Kalpa Publications in Computing
Volume 22, 2025, Pages 558–569
Proceedings of The Sixth International Conference on Civil and Building Engineering Informatics

An LLM Agent–Based Approach for Calculating Floor Areas via Bim IFC Files

Xueying Zhu[1], and Jun Ma[2]

1) Ph.D. Student, Department of Urban Planning and Design, The University of Hong Kong, Hong Kong, SAR. Email: xueyingz@connect.hku.hk
2) Ph.D., Asst. Prof., Department of Urban Planning and Design, The University of Hong Kong, Hong Kong, SAR. Email: junma@hku.hk

Abstract: The accurate and efficient calculation of floor areas is a critical aspect of building design processes, yet it remains a challenging task due to the intricate rules and regulations that must be adhered to. Traditional methods, which rely on manual calculations, are not only time-consuming but also prone to human error, highlighting the need for an automated solution. Building Information Modeling (BIM) can support the automated solution by providing digital representations of building designs. The openBIM workflow enables collaborative work among various stakeholders, allowing them to share and integrate project information seamlessly without being locked into proprietary systems. While Industry Foundation Classes (IFC) files serve as a robust digital representation of BIM, facilitating data exchange across different software platforms, the complexity associated with processing these files poses a significant barrier to ordinary users. To address these gaps, we propose an innovative semi-automated floor area calculation system that leverages state-of-the-art Large Language Model (LLM) agent. This system is designed to efficiently and accurately extract relevant information from IFC files applying specific filtering conditions and then perform the necessary calculations to determine floor areas according to the required standards. Due to LLMs' characteristic of interacting using natural language, our approach significantly reduces the technical barriers faced by non-expert users, making the process of floor area calculation more accessible and user-friendly. We validate the proposed method through the practice of simplified floor area calculation of an academic building, offering a promising solution to streamline BIM workflows and enhance productivity in the architecture, engineering, and construction (AEC) industries.

Keywords: Floor Area Calculation, BIM, IFC Information Extraction, LLM Agent

1. INTRODUCTION

In the architectural design process, manual area calculation has long been a fundamental but labor-intensive task. Traditional methods involve detailed measurements and adherence to complex rules and standards. The rules governing area calculations are often intricate, involving specific criteria such as the inclusion or exclusion of certain spaces based on their function, height, and other attributes. Studies have shown that architects and engineers spend a considerable amount of time on these repetitive and non-creative tasks (Sang et al., 2009), which significantly drains their energy and impacts overall work efficiency. Therefore, there is an urgent need for automated solutions to assist in these tasks and improve productivity.

Several attempts have been made to develop automated systems for area calculation. Early

I. Hkust, J. Cheng and Y. Yantao (eds.), ICCBEI 2025 (Kalpa Publications in Computing, vol. 22), pp. 558–569

efforts focused on rule-based algorithms and expert systems, which could handle specific scenarios but lacked flexibility and scalability (Roanes-Lozano et al., 2000). Later research utilized Dynamo visual programming software to automate floor area calculation and compliance checking within BIM, streamlining the process and improving accuracy and efficiency (Reinhardt & Mathews, 2017). More recent approaches have explored machine learning techniques, particularly supervised learning, to improve the accuracy and adaptability of automated area calculations (Özer & Jacoby, 2022). Some researchers also use traditional programming called high-level implementable methods, such as *getTotalFloorArea()*, to accurately calculate and verify floor areas from building models for code compliance checking (Lee et al., 2023). However, these systems still face challenges in handling the variability and complexity of real-world data.

Building Information Modeling (BIM) is a digital representation of physical and functional characteristics of a building. It serves as a shared knowledge resource for information about a building, forming a reliable basis for decisions during its lifecycle. BIM models contain a wealth of information, including geometric data, material specifications, and performance metrics, which can be leveraged for various purposes, including automated area calculations. Industry Foundation Classes (IFC), a file format standard for BIM data exchange, enabling interoperability between different software platforms, can provide rich data for automated calculation of floor areas. However, the complexity of processing IFC files poses a significant barrier for non-specialist users, as existing software tools require advanced technical skills to operate effectively.

The emergence of Large Language Models (LLMs) offers a new avenue for addressing these challenges. LLMs, such as GPT series, are capable of understanding and generating human-like text, making them suitable for tasks that require natural language interaction (Achiam et al., 2023). In the AEC industry, LLMs have been applied to tasks such as document summarization, code generation, and data extraction (Zheng et al., 2023).

LLM agents as advanced AI systems capable of understanding and processing complex tasks, can handle multi-step reasoning problems and incorporate a reflection capability to ensure the accuracy of the results. They can perform complex operations and solve intricate problems by leveraging their capabilities of planning, finding information, remembering past interactions, and learning from them. This makes them particularly suitable for solving these intricate, double-hop or even multi-hop problems. Recent research has begun to investigate the use of LLM agents for some tasks. Various fields, such as the financial sector (Li et al., 2023) and chemical research (M. Bran et al., 2024), have already begun to harness these advanced technologies. However, their potential for automating area calculations in BIM has not been fully explored.

In this paper, we introduce a novel LLM agent specially designed to semi-automate floor area calculations. To enhance the problem-solving capabilities of LLM agents, techniques such as Chain-of-Thought (CoT) planning (Wei et al., 2022) and Reasoning and Acting (ReAct) frameworks (Yao et al., 2022) have been proposed. CoT planning involves breaking down complex tasks into smaller, manageable steps, allowing the LLM to think through the problem-solving process systematically. The ReAct framework further refines this process by enabling the LLM to reflect on its actions and adjust its strategy based on feedback. Our approach utilizes CoT planning, enabling the LLM to think through the problem-solving process step-by-step, like a human. This method allows the LLM to interpret and process IFC data effectively, extracting the necessary information to calculate floor areas accurately. We employ the ReAct framework to refine the agent's actions and ensure continuous improvement. The ReAct framework enables the agent to reflect on its previous actions and adjust its strategy, accordingly, enhancing the accuracy and reliability of the calculations. Besides, our system consists of two main tools: a calculator tool and a coding tool. The calculator tool performs the area calculations based on the extracted information using the coding tool to generate python codes to do the calculation. We tested our LLM agent on an academic building using a simplified method and achieved positive results. These findings provide new insights and methodologies for future work in automating building area calculations.

The main contributions of this paper are:

- Development of an LLM Agent: We have created a specialized LLM agent that can interpret and process IFC files to semi-automate floor area calculations. Our approach uses CoT planning to enhance the agent's problem-solving capabilities, ensuring accurate and reliable calculations. The ReAct framework allows the agent to continuously improve its performance through iterative refinement.
- Empirical Validation: We have demonstrated the effectiveness of our approach through testing on an academic building model in Revit, achieving positive results.

By addressing the limitations of manual area calculations and the complexities of IFC file processing, our LLM agent offers a promising solution to enhance productivity and efficiency in the architectural design process.

2. METHOD

2.1 Methods for Calculating Floor Area

According to the latest version of the "*Rules for Calculating Floor Area*" issued and implemented by the Ministry of Construction of the People's Republic of China, the main rules for calculating floor area can be divided into two categories: (a) the scope of areas that should be included in the floor area calculation, such as single-story and multi-story buildings with enclosing structures, and (b) the scope of areas that should not be included in the floor area calculation, such as outdoor ladders used for maintenance or fire safety. The scope of areas that should be included in the floor area calculation can further be divided into areas that are calculated in full, and areas that are calculated at 50% of their actual size. Detailed regulations are shown in Figure 1 below.

From Figure 1, we can see that the most significant factor affecting the area calculation coefficient is whether the floor height exceeds 2.2 meters, so in this study, we simplify the area calculation problem and focus only on the simplest scenario where the floor area is calculated in full for buildings with a story height of more than 2.2 meters. Additionally, more complex calculation scenarios, such as balconies and stairs that lie outside the main structural envelope, are not considered at this stage.

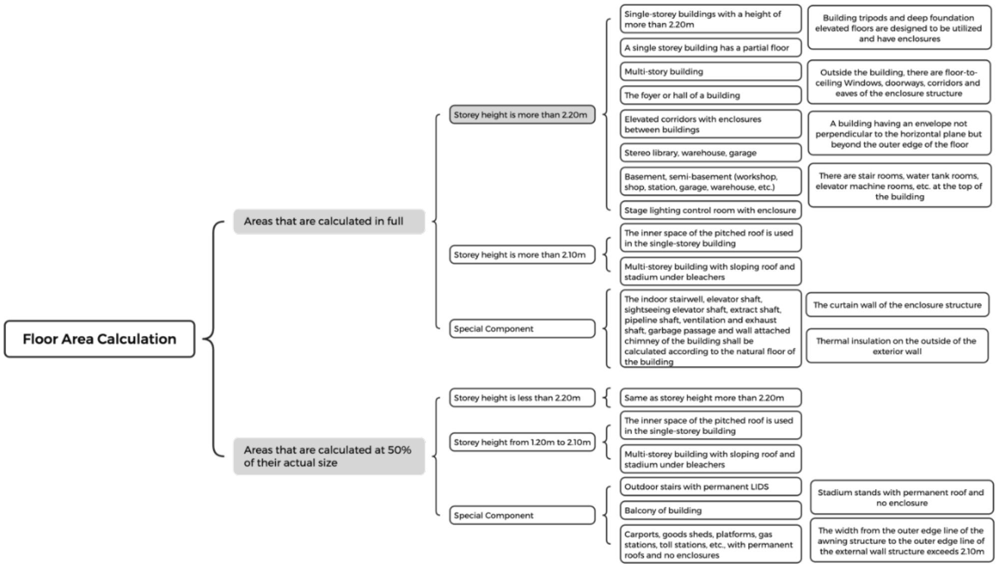


Figure 1. Detailed regulations for floor area calculation

2.2 Floor-Area-Calculation Agent

In this research we propose a specially designed LLM agent to solely solve the floor-area-calculation problems. Leveraging the power of LLMs, this agent efficiently extracts and processes data from IFC files, adhering to the latest regulations, such as the "*Rules for Calculating Floor Area*" issued by the Ministry of Construction of the People's Republic of China. The agent employs CoT planning and the ReAct framework to handle complex, multi-step reasoning tasks, ensuring accurate and reliable results. By simplifying the process and providing a user-friendly interface, the agent makes it accessible for non-specialist users, significantly reducing the time and effort required for manual calculations. This agent is particularly useful for scenarios where the floor height exceeds 2.2 meters, focusing on the most common and straightforward cases while laying the groundwork for more advanced applications in future developments. Figure 2 shows how the agent works.

(1) Planning

The planning component acts like the brain of the agent. It breaks down complex tasks into manageable steps and coordinates the execution of multi-step processes or queries using LLM. The planning component is essential for handling complex, multi-step user requests. At its core, CoT empowers the model to dissect complex problems into a series of more manageable, intermediate steps: First, to extract each floor's information in the given ifc file, which is commonly stored in "IFCSLAB" as this example:

```
#42252=IFCSLAB('0HupypOIXDjguK0hq5TsAf',#18,'\X2\697C677F\X0\:\X2\5BA45185573057
6A\X0\-120mm:310759',$,'\X2\697C677F\X0\:\X2\5BA451855730576A\X0\-
120mm',#42186,#42251,'310759',.FLOOR.)
```

"0HupypOIXDjguK0hq5TsAf" in this example is the IfcGUID of the first-floor slab in the building case. We manually collect each floor's IfcGUID and input them to LLM.

Second, to extract the height of elevations, stored in ifc files as the examples:

```
#102=IFCBUILDINGSTOREY('3Zu5Bv0LOHrPC10026FoQQ',#18,'1F',$,'\X2\68079AD8\X0\:\X
2\6B638D1F96F668079AD8\X0\',#101,$,'1F',.ELEMENT.,0.)
#106=IFCBUILDINGSTOREY('15Z0v90RiHrPC20026FoKR',#18,'2F',$,'\X2\68079AD8\X0\:\X2
\4E0A68075934\X0\',#105,$,'2F',.ELEMENT.,4500.)
```

The numbers "0", "4500" at the end of each line are the height of elevations, which the agent can use to calculate the height of each storey. Third, to calculate the height of storeys. In this case the height of the first storey should be 4500 – 0 = 4500 mm. Fourth, judge whether the height of storey is over 2.2 m. For example, 4.5 m is over 2.2 m. Fifth, to relate each storey and the floors they belong to. As we have already related each floor and its IfcGUID, we can relate each storey height and the floor's IfcGUID. In this case the first floor's IfcGUID is "0HupypOIXDjguK0hq5TsAf", with the storey height of 4.5m. Sixth, to calculate the floor area whose storey is over 2.2m using Calculator tool. The detailed calculation process is explained in the next section.

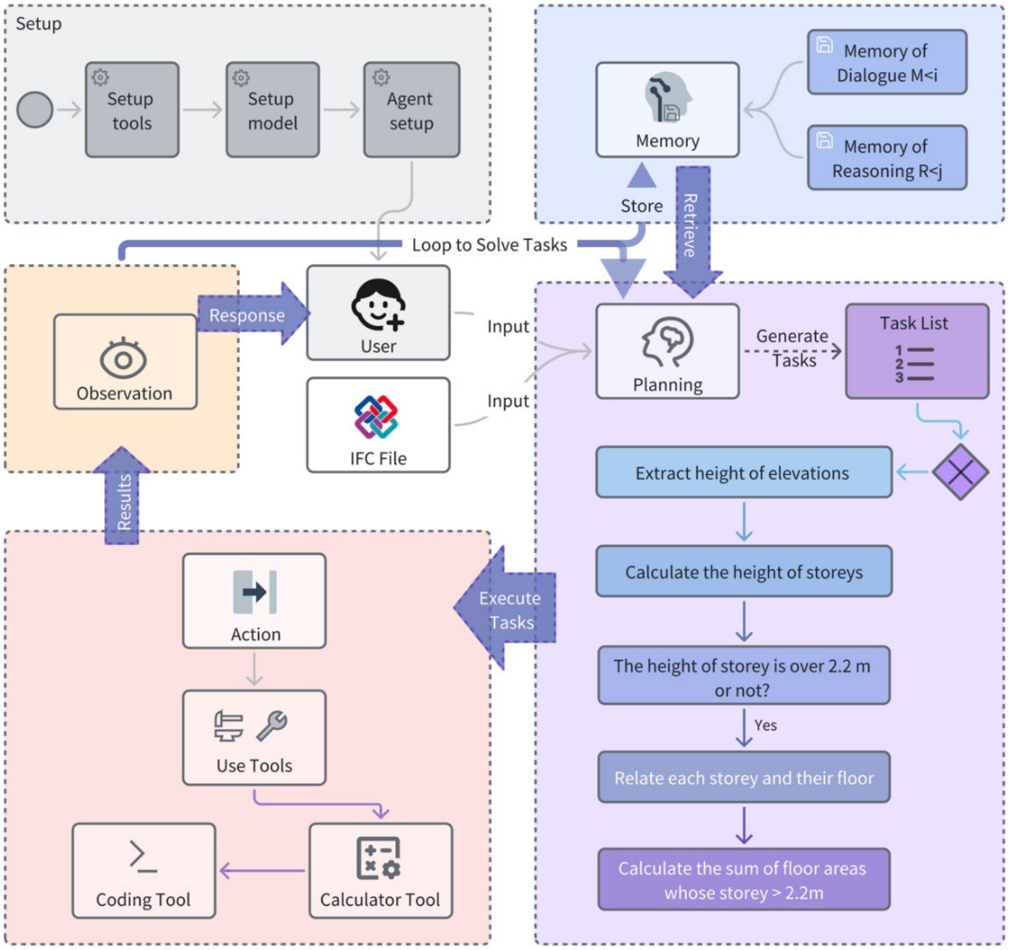


Figure 2. The framework of the proposed floor-area-calculation agent, which has four main components: planning, action, observation and memory.

(2) Action and Observation

Building upon this foundation, the Floor-Area-Calculation Agent incorporates the ReAct framework, which extends the model's capabilities beyond mere reasoning to include interactive decision-making. In the ReAct paradigm, the agent not only deliberates on the problem at hand but also takes decisive actions based on its analytical process. These actions may encompass querying additional information, making informed decisions, or engaging in clarifying dialogues to refine its understanding.

The Action component of the system incorporates two main tools to execute various tasks efficiently - Calculator Tool and Coding Tool. As illustrated in Figure 2, the system employs the Calculator Tool for performing numerical operations and the Coding Tool for generating and executing python codes using LLM to solve calculation tasks. These tools are strategically integrated to enhance the system's capabilities in processing complex queries that may require calculations or code generation. The presence of these diverse tools suggests a flexible and multifaceted approach to problem-solving, allowing the system to adapt to a wide range of user requirements and query types. The system can dynamically select and apply the appropriate tool based on the specific needs of each task, ensuring efficient and accurate execution of the planned actions.

Calculating Logic: Gauss's area formula is used to calculate the areas of floors:

$$S = 0.5 \times \left| \sum_{i=0}^{n-1} (x_i y_{i+1} - y_i x_{i+1}) \right|, \tag{1}$$

where $x_n = x_0$ and $y_n = y_0$, which means the first and last points are connected. Equation (1) can be used to calculate the area of any simple polygon (one that does not intersect itself). This formula is also known as the Shoelace formula.

The geom module from the ifcopenshell library is used, which provides functionality for geometric operations and manipulations on IFC models, enabling the creation, modification, and analysis of 3D building models. Algorithm 1 shows the vectorized implementation of Gauss's area formula using the NumPy library for array operations. Here, the np.roll function cyclically shifts the elements of an array, so that the first element of the y array is multiplied by the last element of the x array, and so on, thus implementing the cross-multiplication in the aforementioned summation. Algorithm 1 is shown as follows:

Algorithm 1

```
# Set geometric parameters
settings = ifcopenshell.geom.settings()
settings.set(settings.USE_WORLD_COORDS, True)

# Get floor geometry data
shape = ifcopenshell.geom.create_shape(settings, slab)

# Extract the vertices of the geometry data
vertices = shape.geometry.verts

def calculate_polygon_area(vertices):
    """
    Calculate the area of a polygon, assuming it is a polygon in a 2D plane.
    """
    x = vertices[::3]    # Get the X coordinates (Every three vertices are X, Y, Z.)
    y = vertices[1::3]    # Get the Y coordinates

    # Use the polygon area formula
    return 0.5 * np.abs(np.dot(x, np.roll(y, 1)) - np.dot(y, np.roll(x, 1)))

# Calculate floor area
area = calculate_polygon_area(vertices)
print(f" Floor area: {area} square meters ")
```

Figure 3 illustrates how a polygon floor slab is divided into triangles, with its area calculated using Gauss's area formula.

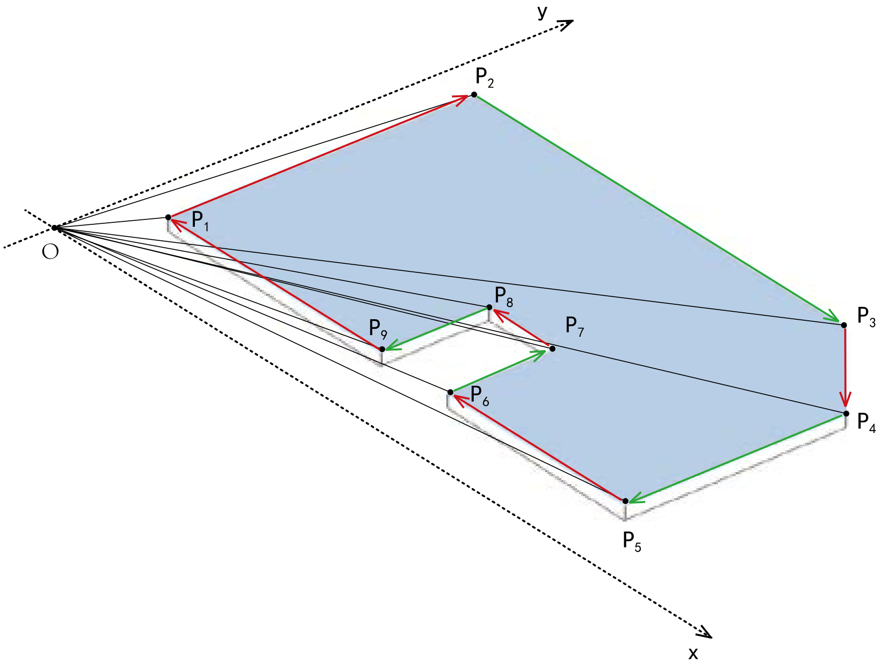


Figure 3. Triangle form of Gauss's area formula.
The color of the edges indicate, which triangle area is positive (green) and negative (red) respectively.

As for the Observation component, the primary purpose of it is to gather and analyze data from the environment or the task at hand. This data can include various types of information, such as input data, intermediate results, and feedback from previous actions. By thoroughly observing the current state, the agent can make informed decisions and adjust its strategy as needed until finally get the perfect answer.

(3) Memory

The Memory module is divided into two key sub-components: (a) Memory of Dialogue: This component stores the history of the conversation between the user and the AI system. (b) Memory of Reasoning: This sub-component retains the system's step-by-step reasoning process from previous interactions. The Memory module interacts with the Planning component through two processes: Store, which allows new information and reasoning steps to be added to the memory for future reference in the LLM's prompt input, and Retrieve, which enables the system to access stored information when planning new actions or responding to user queries. By maintaining this dual memory structure, the system can provide context-aware responses, build upon previous interactions, and apply learned reasoning strategies to new problems.

3. CASE STUDY

To further illustrate the effectiveness and versatility of our proposed framework, we conducted a case study involving the development of an academic building.

Figure 4 is the axonometric drawing of the academic building. The model consists of 3 floors. Each floor has been meticulously designed to accommodate a variety of room types with specific educational purposes. The overall height of the building stands at 12.3 meters, with the first floor measuring 4.5 meters, and both the second and third floors standing at 3.9 meters each. The skylight in the middle of the roof ensures optimal natural lighting and ventilation throughout the building. The meticulous planning of each floor's dimensions reflects a thoughtful approach to creating an environment conducive to learning and collaboration. Whether it be for lectures, seminars, or individual study sessions, this academic building is poised to serve as a hub of intellectual activity. As explained in Section 2.1, in this case study we only consider the calculations to the main floor slab areas and do not consider special areas such as balconies and stairs that lie outside the main structural envelope. Future in-depth studies will address these more complex scenarios.

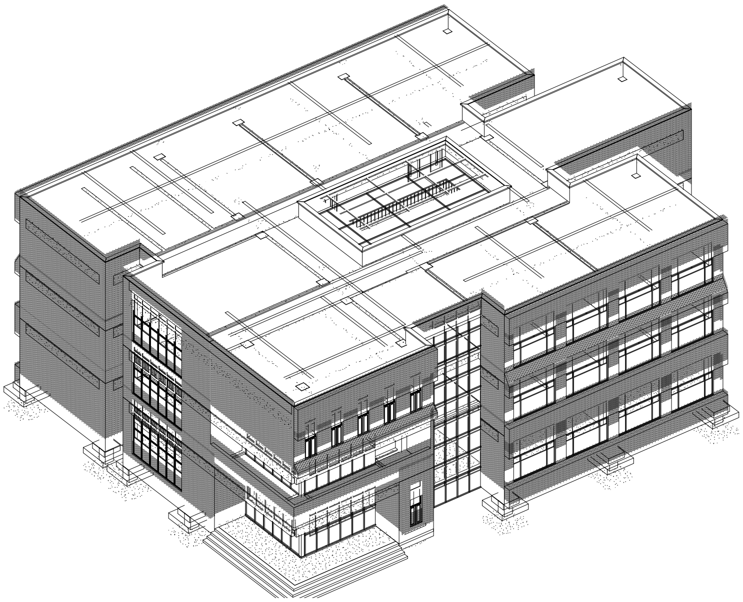
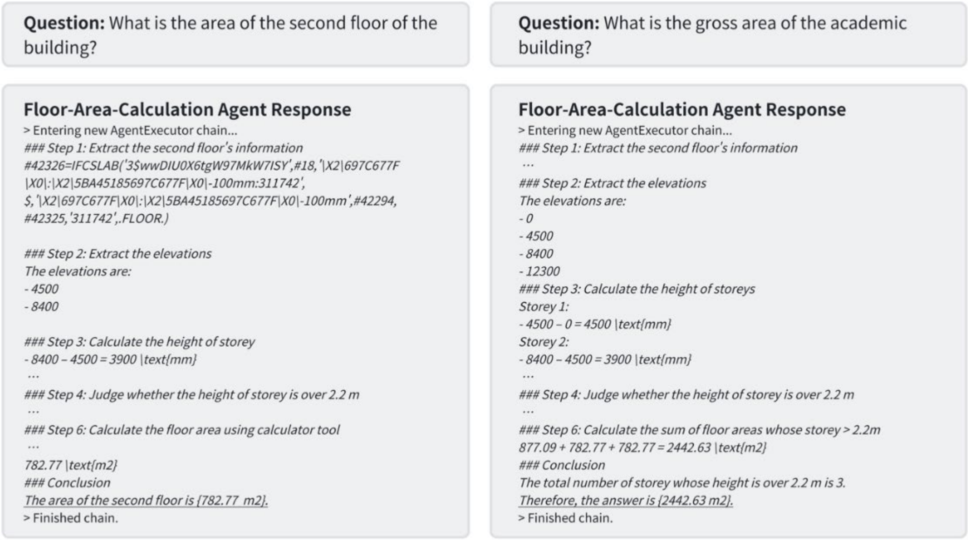


Figure 4. Axonometric drawing of the academic building, northeast view.

We asked the floor-area-calculation agent about the areas of each floor separately. The results are shown in Figure 5. For both single floor and gross area, the agent gave correct answers in the simplified case. In this case of the study, we only consider the simplest situation that is whether the storey height exceeds 2.2 meters. If it does, we calculate the full floor plate area for that level, without considering other factors that might affect the total floor area. We will delve deeper into these other factors in future research.



Question: What is the area of the second floor of the building?

Floor-Area-Calculation Agent Response

```
> Entering new AgentExecutor chain...
### Step 1: Extract the second floor's information
#42326=IFCSLAB('3$wwDIU0X6tgW97MkW7ISY',#18,'\X2\697C677F
\X0\:\X2\5BA45185697C677F\X0\-100mm:311742',
$,'\X2\697C677F\X0\:\X2\5BA45185697C677F\X0\-100mm',#42294,
#42325,'311742',.FLOOR.)

### Step 2: Extract the elevations
The elevations are:
- 4500
- 8400

### Step 3: Calculate the height of storey
- 8400 – 4500 = 3900 \text{mm}
 ...
### Step 4: Judge whether the height of storey is over 2.2 m
 ...
### Step 6: Calculate the floor area using calculator tool
 ...
782.77 \text{m2}
### Conclusion
The area of the second floor is {782.77  m2}.
> Finished chain.
```

Question: What is the gross area of the academic building?

Floor-Area-Calculation Agent Response

```
> Entering new AgentExecutor chain...
### Step 1: Extract the second floor's information
 ...
### Step 2: Extract the elevations
The elevations are:
- 0
- 4500
- 8400
- 12300
### Step 3: Calculate the height of storeys
Storey 1:
- 4500 – 0 = 4500 \text{mm}
Storey 2:
- 8400 – 4500 = 3900 \text{mm}
 ...
### Step 4: Judge whether the height of storey is over 2.2 m
 ...
### Step 6: Calculate the sum of floor areas whose storey > 2.2m
877.09 + 782.77 + 782.77 = 2442.63 \text{m2}
### Conclusion
The total number of storey whose height is over 2.2 m is 3.
Therefore, the answer is {2442.63 m2}.
> Finished chain.
```

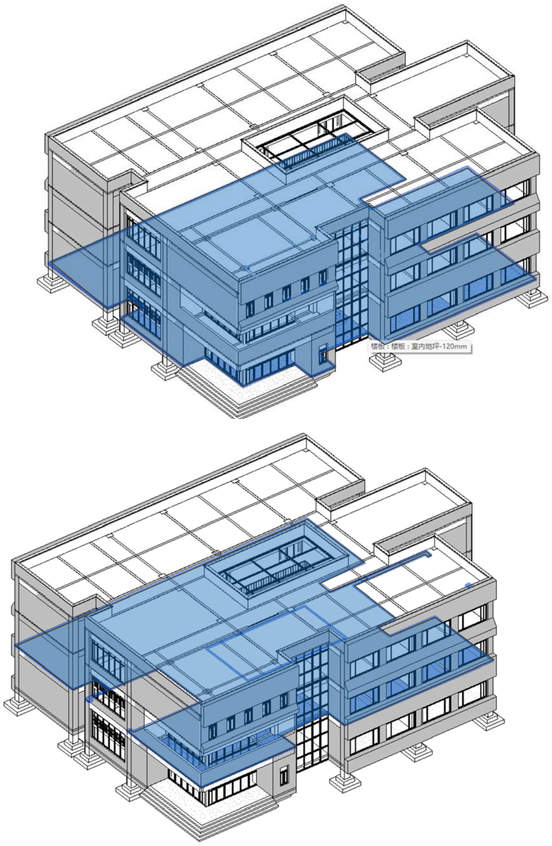
Figure 5. The results of two questions about the floor area and gross area of the academic building separately.

Table 1 presents the test results of the floor-area-calculation agent across different floors. For the first floor (1F), the actual area measured 868.73 square meters, while the calculated area was slightly higher at 877.09 square meters, resulting in an error rate of 0.96%. On the second floor (2F) and third floor (3F), both had identical actual areas of 796.19 square meters, with calculated areas of 782.77 square meters, leading to an error rate of 1.69% for both floors. Overall, when considering all floors combined, the total actual area was 2461.11 square meters, compared to a calculated area of 2442.63 square meters, yielding a total error rate of 0.75%. At this stage, the algorithm is still quite simplified, which leads to a certain degree of error in the calculation results. From the test results, the error rates are greater for the second and third floors, which may be due to the irregular shapes caused by the cutouts for the stairwells on these levels. The next steps in the research will involve a deeper investigation into the causes of these errors and further refinement of the floor area calculation method.

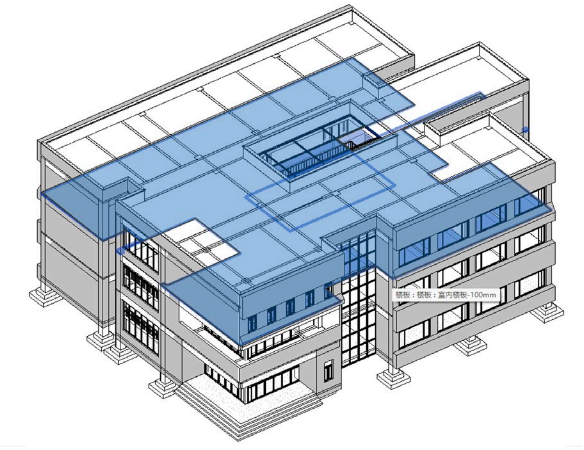
Table 1. Test results of the floor-area-calculation agent

	Ground Truth Area (m^2)	**Calculated Area (m^2)**	**Error Rate (%)**
1 F	868.73	877.09	0.96
2 F	796.19	782.77	1.69
3 F	796.19	782.77	1.69
Total	2461.11	2442.63	0.75

Figure 6 (a), (b) and (c) respectively show the shapes of the first, second, and third floor slabs of this academic building, to help readers better understand the case. Areas with special calculation rules, such as balconies, are not specifically considered in half size in this study.



(a) The first-floor slab of the case. (b) The second-floor slab of the case.



(c) The third-floor slab of the case.

Figure 6. The shapes of the first, second, and third floor slabs of this academic building in northeast view.

4. DISCUSSION

The proposed LLM agent for semi-automated floor area calculation has demonstrated significant potential in improving the efficiency and accuracy of the process. By leveraging CoT planning and the ReAct framework, the agent is capable of handling complex, multi-step reasoning tasks and ensuring the accuracy of the results. The empirical validation on the academic building showed positive outcomes, validating the effectiveness of the approach.

One of the key advantages of the LLM agent is its user-friendly interface, which makes it accessible to non-specialist users. Traditional methods and software tools often require advanced technical skills, which can be a barrier to widespread adoption. By simplifying the process and providing clear, understandable outputs, the LLM agent can be easily integrated into the workflow of architects and engineers, thereby reducing the learning curve and increasing productivity.

While the initial results are promising, there are several limitations that need to be addressed in future research. One limitation is the current focus on the simplest case, where the floor height exceeds 2.2 meters. Future work will involve expanding the agent's capabilities to handle more complex scenarios, such as the calculation of areas with varying heights, special features like stairwells, and other factors that affect the total floor area.

What's more, for slabs that have holes in them, which is a very common situation in real projects, the agent should be able to distinguish between the vertices of the holes and the vertices of the slab edges, and then calculate the correct area. This is also an issue to be addressed in the next steps of the research.

Finally, the reliance on LLMs introduces inherent biases and potential inaccuracies. The performance of the LLM agent in larger and more complex building projects needs to be evaluated to ensure its robustness and scalability. Future research should also explore the robustness and scalability of using AI in the AEC industry and develop guidelines to address these concerns.

5. CONCLUSIONS

The potential benefits of the developed tool are substantial, offering several advantages over existing software solutions. Unlike traditional methods that require specialized knowledge and can be cumbersome to use, our LLM agent simplifies the workflow, making it more accessible to non-expert users. The system's ability to interpret and process IFC files, combined with its natural language interaction capabilities, significantly lowers the technical barriers associated with BIM data manipulation.

This tool has the potential to transform the AEC industry by enabling more efficient and accurate floor area calculations, which in turn can lead to better-informed decision-making, improved project cost estimation, and enhanced productivity. In practical application, the tool can significantly reduce the time architects and engineers spend on area calculations, allowing them to focus more on creative design and other high-value activities. Additionally, for project management and cost estimation, having quick and accurate floor area data is crucial, and this tool can help enhance overall productivity and efficiency in the AEC industry.

As technology advances and research deepens, it is anticipated that the LLM agent will further optimize its performance to address a broader spectrum of scenarios and challenges. Future enhancements may extend the tool's capabilities to handle calculations involving complex features such as floor slabs with holes, curved surfaces, and distinct areas like balconies and stairs that lie outside the main structural envelope, as well as regions with height variations. Additionally, the tool could adapt to align with diverse regional or national regulations. Such advancements would render the LLM agent a more comprehensive and flexible solution, driving the standardization and automation of BIM workflows and setting new benchmarks for efficiency and accuracy in the AEC industry.

In conclusion, this LLM-based semi-automated floor area calculation system not only improves the efficiency and accuracy of area calculations in AEC industry by reducing reliance on manual calculations, which are often time-consuming and prone to human error, but also showcases significant industry potential with its user-friendliness and flexibility. As it iterates and improves, the tool is poised to revolutionize how floor area calculations are performed in the AEC industry, promoting overall workflow efficiency.

REFERENCES

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., & Anadkat, S. (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Lee, J.-K., Cho, K., Choi, H., Choi, S., Kim, S., & Cha, S. H. (2023). High-level implementable methods for automated building code compliance checking. *Developments in the Built Environment*, *15*, 100174.

Li, Y., Wang, S., Ding, H., & Chen, H. (2023). Large language models in finance: A survey. Proceedings of the fourth ACM international conference on AI in finance,

M. Bran, A., Cox, S., Schilter, O., Baldassari, C., White, A. D., & Schwaller, P. (2024). Augmenting large language models with chemistry tools. *Nature Machine Intelligence*, 1-11.

Özer, S., & Jacoby, S. (2022). Dwelling size and usability in London: a study of floor plan data using machine learning. *Building Research & Information*, *50*(6), 694-708.

Reinhardt, J., & Mathews, M. (2017). The automation of BIM for compliance checking: a visual programming approach.

Roanes-Lozano, E., Laita, L., Roanes-Macías, E., Maojo, V., Corredor, S., De La Vega, A., & Zamora, A. (2000). A Gröbner bases-based shell for rule-based expert systems development. *Expert Systems with Applications*, *18*(3), 221-230.

Sang, K. J., Ison, S. G., & Dainty, A. R. (2009). The job satisfaction of UK architects and relationships with work‐life balance and turnover intentions. *Engineering, Construction and Architectural Management*, *16*(3), 288-300.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, *35*, 24824-24837.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2022). React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Zheng, Z., Chen, K.-Y., Cao, X.-Y., Lu, X.-Z., & Lin, J.-R. (2023). Llm-funcmapper: Function identification for interpreting complex clauses in building codes via llm. *arXiv preprint arXiv:2308.08728*.