

# RESEnv: A Realistic Earthquake Simulation Environment based on Unreal Engine

Yitong Sun<sup>†</sup><sup>1</sup>, Hanchun Wang<sup>2</sup>, Zhejun Zhang<sup>1</sup>, Cyriel Diels<sup>1</sup>, and Ali Asadipour<sup>‡</sup><sup>1</sup>

<sup>1</sup>Royal College of Art  
<sup>2</sup>Imperial College London

## Abstract

Earthquakes have a significant impact on societies and economies, driving the need for effective search and rescue strategies. With the growing role of AI and robotics in these operations, high-quality synthetic visual data becomes crucial. Current simulation methods, mostly focusing on single building damages, often fail to provide realistic visuals for complex urban settings. To bridge this gap, we introduce an innovative earthquake simulation system using the Chaos Physics System in Unreal Engine. Our approach aims to offer detailed and realistic visual simulations essential for AI and robotic training in rescue missions. By integrating real seismic waveform data, we enhance the authenticity and relevance of our simulations, ensuring they closely mirror real-world earthquake scenarios. Leveraging the advanced capabilities of Unreal Engine, our system delivers not only high-quality visualisations but also real-time dynamic interactions, making the simulated environments more immersive and responsive. By providing advanced renderings, accurate physical interactions, and comprehensive geological movements, our solution outperforms traditional methods in efficiency and user experience. Our simulation environment stands out in its detail and realism, making it a valuable tool for AI tasks such as path planning and image recognition related to earthquake responses. We validate our approach through three AI-based tasks: similarity detection, path planning, and image segmentation.

## CCS Concepts

• **Computing methodologies** → *Simulation environments; Computer vision*; • **Human-centered computing** → *Visualization toolkits*;

## 1. Introduction

Earthquakes, as recurrent natural disasters, profoundly impact human life and economic activities [Jos22]. To mitigate the adverse effects and enhance the efficiency of post-earthquake rescue operations, a new paradigm shift towards the integration of artificial intelligence (AI) and robotics has been observed [NZ20]. These technologies, encompassing tasks such as path planning, automatic obstacle avoidance, and image recognition, bring about significant improvements in earthquake response activities [MPS\*19]. Yet, given the unpredictable and dynamic nature of earthquakes, creating a platform to iteratively train these technologies across a myriad of realistic earthquake scenarios becomes a pressing need.

While virtual game engines have emerged as a potent tool for simulating various disaster situations, like firestorms or flash floods [BC20], their application for earthquake scenarios, despite their potential, remains largely unexplored. These engines not only offer reduced operational complexity but also ensure shorter and more

efficient training cycles than their traditional counterparts. Furthermore, the integration of cutting-edge realistic rendering techniques, especially ray tracing, is paving the way for substituting real-world data, thereby revolutionising visual recognition research [GBB\*22]. However, harnessing these advanced techniques and tools specifically for earthquake simulations within a virtual environment has been a gap in the current research landscape.

To address this gap, we present RESEnv—an environment tailored for earthquake simulation, leveraging the capabilities of the Chaos physics engine within Unreal Engine 5 (UE5). By using actual seismic waveform data sourced from online repositories and integrating it into UE5, RESEnv aims to offer a nuanced simulation of building destruction during earthquakes. The overarching goal of this novel environment is to furnish high-resolution, in-depth visual, and scenario simulations. This not only aids in search and rescue mission planning but also emerges as an invaluable synthetic data reservoir for AI training in diverse applications such as path planning and visual recognition.

Our key contributions in this study include:

1. We introduce a method based on UE5, aiming to provide a realistic earthquake simulation. This approach simulates multi-

<sup>†</sup> yitong.sun@network.rca.ac.uk

<sup>‡</sup> ali.asadipour@rca.ac.uk

building scenarios by capitalising on actual seismic data sourced online.

2. The endorsement of the applicability and efficiency of RESEnv via three AI-centric tasks, namely similarity detection, path planning, and image segmentation. This solidifies RESEnv's potential as an exemplary environment that equips AI models with a rich dataset.
3. A paradigmatic shift from conventional methods to a more streamlined, intuitive, and accessible approach, courtesy of the game engine's versatility, effectively dismantling domain-specific barriers and democratising access.

## 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 [MP22], has seen significant advancements due to recent progress in computer hardware. This has enabled more sophisticated modelling of earthquakes and consequent building damage using numerical simulation techniques [XWLJ20, SFN\*22]. Stress simulations of individual buildings, initially aimed at analysing seismic stress-induced deformation and structural optimisation, have matured, and some researchers have employed finite element analysis (FEA) for assessing earthquake-induced building damage and exploring risk mitigation strategies [XZL\*18, McK11]. 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 [McE22, Ole22].

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) [SS06]. Based on standardised 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, utilising multi-dimensional data fusion calculation methods [Hor11]. Subsequently, Turkish researchers developed a regional building simulation method for Istanbul using MATLAB, based on the IES framework [SSAH16]. 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 [MPM\*21], the focus of these research predominantly lies in calculating the complexity

of geological structures, with scant attention paid to the fidelity of building structures.

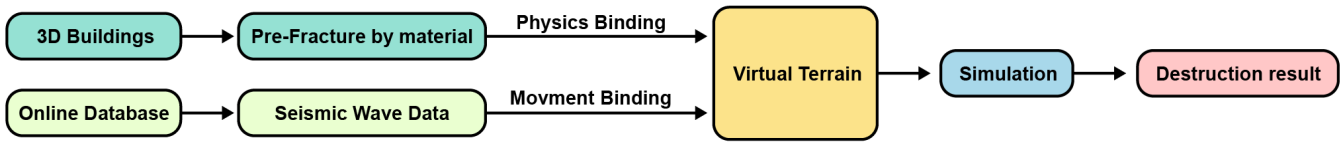
In our approach, we utilise the Chaos destruction system (<https://docs.unrealengine.com/5.1/en-US/destruction-overview>) and Nanite visualisation system (<https://docs.unrealengine.com/5.1/en-US/nanite-virtualized-geometry-in-unreal-engine>) within the UE 5 game engine, achieving previously challenging fracture and fragmentation simulations for different materials and complex hybrid structures, accurate physical collisions, and three DOF geological motion. 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. Benefiting from the user-friendliness of the game engine, we have established a pre-fabricated library of materials and structures readily available to users by creating pre-set programs, greatly simplifying operational difficulties compared to traditional simulation frameworks, and thus enabling researchers from various fields to use our approach with ease.

### 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 [CS19, QTP\*20]. For instance, the study by LinLin et al. utilised the SBMPC algorithm to investigate path planning problems for search and rescue robots [WP20]. Xuexi et al. explored indoor search and rescue using Simultaneous Localisation and Mapping (SLAM) and Light Detection and Ranging (LiDAR) methods [ZLX\*20]. Farzad's research introduced the application of deep reinforcement learning (DRL) methods in search and rescue robot tasks [NZKN19]. 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 [BHB\*19]. 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 [SR10], USARSim [PCFN07], and the BCB environment developed by Laurea University of Applied Sciences [GKS\*18], 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



**Figure 1:** Flowchart of RESEnv for earthquake simulation. There are three steps: scenario preparation, data binding, and simulation. During the data preparation phase, 3D building models are imported into UE, which are then pre-fractured by material groups. Actual seismic wave data was acquired from the IRIS online database and imported via a graphical UI system. In the data binding phase, the 3D buildings are bound to the virtual terrain by PCA through an automated analysis program. The seismic wave data is converted into terrain displacements for binding. During the simulation phase, RESEnv is run at a specified frame rate, 40 FPS for desktop computer simulation, 90 FPS for VR training, and 240 FPS for high frame rate sensor training. As the simulation begins, the seismic wave data displaces the terrain, which in turn causes the pre-fractured 3D buildings to be destroyed. RESEnv is interactable throughout the simulation.

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

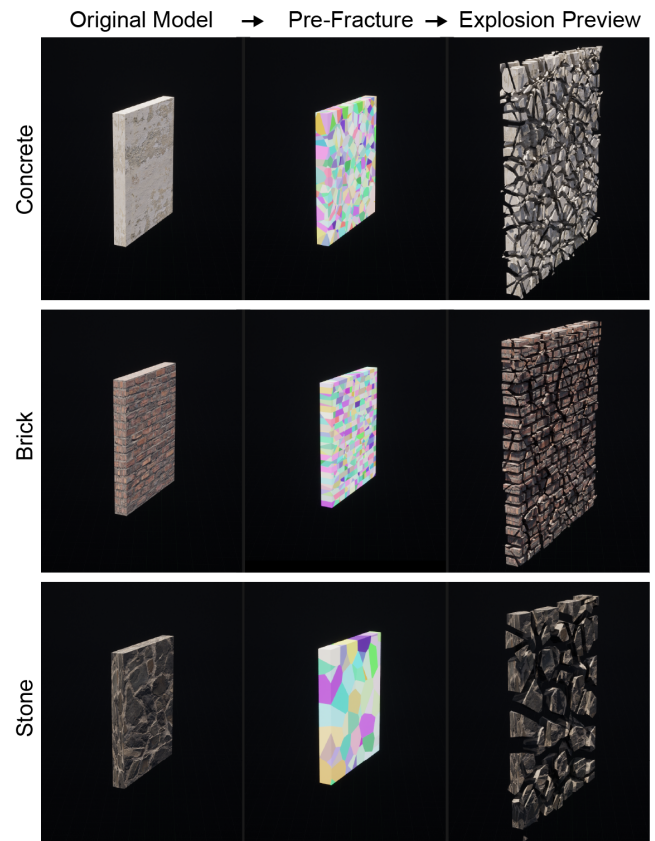
RESEnv executes the earthquake simulation in a three-stage process: **data preparation**, **data binding**, and **simulation**. Figure 1 describes the flowchart of the method.

#### 3.1. Data Preparing

**Virtual Building Processing** Due to the flexibility in model importation within UE, virtual building models based on Polygon Mesh can be acquired from various sources. For instance, manual creation using modelling software like Blender, computation from GIS data in CityEngine software [BEY22], or generation via AI methods [Cha19]. However, to ensure that the building models can be effectively simulated, the models first need to be pre-processed before importing them into our method through UE. Initially, the size units of the models need to be standardised. Typically, Polygon Mesh-based models do not have a unified scale unit like NURBS Surface models; the models need to be pre-scale to align to the cross-platform unified units. In RESEnv, the default length unit in UE5 is adopted (centimetre). Secondly, building models need to be segmented according to distinct materials, such as concrete, bricks, and wooden structures, to set up pre-fracture settings (<https://docs.unrealengine.com/5.1/en-US/destruction-overview/>) separately after importing into UE.

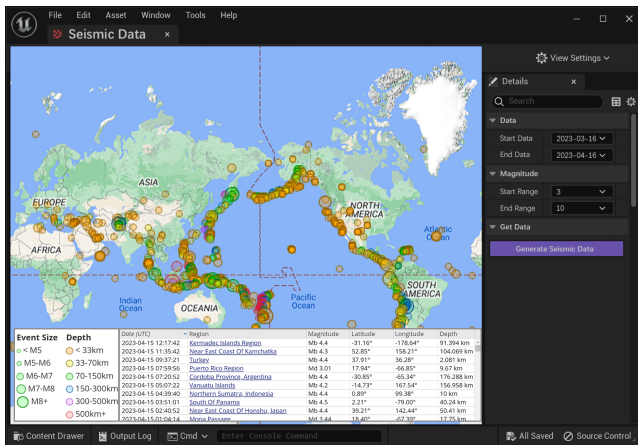
In the process of calibrating our pre-fracture settings, the material mechanics data published by MatWeb (<https://www.matweb.com>) and Material Data Repository (<https://material.sdata.nist.gov>) served as crucial references. Specifically, we employed Ansys to incorporate this data, setting up analogous conditions within UE to discern the optimal pre-fracture parameters. By aligning the mechanical properties derived from these databases with UE's pre-fracturing capabilities, we ensured that our simulations were as realistic and accurate as possible. Additionally, to facilitate usability, we have meticulously curated a comprehensive material library, with plans for its continuous expansion. For users inclined towards customising the pre-fracture settings further, we recommend initially consulting our material library settings as a

baseline, and subsequently referring to the official Unreal Engine documentation for tailored adjustments and configurations. This strategy not only ensures that our method remains grounded in val-



**Figure 2:** Three examples of pre-fracturing of walls in different materials. Once the building has been imported into UE, the 3D model needs to be pre-fractured in order to be damaged accordingly in the earthquake simulation. The pre-fracture is set according to the material properties of the building. The three instances show the set-up process, segmentation methods, and explosion view for concrete, brick wall, and stone structures, respectively.



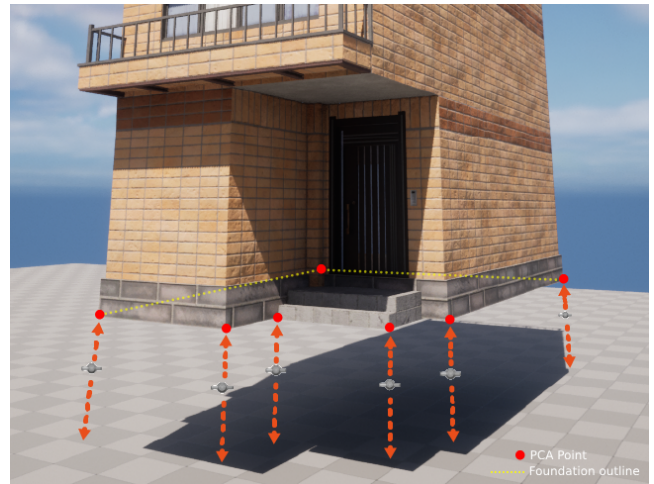


**Figure 3:** The RESEnv user interface for acquiring IRIS online seismic data. The user interface contains an interactive world map that can be clicked on to select the seismic data to be acquired. The column on the right side allows to define time ranges and earthquake levels as a filter.

idated material science but also offers flexibility for advanced users to fine-tune their simulations.

In contrast to traditional earthquake simulation platforms that only simulate the physical collision between blocks and joints, our approach based on the Chaos Physics System in UE5 can achieve effects similar to real building destruction. Figure 2 presents examples of pre-fracture settings for three different wall materials. As seen in the figure, different segmentation strategies and levels can be applied according to distinct materials. The fractured geometry collection can be set with different break-force thresholds to represent the strength of the materials.

**Seismic Wave Data Acquisition** The seismic 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 synthesised through algorithms [MMNM18, MNMM20]. In seismic resistance testing of buildings, researchers have developed various methods to synthesise 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 (<https://ds.iris.edu/ds/nodes/dmc/data/>). The acquired seismic waveform data records three DOF of geological movement, named "BH1" (east-west direction), "BH2" (north-



**Figure 4:** An exemplification of a 3D building, autonomously anchored to a virtual terrain through the PCA placement program.

south direction), and "BHZ" (vertical direction). We have implemented a user-friendly user interface (UI) and Python-based automatic format conversion program in UE (Figure 3), enabling users to directly obtain seismic 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.

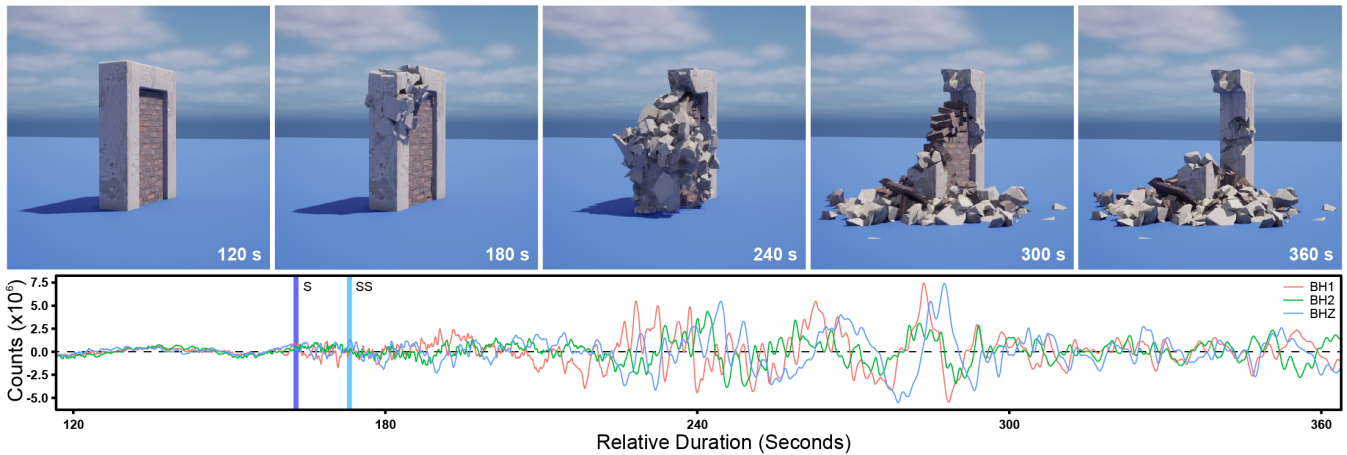
### 3.2. Data Binding with Virtual Terrain

In the real world, structures are anchored to the ground, responding dynamically to forces from seismic activities. Mimicking this, our method endeavours 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.

The choice of using the "Physics Constraint Actor" (PCA) (<https://docs.unrealengine.com/5.1/en-US/physics-constraints-in-unreal-engine/>) in the UE physics system for binding buildings to the virtual terrain is motivated by its robust capability to simulate real-world constraint forces. It accurately mirrors the building-ground interaction during earthquakes based on their physical attributes. This binding mechanism establishes a series of physical anchor force constraints between the structure and the terrain, wherein the building's linear motion threshold and rotational thresholds within the Cartesian coordinate system are determined by the building's material properties and volume.

A cornerstone of our approach is the foundation weight map, which is derived from the building's three-dimensional volume. Larger volume regions bear higher weight, ensuring that the PCA's anchoring force corresponds authentically to the actual size and shape of the building. For instance, during minor seismic activities, the building's foundation and walls will undergo nominal displace-





**Figure 5:** 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 seismic station code-named TEIG. The entire simulation lasted 360 seconds. Five frames were extracted to demonstrate wall damage changes during the simulation.

ments without detaching from the terrain. However, during more intense quakes, the building components will fracture and potentially collapse, aligning with the intrinsic architectural design.

One salient challenge in placing the PCAs is preventing an overwhelming concentration of constraint forces within a singular area, which might result in unrealistic building movements. Meanwhile, too dense constraint placement will cause serious waste of computing resources. To surmount this, we've devised an algorithm that dynamically identifies optimal PCA positions based on the foundation's contours and inflection points. Furthermore, the RESEnv deploys a C++ program to automate the analysis of the building foundation's shape, primarily its planar geometry. By identifying edge transitions, we systematically place the PCA while ensuring a minimum distance between them. Appendix B lists pseudocode for performing PCA binding. Figure 4 demonstrates the distribution of binding points when a building is automatically bound to the ground in our approach.

To facilitate the simulation of terrains moving akin to the Earth's crust during seismic events, we've 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 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 [YFMA07].

### 3.3. Simulation

Upon completion of the data binding, our method can be executed in UE in Simulate In Editor (SIE) mode (<https://docs.unrealengine.com/5.1/en-US/in-editor-testing-play-and-simulate-in-unreal-engine>). 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. Figure 5 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 seismic 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 seconds 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 neighbourhood 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.



**Figure 6:** 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.

#### 4.1. 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. Figure 6 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 [GID22, noa], and two were natural earthquakes [Mic23, Tea23].

We obtained the building data and seismic wave data of the two laboratory-simulated scenarios via email correspondence. The seismic 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 view-

points using multiple online references. All four buildings were constructed in Blender, and surface textures were obtained from the Quixel Megascans material library (<https://quixel.com/megascans>). The simulation was carried out in UE 5.1.1 on a Razor laptop with an RTX-3070 GPU, an AMD Ryzen 6900HX CPU, and 16GB of RAM as the minimal requirement. The initial UE scenario was set to the default configuration.

To address the challenges arising from the inherent complexities of real-world building materials and structures, which are difficult to replicate perfectly in simulations, we adopted a more exhaustive approach for similarity detection. Recognising that a single simulation might not capture the full extent of potential damages, we conducted 100 random pre-fracture simulations for each building scenario, ensuring consistent material properties across iterations.

Subsequent to this, iterative similarity checks were performed using our algorithm. The results presented are based on the highest similarity scores obtained, which, we believe, best represent the capabilities of our proposed method. This strategy not only helps in capturing the variability in damage patterns but also underscores the robustness of our approach in achieving close resemblance despite the inherent unpredictability and complexities of real-world scenarios.

To address the challenges arising from the inherent complexities of real-world building materials and structures, such as material ageing, non-uniformity, and construction errors, we adopted a more exhaustive approach for similarity detection. Recognising 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. [OH22]. 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.

**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 /100	83 /100	22 /100	48 /100

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 seismic events in buildings. The observed discrepancies in the outcomes could be attributed to variations in the pre-fracture parameters 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.

## 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 neighbourhood were obtained via OpenStreetMap and subsequently converted into a 3D scene utilising

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<sup>®</sup>, was positioned in the scene and equipped with simulated RGB and depth camera sensors (Figure 7.a, d, e). The robot was assigned two tasks: 1) Utilising a DRL model based on SLAM, as proposed by Shuhuan et al. [WZY\*20], the robot was instructed to randomly select 20 coordinates as endpoints for path planning and obstacle avoidance testing within the simulated environment (Figure 7.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) [KMR\*23] was selected as the state-of-the-art for generalised image segmentation model to collect data from RGB sensors for object segmentation (Figure 7.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.8. 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.8. 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 Tabel 2, full targets in Appendix C Tabel 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  $1550 \times 1162$  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.

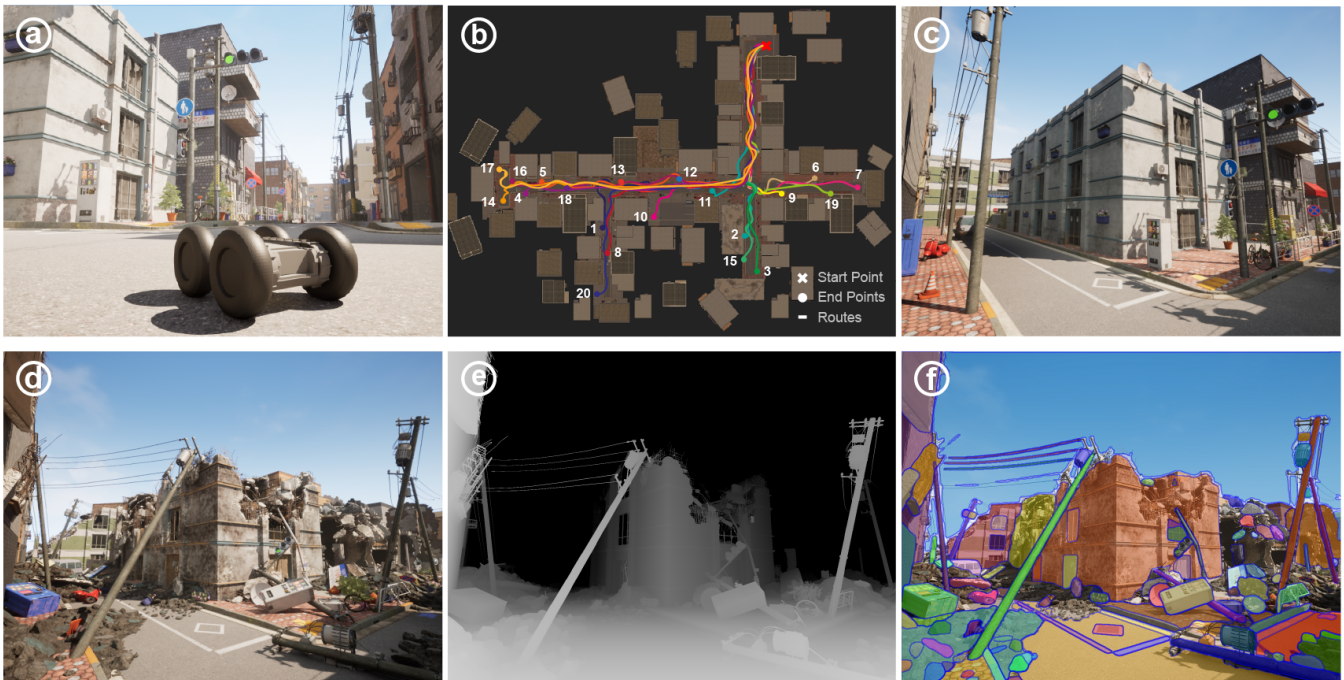
**Table 2:** Success Rates of DRL SLAM Path Planning and Accuracy of Image Segmentation edges for 4 typical out of 20 Targets

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%

**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 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





**Figure 7:** 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 travelling. 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.

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.8. 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 multifactorial 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.

**Limitation** Our work, while pioneering in many aspects, inevitably has some constraints. Primarily, our simulation predom-

inantly models the seismic repercussions on edifices, but the intricate dynamics of foundation flexing and land movements during such seismic activities are abstracted. This means our model might not reflect the real-world nuances of how foundations crumble or shift during earthquakes. Moreover, the parameters representing building materials in our simulations are extracted from idealistic configurations. This could raise concerns when we think of the myriad of building types, architectures, and materials that are influenced by cultural, geographical, and technological diversities. Lastly, our virtual scenario does not factor in various environmental conditions like fluctuating lighting, or obstructive elements such as smoke and dust. These elements could critically influence the behaviour and performance of AI algorithms within such scenarios.

**Future Works** We aim to address these limitations to enhance RESEnv’s realism. A planned extension involves simulating environmental factors, emphasising dynamic lighting, smoke, dust, and their interactions. This will amplify the authenticity, especially in disaster aftermath scenarios. Additionally, we’ll incorporate diverse building models and materials, possibly using perceptual similarity metrics, to improve realism. We’ll also extend our approach to other disasters, including floods and hurricanes. The ultimate goal is to cultivate a challenging environment, enhancing the robustness of AI models trained within RESEnv.

## References

- [BC20] BOURHIM E. M., 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* (2020), Springer, pp. 955–966. 1
- [BEY22] BADWI I. M., ELLAITHY H. M., YOUSSEF H. E.: 3d-gis parametric modelling for virtual urban simulation using cityengine. *Annals of GIS* 28, 3 (2022), 325–341. 3
- [BHB\*19] BISCHKE B., HELBER P., BRUGMAN S., BASAR E., ZHAO Z., LARSON M. A., POGORELOV K.: The multimedia satellite task at mediaeval 2019. In *MediaEval* (2019). 2
- [Cha19] CHAILLOU S.: *Ai & architecture*, 2019. 3
- [CS19] COSTA M. M., SILVA M. F.: A survey on path planning algorithms for mobile robots. In *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)* (2019), IEEE, pp. 1–7. 2
- [GBB\*22] GREFF K., BELLETTI F., BEYER L., DOERSCH C., DU Y., DUCKWORTH D., FLEET D. J., 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), pp. 3749–3761. 1
- [GID22] GIARETTON M., INGHAM J. M., DIZHUR D.: Shaking table testing of two storey as-built and retrofitted scaled urm buildings. In *Structures* (2022), vol. 41, Elsevier, pp. 260–269. 6
- [GKS\*18] GRUNWALD C., KHALIL A. A., SCHAUFELBERGER B., RICCIARDI E. M., PELLECCIA C., DE IULIIS E., RIEDEL W.: Reliability of collapse simulation—comparing finite and applied element method at different levels. *Engineering Structures* 176 (2018), 265–278. 2
- [Hor11] HORI M.: *Introduction to computational earthquake engineering*. World Scientific, 2011. 2
- [Jos22] JOSEPH I.-L.: The effect of natural disaster on economic growth: Evidence from a major earthquake in haiti. *World Development* 159 (2022), 106053. 1
- [KMR\*23] KIRILLOV A., MINTUN E., RAVI N., MAO H., ROLLAND C., GUSTAFSON L., XIAO T., WHITEHEAD S., BERG A. C., LO W.-Y., ET AL.: Segment anything. *arXiv preprint arXiv:2304.02643* (2023). 7
- [McE22] MCENTEE P.: Simpson strong-tie research: The tie gilb lab, Jun 2022. URL: <https://seblog.strongtie.com/2012/05/simpson-strong-tie-research-the-tye-gilb-lab/>. 2
- [McK11] MCKENNA F.: Opensees: a framework for earthquake engineering simulation. *Computing in Science & Engineering* 13, 4 (2011), 58–66. 2
- [Mic23] MICHAELSON R.: Earthquake in Turkey and Syria kills thousands and devastates cities. *The Guardian* (Feb. 2023). URL: <https://www.theguardian.com/world/2023/feb/06/turkey-earthquake-2023-gaziantep>. 6
- [MMNM18] MOSELEY B., MARKHAM A., NISSEN-MEYER T.: Fast approximate simulation of seismic waves with deep learning. *arXiv preprint arXiv:1807.06873* (2018). 4
- [MNM20] MOSELEY B., NISSEN-MEYER T., MARKHAM A.: Deep learning for fast simulation of seismic waves in complex media. *Solid Earth* 11, 4 (2020), 1527–1549. 4
- [MP22] MATIN S. S., PRADHAN B.: Challenges and limitations of earthquake-induced building damage mapping techniques using remote sensing images—a systematic review. *Geocarto International* 37, 21 (2022), 6186–6212. 2
- [MPM\*21] 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. *Earthquake Spectra* 37, 2 (2021), 736–761. 2
- [MPS\*19] 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* (2019), Springer, pp. 15–26. 1
- [noa] Seismic testing of seismic intensity 6. URL: <https://www.youtube.com/watch?v=ddr5KT1nX7M>. 6
- [NZ20] NAZAROVA A. V., ZHAI M.: The application of multi-agent robotic systems for earthquake rescue. *Robotics: Industry 4.0 Issues & New Intelligent Control Paradigms* (2020), 133–146. 1
- [NZKN19] NIRoui F., ZHANG K., KASHINO Z., NEJAT G.: Deep reinforcement learning robot for search and rescue applications: Exploration in unknown cluttered environments. *IEEE Robotics and Automation Letters* 4, 2 (2019), 610–617. 2
- [OH22] 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), pp. 1–9. 7
- [Ole22] OLESON T.: Faking quakes at full scale: Giant shake tables simulate earthquakes to make buildings safer, Jun 2022. URL: <https://www.earthmagazine.org/article/faking-quakes-full-scale-giant-shake-tables-simulate-earthquakes-make-buildings-safer/>. 2
- [PCFN07] 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* (2007), Springer, pp. 491–498. 2
- [QTP\*20] QUERALTA J. P., TAIPALMAA J., PULLINEN B. C., SARKER V. K., GIA T. N., TENHUNEN H., GABBOUJ M., RAITOHARJU J., WESTERLUND T.: Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision. *Ieee Access* 8 (2020), 191617–191643. 2
- [SFN\*22] SHAW B. E., FRY B., NICOL A., HOWELL A., GERSTENBERGER M.: An earthquake simulator for new zealand. *Bulletin of the Seismological Society of America* 112, 2 (2022), 763–778. 2
- [SR10] 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), pp. 1647–1648. 2
- [SS06] SCHNEIDER P. J., SCHAUER B. A.: Hazus—its development and its future. *Natural Hazards Review* 7, 2 (2006), 40–44. 2
- [SSAH16] SAHIN A., SISMAN R., ASKAN A., HORI M.: Development of integrated earthquake simulation system for istanbul. *Earth, Planets and Space* 68, 1 (2016), 1–21. 2
- [Tea23] TEAM V. J.: Turkey earthquake: Before and after pictures show extent of destruction. *BBC News* (Feb. 2023). URL: <https://www.bbc.com/news/world-europe-64544998>. 6
- [WP20] WANG L.-L., PAN L.-X.: Research on sbmpc algorithm for path planning of rescue and detection robot. *Discrete Dynamics in Nature and Society* 2020 (2020), 1–11. 2
- [WZY\*20] 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. *Intelligent Service Robotics* 13 (2020), 263–272. 7
- [XWLJ20] XU Z., WU Y., LU X., JIN X.: Photo-realistic visualization of seismic dynamic responses of urban building clusters based on oblique aerial photography. *Advanced engineering informatics* 43 (2020), 101025. 2

- [XZL\*18] 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. *Automation in Construction* 90 (2018), 9–22. [2](#)
- [YFMA07] YU Z., FERGUSON J., MCMECHAN G., ANNO P.: Wavelet-radon domain dealiasing and interpolation of seismic data. *Geophysics* 72, 2 (2007), V41–V49. [5](#)
- [ZLX\*20] ZHANG X., LAI J., XU D., LI H., FU M.: 2d lidar-based slam and path planning for indoor rescue using mobile robots. *Journal of Advanced Transportation* 2020 (2020), 1–14. [2](#)