vectors of parameters, the weighted combination can be extended as:

$$\sigma_i^{\text{child}} = w_i \cdot \sigma_i^1 + \left(1 - w_i\right) \cdot \sigma_i^2, \quad \forall i \in \{1, \dots, k\}$$
$$N_i^{\text{child}} = w_i \cdot N_i^1 + \left(1 - w_i\right) \cdot N_i^2, \quad \forall i \in \{1, \dots, k\}$$

where each w_i is independently selected from a uniform distribution $w_i \sim \mathcal{U}(0,1)$. This approach ensures that the offspring values lie within the range defined by the parent values, promoting diversity while retaining characteristics of both parents.

To ensure sufficient genetic diversity and avoid local optima, we incorporate a Mutation operation with a predefined low-probability P_{mut}. The Mutation perturbs the offspring's genes by adding a small random value δ to both σ and N. The perturbation δ is drawn from a Gaussian distribution $\mathcal{N}(0, \sigma_{\text{mut}}^2)$ or $\mathcal{N}(0, N_{\text{mut}}^2)$, where σ_{mut} and N_{mut} are the standard deviations that control the magnitude of mutation.

The genetic algorithm iteratively evolves the population by evaluating the fitness of each new generation, selecting the most promising individuals, and applying crossover and mutation operations until convergence criteria are met. In our experiment, we set the terminate criteria to be The final output of this process is the set of parameters $\{\sigma^*, N^*\}$ that maximize the fitness function, thereby ensuring that the UE fracture simulation closely aligns with the Ansys benchmark. We use $N_{\text{pop}} = 100$, $P_{\text{mut}} = 0.05$, $G_{\max} = 200$. Algorithm 1 summarizes the process of optimizing UE material parameters using our designed genetic algorithm.

Algorithm 1 Genetic Algorithm for Parameter Optimization with Mathematical Notation

Input: Population size N_{pop}, Mutation probability P_{mut}, Max generations $G_{\max}$
Output: Optimized parameters (σ^*, N^*)
Initialize population $\mathcal{P}_0 = \{(\sigma_i, N_i)\}_{i=1}^{N_{\text{pop}}}$
for $g = 1$ to $G_{\max}$ **do**
  for each individual $(\sigma_i, N_i) \in \mathcal{P}_{g-1}$ **do**
    Compute fitness:

$$F(\sigma_i, N_i) = -\sum_j \left| \phi_{\text{UE}}(T, x_j; \sigma_i, N_i) - \phi_{\text{ansys}}(T, x_j) \right|^2$$

  end for
  Rank individuals in $\mathcal{P}_{g-1}$ by $F(\sigma_i, N_i)$ and assign selection probabilities $P(r_i)$
  $P(r_i) = \frac{2\times(N_{\text{pop}} - r_i + 1)}{N_{\text{pop}} \times (N_{\text{pop}} + 1)}$ ▷ where r_i is the rank of individual i
  Select N_{pop} individuals from $\mathcal{P}_{g-1}$ based on $P(r_i)$ to form the mating pool
  for each selected pair (σ^1, N^1) and (σ^2, N^2) **do**
    Perform Two-Point Crossover:

$$\sigma_i^{\text{child}} = w_i \cdot \sigma_i^1 + \left(1 - w_i\right) \cdot \sigma_i^2, \quad \forall i \in \{1, \dots, k\}$$
$$N_i^{\text{child}} = w_i \cdot N_i^1 + \left(1 - w_i\right) \cdot N_i^2, \quad \forall i \in \{1, \dots, k\}$$

    With probability P_{mut}, apply mutation:

$$\sigma^{\text{mut}} = \sigma^{\text{child}} + \delta_\sigma, \quad \delta_\sigma \sim \mathcal{N}(0, \sigma_{\text{mut}}^2)$$
$$N^{\text{mut}} = N^{\text{child}} + \delta_N, \quad \delta_N \sim \mathcal{N}(0, N_{\text{mut}}^2)$$

    Add $(\sigma^{\text{mut}}, N^{\text{mut}})$ to the new population $\mathcal{P}_g$
  end for
  if stopping criteria are met (e.g., $\max F(\sigma_i, N_i)$ is stable or $g = G_{\max}$) **then**
    break
  end if
end for
Return $(\sigma^*, N^*) = \underset{(\sigma_i, N_i) \in \mathcal{P}_g}{\arg\max}\ F(\sigma_i, N_i)$

The RESenv Material Calibration Bridge is developed in C++ and can be loaded as a plugin into the Unreal Engine UI. This facilitates the bridging of data from Ansys and the management of the UE Fracture material library. It greatly simplifies the process for developers to customize and calibrate additional materials while avoiding the need to install external programs and configure data interfaces.

The Material Calibration Bridge represents a crucial step towards enabling high-fidelity simulations within the UE Destruction system, focusing specifically on material properties. Through parameter bridging, RESenv maintains UE's real-time interactivity and authentic visual rendering, while incorporating the physical simulation capabilities of Ansys.

3.2. Data preparing

This phase involves importing architectural models into the UE environment, applying fracture settings from the RESenv material library to the architectural models and acquiring earthquake wave data from online database.

Virtual Building Processing Due to the flexibility in model importation within UE, virtual building models represented as polygonal meshes can be acquired from various sources. For instance, they can be manually created using modeling software like Blender, computed from GIS data in CityEngine software [39], or generated via AI methods [40]. However, to ensure that the building models can be effectively simulated, the models first need to be pre-processed before being imported into our method through UE. Initially, the size units of the models need to be standardized. Typically, Polygon Mesh-based models do not have a unified scale unit like NURBS Surface models; therefore, the models need to be scaled to align with the cross-platform unified units. In RESenv, the default length unit in UE5, the centimeter, is adopted. Subsequently, once the architectural models are imported into UE, the pre-calibrated fracture settings are retrieved from the RESenv material library and applied to the geometries based on the materials of the buildings. As shown in Fig. 3, we have specifically developed a material preview visualization interface for the RESenv Material Calibration Bridge plugin to enable developers to intuitively select and fine-tune materials.

Earthquake Wave Data Acquisition The earthquake waveforms used by earthquake simulation can be divided into two categories. (1) waveforms recorded from actual earthquakes that have already occurred. These waveforms can be obtained from publicly available datasets online. An earthquake event is often recorded by multiple seismometers located at different geographical locations; by cross-comparing and applying algorithms for noise reduction, the absolute motion of the Earth's surface can be authentically reproduced in simulation platforms. (2) waveforms synthesized through algorithms [41, 42]. In earthquake resistance testing of buildings, researchers have developed various methods to synthesize earthquake waveforms in order to assess the impact of different levels and types of earthquakes on building structures. This enables the simulation platform to carry out unlimited iterations of earthquake tests in any conditions. Our method primarily aims to simulate the damage sustained by urban buildings in actual earthquakes to provide realistic datasets for AI visual-based training; therefore, the method initially implements the simulation of global earthquake waveforms obtained from the IRIS online database [43]. The acquired earthquake waveform data records three DOF of geological movement, named "BH1" (east–west direction), "BH2" (north-south direction), and "BHZ" (vertical direction). We have implemented a user-friendly user interface (UI) and Python-based automatic format conversion program in UE (Fig. 4), enabling users to directly obtain earthquake waveforms by clicking on the geographical location and event occurrence time on the global map without the need for complex data searches and imports. Once the user selects the required data, the waveforms are automatically converted into a "DataTable" file supported by UE.

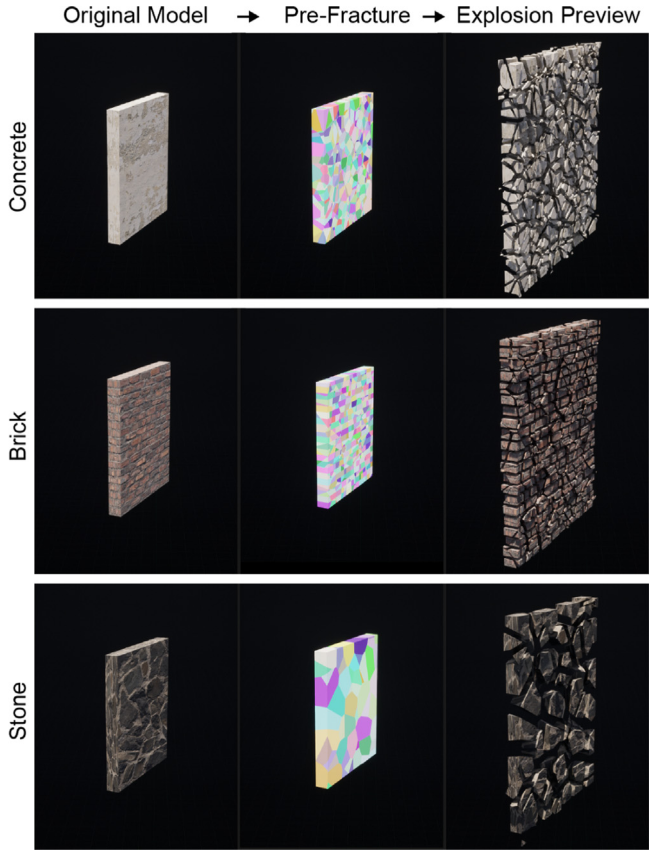

Fig. 3. Material Preview UI of the RESenv Calibration Bridge Plugin. The image displays three material examples: concrete, brick wall, and stone. Each example showcases texture rendering previews, fracture segmentation previews, and exploded views of fragments after applying fracture settings from the RESenv material library. The interface allows developers to intuitively select and finely adjust parameters for each material type.

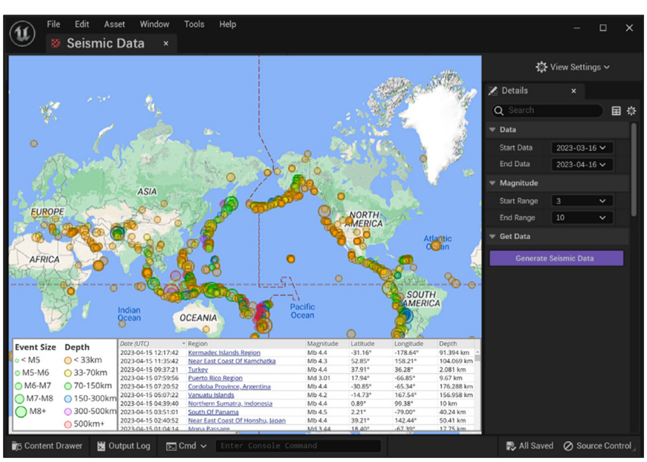

Fig. 4. The RESenv user interface for acquiring IRIS online earthquake data. The user interface contains an interactive world map that can be clicked on to select the earthquake data to be acquired. The column on the right side allows to define time ranges and earthquake levels as a filter.

3.3. Data binding with virtual terrain

In the real world, structures are anchored to the ground, responding dynamically to forces from earthquake activities. Mimicking this, our method endeavors to realistically simulate the anchor forces exerted on buildings during ground movements. Consequently, the RESenv necessitates the presence of a virtual terrain within UE to serve as the anchoring ground, binding the movements generated by the earthquake data.

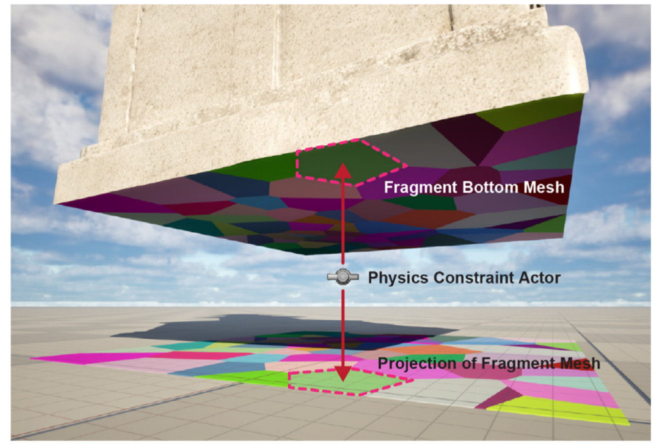

Fig. 5. Simplified Principles of Building Binding in RESenv. The algorithm developed identifies fragments at the base of the building geometry and their contact meshes with the ground. A UE Physics Constraint Actor (UEPCA) is then created for each contact mesh and bound to the virtual terrain.

RESenv employs the UE "Physics Constraint Actor" (UEPCA) [44] from the UE physics system to bind individual buildings to the virtual terrain. This method effectively constraints linear movements and rotations within the Cartesian coordinate system and allows for custom detachment thresholds, aligning with the requirements of simulating building-ground interactions during earthquakes.

Simplified Principles of Foundation Binding in RESenv: As Demonstrated in Fig. 5, the foundation binding in RESenv operates on a simplified principle. Specifically, we have developed a algorithm that identifies the fragments at the base of the building geometry and their contact meshes with the ground. The algorithm then creates a UEPCA for each of these meshes and binds them to the virtual terrain. This design offers three advantages:

1. *Customizable Binding Rules*: Developers can adjust the detachment stress and movement constraints based on different building types, for instance, wooden or concrete structures. This flexibility supports secondary development of constraint rules tailored to specific architectural needs.
2. *Realistic Simulation of Earthquake Effects*: The design accommodates the propagation of earthquake waves and the resulting flexible deformations of the terrain, leading to non-uniform stress changes on the building foundations, thus enhancing the realism of the simulations.
3. *Support for Three-dimensional Terrain Fluidity*: The system supports the transformation of virtual terrain into a three-dimensional fluid model. It allows for the importation of buildings with foundations without the need to reconfigure the binding program.

To facilitate the simulation of terrains moving akin to the Earth's crust during earthquake events, we have harnessed a C++ program. This program associates the previously extracted earthquake waveform data — "BH1", "BH2", and "BHZ" — with the "X", "Y", and "Z" axes, governing the terrain's motion. Earthquake waveform data from the IRIS database exhibits a frequency of 40 Hz, implying 40 recorded samples every second. In our simulation, RESenv offers three distinct frame rates in UE: 40 FPS, 90 FPS, and 240 FPS. These rates cater to different applications, including desktop simulations, VR training, and high frame rate sensor data synthesis. For handling the data at 90

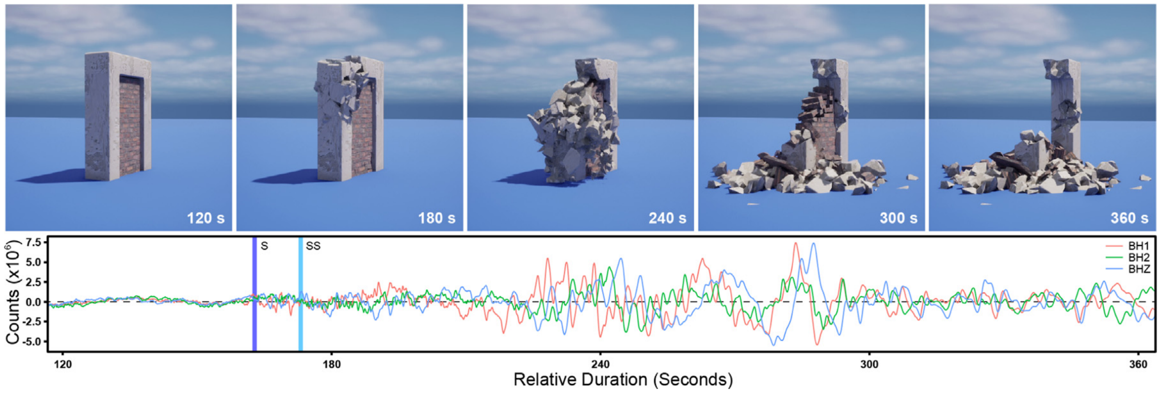

Fig. 6. An example showing the process of earthquake simulation by RESenv. In this example, a wall with a concrete and brick structure is pre-fractured and bound to a virtual terrain. The data taken from the 7.4 magnitude earthquake in Oaxaca, Mexico, on 23 June 2020, recorded by the earthquake station code-named TEIG. The entire simulation lasted 360 s. Five frames were extracted to demonstrate wall damage changes during the simulation.

and 240 FPS rates, we employed the wavelet interpolation algorithm, a method renowned for its ability to preserve signal details while increasing the sample rate [45].

3.4. Simulation

Upon completion of the data binding, our method can be executed in UE in Simulate In Editor (SIE) mode [46]. It is worth noting that, unlike traditional simulation approaches, our method inherits features from Unreal Engine, allowing all virtual assets and fractured models to be interactive during run-time. Applications such as VR search and rescue training and robotic dynamic obstacle avoidance will transition from static scene training to dynamic training with time-varying properties. Fig. 6 displays an example of a complete simulation process. In this instance, a concrete frame and a brick wall are bound to a flat virtual ground. The earthquake data is sourced from the magnitude 7.4 earthquake in Oaxaca, Mexico, on June 23, 2020, recorded by the seismograph station coded as TEIG. The simulation lasts 360 s and runs at a frame rate of 40 FPS.

4. Experiment

In order to verify the efficacy of our approach, two earthquake simulation experiments with three tasks were designed with the aim of providing synthetic visual data for AI training. (1) Two historical earthquake events and two laboratory experiments were reproduced and simulated with damage to buildings with four different materials. The similarity between real and simulated damaged buildings was assessed using a pre-trained Vision Transformer (ViT) model. (2) Using GIS data, we recreated a Japanese neighborhood and then conducted a 20 random endpoints robot path planning test in the simulated post-earthquake area based on synthetic visual data. The completion rates of the robot's path and the success rates of visual recognition en route are counted.

4.1. Realistic simulation of buildings

Four real-world earthquake-induced building damage scenarios are selected and re-created the destruction in a simulated environment using RESenv to compare the accuracy of our approach. Fig. 7 presents the original references and simulation results of the four scenarios. Each scenario simulates a building structure based on different materials. Two were simulated on a laboratory shake table [47,48], and two were natural earthquakes [49,50].

We obtained the building data and earthquake wave data of the two laboratory-simulated scenarios via email correspondence. The earthquake wave data of the two real-world scenarios were directly obtained from the IRIS database through the RESenv UI system, while the two buildings were reconstructed from multiple viewpoints using multiple online references. All four buildings were constructed in Blender, and surface textures were obtained from the Quixel Megascans material library [51]. The simulation was carried out in UE 5.4 on a Razor laptop with an RTX-3070 GPU, an AMD Ryzen 6900HX CPU, and 16 GB of RAM as the minimal requirement. The initial UE scenario was set to the default configuration.

Table 1
Similarity of the four scenarios simulated with RESenv over 100 attempts, with the best simulation run indicated.

Scenario	Brick	Wood	Stone	Concrete
Similarity	94.80%	90.07%	92.25%	89.34%
Best Attempt	57	83	22	48

To address the challenges arising from the inherent complexities of real-world building materials and structures, such as material aging, non-uniformity, and construction errors, we adopted a more exhaustive approach for similarity detection. Recognizing that a single simulation might not capture the full extent of potential damages, we conducted 100 random pre-fracture simulations for each building scenario to approximate the properties of real building materials iteratively. Following this exhaustive pre-fracture simulation, to ascertain the similarity between the simulated results and actual structural damage — especially in the context of robotic visual recognition tasks — we employed a verified ViT deep learning model designed for feature similarity assessment. This model was pre-trained on the widely-used ImageNet-21K dataset, and its efficacy has been validated in research by Omini et al. [52]. Images of real-world damaged buildings and those from our simulations were independently input into the ViT model to compute their similarity. The results, based on the highest similarity scores achieved, are believed to best represent the capabilities of our proposed method. This strategy not only captures the variability in damage patterns but also underscores our approach's robustness in mimicking real-world scenarios despite their inherent unpredictability and complexities. The final computational outcomes are presented in Table 1.

The results demonstrate that the simulations for all four structures attained a considerable degree of resemblance to real-world scenarios. Our proposed earthquake simulation technique exhibits a robust capability to accurately replicate the damage patterns induced by actual earthquake events in buildings. The observed discrepancies in the outcomes could be attributed to variations in the pre-fracture random factor setting of the 3D building materials, as compared to those of the reference structures. Consequently, these disparities give rise to differences in the morphology and movement trajectories of the fragmented masses within the simulation.

Fig. 7. Earthquake simulation using RESenv for four actual scenarios. Two laboratory experiments (columns 1–2) and two actual buildings (columns 3–4) were chosen. Row 1: The original forms of four buildings before the earthquake events. Row 2: Destruction of buildings following the earthquake events. Row 3: The 3D models are recreated based on the actual buildings and are given textures and pre-fracture settings in RESenv. Row 4: Destruction results of four buildings after RESenv earthquake simulation.

4.2. Scenario simulation and robotic training

To evaluate the effectiveness of RESenv in conducting earthquake simulations within urban settings featuring clusters of buildings for robotic training, two distinct tasks are designed. Initially, the GIS data from a Japanese neighborhood were obtained via OpenStreetMap and subsequently converted into a 3D scene utilizing CityEngine. The scene was then imported into UE. All buildings within the scene were automatically anchored to the terrain using the program in RESenv, while the terrain was linked to earthquake data. A robot model, sourced from RoverRobotics®, was positioned in the scene and equipped with simulated RGB and depth camera sensors (Fig. 8.a, d, e). The robot was assigned two tasks: (1) Utilizing a DRL model based on SLAM, as proposed by Shuhuan et al. [53], the robot was instructed to randomly select 20 coordinates as endpoints for path planning and obstacle avoidance testing within the simulated environment (Fig. 8.b). The ratio of the completed length of each path to its total length is recorded. This test aimed to verify whether our simulation framework could provide effective earthquake scenarios for AI path-planning methods with demonstrated efficacy. (2) Concurrently with Task 1, the Segment-Anything Model (SAM)(model: ViT-H) [54] was selected as the state-of-the-art for generalized image segmentation model to collect data from RGB sensors for object segmentation (Fig. 8.f). Image segmentation and its edge detection serve as the foundation for training AI models and path-planning tasks. For each frame from the camera, we compared the alignment of edges between the built-in UE segmentation (ground truth), SAM-processed ground truth, and SAM segmentation (Appendix A Fig. 9. Row 2–3). By using the Canny algorithm with edge dilation for pixel-level alignment deviation, the segmentation score was calculated as the proportion of overlapping edge pixels to the total edge pixels in the UE segmentation (Appendix A Fig. 9. Row 4–6). The final success rate for each path is derived from the average of all frames.

Upon completing the aforementioned tasks, the results (4 typical in Table 2, full targets in Appendix B Table 3) revealed that in Task 1, pertaining to path planning, 80% of path-planning trials achieved a 100% completion rate. When the input images have a resolution of 1550×1162 with a dilation kernel of 50 pixels, SAM achieved an overall 95% accuracy in detecting edges when compared to SAM-processed ground truth. These results prove our simulated post-earthquake scenario can furnish an effective image segmentation data source for visual recognition, thereby facilitating the training of various visual AI models.

Key Findings and Unforeseen Challenges The findings indicate that our proposed earthquake simulation approach effectively generates realistic urban destruction scenarios for robot training. The high

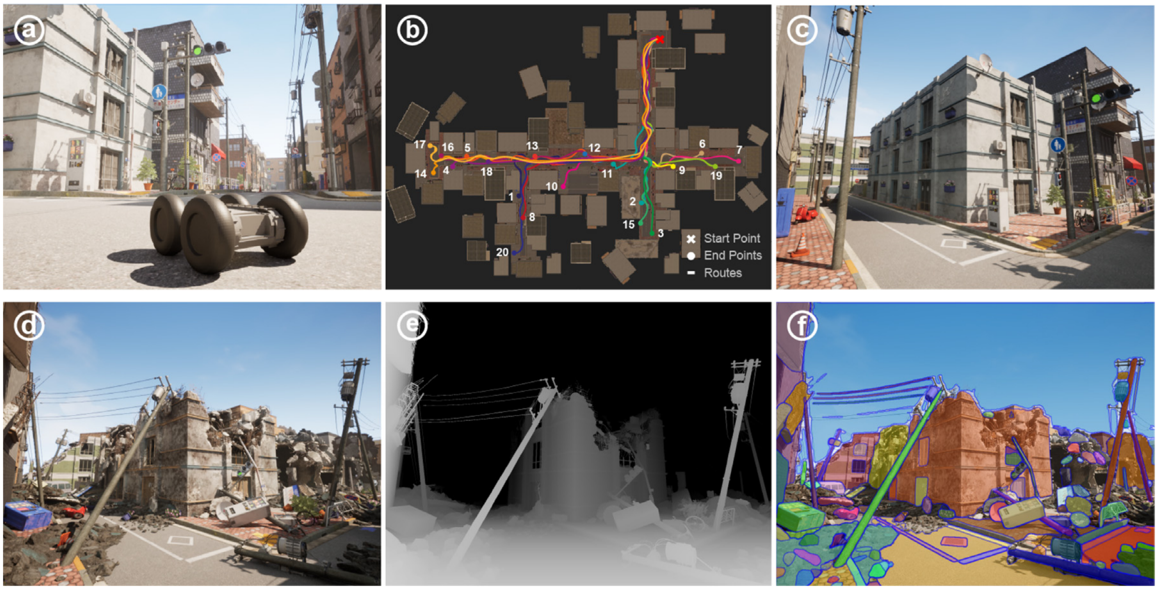

Fig. 8. Multi-building scenario earthquake simulation experiments in RESenv. Task 1 is to perform a robot path planning and obstacle avoidance test using the pre-trained DRL SLAM to verify the effectiveness of the earthquake simulation scenario for robot training. Task 2 is to use SAM to perform image segmentation detection on synthetic data from a RGB camera while the robot is traveling. Ultimately, the segmentation success rate will be counted. a: buildings generated in CityEngine using GIS data. A rover robot is placed in the scenario to perform path-planning tasks based on SLAM DRL. b: A record of 20 randomly selected endpoints for the path planning task. c: simulated RGB camera view of the original scenario in UE. d: RGB camera view of the scenario after an earthquake simulation. e: simulated depth camera view for SLAM DRL algorithm data input. f: RGB camera view with SAM image segmentation.

Table 2
Success rates of DRL SLAM path planning and accuracy of image segmentation edges for 4 typical out of 20 Targets.

Tgt. Pt.	Path plan.	Edges Acc.			
		UE Segment		UE + SAM	
		Ker. 25	Ker. 50	Ker. 25	Ker. 50
1	Complete	81.2%	91.9%	89.9%	96.0%
8	Complete	76.9%	87.3%	96.4%	99.1%
14	96.55%	78.8%	90.1%	87.9%	95.2%
20	85.80%	74.2%	86.2%	90.4%	95.5%

completion rates in Task 1 suggest that our simulation environment is capable of providing challenging yet achievable path planning and obstacle avoidance test scenarios for AI algorithms.

Moreover, in Task 2, we observed lower completion rates for some paths. Upon further inspection, we attributed this not to the complexity of the urban scenario itself, but to the *glare from the virtual sun* interfering with the simulated RGB camera used by the robot (Appendix A Fig. 9. Path-14, Path-20), leading to visual recognition difficulties. This situation has been overlooked in studies using ideal laboratory conditions and similar simulation platforms. This highlights the importance of considering the complexity and multi-factorial nature of real-world environments when designing and testing AI algorithms for disaster response and recovery, rather than focusing solely on object simulation.

5. Discussion and future work

Discussion Our study introduces an innovative earthquake simulation environment, designed to generate realistic urban scenarios for VR and robot training in the context of disaster response and recovery. Using computer vision techniques such as ViT, DRL SLAM, SAM, and our proposed earthquake simulation method, we have demonstrated the effectiveness of our approach by completing three distinct tasks: similarity, path-finding success rate, and segmentation edge accuracy. Our results show the environment is feasible for the deployment of downstream tasks.

This paper significantly expands upon the conference paper by providing a detailed description and discussion of the RESenv Material Calibration Workflow, and it refines the method for binding building foundations using the UEPCA in UE5 [8]. These enhancements have led to further improvements in the accuracy of building fracture simulations in RESenv.

Limitation While our work introduces pioneering advancements in earthquake simulation, it is subject to certain limitations. Primarily, our simulations focus mainly on the impacts of earthquakes on buildings, abstracting more complex dynamics such as foundation flexing and land movements. Consequently, our model may not fully capture the real-world nuances of how foundations degrade or shift during earthquake events currently. Furthermore, the parameters used to represent building materials are derived from idealized configurations, which may not accurately reflect the diversity of building types, architectures, and materials influenced by varying cultural, geographical, and technological factors. Lastly, our simulation environment does not account for variable environmental conditions such as changing lighting or obstructive elements like smoke and dust, which could significantly affect the performance and behavior of AI algorithms in such scenarios.

Future Works Our future endeavors in developing RESenv are aimed at addressing three main areas to enhance the platform's capability and accessibility. Firstly, we plan to establish an online database for calibrated materials used within RESenv, expanding it to include a comprehensive range of commonly used building materials. This will facilitate the use of RESenv by developers who are not familiar with Ansys, eliminating the need for them to perform their own material calibrations.

Secondly, we aim to focus on improving the current static terrain model by developing a two-dimensional dynamic terrain that accurately reflects the deformations caused by seismic waves. This advancement will significantly enhance the realism and scientific accuracy of our earthquake simulations.

Thirdly, we intend to enrich the simulation environment in RESenv by incorporating dynamic environmental factors such as changing lighting conditions, smoke, and dust. These elements will add to the realism

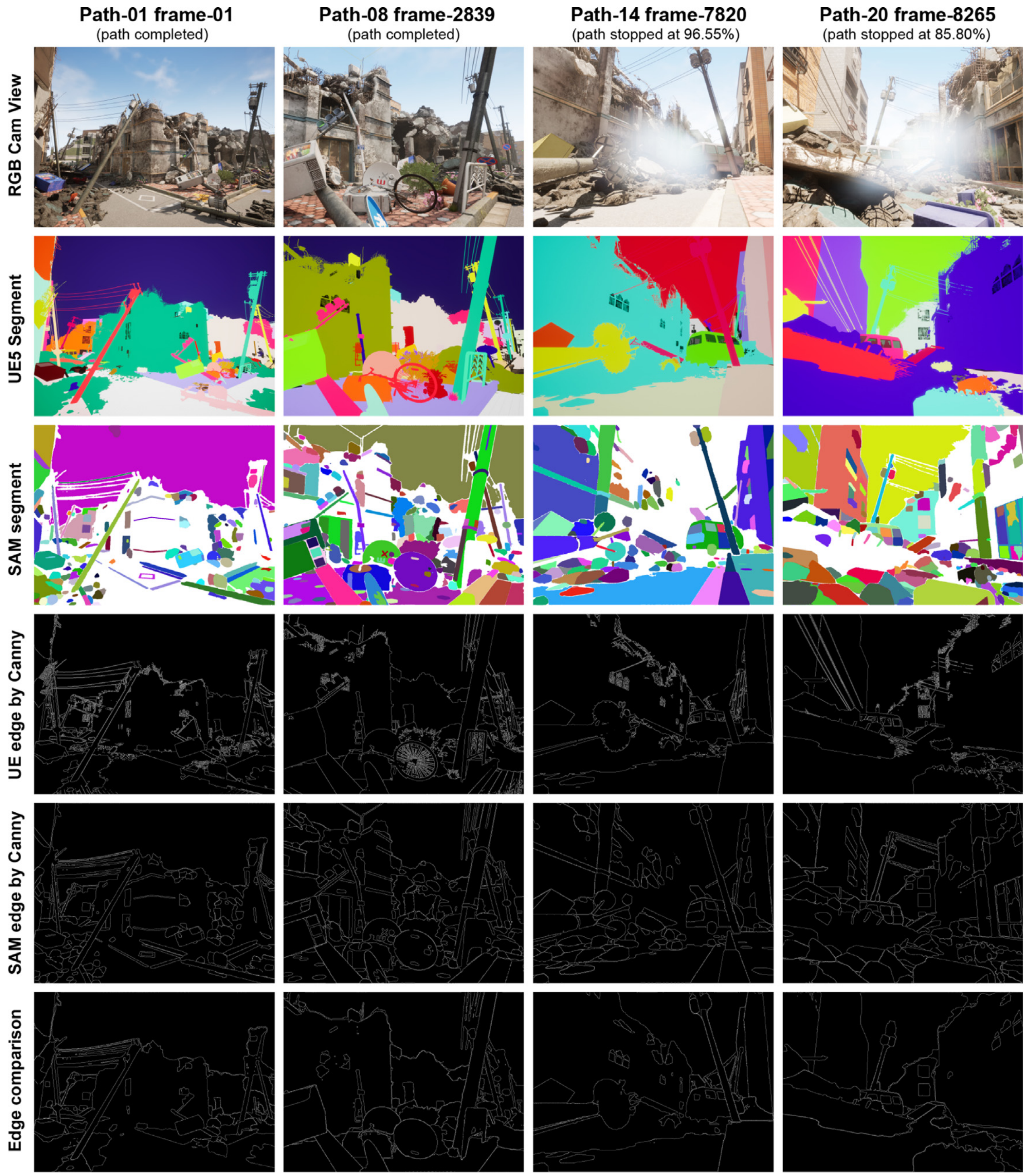

Fig. 9. Comparison of image segmentation and score calculation for 4 typical paths. **Row 1:** Synthetic RGB camera shot. **Row 2:** Object image segmentation using UE5 built-in camera post-processing material (ground truth). **Row 3:** Image segmentation of the RGB shot image using the SAM model. **Row 4:** UE segmentation edge map calculated using the Canny algorithm with edge dilation. **Row 5:** SAM segmentation edge map calculated using the Canny algorithm with edge dilation. **Row 6:** The segmentation score was calculated as the proportion of overlapping edge pixels to the total edge pixels in the UE segmentation.

of the scenarios and are expected to improve the robustness of AI and robotics training within these complex and variable conditions.

In the long term, we plan to extend our methodology to other types of disasters, including floods and hurricanes. This expansion will not only broaden the scope of RESenv's applicability but also contribute valuable insights into the simulation and mitigation of various natural disasters.

CRediT authorship contribution statement

Yitong Sun: Writing – review & editing, Writing – original draft, Visualization, Software, Resources, Methodology. **Hanchun Wang:** Validation, Software. **Zhejun Zhang:** Visualization. **Cyriel Diels:** Supervision. **Ali Asadipour:** Writing – review & editing, Supervision.

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

Data will be made available on request.

Appendix A

See Fig. 9.

Table 3
Success rates of DRL SLAM path planning and accuracy of image segmentation edges for 20 targets.

Tgt. Pt.	Path plan.	Edges Acc.			
		UE Segment		UE + SAM	
		Ker. 25	Ker. 50	Ker. 25	Ker. 50
1	Complete	81.2%	91.9%	89.9%	96.0%
2	Complete	76.3%	86.7%	88.6%	93.5%
3	Complete	79.2%	89.8%	87.9%	94.2%
4	Complete	75.6%	85.9%	86.8%	92.6%
5	Complete	82.5%	92.7%	90.8%	97.5%
6	Complete	71.8%	82.4%	85.0%	91.3%
7	85.4%	70.4%	81.1%	84.7%	90.8%
8	Complete	7k6.9%	87.3%	96.4%	99.1%
9	Complete	79.9%	90.1%	89.5%	96.7%
10	Complete	81.0%	91.4%	90.6%	96.3%
11	Complete	80.2%	91.6%	90.4%	96.4%
12	Complete	81.6%	91.5%	89.9%	96.5%
13	Complete	80.7%	92.2%	90.2%	96.6%
14	96.55%	78.8%	90.1%	87.9%	95.2%
15	Complete	81.3%	91.8%	90.0%	96.8%
16	Complete	80.9%	91.7%	90.5%	96.1%
17	92.21%	72.4%	85.0%	89.3%	94.9%
18	Complete	80.6%	91.4%	89.8%	96.7%
19	Complete	80.8%	91.9%	90.7%	96.3%
20	85.80%	74.2%	86.2%	90.4%	95.5%

Appendix B

See Table 3.

References

[1] Joseph I-L. The effect of natural disaster on economic growth: Evidence from a major earthquake in Haiti. World Dev 2022;159:106053.

[2] Favero DD, Thurow S, Ostwald MJ, Frohne U, editors. Climate disaster preparedness, first ed.. Arts, research, innovation and society, Cham: Springer Cham; 2024, http://dx.doi.org/10.1007/978-3-031-56114-6.

[3] Nazarova AV, Zhai M. The application of multi-agent robotic systems for earthquake rescue. Robot: Industry 4.0 Issues & New Intelligent Control Paradigms 2020;133–46.

[4] Magid E, Pashkin A, Simakov N, Abbyasov B, Suthakorn J, Svinin M, Matsuno F. Artificial intelligence based framework for robotic search and rescue operations conducted jointly by international teams. In: Proceedings of 14th international conference on electromechanics and robotics "zavalishin's readings" ER (ZR) 2019, kursk, Russia, 17-20 April 2019. Springer; 2019, p. 15–26.

[5] Duan J, Yu S, Tan HL, Zhu H, Tan C. A survey of embodied ai: From simulators to research tasks. IEEE Trans Emerg Top Comput Intell 2022;6(2):230–44.

[6] Bourhim EM, Cherkaoui A. Selection of optimal game engine by using AHP approach for virtual reality fire safety training. In: Intelligent systems design and applications: 18th international conference on intelligent systems design and applications (ISDA 2018) held in vellore, India, December 6-8, 2018, volume 1. Springer; 2020, p. 955–66.

[7] Greff K, Belletti F, Beyer L, Doersch C, Du Y, Duckworth D, Fleet DJ, Gnanapragasam D, Golemo F, Herrmann C, et al. Kubric: A scalable dataset generator. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 3749–61.

[8] Sun Y, Wang H, Zhang Z, Diels C, Asadipour A. Resenv: A realistic earthquake simulation environment based on unreal engine. 2023, arXiv preprint arXiv:2311.07239.

[9] Matin SS, Pradhan B. Challenges and limitations of earthquake-induced building damage mapping techniques using remote sensing images-A systematic review. Geocarto Int 2022;37(21):6186–212.

[10] Xu Z, Wu Y, Lu X, Jin X. Photo-realistic visualization of seismic dynamic responses of urban building clusters based on oblique aerial photography. Adv Eng Inform 2020;43:101025.

[11] Shaw BE, Fry B, Nicol A, Howell A, Gerstenberger M. An earthquake simulator for New Zealand. Bull Seismol Soc Am 2022;112(2):763–78.

[12] Xu Z, Zhang Z, Lu X, Zeng X, Guan H. Post-earthquake fire simulation considering overall seismic damage of sprinkler systems based on BIM and FEMA P-58. Autom Constr 2018;90:9–22.

[13] McKenna F. OpenSees: a framework for earthquake engineering simulation. Comput Sci Eng 2011;13(4):58–66.

[14] McEntee P. Simpson strong-tie research: The tye gilb lab. SE Blog 2022. URL https://seblog.strongtie.com/2012/05/simpson-strong-tie-research-the-tye-gilb-lab/.

[15] Oleson T. Faking quakes at full scale: Giant shake tables simulate earthquakes to make buildings safer. In: Faking quakes at full scale: Giant shake tables simulate earthquakes to make buildings safer. 2022, https://www.earthmagazine.org/article/faking-quakes-full-scale-giant-shake-tablessimulate-earthquakes-make-buildings-safer/.

[16] Schneider PJ, Schauer BA. HAZUS—Its development and its future. Nat Hazards Rev 2006;7(2):40–4.

[17] Hori M. Introduction to computational earthquake engineering. World Scientific; 2011.

[18] Sahin A, Sisman R, Askan A, Hori M. Development of integrated earthquake simulation system for Istanbul. Earth, Planets Space 2016;68(1):1–21.

[19] McCallen D, Petrone F, Miah M, Pitarka A, Rodgers A, Abrahamson N. EQSIM—A multidisciplinary framework for fault-to-structure earthquake simulations on exascale computers, part II: Regional simulations of building response. Earthq Spectr 2021;37(2):736–61.

[20] Taylan D, Aydın T. Analysis of dynamic behavior of Darideresi-II dam by ANSYS. Nat Hazards 2018;90:1223–35.

[21] AlMusbahi S, Güngör A. A composite building isolation system for earthquake protection. Eng Sci Technol, Int J 2019;22(2):399–404.

[22] Sevim B, Bayraktar A, Altunişik AC, Atamtürktür S, Birinci F. Finite element model calibration effects on the earthquake response of masonry arch bridges. Finite Elem Anal Des 2011;47(7):621–34.

[23] Chaos Destruction Overview; 2024. [Accessed: 2024-05-19], https://dev.epicgames.com/documentation/en-us/unreal-engine/destruction-overview.

[24] Nanite Virtualized Geometry in Unreal Engine; 2024. [Accessed: 2024-05-19], https://dev.epicgames.com/documentation/unreal-engine/nanite-virtualized-geometry-in-unreal-engine.

[25] Costa MM, Silva MF. A survey on path planning algorithms for mobile robots. In: 2019 IEEE international conference on autonomous robot systems and competitions. ICARSC, IEEE; 2019, p. 1–7.

[26] Queralta JP, Taipalmaa J, Pullinen BC, Sarker VK, Gia TN, Tenhunen H, Gabbouj M, Raitoharju J, Westerlund T. Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision. IEEE Access 2020;8:191617–43.

[27] Wang L-L, Pan L-X. Research on sbmpc algorithm for path planning of rescue and detection robot. Discrete Dyn Nat Soc 2020;2020:1–11.

[28] Zhang X, Lai J, Xu D, Li H, Fu M. 2D lidar-based slam and path planning for indoor rescue using mobile robots. J Adv Transp 2020;2020:1–14.

[29] Niroui F, Zhang K, Kashino Z, Nejat G. Deep reinforcement learning robot for search and rescue applications: Exploration in unknown cluttered environments. IEEE Robot Autom Lett 2019;4(2):610–7.

[30] Bischke B, Helber P, Brugman S, Basar E, Zhao Z, Larson MA, Pogorelov K. The multimedia satellite task at MediaEval 2019.. In: MediaEval. 2019.

[31] Skinner C, Ramchurn S. The robocup rescue simulation platform. In: Proceedings of the 9th international conference on autonomous agents and multiagent systems: volume 1-volume 1. 2010, p. 1647–8.

[32] Polverari G, Calisi D, Farinelli A, Nardi D. Development of an autonomous rescue robot within the usarsim 3d virtual environment. In: RoboCup 2006: robot soccer world cup x 10. Springer; 2007, p. 491–8.

[33] Grunwald C, Khalil AA, Schaufelberger B, Ricciardi EM, Pellecchia C, De Iuliis E, Riedel W. Reliability of collapse simulation–Comparing finite and applied element method at different levels. Eng Struct 2018;176:265–78.

[34] Fracturing Geometry Collections User Guide; 2024. [Accessed: 2024-05-19], https://dev.epicgames.com/documentation/unreal-engine/fracturing-geometry-collections-user-guide.

[35] ASTM International. Standard test method for flexural strength of concrete (using simple beam with center-point loading). ASTM International; 2021, Available from: https://www.astm.org/Standards/C293.htm, ASTM C293/C293M-16, West Conshohocken, PA.

[36] ASTM International. Standard test method for diagonal tension (shear) in masonry assemblages. ASTM International; 2022, Available from: https://www.astm.org/Standards/E519.htm, ASTM E519/E519M-17, West Conshohocken, PA.

[37] Riedel W, Kawai N, Kondo K-i. Numerical assessment for impact strength measurements in concrete materials. Int J Impact Eng 2009;36(2):283–93.

[38] Parker C, Tanner J, Varela J. Evaluation of ASTM methods to determine splitting tensile strength in concrete, masonry, and autoclaved aerated concrete introduction and background on ASTM tests to evaluate splitting tensile strength. 2019.

[39] Badwi IM, Ellaithy HM, Youssef HE. 3D-GIS parametric modelling for virtual urban simulation using CityEngine. Ann GIS 2022;28(3):325–41.

[40] Chaillou S. AI & architecture. 2019.

[41] Moseley B, Markham A, Nissen-Meyer T. Fast approximate simulation of seismic waves with deep learning. 2018, arXiv preprint arXiv:1807.06873.

[42] Moseley B, Nissen-Meyer T, Markham A. Deep learning for fast simulation of seismic waves in complex media. Solid Earth 2020;11(4):1527–49.

[43] IRIS Data Services; 2024. [Accessed: 2024-05-19], https://ds.iris.edu/ds/nodes/dmc/data/.

[44] Physics constraints in unreal engine; 2024. [Accessed: 2024-05-19], https://dev.epicgames.com/documentation/unreal-engine/physics-constraints-in-unreal-engine.

[45] Yu Z, Ferguson J, McMechan G, Anno P. Wavelet-radon domain dealiasing and interpolation of seismic data. Geophysics 2007;72(2):V41–9.

[46] In-Editor testing, play and simulate in unreal engine; 2024. [Accessed: 2024-05-19], https://dev.epicgames.com/documentation/unreal-engine/in-editor-testing-play-and-simulate-in-unreal-engine.

[47] Giaretton M, Ingham JM, Dizhur D. Shaking table testing of two storey as-built and retrofitted scaled URM buildings. In: Structures. 41, Elsevier; 2022, p. 260–9.

[48] Seismic testing of seismic intensity 6; 2020. URL https://www.youtube.com/watch?v=ddr5KT1nX7M.

[49] Michaelson R. Earthquake in Turkey and Syria kills thousands and devastates cities. Guardian 2023. URL https://www.theguardian.com/world/2023/feb/06/turkey-earthquake-2023-gaziantep.

[50] Team VJ. Turkey earthquake: Before and after pictures show extent of destruction. BBC News 2023. URL https://www.bbc.com/news/world-europe-64544998.

[51] Quixel Megascans; 2024. [Accessed: 2024-05-19], https://quixel.com/megascans.

[52] Omori H, Hanyu K. Predict inter-photo visual similarity via pre-trained computer vision models. In: Proceedings of the 2022 5th international conference on algorithms, computing and artificial intelligence. 2022, p. 1–9.

[53] Wen S, Zhao Y, Yuan X, Wang Z, Zhang D, Manfredi L. Path planning for active SLAM based on deep reinforcement learning under unknown environments. Intell Serv Robot 2020;13:263–72.

[54] Kirillov A, Mintun E, Ravi N, Mao H, Rolland C, Gustafson L, Xiao T, Whitehead S, Berg AC, Lo W-Y, et al. Segment anything. 2023, arXiv preprint arXiv:2304.02643.