

# Kopplung von Building Information Modeling mit Behavior Trees als Grundlage für automatisierte Bauwerksinspektionen

Aditya Tandon<sup>id</sup>, Jan Stührenberg<sup>id</sup> and Saurabh Sawant

Institut für Digitales und Autonomes Bauen, Technische Universität Hamburg, Hamburg, Deutschland

E-mail(s): aditya.tandon@tuhh.de

**Abstract:** Das Monitoring und regelmäßige Wartungsinspektionen von Gebäuden und Infrastrukturbawerken sind unerlässlich, um Qualitätsstandards aufrechtzuerhalten, die die strukturelle Sicherheit und Integrität gewährleisten. Da manuelle Inspektionen zeitaufwändig und teuer sein können, wird der Einsatz von Robotern erforscht, um Gebäudeinspektionen zu automatisieren und Wartungsprozesse zu optimieren. Die Einführung robotergestützter Inspektionsmethoden wird jedoch durch eine fehlende Kopplung mit der Bauwerksinformationsmodellierung (Building Information Modeling, BIM) für eine situationsbezogene Navigation sowie eine mangelnde Anpassungsfähigkeit der Inspektionsplanung erschwert. In diesem Beitrag wird ein BIM-basiertes Framework für automatisierte Bauwerksinspektionen vorgestellt, um die genannten Herausforderungen zu bewältigen. Räumliche und semantische Daten werden aus BIM-Modellen extrahiert, um die Roboternavigation zu verbessern und zu inspizierende Gebäudeelemente zu identifizieren. Behavior Trees werden für eine anpassungsfähige und automatisierte Planung und Ausführung von Roboterinspektionsaufgaben genutzt. Das BIM-basierte Framework für automatisierte Bauwerksinspektionen wird auf vierbeinigen Robotern implementiert und durch die Durchführung von Inspektionsaufgaben in Bürogebäuden validiert. Die Ergebnisse zeigen, dass die Kopplung von BIM und Behavior Trees die Genauigkeit und Effizienz von automatisierten Gebäudeinspektionen verbessert. Zusammenfassend lässt sich sagen, dass das BIM-basierte Framework für automatisierte Bauwerksinspektionen einen Schritt in Richtung der Einführung von robotergestützter Gebäudeinspektion in der Praxis darstellt.

**Keywords:** Gebäudeinspektionen, Bauwerksinspektionen, Bauwerksinformationsmodellierung (BIM), Behavior Trees, vierbeinige Roboter, autonome Navigation



DOI: 10.18154/RWTH-CONV-254872. Published in the conference proceedings of the 36. Forum Bauinformatik 2025, Aachen, Germany.  
© 2025 The copyright for this article lies with the authors. This publication, except for quotations and otherwise indicated parts, is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) license.

## 1 Einführung

Regelmäßige Inspektionen von Gebäuden sind für die Aufrechterhaltung von Qualitätsstandards, die die strukturelle Sicherheit und Integrität gewährleisten, unerlässlich [1]. Gebäudeinspektionen werden üblicherweise manuell durchgeführt, wobei geschulte Inspektoren visuelle und strukturelle Bewertungen vornehmen [2]. Manuelle Inspektionen sind jedoch oft zeitaufwändig, arbeitsintensiv und

gefährlich und neigen zu subjektiven Qualitätsbewertungen [3]. Deshalb haben sich mobile Roboter als praktikable Lösung für die Automatisierung von Wartungsinspektionen von Gebäuden erwiesen. Sie bieten das Potenzial für sicherere, häufigere und objektivere Qualitätsbewertungen im Vergleich zu herkömmlichen manuellen Inspektionen [4]. Die Bauwerksinformationsmodellierung (Building Information Modeling, BIM) wird zunehmend eingesetzt, um standardisierte digitale Gebäudemodelle über den gesamten Lebenszyklus von Gebäuden zu nutzen [5]. BIM-Modelle enthalten sowohl semantische als auch geometrische Informationen, die für die Lokalisierung von Robotern innerhalb von Gebäuden, die Identifizierung von inspizierenden Gebäudeelementen, die Planung optimaler Pfade zu den Gebäudeelementen und die Durchführung von Inspektionsaufgaben genutzt werden können [6].

Kim und Peavy [7] haben ein Framework entwickelt, das BIM-Daten in das Universal Robot Description Format konvertiert, um Inspektionen zu planen und durchzuführen. Die Validierung wird jedoch ausschließlich in simulierten Umgebungen durchgeführt. Chen et al. [8] haben ein BIM-basiertes System für automatisierte Gebäudeinspektionen mit Robotern unter Annahme einer geschlossenen Welt vorgeschlagen. Insgesamt wird die Interoperabilität zwischen BIM und Robotern nur selten berücksichtigt, und die Anpassungsfähigkeit an reale dynamische Umgebungen oder die Neuplanung von Inspektionen ist begrenzt [9]. Darüber hinaus können Abweichungen zwischen dem geplanten Zustand, der in BIM-Modellen widerspiegelt wird, und dem tatsächlichen Zustand die Aufgaben- und Bewegungsplanung von Robotern bei der Automatisierung von Gebäudeinspektionen beeinträchtigen. Um die Anpassungsfähigkeit von Robotern in dynamischen Umgebungen zu verbessern, haben Behavior Trees an Popularität gewonnen. Der hierarchische Aufbau von Behavior Trees ermöglicht eine adaptive und reaktive Aufgabenplanung in Verbindung mit flexiblen Strategien zur Aufgabenausführung in unsicheren und dynamischen Umgebungen [10].

In diesem Beitrag wird ein BIM-basiertes Inspektionsframework präsentiert, das BIM und Behavior Tree (BT)-Kontrollstrukturen verbindet, um automatisierte Gebäudeinspektionen mit vierbeinigen Robotern durchzuführen. Zu inspizierende Gebäudeelemente werden aus BIM-Modellen extrahiert. Um zu den Gebäudeelementen zu navigieren, werden künstliche Marker eingesetzt, um die BIM-Modelle an Karten für die Roboternavigation auszurichten. Inspektionsaufträge werden als BTs generiert, basierend auf der Art der Gebäudeelemente und den Fähigkeiten der Roboter zur Durchführung von Inspektionen.

Der Rest des Beitrags ist wie folgt gegliedert. Zunächst wird der Entwurf des BIM-basierten Inspektionsframeworks vorgestellt. Daraufhin werden die Implementierung und Validierung des Frameworks beschrieben und die Ergebnisse der Validierungstests diskutiert. Abschließend werden die Ergebnisse zusammengefasst und mögliche zukünftige Forschungsrichtungen vorgeschlagen.

## **2 Ein BIM-basiertes Inspektionsframework mit BT-Kontrollstrukturen**

In diesem Abschnitt wird das Design des BIM-basierten Inspektionsframeworks vorgestellt. Zunächst wird die Extraktion von Gebäudeelementen aus BIM-Modellen für die Generierung von Inspektionsaufgaben vorgestellt, gefolgt von der Ausrichtung von BIM-Modellen mit Karten für die Roboternavigation und der automatischen Generierung von Behavior-Trees aus Gebäudeelementen.

## 2.1 Extraktion von Gebäudeelementen aus BIM-Modellen

Als Grundlage für zuverlässige und präzise Inspektionen in Gebäuden werden Informationen, wie etwa Abmessungen und Positionen, über spezifische zu inspizierende Gebäudeelemente („Zielelemente“) aus BIM-Modellen extrahiert. Inspektionsposen werden in festgelegtem Abstand zu den Zielelementen definiert und so ausgerichtet, dass sie die Zielelemente frontal erfassen und somit eine optimale Inspektion ermöglichen. Die Inspektionspose  ${}^B_I\mathbf{P}$  im Bezugssystem des BIM-Modells  $\mathbf{B}$ , für das  $n$ -te Gebäudeelement vom Typ  $\mathbf{T}$ , wird folgendermaßen berechnet:

$${}^B_I\mathbf{P}_{T_n} = \mathbf{H}_T {}^B_E\mathbf{P}_{T_n} \in SE(3) \quad (1)$$

Dabei bezeichnet  ${}^B_I\mathbf{P}_{T_n}$  die Inspektionspose für das  $n$ -te Element vom Typ  $\mathbf{T}$  in  $\mathbf{B}$ ,  $\mathbf{H}_T$  die homogene Transformationsmatrix, die den festen Abstand und die Ausrichtung für Elemente des Typs  $\mathbf{T}$  enthält, und  ${}^B_E\mathbf{P}_{T_n}$  die Pose des Gebäudeelements für das  $n$ -te Element vom Typ  $\mathbf{T}$  in  $\mathbf{B}$ .

## 2.2 Ausrichtung von BIM-Modellen mit Karten für die Roboternavigation

Die Zielelemente sind im Bezugssystem des BIM-Modells  $\mathbf{B}$  definiert. Bei herkömmlichen Roboternavigationsansätzen werden jedoch Karten zur Lokalisierung und Pfadplanung eingesetzt, die üblicherweise in einem von  $\mathbf{B}$  separaten Koordinatensystem  $\mathbf{M}$  definiert sind. Daher ist eine Transformation  ${}^B\mathbf{T}_M$  erforderlich, um das Bezugssystem des BIM-Modells  $\mathbf{B}$  und das Kartenkoordinatensystem  $\mathbf{M}$  auszurichten.

Das BIM-basierte Inspektionsframework berechnet die starre Transformation  ${}^B\mathbf{T}_M$  anhand mehrerer Korrespondenzen von leicht zu erkennenden Referenzmarken. Als Referenzmarken werden AprilTags verwendet [11], die an denselben Stellen im realen Gebäude und im entsprechenden BIM-Modell platziert werden. Durch die Platzierung "virtueller" AprilTags im BIM-Modell können die Positionen der AprilTags in Bezug auf  $\mathbf{B}$  abgerufen werden. Die Beobachtung der in realen Gebäuden platzierten AprilTags mit an Robotern montierten Kameras ermöglicht die Berechnung von Transformationen von den Robotern zu den AprilTags. Kombiniert man die Transformationen mit den aktuellen Posen der Roboter in der Karte, erhält man die Posen der AprilTags in Bezug auf  $\mathbf{M}$ . Da die AprilTags ID-Nummern enthalten, lassen sich die entsprechenden AprilTags in BIM-Modellen und in realen Gebäuden schnell und eindeutig identifizieren. Während die Roboter Gebäude durchqueren und AprilTags beobachten, werden Korrespondenzen von realen und virtuellen AprilTags erfasst. Wenn das erste AprilTag beobachtet wird, wird die erste Korrespondenz erfasst, und die starre Transformation  ${}^B\mathbf{T}_M$  kann wie folgt berechnet werden:

$${}^B\mathbf{T}_M = {}^B\mathbf{T}_{A_i} {}^M_t\mathbf{T}_{A_i}^{-1} \in SE(3) \quad \text{mit} \quad {}^M_t\mathbf{T}_{A_i} = {}^M_t\mathbf{T}_R {}^R_t\mathbf{T}_{A_i} \quad (2)$$

Dabei bezeichnet  ${}^B\mathbf{T}_{A_i}$  die Pose des AprilTags mit der ID  $i$  im Bezugssystem des BIM-Modells  $\mathbf{B}$ ,  ${}^M_t\mathbf{T}_{A_i}^{-1}$  die Transformationsschätzung des AprilTags mit der ID  $i$  in das Kartenkoordinatensystem  $\mathbf{M}$  zum Zeitpunkt  $t$ ,  ${}^M_t\mathbf{T}_R$  die Pose des Roboters in  $\mathbf{M}$  zum Zeitpunkt  $t$  und  ${}^R_t\mathbf{T}_{A_i}$  die Transformation vom durch die Kamera beobachteten AprilTag mit der ID  $i$  in das Roboterkoordinatensystem  $\mathbf{R}$  zum Zeitpunkt  $t$ . Die Genauigkeit von  ${}^M_t\mathbf{T}_{A_i}$  und folglich auch die Genauigkeit von  ${}^B\mathbf{T}_M$  wird jedoch durch Messrauschen und die Bewegung des Roboters beeinflusst und können zu einer ungenauen

Ausrichtung des BIM-Modells und der Karte führen. Um eine genaue Transformation  ${}^B\mathbf{T}_M$  zu erhalten, werden  $m$  Detektionen von AprilTags in aufeinanderfolgenden Kamerabildern berücksichtigt, um die mittlere starre Transformation  ${}^M\overline{\mathbf{T}}_{A_i}$  zu berechnen. Die Varianz dieser Transformationen wird auf Basis des Root Mean Square Error (RMSE) bewertet. Transformationen geringer Qualität, die möglicherweise zu Fehlausrichtungen führen, werden identifiziert, wenn der RMSE den Schwellwert  $\tau_{RMSE}$  überschreitet, und von weiteren Berechnungen ausgeschlossen.

Zusätzlich wird die mittlere Transformation  ${}^B\overline{\mathbf{T}}_M$  anhand mehrerer AprilTag-Korrespondenzen berechnet, um mögliche Mess- und Platzierungsfehler der AprilTags auszugleichen und eine genaue Ausrichtung zu gewährleisten. Die Transformation  ${}^B\overline{\mathbf{T}}_M$  wird neu berechnet, sobald eine neue AprilTag-Korrespondenz mit einer hochwertigen Transformation  ${}^M\overline{\mathbf{T}}_{A_i}$  akzeptiert wurde.

### 2.3 Automatisierte Generierung von BTs aus Gebäudeelementen

Eine zuverlässige und genaue Planung von Inspektionsaufgaben erfordert Strukturen, die den Ausführungsablauf von Robotern dynamisch adaptieren. In dieser Studie werden Behavior Trees aufgrund ihrer Modularität, Anpassungsfähigkeit und Wiederverwendbarkeit verwendet. Im Wesentlichen werden die Zielelemente aus Abschnitt 2.1 analysiert, um die Art und den Ort der für die Gebäudeinspektion erforderlichen Inspektionsaufgaben zu bestimmen. Für jedes Zielelement wird ein eindeutiger Satz von Inspektionsaufgaben definiert, der auf funktionalen und betrieblichen Anforderungen basiert. Die Inspektionsaufgaben werden den spezifischen Roboterfähigkeiten zugeordnet, um sicherzustellen, dass die Roboter die Anforderungen erfüllen können. Auf der Grundlage der Zuordnung von Inspektionsaufgaben und Roboterfähigkeiten werden BT-Subtrees generiert, die die Logik, Parameter und Ausführungssequenzen enthalten.

In dieser Studie werden der Navigations-Subtree und der visuelle Inspektions-Subtree entwickelt. Der in Abbildung 1 dargestellte Navigations-Subtree erleichtert die autonome Planung und Ausführung von Navigationspfaden und die Fehlerbehebung im Falle von Navigationsfehlern. Der Navigations-Subtree wertet die Bedingungen "Zielpose erhalten" und "Zielpose erreicht" aus, bevor die Navigation erfolgt. Ziele werden mithilfe von "Ziel planen" geplant und "Zum Ziel navigieren" ausgeführt.

Der Subtree zur visuellen Inspektion, dargestellt in Abbildung 2 führt eine Reihe von visuellen Inspektionen innerhalb der Betriebsumgebung aus. Nach der Auswertung der Bedingung zum Start der Inspektion aktiviert der Subtree die für die visuelle Inspektion notwendigen Module (z.B. Kamera und Objekterkennung) und erfasst Bilder zur Dokumentation.

Um größere Abfolgen von Inspektionsaufgaben zu erstellen, werden der Navigations-Subtree und der Subtree zur visuellen Inspektion in Steuerstrukturen für die Ausführung von BTs miteinander verknüpft. Dadurch können Roboter zu Inspektionspunkten navigieren, visuelle Inspektionen durchführen und zu nachfolgenden Inspektionspunkten weiterziehen. Im folgenden Abschnitt werden die Implementierung und Validierung des BIM-basierten Inspektionsframeworks beschrieben und die Ergebnisse diskutiert.

## 3 Implementierung und Validierung des Inspektionsframeworks

In diesem Abschnitt wird die Implementierung und Validierung des BIM-basierten Inspektionsframeworks vorgestellt. Schließlich werden die Ergebnisse der Validierungstests vorgestellt und diskutiert.

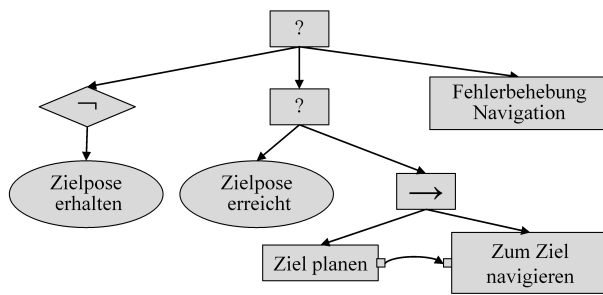


Abbildung 1: Navigations-Subtree

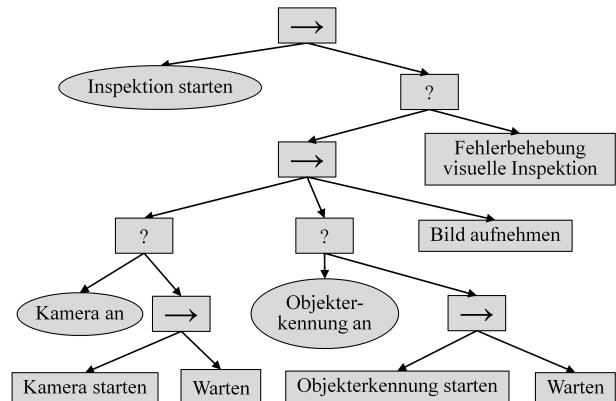


Abbildung 2: Visueller Inspektions-Subtree

### 3.1 Implementierung

Um die Extraktion der Gebäudeelemente aus BIM-Modellen zu ermöglichen, wird das offene und standardisierte Industry Foundation Classes (IFC)-Schema verwendet. Zur Implementierung des BIM-basierten Inspektionsframeworks wird die Open-Source-Bibliothek *IfcOpenShell* verwendet, die das Parsen von IFC-Dateien ermöglicht. Die Extraktion von Gebäudeelementen erfolgt über die Python-API der *IfcOpenShell*-Bibliothek. Sobald die Zielelemente ermittelt sind, werden entsprechende Inspektionsposen für die Roboter in vordefinierten Abständen zu den Zielelementen in **B** berechnet.

Um das BIM-Modell und die Karten zur Navigation auszurichten, wird eine benutzerdefinierte, mit Autodesk Revit kompatible AprilTag-Familie verwendet, die im BIM-Modell platziert wird. Aus dem BIM-Modell werden die Tags als *IfcBuildingElementProxy*-Elemente in das IFC-Format exportiert, und die Posen der AprilTags  ${}^B\mathbf{T}_{A_i}$  werden mithilfe der *IfcOpenShell*-Bibliothek automatisch extrahiert. Im realen Gebäude werden die AprilTags auf Papier gedruckt und entsprechend den Positionen im BIM-Modell platziert, und die Bibliothek *apriltag\_ros* wird verwendet, um die Tags im realen Gebäude zu erkennen. Das Optimierungsproblem der Berechnung der mittleren Transformationen  ${}^M\overline{\mathbf{T}}_{A_i}$  und  ${}^B\overline{\mathbf{T}}_M$  ist in Python unter Verwendung der *scipy.optimize.least\_squares* Bibliothek implementiert.

Die Kontrollstrukturen für die Ausführung von BTs werden mithilfe von Python im Robot Operating System (ROS) implementiert. Zur Generierung von BTs für Inspektionsmissionen wird das *ros\_bt\_py*-Framework [12] erweitert, um benutzerdefinierte Knoten zu erstellen. Die Navigations- und visuellen Inspektions-Subtrees werden über eine Programmierschnittstelle (API) mit einer vordefinierten logischen Struktur implementiert. Die Subtrees werden mithilfe von ROS-Serviceaufrufen miteinander verkettet. Für weitere Informationen zur Roboternavigation sei auf [3] verwiesen.

### 3.2 Validierung

Die Validierungstests werden in einer Büroumgebung mit den Abmessungen 39 m x 16 m und einem entsprechenden BIM-Modell durchgeführt (Abbildung 3). Die Tests dienen dazu, die Genauigkeit und Zuverlässigkeit des BIM-basierten Inspektionsframeworks zu validieren. Die Genauigkeit wird durch Messung der Abstände zwischen den geplanten sowie den von den Robotern erreichten Inspektionsposen und Berechnung des RMSE bewertet. Die Zuverlässigkeit des BIM-basierten Inspektionsframeworks wird anhand der Fähigkeit bewertet, Inspektionsaufgaben automatisch ohne

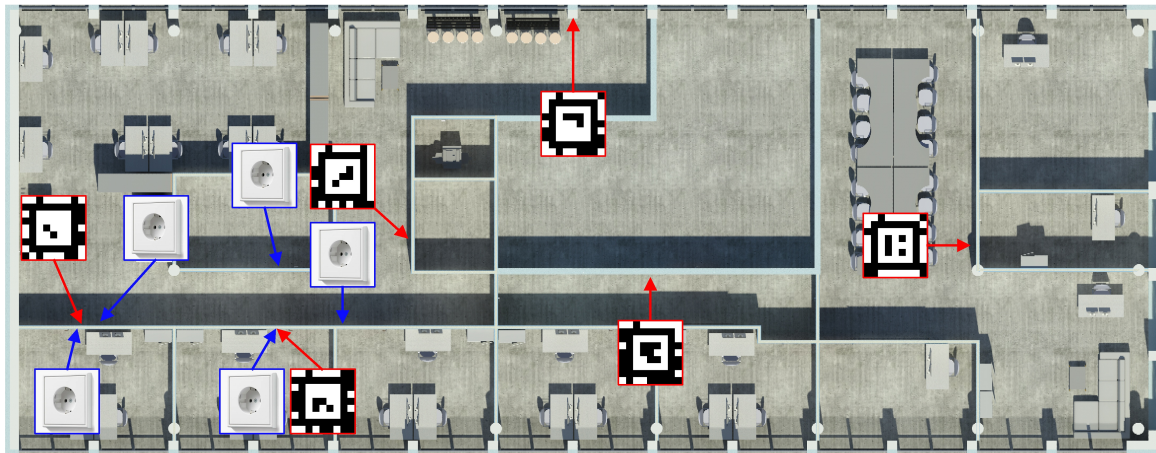


Abbildung 3: BIM-Modell der Büroumgebung mit Steckdosen und AprilTags

menschliches Eingreifen durchzuführen, d. h. (i) Navigation zu den Inspektionsposen, (ii) Ausführung der BT-Strukturen und (iii) erfolgreiche Erkennung und Dokumentation von Zielelementen.

Die Validierungstests werden wie folgt durchgeführt: Elektrische Steckdosen werden als Zielelemente für visuelle Inspektionen ausgewählt. Das in das IFC-Format exportierte BIM-Modell der Büroumgebung wird nach dem entsprechenden IFC-Element `IfcOutlet` durchsucht. Dann werden die Inspektionsmissionen, bestehend aus Navigations- und Sichtprüfungsaufgaben, geplant. Die Inspektionsposen werden mit 1 m Abstand zu den Steckdosen geplant. Für eine genaue Ausrichtung des BIM-Modells und der Karten werden sechs AprilTags strategisch an den Wänden der Büroumgebung und im BIM-Modell platziert (Abbildung 3).

Die Validierungstests werden mit einem vierbeinigen Roboter, dem in Abbildung 4 gezeigten "Intelligent Documentation Gadget" (IDOG), durchgeführt, der bereits in früheren Studien erfolgreich zur Durchführung von Inspektionen zur strukturellen Gesundheitsüberwachung eingesetzt wurde [13]. Der IDOG verwendet das in [3] beschriebene Navigationsframework. Physikalische Inspektionsposenmarker werden an den entsprechenden Inspektionsposen im realen Gebäude platziert. Es werden drei Inspektionseinsätze durchgeführt, um die Genauigkeit anhand der Abweichung zwischen den geplanten Inspektionsposen und den tatsächlich erreichten Posen des IDOG zu testen. Für die Zuverlässigkeit des BIM-basierten Inspektionsframeworks werden die Anzahl der manuellen Eingriffe aufgezeichnet. Die Ergebnisse werden im folgenden Unterabschnitt zusammengefasst.

### 3.3 Ergebnisse

Alle fünf Inspektionsposen werden erfolgreich erzeugt. Die Ausrichtung des BIM-Modells und der Karte für die Roboternavigation wird mit geringen Abweichungen durchgeführt. Die Ergebnisse der Validierungstests hinsichtlich Genauigkeit sind in Tabelle 1 dargestellt. Im RMSE zusammengefasst, wird ein Fehler von 0,129 m beobachtet. Der IDOG navigiert zu den Inspektionsposen ohne manuellen Eingriff (Abbildung 5). Das Inspektionsmodul ist in der Lage, die BT-Strukturen zuverlässig auszuführen und die Zielelemente, d. h. die Steckdosen, zu erkennen und den Zustand erfolgreich durch die Aufnahme von Bildern inklusive Zeitstempel zu dokumentieren.

Tabelle 1: Abweichungen zwischen geplanten und tatsächlichen Posen des IDOGs in Metern

| Test Nr. | Pos 1 | Pos 2 | Pos 3 | Pos 4 | Pos 5 |
|----------|-------|-------|-------|-------|-------|
| 1        | 0,18  | 0,12  | 0,18  | 0,14  | 0,11  |
| 2        | 0,09  | 0,06  | 0,16  | 0,07  | 0,22  |
| 3        | 0,13  | 0,06  | 0,13  | 0,07  | 0,07  |

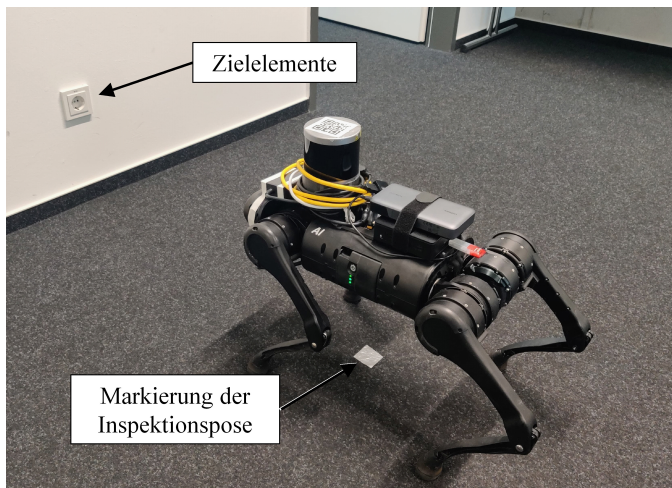


Abbildung 4: Vierbeiniger Roboter "IDOG" in den Validierungstests

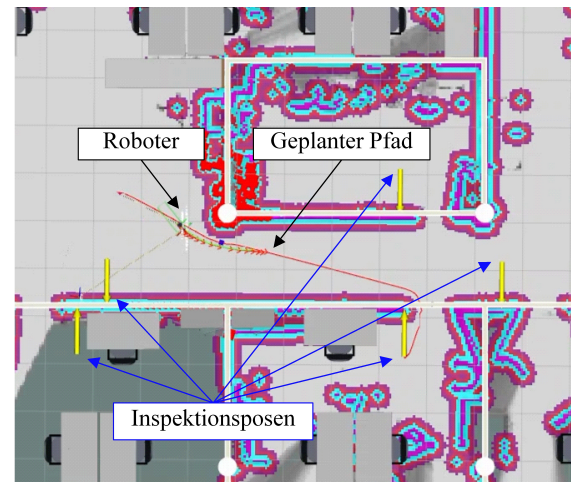


Abbildung 5: IDOG navigiert zu Inspektionsposen

### 3.4 Diskussion

Die Abweichungen zwischen den geplanten und den vom IDOG erreichten Inspektionsposen werden von der Genauigkeit der Platzierung der AprilTags und der Zielelemente im BIM-Modell beeinflusst. Abweichungen zwischen realen und der im BIM-Modell platzierten virtuellen AprilTags führen zu Fehlausrichtungen der Karte und des BIM-Modells. Abweichungen zwischen den Posen der realen und virtuellen Zielelemente im BIM-Modell führen zu Messfehlern. Insgesamt ist die Genauigkeit mit einem RMSE von 0,129 m für visuelle Inspektionen ausreichend. Das BIM-basierte Inspektionsframework hat sich als zuverlässig erwiesen, da alle Inspektionsaufträge von den IDOGs ohne menschliches Eingreifen ausgeführt wurden. Im nächsten Abschnitt wird die Arbeit zusammengefasst und es werden Schlussfolgerungen gezogen.

## 4 Zusammenfassung und Schlussfolgerungen

BIM-Modelle enthalten wertvolle semantische und geometrische Informationen, um automatisierte Gebäudeinspektionen mit mobilen Robotern zu verbessern. In diesem Beitrag wurden die Herausforderungen der Interoperabilität angegangen, indem künstliche Referenzmarken zur Kartenausrichtung sowie BT-Kontrollstrukturen eingesetzt wurden. Das entwickelte BIM-basierte Inspektionsframework wurde per automatischer visueller Inspektion von Steckdosen mit einem vierbeinigen Roboter in einer Büroumgebung validiert. Die Validierungstests zeigen die Fähigkeit des Systems, genaue und zuverlässige Gebäudeinspektionen zu ermöglichen. Die Extraktion von Gebäudedaten aus BIM-Modellen ermöglichte eine kontextbewusste Navigation, die Integration von künstlichen Referenzmarken ermöglichte eine genaue Kartenausrichtung und die BT-Kontrollstrukturen boten einen

robusten Mechanismus zur Anpassung von Inspektionsaufgaben an reale Umgebungen. In zukünftigen Arbeiten könnte das Framework erweitert werden, um verschiedene Arten von Gebäudeelementen und Inspektionsaufgaben zu integrieren. Weiterhin könnte mehreren Robotern die gemeinsame Durchführung von Inspektionsaufgaben ermöglicht werden.

## Literatur

- [1] S. Halder und K. Afsari, »Robots in inspection and monitoring of buildings and infrastructure: A systematic review«, *Applied Sciences*, Jg. 13, Nr. 4, S. 2304, 2023.
- [2] M. H. Shariq und B. R. Hughes, »Revolutionising building inspection techniques to meet large-scale energy demands: A review of the state-of-the-art«, *Renewable and Sustainable Energy Reviews*, Jg. 130, S. 109 979, 2020.
- [3] A. Tandon, J. Stührenberg, K. Dragos und K. Smarsly, »Autonomous navigation of quadruped robots for monitoring and inspection of civil infrastructure«, in *Advances in Information Technology in Civil and Building Engineering*, A. Francis, E. Miresco und S. Melhado, Hrsg., Bd. 629, Cham: Springer Nature Switzerland, 2025, S. 347–360.
- [4] K. Smarsly, K. Dragos, J. Stührenberg und M. Worm, »Mobile structural health monitoring based on legged robots«, *Infrastructures*, Jg. 8, Nr. 9, S. 136, 2023.
- [5] M. Theiler und K. Smarsly, »IFC Monitor – An IFC schema extension for modeling structural health monitoring systems«, *Advanced Engineering Informatics*, Jg. 37, S. 54–65, 2018.
- [6] J. Stührenberg und K. Smarsly, »LIO-BIM – Coupling lidar inertial odometry with building information modeling for robot localization and mapping«, *Advanced Engineering Informatics*, Jg. 66, S. 103 477, 2025.
- [7] K. Kim und M. Peavy, »BIM-based semantic building world modeling for robot task planning and execution in built environments«, *Automation in Construction*, Jg. 138, S. 104 247, 2022.
- [8] J. Chen, W. Lu, Y. Fu und Z. Dong, »Automated facility inspection using robotics and BIM: A knowledge-driven approach«, *Advanced Engineering Informatics*, Jg. 55, S. 101 838, 2023.
- [9] A. Tandon, J. Stührenberg und K. Smarsly, »Automated building inspections coupling building information modeling and behavior trees«, in *Proceedings of the 2025 European Conference on Computing in Construction (EC3)*, Porto, Portugal, Juli 2025.
- [10] R. Ghzouli, T. Berger, E. B. Johnsen, A. Wasowski und S. Dragule, »Behavior trees and state machines in robotics applications«, *IEEE Transactions on Software Engineering*, Jg. 49, Nr. 9, S. 4243–4267, 2023.
- [11] E. Olson, »AprilTag: A robust and flexible visual fiducial system«, in *2011 IEEE international conference on robotics and automation*, IEEE, 2011, S. 3400–3407.
- [12] G. Heppner, N. Berg, D. Oberacker, N. Spielbauer, A. Roennau und R. Dillmann, »Distributed behavior trees for heterogeneous robot teams«, in *2023 IEEE 19th International Conference on Automation Science and Engineering (CASE)*, 2023, S. 1–8.
- [13] S. Johann u. a., »Implementation and validation of robot-enabled embedded sensors for structural health monitoring«, *The e-journal of nondestructive testing & ultrasonics*, 2024.