A Comparative Study of Deep Learning and Evac4BIM in Building Evacuation Design

Patrick Berggold[1], Ann-Kristin Dugstad[1,2] and Mohab Hassaan[1]
[1]Chair of Computational Modeling and Simulation, Technical University of Munich, Arcisstraße 21, 80333 Munich, Germany
[2]Division of Fire Safety Engineering, Lund University, John Ericssons väg 1, 22363 Lund, Sweden
E-mail(s): patrick.berggold@tum.de, ann-kristin.dugstad@brand.lth.se, mohab.hassaan@tum.de

Abstract: The integration of pedestrian simulations into the design process of a building is an inevitable step to improve its evacuation performance. This becomes particularly important during the initial design stages, when countless building configurations are explored and evaluated through interdisciplinary discussions. However, conventional simulators typically operate as stand-alone tools, necessitating several manual export and conversion steps and time-consuming runtimes. To address these challenges, recent developments have led to the integration of Building Information Modeling (BIM) with the widely-used pedestrian simulator Pathfinder in an open source tool called Evac4BIM. In this study, we explore the potential for further accelerating the generation and integration of evacuation simulation results through neural networks, trained on a comprehensive, synthetic dataset to enable real-time predictions. Subsequently, we analyze and compare the applicability of Evac4BIM and our deep learning approach in terms of practicality and accuracy.

Keywords: Pedestrian simulation, Building information modeling, Deep learning, Evacuation performance

Erschienen in Tagungsband 35. Forum Bauinformatik 2024, Hamburg, Deutschland, DOI: will/be/assigned/by/editor

1 Introduction

The simulation of building evacuations is vital for enhancing building design and preparing for disaster scenarios. However, these simulations are oftentimes computationally intensive, especially when using microscopic, agent-based models that make it unfeasible to investigate the evacuation performance of every single building variant that emerges throughout the planning and design stages of a building [1], [2]. To mitigate this computational burden, researchers have recently explored neural networks trained on simulation results to produce fast predictions that enable real-time evacuation assessment, thus aiding the decision-making during building design.

Moreover, Building Information Modeling (BIM) has gained widespread popularity due to its ability to provide a comprehensive digital representation of a building. Such a digital model facilitates the sharing of information among various stakeholders, making it easier to explore different design options from a multidisciplinary perspective. As a result, BIM enables more efficient collaboration and decision-making during the design and planning stages [3]. BIM Models can be exported in standard file formats such as the Industry Foundation Classes (IFC). This capability is particularly advantageous for investigating evacuation performance, as IFC files can be imported into evacuation simulators. Recently, there have been significant advancements in integrating IFC with evacuation tools and employing deep learning approaches using floorplan or layout inputs.

In this study, we aim to investigate and compare the efficacy of a traditional simulation-based tool and a deep learning approach for predicting evacuation performance. Specifically, we develop a parametric model using a BIM-authoring tool to generate a comprehensive synthetic dataset for training our neural network. Distinct from previous studies, our approach relies solely on image-related inputs, encoding occupant information in input tensors without additional use case-specific inputs. In addition, we examine a novel IFC-based evacuation tool, namely Evac4BIM [4], which employs Pathfinder [5] to produce evacuation simulation results. We conduct a comparative analysis of these two methodologies, assessing their practicality and effectiveness in predicting evacuation performance. Through this investigation, we aim to provide insights into the potential advantages and limitations of each approach, contributing to the optimization of building design and safety planning.

2 State of the art

2.1 Evacuation modeling

Three primary methodologies are utilized to model evacuation behavior: macroscopic, microscopic, and mesoscopic modeling [6]. A macroscopic modeling approach simulates the movement of the whole crowd as a continuous flow, often by subdividing the area into cells and defining the in- and output flow of pedestrians per cell, similar to how eularian transport is modeled in particle flow models. A microscopic modeling approach focuses on the individual itself, by simulating the movement of individual agents with specific characteristics and behaviours. This approach is comparable to Lagrangian transport in transport models [2]. While the microscopic approach - the models employing this approach are often so called agent-based models - is more suitable for detailed studies, it also is more computationally intensive. Alternatively, a solution inbetween the macroscopic and microscopic model can be employed, the so called mesoscopic modeling approach, which represent groups of people with similar characteristics as single entities.

2.2 BIM-based evacuation tools

BIM is a methodology widely utilized in the architecture and construction industry to create and manage building models enriched with comprehensive information throughout the entire lifecycle of a structure [3]. A common format to store those three dimensional information rich models is IFC. BIM-models in IFC format can then be used in evacuation dynamics research to analyse evacuation dynamics in buildings. Pathfinder is one evacuation dynamics simulator which can import geometrical features such as walls, floors and stairs from IFC files [7]. However, those IFC files must be enriched manually by the user. [4] aimed to fill this interface gap between Revit and Pathfinder by developing the add-in Evac4BIM, which allows to export simulation-relevant BIM data such as alarm times and pre-evacuation times from Revit and import of Pathfinder's simulation results such as evacuation times and occupancy flow rates into Revit. All parameters which are included in this Add-in are displayed in Table 1.

Table 1: Evac4BIM's analysis on the Simulation Input and Output Parameters from Pathfinder [8].

Simulation input	**Simulation output**
Alarm time	Simulation Brief
Pre evacuation time	Initial Occupant Number
Number of occupants	Evacuation Time
Occupant load	Overall Evacuation Time
Peak number of occupants	Occupant History
Building occupancy day/night	Travel distances
Component status	First occupant in
Required door flow rate	Last occupant out
Occupant profiles	Total use
Admitted profiles	Door Flow rate History
	Average occ. flow rate

2.3 Machine learning in evacuation planning

Machine learning, and particularly deep learning, are increasingly used to enhance building evacuation predictions – for instance by estimating total evacuation time (TET). Recent studies have demonstrated the ability to predict TET from vector inputs that encompass environment or crowd configuration parameters, using different regression methods [9] or an artificial neural network (ANN) [10]. Similarly, Testa et al. [11] estimate per-room evacuation time from vector input with an ANN, and aggregate these results for the overall TET prediction. Other research has focused on converting floorplans into images to capture the full complexity of building geometries. For instance, [1] predict TET from train station platforms by combining a convolutional neural network (CNN) with supplementary vector inputs describing the layout geometry. Extending this methodology, further studies simultaneously predict TET and other evacuation-related metrics to aid in building design. For example, Berggold et al. [12] utilize a single neural network to predict TET and density over time from image and vector inputs. Similarly, Nourkojouri et al. [13] separate density and TET predictions through a neural network and Extreme Gradient Boosting, respectively. Further works integrate trajectory prediction into the design process, from which TET can be derived [14].
In contrast to these approaches, we propose a general, streamlined neural network model that predicts TET exclusively from image-based inputs. This model eliminates the need for supplementary vector inputs, which are often specific to particular use cases and architectural descriptions, thus simplifying the prediction process and enhancing its applicability across diverse building designs.

3 Methodology

We aim at predicting evacuation time via a deep neural network directly from the BIM model, provided initialization information. Thus, in the following, we outline our proposed methodology in Figure 1, incorporating the dataset generation from a parametric BIM model, its associated export as semantic map, and the neural network training. As a result, once the network is trained, it can be called directly called from the Dynamo script for an instant evacuation time prediction.

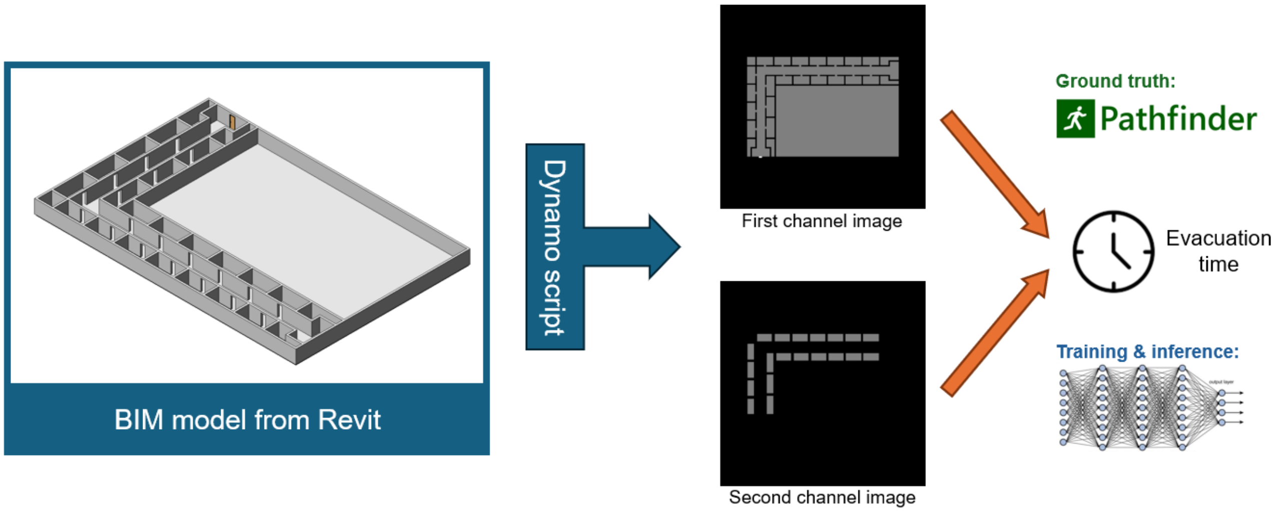

Figure 1: Our methodology, including the BIM model generation based on the parameters in Table 2, its image export via Dynamo, and the resulting tensor map with two channels. The first image channel encompasses walkable areas visualized in gray, the non-walkable areas in black, and the destination in white. The second channel resembles per-room densities at initialization.

3.1 Dataset generation

To train our neural network, we utilize a BIM model that is based on the input parameters in Table 2. Specifically, we present 256 samples, where each represents a unique combination of those input parameters. We use two options for the hallway width, four options for the room length, and four options both for site width and length. Additionally, we incorporate eight options for the initial number of occupants per room, ranging from single agents to dense crowds. Consequently, we investigate high-risk evacuation scenarios, where bottlenecks may form during an emergency situation with only one exit.

Table 2: The input parameters of the BIM model, resulting in 256 unique combinations.

	Number of Configurations	**Values**
Hallway Width (m)	2	[2, 3]
Room Length (m)	4	[5, 6, 7, 8]
Site dimensions [width $\times$ length] (m$\times$m)	4	$[32\times32, 64\times64, 48\times64, 32\times48]$
Number of Occupants per room	8	[1, 4, 7, 10, 13, 16, 19, 22]

The models were generated using Dynamo, which is Revit's integrated parametric modeling system. The Dynamo script creates a single-story, parametric model given the parameters detailed in Table 2, as depicted in Figure 1. Additionally, the script includes the number of occupants as an IFC parameter and generates an exit door. Notably, the number of occupants per room is constant across each dataset sample.

After model generation inside Revit, we export the model as an IFC file and run evacuation simulations using Pathfinder, as presented in Figure 2. As a result, we obtain the evacuation time for each dataset configuration. Intuitively, rising occupant numbers correspond to significantly higher evacuation times due to congestion effects both at the door of each room and at the final exit of the building.

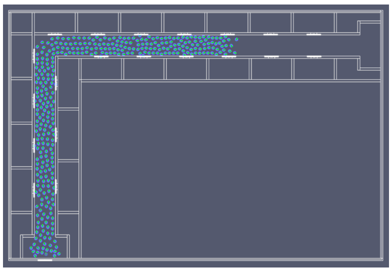

Figure 2: An evacuation simulation with only one exit inside Pathfinder.

3.2 Deep neural network

Different from previous works, we propose an image-only approach, without adding any supplementary parameters to the network. The neural network architecture is visualized in Figure 3. Initially, we process both channels individually through convolutional blocks, extracting geometry and density information, respectively. One convolutional block (*C-Block*) is made up of a batch norm and ReLU activation, with a preceding two-dimensional convolution that includes different input and output channels, as well as kernel size and stride. Subsequently, the extracted features are flattened and concatenated, and a two-layer MLP merges the vectors for the final prediction. The hidden layers in the MLP have the input dimensions 128 and 64, respectively, with a one-dimensional output that is the evacuation time prediction.

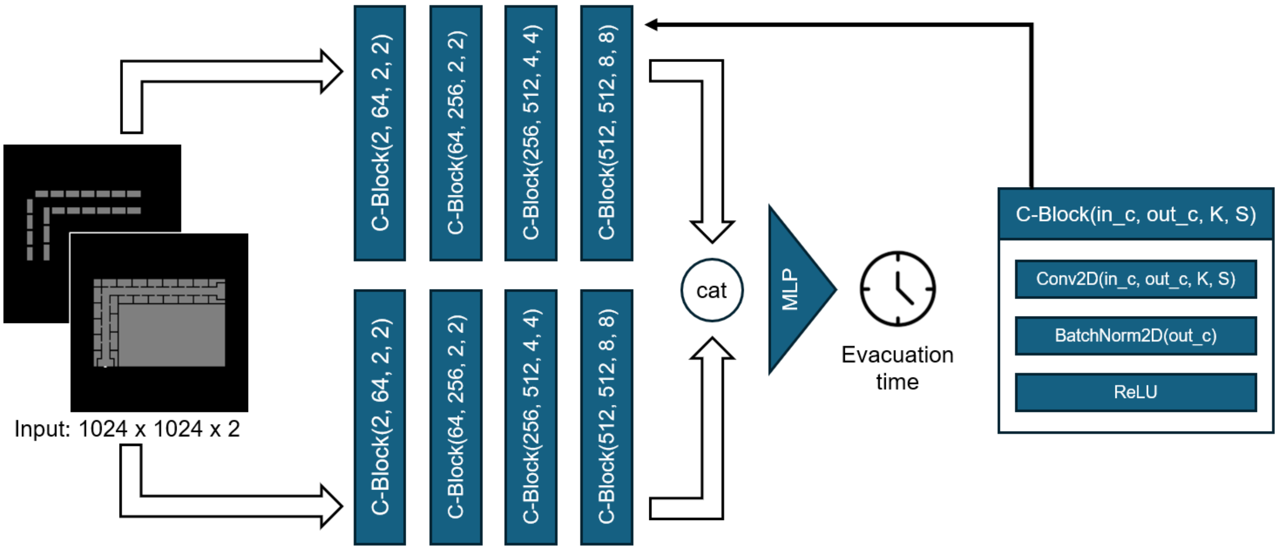

Figure 3: The neural network architecture.

3.3 Training results

We train the neural network by splitting our dataset into 70% training samples, while 15% are used both for validation and for testing. The resulting plots of training and validation are displayed in Figure 4, utilizing the Mean-Squared-Error (MSE) as loss function to effectively reduce the number of outlier predictions. Furthermore, in the following, we present the mean absolute and the relative errors on the test set, which are more intuitive than MSE.

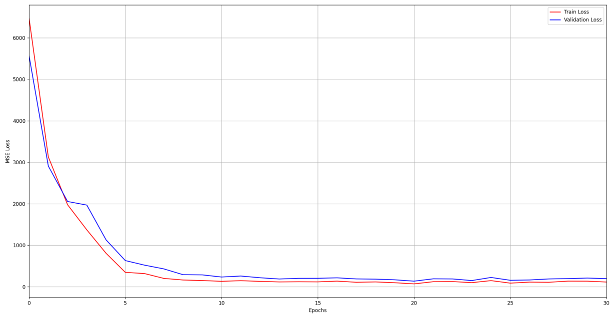

Figure 4: The result of our training process, including training (red line) and validation (blue line) results.

The evaluation of the neural network follows on the test set. We assess the results in terms of the absolute error (AE_i) between true evacuation time $T_{evac,i}$ from the simulation, and neural network prediction $\hat{T}_{evac,i}$. Specifically, we provide the mean (MAE), the relative error (RE) and the standard deviation (SD) of the absolute error:

$$\mathrm{MAE} = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \left| T_{evac,i} - \hat{T}_{evac,i} \right| = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \mathrm{AE}_i = 8.31\,\mathrm{sec}$$

$$\mathrm{RE} = \frac{1}{N_{test}} \sum_{i=1}^{N_{test}} \frac{\mathrm{AE}_i}{T_{evac,i}} = 7.38\% \;\text{ and }\; \mathrm{SD} = \sqrt{\frac{1}{N_{test}} \sum_{i=1}^{N_{test}} (\mathrm{AE}_i - \mathrm{MAE})^2} = 9.93\,\mathrm{sec}$$

Based on these results, it is evident that our deep learning approach serves as an effective approximation, aligning with its intended role as a supportive tool in the design process. Nonetheless, in edge cases where higher precision is required, a real simulation remains necessary. In addition, the current dataset size is constrained due to the time-intensive nature of its creation. To enhance the applicability of this approach in real-world scenarios, expanding the dataset is essential.

4 Comparison between deep learning and Evac4BIM

In the following section, we compare the practicality of using our straightforward Dynamo script and the neural network, to Evac4BIM for generating simulation results.
With respect to the deep learning approach, as detailed in Section 3, each dataset sample requires an IFC export from the Dynamo script, including the building's geometry and its simulation parameters, namely the number of occupants per room. Subsequently, the IFC files can be imported into Pathfinder to execute the evacuation simulations – the results of which constitute our synthetic dataset, utilized for training the neural network depicted in Figure 3. The major advantage of this method is the neural network's inference time, which is less than one second, thereby enabling real-time evacuation assessments. The critical limitation is the reliability of the network's performance, which is intrinsically linked to the quality and comprehensiveness of the dataset. To ensure the practical applicability across diverse scenarios and building layouts, the dataset must be substantially expanded to facilitate interpolation on previously unseen samples.

In comparison, the usage of Evac4BIM involves some additional, manual steps, such as initializing the geometry elements, adding occupant profiles, assigning room categories, or pairing doors to their adjacent rooms, among others. Subsequently, the IFC file can be exported to be used as input to Pathfinder. While Evac4BIM's approach is thorough, for the purpose of generating this dataset, it is quite excessive; many of the steps are not required to generate our dataset. As an alternative, we used the Dynamo script method described in Section 3. The script is used to assign all the required parameters, resulting in lower effort. Comparing the average time taken by both approaches to generate one IFC file, we report that the Dynamo script method is significantly faster than the usage of Evac4BIM, taking us 17 seconds and 109 seconds, respectively.

5 Conclusion

In this study, we compared a traditional simulation-based method for predicting evacuation performance in buildings using Revit, Pathfinder, and the Revit Add-in Evac4BIM with our deep learning approach. By employing both the traditional method and a custom Dynamo script, we generated a synthetic dataset to train a neural network that predicts TET based solely on image-related inputs. This neural network offers rapid inference, making it suitable for real-time evacuation assessments during the building design process.

Our comparative analysis revealed that, while the neural network's performance is highly dependent on the dataset's quality and comprehensiveness, it significantly reduces prediction time. Conversely, the IFC-based tool Evac4BIM, despite requiring extensive manual input, provides detailed and accurate simulation results. Additionally, generating IFC files using a Dynamo script proved to be significantly faster than using Evac4BIM, highlighting its practicality for large dataset creation.

In conclusion, the neural network approach offers quick and efficient evacuation predictions, whereas traditional tools like Evac4BIM deliver detailed and reliable results. Future research should focus on expanding and enhancing the training dataset for neural networks and exploring the integration of both approaches to improve building design and safety planning.

Acknowledgements

This research was supported by the project FORWARD, funded by the TUM Georg Nemetschek Institute. We would like to thank Enrico Ronchi for his guidance in leveraging this project.

References

[1] J. Clever, J. Abualdenien, R. K. Dubey, and A. Borrmann, "Predicting occupant evacuation times to improve building design", in *Proc. of European Conference on Product and Process Modeling 2022*, Sep. 2022.

[2] E. J. Charlesworth, A.-K. Dugstad, F. Fritsch, P. Jöckel, and F. Plöger, "Impact of lagrangian transport on lower-stratospheric transport timescales in a climate model", *Atmospheric Chemistry and Physics*, vol. 20, no. 23, pp. 15 227–15 245, 2020. DOI: 10.5194/acp-20-15227-2020. [Online]. Available: https://acp.copernicus.org/articles/20/15227/2020/.

[3] A. Borrmann, M. König, C. Koch, and J. Beetz, *Building Information Modeling: Technologische Grundlagen und industrielle Praxis*. Springer-Verlag, 2015.

[4] N. Yakhou, P. Thompson, A. Siddiqui, J. Abualdenien, and E. Ronchi, “The integration of building information modelling and fire evacuation models”, *Journal of Building Engineering*, vol. 63, p. 105 557, 2023, ISSN: 2352-7102. DOI: https://doi.org/10.1016/j.jobe.2022.105557. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352710222015637.

[5] T. Engineering, *Pathfinder*, Accessed: 2024-07-06, 2024. [Online]. Available: https://www.thunderheadeng.com/pathfinder/.

[6] A. Rohaert, N. Janfeshanaraghi, E. Kuligowski, and E. Ronchi, “The analysis of traffic data of wildfire evacuation: The case study of the 2020 glass fire”, *Fire Safety Journal*, vol. 141, p. 103 909, 2023, ISSN: 0379-7112. DOI: https://doi.org/10.1016/j.firesaf.2023.103909. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0379711223001777.

[7] T. Engineering, *Ifc import to pathfinder (and pyrosim)*, Accessed: 2024-07-06, 2024. [Online]. Available: https://support.thunderheadeng.com/tutorials/pathfinder/ifc-import/.

[8] N. Yakhou, P. Thompson, and E. Ronchi, “Development of the ’evac4bim’ add-in for revit”, in *Proceedings of the Fire and Evacuation Modeling Technical Conference (FEMTC)*, Accessed: 2024-07-06, Brno, Czech Republic, 2022. [Online]. Available: https://www.femtc.com/events/2022/d1-08-yakhou/.

[9] M. Abadeer, F. Ebeid, and S. Gorlatch, “Integration of machine learning methods into agent-based simulations for predicting evacuation time in disaster scenarios”, in *2022 3rd Asia Symposium on Signal Processing (ASSP)*, IEEE, 2022, pp. 71–81.

[10] K. Hu, S. Yoon, V. Pavlovic, P. Faloutsos, and M. Kapadia, “Predicting crowd egress and environment relationships to support building design optimization”, *Computers & Graphics*, vol. 88, pp. 83–96, 2020.

[11] E. Testa, R. C. Barros, and S. R. Musse, “Crowdest: A method for estimating (and not simulating) crowd evacuation parameters in generic environments”, *The Visual Computer*, vol. 35, pp. 1119–1130, 2019.

[12] P. Berggold, S. Nousias, R. K. Dubey, and A. Borrmann, “Towards predicting pedestrian evacuation time and density from floorplans using a vision transformer”, *Proc. of the 30th Int. Conference on Intelligent Computing in Engineering (EG-ICE)*, 2023.

[13] H. Nourkojouri, A. Nikkhah Dehnavi, S. Bahadori, and M. Tahsildoost, “Early design stage evaluation of architectural factors in fire emergency evacuation of the buildings using pix2pix and explainable xgboost model”, *Journal of Building Performance Simulation*, vol. 16, no. 4, pp. 415–433, 2023.

[14] P. Berggold, S. Nousias, and A. Borrmann, “Integrating pedestrian simulation into early building design: A deep learning approach for trajectory prediction”, in *i3CE 2024: 2024 ASCE International Conference on Computing in Civil Engineering*, Jul. 2024.